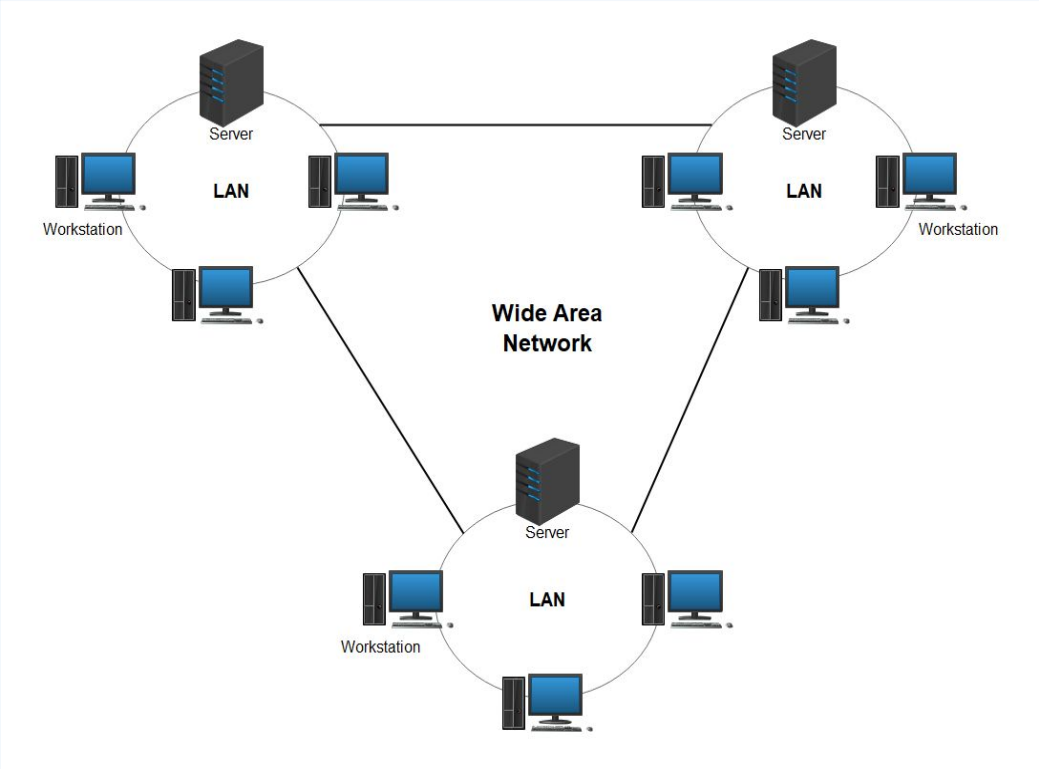
Протокол IP

Сетевой уровень стека TCP/IP

Задача межсетевого взаимодействия

- **Канальный уровень** (Ethernet) доставляет кадры в пределах одного сегмента сети (LAN).
- **Сетевой уровень** (IP) отвечает за доставку пакетов между разными сетями (WAN).



Модели OSI и TCP/IP

Модель OSI

- Уровень 7: Прикладной
- Уровень 6: Представительский
- Уровень 5: Сеансовый
- Уровень 4: Транспортный
- **-> Уровень 3: Сетевой -> IP**
- Уровень 2: Канальный
- Уровень 1: Физический

Модель TCP/IP

- Уровень 4: Прикладной
- Уровень 3: Транспортный
- **-> Уровень 2: Межсетевое взаимодействие -> IP**
- Уровень 1: Сетевой интерфейс

IP-адреса

Форма представления:

- 4 десятичных числа 0-255, разделенных точками (октеты)
- всего 32 бита

213.180.193.3

11010101.10110100.11000001.00000011

Структура IP-адреса

Структура IP-адреса:

- Номер подсети - старшие биты
- Номер хоста - младшие биты

Пример:

- IP-адрес: 213.180.193.3
- Номер подсети: 213.180.193.0
- Номер хоста: 3 (0.0.0.3)

Маска подсетей

Определение: 32-битное число, которое разделяет IP-адрес на две части:

- **Сеть:** Биты, покрытые 1 в маске.
- **Хост:** Биты, покрытые `0` в маске.

Принцип:

- **1 в маске** = бит номера сети.
- **`0` в маске** = бит номера хоста.

Формат записи CIDR (Classless Inter-Domain Routing):

- **/число** (например, `/24`) — это количество единичных битов в маске.
- **Пример:** `/24` = `255.255.255.0`

Пример

172.16.6.27 - IP-адрес
255.255.0.0 (/16) - маска подсети

10101100.00010000.00000110.00011011 - IP-адрес
111111111.111111111.00000000.00000000 - маска подсети
10101100.00010000.00000000.00000000 - адрес сети

172.16.0.0 - IP-адрес

Система адресации IPv4: Классы сетей

Класс	Первые биты	Октеты (С - сеть, Х - хост)	Число возможных адресов сетей	Маска подсети	Число возможных адресов хостов	Диапазон
A	0	C.X.X.X	126	255.0.0.0	$2^{24} - 2 =$ 16 777 214	1.0.0.0 - 127.255.255.255
B	10	C.C.X.X	16 384	255.255.0.0	$2^{16} - 2 =$ 65 534	128.0.0.0 - 191.255.255.255
C	110	C.C.C.X	2 097 152	255.255.255.0	$2^8 - 2 = 254$	192.0.0.0 - 223.255.255.255
D	1110	Групповой адрес				224.0.0.0 - 239.255.255.255
E	1111	Зарезервировано				240.0.0.0 - 255.255.255.255

Специальные IPv4-адреса

Loopback (обратная связь):

- Диапазон: `127.0.0.0/8` (чаще `127.0.0.1`).
- Назначение: Взаимодействие процессов, тестирование сетевого стека на самом хосте. Пакет не покидает устройство.

Private (частные):

- `10.0.0.0/8`, `172.16.0.0/12`, `192.168.0.0/16`.
- Назначение: Использование в закрытых сетях. Не маршрутизируются в Интернете.

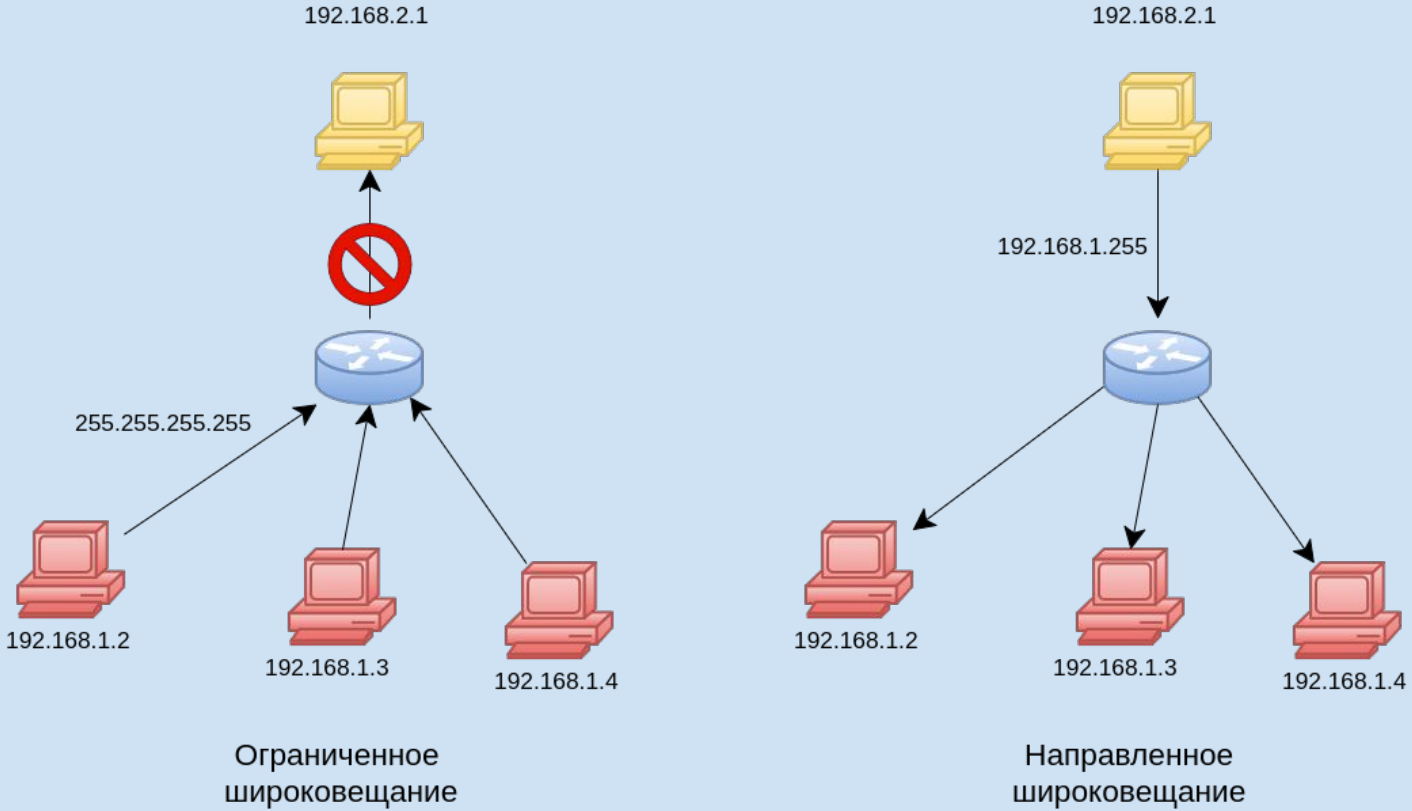
Broadcast (широковещательный):

- В номере хоста все единицы
- Пример: `192.168.1.255` для сети `192.168.1.0/24`.
- Назначение: Обращение ко всем узлам в локальной сети.

Multicast (групповой):

- Диапазон: `224.0.0.0/4` (Класс D).
- Назначение: Обращение к группе узлов
- Примеры известных адресов:
  - `224.0.0.1` — Все хосты в данной подсети
  - `224.0.0.2` — Все маршрутизаторы в данной подсети

Два широковещательных адреса



VLSM (Variable Length Subnet Mask)

Суть технологии

Создание подсетей **разного размера** внутри одной исходной сети. Позволяет проводить многоуровневую, иерархическую разбивку адресного пространства.

Ключевое требование

Для работы требуется использование **протоколов маршрутизации с поддержкой бесклассовой адресации (CIDR)**, которые передают в своих обновлениях не только адрес сети, но и его маску.

- **Примеры:** OSPF, EIGRP, RIPv2, IS-IS.
- **Не поддерживает:** RIPv1 (передает только классовые адреса без маски).

Преимущества

- **Максимальная эффективность** использования IP-адресов.
- **Устранение "пустых" IP-пространств** за счет соответствия размера подсети реальным потребностям.
- **Создание иерархической структуры сети**, что упрощает маршрутизацию и управление.

VLSM Пример

Дано: Сеть 192.168.1.0/24
Требуется создать подсети для:

- Отдел продаж: 60 хостов
- Отдел маркетинга: 28 хостов
- Бухгалтерия: 12 хостов
- Связь между маршрутизаторами: 2 хоста

Решение с помощью VLSM

Пример расчета для бухгалтерии

IP-адрес: 192.168.1.100
Маска: 255.255.255.240

`11000000.10101000.00000001.01100100` - IP-адрес
`11111111.11111111.11111111.11110000` - маска подсети
`11000000.10101000.00000001.01100000` - адрес сети

192.168.1.96 - адрес сети подсети бухгалтерии

Отдел продаж:	192.168.1.0 / 26	255.255.255.192
Отдел маркетинга	192.168.1.64 / 27	255.255.255.224
Бухгалтерия	192.168.1.96 / 28	255.255.255.240
Связь между маршрутизаторами	192.168.1.112 / 30	255.255.255.252

Бесклассовая адресация (CIDR)

Основная концепция

Полный **отказ от жесткой классовой системы (Class A, B, C)**. Граница между сетевой и хостовой частью стала гибкой и определяется исключительно маской (префиксом) переменной длины.

Нотация CIDR

- **Формат:** `IP-адрес/длина_префикса`
- **Пример:** `192.168.1.0/24`
- **Длина префикса:** Количество бит в маске подсети, отведенное под адрес сети.

Супернетинг (Агрегация маршрутов)

Объединение нескольких смежных сетей в один объявленный маршрут (суперсеть).

CIDR Пример

- **Провайдер** выдал 4 сети класса C своим клиентам:
  - `192.168.0.0/24`
  - `192.168.1.0/24`
  - `192.168.2.0/24`
  - `192.168.3.0/24`
- **Результат:** В глобальную маршрутизацию передаётся **4 отдельных маршрута**.

Решение: Агрегация маршрутов (CIDR)

- Смотрим на адреса в **двоичном виде**:

192.168.0.0 → 11000000.10101000.000000**00**.00000000
192.168.1.0 → 11000000.10101000.000000**01**.00000000
192.168.2.0 → 11000000.10101000.000000**10**.00000000
192.168.3.0 → 11000000.10101000.000000**11**.00000000

- Первые 22 бита у всех сетей одинаковые.

Итог: Все 4 сети можно описать **одной записью**:

192.168.0.0/22

Преимущество: Радикальное сокращение записей в таблицах маршрутизации.

Утилита ipcalc - расчет параметров сети

Назначение утилиты

- Расчет широковещательного адреса, адреса сети, диапазона допустимых адресов хостов.
- Преобразование масок между разными форматами (десятичный, CIDR, бинарный).
- Визуализация принадлежности адреса к классовой сети и частным диапазонам.
- Проверка корректности введенных данных.

Демонстрация ipcalc

```
$ ipcalc 172.16.64.10/18
Address:   172.16.64.10          10101100.00010000.01 000000.00001010
Netmask:   255.255.192.0 = 18    11111111.11111111.11 000000.00000000
Wildcard:  0.0.63.255            00000000.00000000.00 111111.11111111
=>
Network:   172.16.64.0/18        10101100.00010000.01 000000.00000000
HostMin:   172.16.64.1           10101100.00010000.01 000000.00000001
HostMax:   172.16.127.254        10101100.00010000.01 111111.11111110
Broadcast: 172.16.127.255        10101100.00010000.01 111111.11111111
Hosts/Net: 16382                  Class B, Private Internet
```

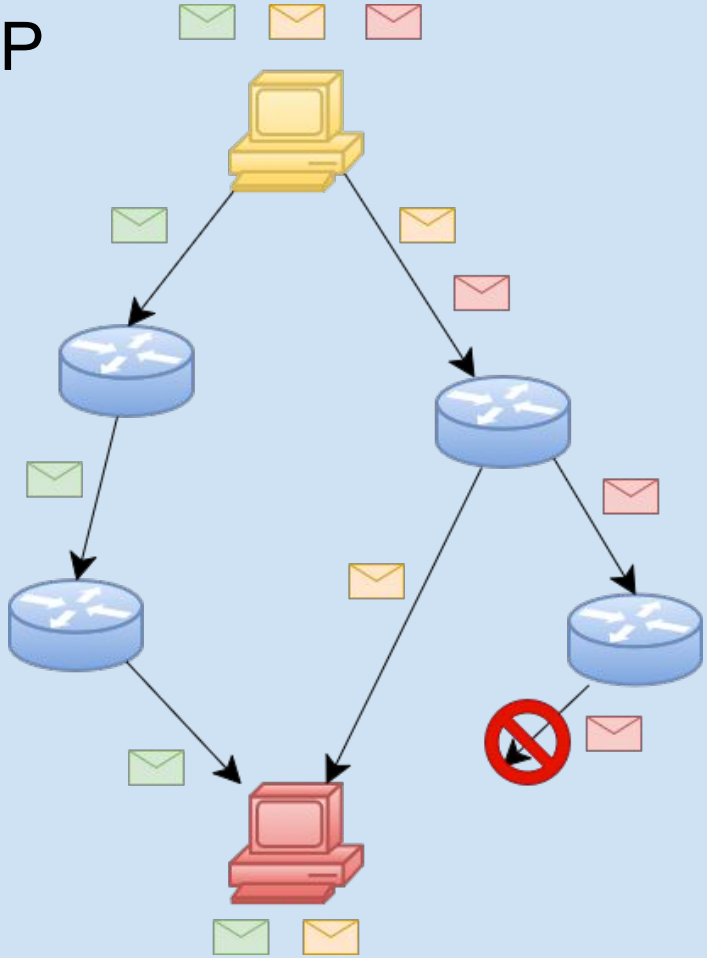
Демонстрация ipcalc

```
$ ipcalc 199.201.34.45
Address:   199.201.34.45        11000111.11001001.00100010. 00101101
Netmask:   255.255.255.0 = 24   11111111.11111111.11111111. 00000000
Wildcard:  0.0.0.255            00000000.00000000.00000000. 11111111
=>
Network:   199.201.34.0/24      11000111.11001001.00100010. 00000000
HostMin:   199.201.34.1         11000111.11001001.00100010. 00000001
HostMax:   199.201.34.254       11000111.11001001.00100010. 11111110
Broadcast: 199.201.34.255       11000111.11001001.00100010. 11111111
Hosts/Net: 254                   Class C
```

Общая характеристика протокола IP

- **Задачи:**
  - Объединение сетей
  - Маршрутизация
- **Дейтаграммный режим:**
  - Каждый пакет (дейтаграмма) независим.
  - Маршруты и порядок доставки могут различаться.
- **Отсутствие гарантий доставки:**
  - Не подтверждает доставку пакетов.
  - Не осуществляет повторную передачу потерянных пакетов.
- **Отсутствие установления соединения:**
  - Не создает виртуальный канал перед передачей данных.

Примечание: Надежность обеспечивается протоколами транспортного уровня (например, TCP)



Формат заголовка IPv4-пакета

<table>
<tr><td>Version</td><td>Length (IHL)</td><td>Type of Service (TOS)</td><td colspan="2">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flag</td><td>Fragment Offset</td></tr>
<tr><td colspan="2">Time To Live (TTL)</td><td>Protocol</td><td colspan="2">Header Checksum</td></tr>
<tr><td colspan="5">Source IP Address</td></tr>
<tr><td colspan="5">Destination IP Address</td></tr>
<tr><td colspan="5">Options</td></tr>
</table>

Формат заголовка IPv4-пакета

Основная информация:

- Минимальный размер: **20 байт** (без опций)
- Максимальный размер: **60 байт** (с опциями)
- Выравнивание: Поля выровнены по 32-битной границе

- **Version (4 бита):** Версия протокола = 4
- **IHL (4 бита):** Длина заголовка в 32-битных словах
- **Type of Service (8 бит):** Приоритет и тип обслуживания (QoS)
- **Total Length (16 бит):** Общая длина пакета (заголовок + данные)
- **Identification (16 бит):** Идентификатор для сборки фрагментов
- **Flags (3 бита):** Флаги фрагментации (MF, DF)
- **Fragment Offset (13 бит):** Смещение фрагмента
- **Time to Live (8 бит):** Время жизни пакета (макс. число хопов)
- **Protocol (8 бит):** Вышележащий протокол (1=ICMP, 6=TCP, 17=UDP)
- **Header Checksum (16 бит):** Контрольная сумма заголовка
- **Source/Destination Address (32 бита):** IP-адреса отправителя и получателя
- **Options (переменная):** Дополнительные опции
- **Padding:** Выравнивание до 32-битной границы

Опции

Опции используются для управления маршрутизацией, диагностикой и мониторингом, но не являются обязательными.

- Record Route (Запись маршрута)
- Strict Source and Record Route (SSRR) — Строгая маршрутизация от источника
- Loose Source and Record Route (LSRR) — Свободная маршрутизация от источника
- Internet Timestamp (Временные метки)

Фрагментация

Протокол IP работает на сетевом уровне

- Объединение сетей, построенных на основе разных технологий

Различия в сетях

- Максимальный размер передаваемых данных (MTU, Maximum Transmission Unit)
- Ethernet - 1500 байт
- Token Ring - 4464 байта

Фрагментация - разделение пакета на несколько частей (**фрагментов**) для передачи по сети с маленьким MTU

Флаги

Размер поля флаги - 3 бита

Значение полей

- Первый бит зарезервирован и не используется
- DF (Don’t Fragment) - не фрагментировать
- MF (More Fragments) - есть еще фрагменты

Смещение фрагмента

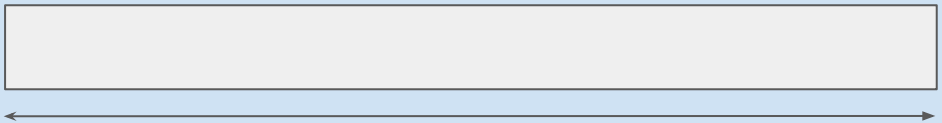
Смещение фрагмента в поле данных исходного пакета

- Измеряется в 8-байтовых блоках

Пример

- Исходный пакет 4000 байт (заголовок 20 байт, данные 3980 байт)
- MTU сети 1500 байт (заголовок 20 байт, данные 1480 байт)
- Три фрагмента данных: 0-1479, 1480-2959, 2960-3980
- Смещение фрагментов: 0, 185, 370

Фрагментация



4000 байт

Смещение фрагмента

Номер пакета

More fragments

81	0	1	

1500 байт

81	185	1	

1500 байт

81	370	0	

1040 байт

Флаг DF

Флаг DF (Don’t Fragment) запрещает фрагментация пакета

Если MTU сети меньше размера пакета

- Маршрутизатор отбрасывает пакет
- Получателю отправляется ICMP сообщение Тип 3, Код 4 (Destination Unreachable, Fragmentation required, and DF flag set)

Протокол ICMP

ICMP (Internet Control Message Protocol) - протокол межсетевых управляющих сообщений

Протокол IP передает данных без гарантии доставки

- В случае ошибки при передаче пакета никаких действий не предпринимается

Функция ICMP

- Оповещение об ошибках на сетевом уровне
- Тестирование работоспособности сети

Сообщение об ошибках ICMP не обязательно должны обрабатываться

Протокол ICMP - формат заголовка

	Bits 0–7	Bits 8–15	Bits 16–23	Bits 24–31
IP Header (20 bytes)	Version/IHL	Type of service	Length	
	Identification		*flags* and *offset*	
	Time To Live (TTL)	Protocol	Checksum	
	Source IP address			
	Destination IP address			
ICMP Header (8 bytes)	Type of message	Code	Checksum	
	Header Data			
ICMP Payload (*optional*)	Payload Data			

Протокол ICMP - формат заголовка

Type (8 бит): Тип ICMP-сообщения

- **0** - Echo Reply
- **8** - Echo Request
- **3** - Destination Unreachable
- **11** - Time Exceeded

Code (8 бит): Уточнение типа сообщения

- **Для Type=3 (Unreachable):**
  - 0 - Network unreachable
  - 1 - Host unreachable
  - 3 - Port unreachable

Checksum (16 бит): Контрольная сумма всего ICMP-сообщения

Data (переменная): Зависит от типа сообщения

- Для Echo Request/Reply: содержит Identifier, Sequence Number и произвольные данные
- Для ошибок: содержит заголовок исходного IP-пакета, вызвавшего ошибку

Типы ICMP - сообщений

Тип	Назначение сообщения
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

Коды ICMP - сообщений (для типа 3)

Тип	Назначение сообщения
0	Сеть недостижима
1	Узел недостижим
2	Протокол недостижим
3	Порт недостижим
4	Ошибка фрагментации
5	Ошибка в маршруте источника
6	Сеть назначения неизвестна
7	Узел назначения неизвестен
8	Узел-источника изолирован
9	Административный запрет

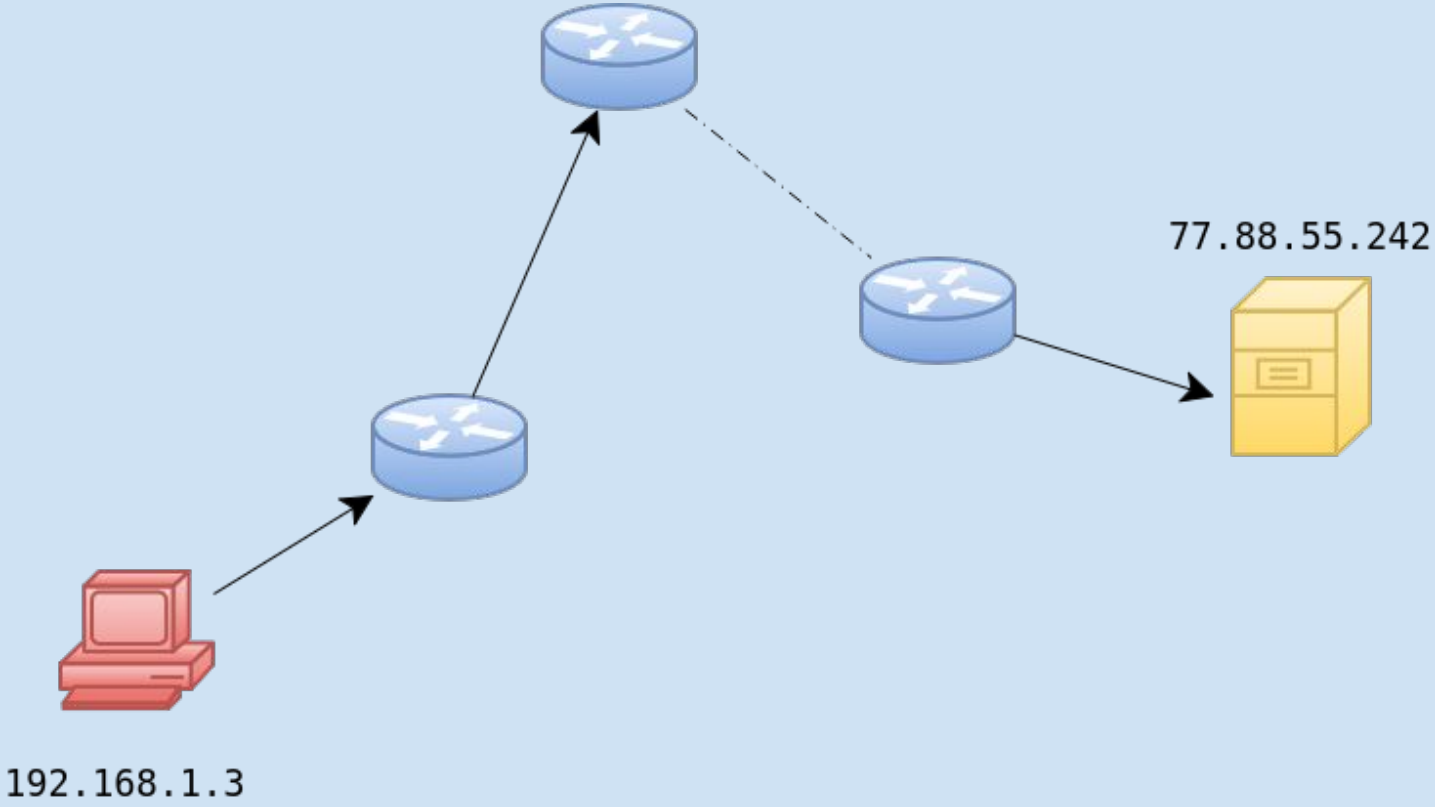
Утилита ping - проверка доступности хоста

Задача: Проверить, доступен ли удаленный узел в сети, и измерить время прохождения пакетов (RTT - Round Trip Time).

Принцип работы:

- Использует протокол **ICMP (Internet Control Message Protocol)**.
- Отправляет пакет **ICMP Эхо-запрос** (Тип = 8, Код = 0) на целевой IP-адрес.
- Ожидает ответ **ICMP Эхо-ответ** (Тип = 0, Код = 0) от целевого узла.

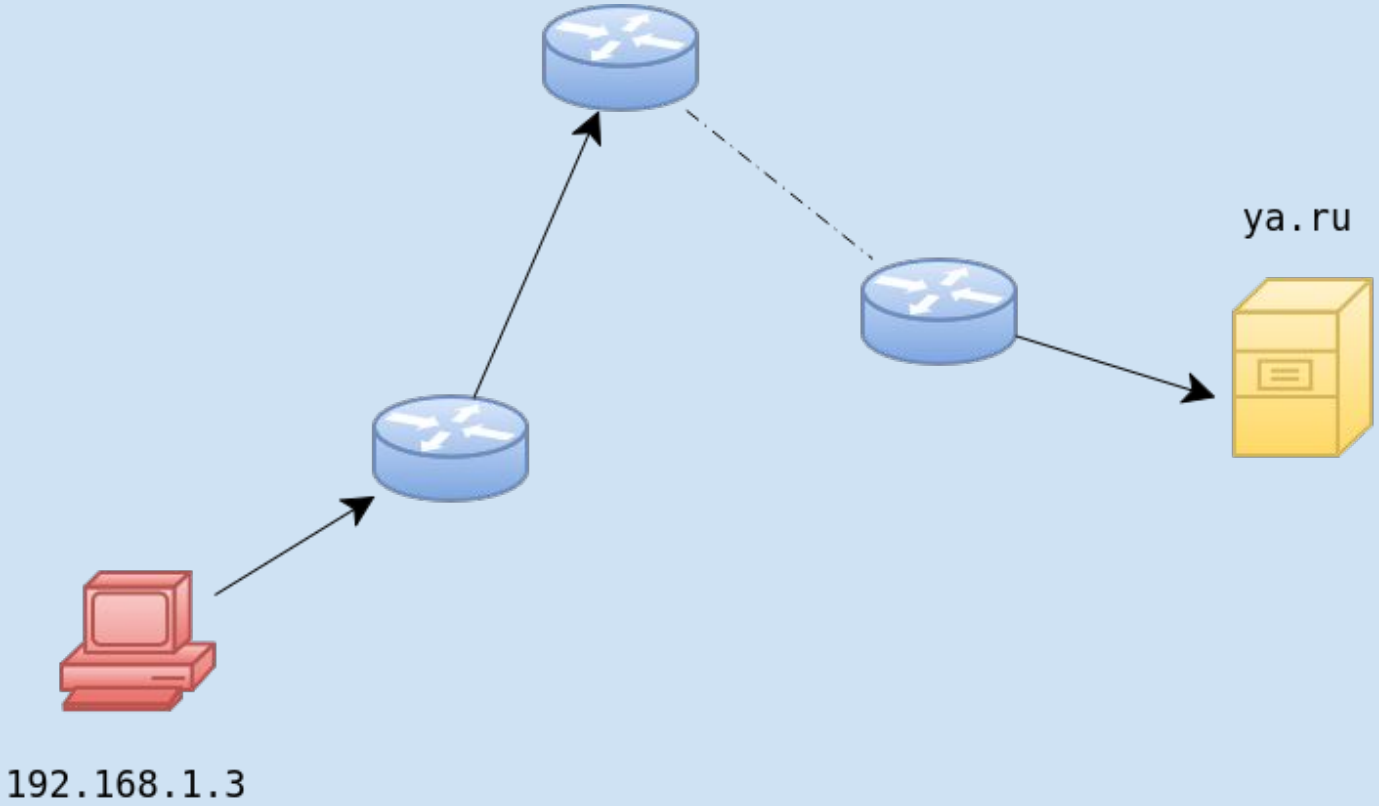
Демонстрация ping - по IP адресу



Демонстрация ping - по IP адресу

```
$ ping -c 5 77.88.55.242
PING 77.88.55.242 (77.88.55.242) 56(84) bytes of data.
64 bytes from 77.88.55.242: icmp_seq=1 ttl=53 time=35.2 ms
64 bytes from 77.88.55.242: icmp_seq=2 ttl=53 time=32.2 ms
64 bytes from 77.88.55.242: icmp_seq=3 ttl=53 time=32.2 ms
64 bytes from 77.88.55.242: icmp_seq=4 ttl=53 time=32.2 ms
64 bytes from 77.88.55.242: icmp_seq=5 ttl=53 time=32.3 ms

--- 77.88.55.242 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 32.157/32.795/35.186/1.195 ms
```

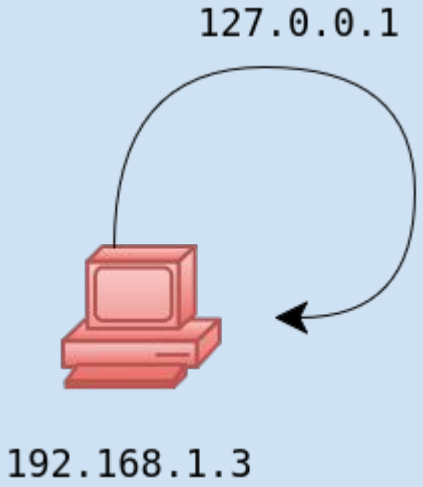
Демонстрация ping - по доменному имени



Демонстрация ping - по доменному имени

```
$ ping -c 5 ya.ru
PING ya.ru (77.88.55.242) 56(84) bytes of data.
64 bytes from ya.ru (77.88.55.242): icmp_seq=1 ttl=53 time=32.8 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=2 ttl=53 time=32.0 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=3 ttl=53 time=32.7 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=4 ttl=53 time=32.3 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=5 ttl=53 time=32.2 ms

--- ya.ru ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 31.971/32.384/32.843/0.322 ms
```

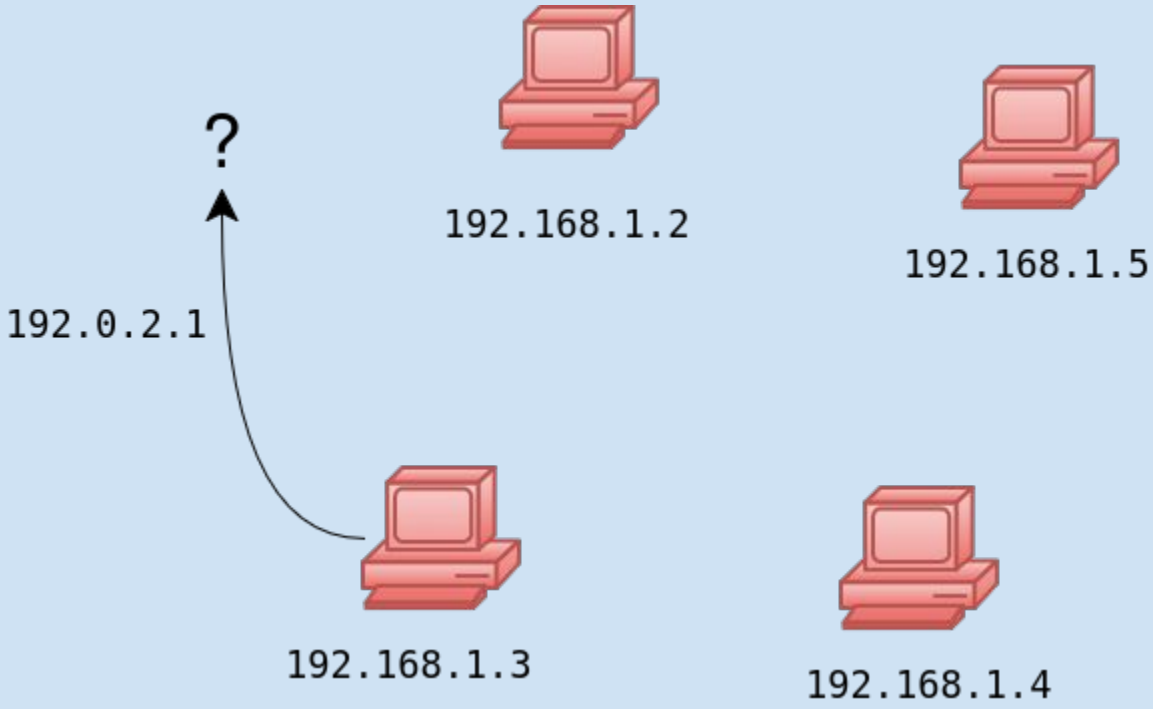
Демонстрация ping - loopback



Демонстрация ping - loopback

```
$ ping -c 3 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.025 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.053 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.052 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2044ms
rtt min/avg/max/mdev = 0.025/0.043/0.053/0.013 ms
```

Демонстрация ping - неудачный пинг



Демонстрация ping - неудачный пинг

```
$ ping -c 4 192.0.2.1
PING 192.0.2.1 (192.0.2.1) 56(84) bytes of data.

--- 192.0.2.1 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3109ms
```

Утилита ping - проверка доступности хоста

Многие хосты в интернете не отвечают на ICMP Echo Request по соображениям безопасности или политики.

Причины, по которым `ping` может не работать, даже если хост жив:

1. **Файрволы (Межсетевые экраны):**
   - Администраторы часто блокируют входящие ICMP-пакеты на границе сети, чтобы скрыть внутренние узлы от простого сканирования.
   - **Пример:** Корпоративный файрвол может разрешать только веб-трафик (порт 80/443), а весь ICMP — запрещать.
2. **Настройки на самом хосте:**
   - **Windows:** В брандмауэре Windows есть отдельная настройка "Разрешить проверку связи". Если она выключена, компьютер не будет отвечать на `ping`.
   - **Linux:** С помощью `sysctl` (`net.ipv4.icmp_echo_ignore_all`) можно заставить ядро игнорировать Echo Request.
3. **Проблемы с обратным маршрутом:** Даже если пакет дошел до цели, ответу может не хватить TTL или может быть заблокирован на пути назад.

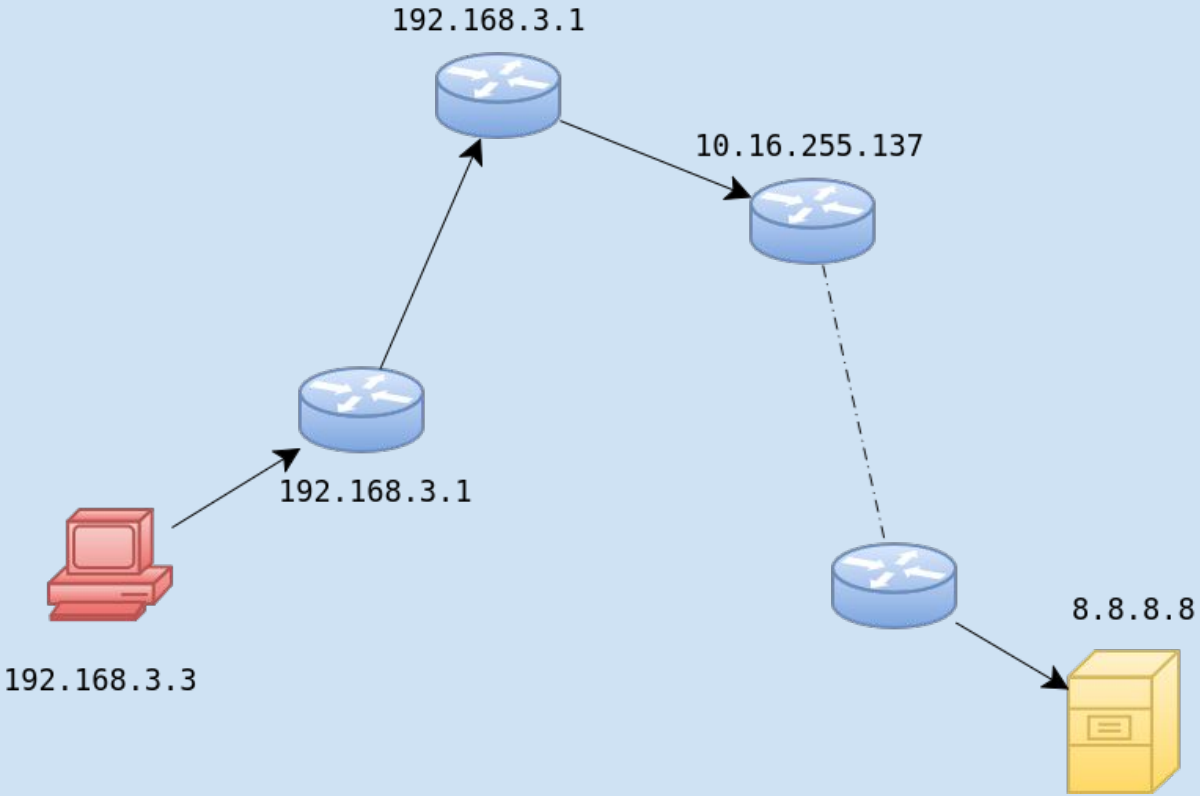
Утилита traceroute

Назначение:
`traceroute (tracert` на **Windows**) — это утилита для определения пути, который пакеты проходят от вашего компьютера до целевого узла в сети. Она показывает каждый промежуточный маршрутизатор на пути.

Принцип работы (основан на поле TTL):

1. `traceroute` отправляет серию пакетов (обычно UDP, ICMP или TCP) к целевому хосту.
2. **Первая серия** пакетов отправляется с **TTL=1**. Первый же маршрутизатор уменьшает TTL до 0, отбрасывает пакет и отправляет обратно сообщение **"Время истекло”** (Тип = 11, Код = 0).
3. `traceroute` фиксирует адрес этого маршрутизатора и время отклика.
4. **Следующая серия** пакетов отправляется с **TTL=2**. Они проходят первый маршрутизатор (TTL уменьшается до 1), доходят до второго, где TTL обнуляется, и он отправляет "Время истекло”.
5. Процесс повторяется с увеличением TTL до тех пор, пока пакет не достигнет цели.

Демонстрация traceroute - базовое использование



Демонстрация traceroute - базовое использование

```
$ traceroute -n 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  192.168.3.1  4.976 ms  4.901 ms  5.358 ms
 2  10.16.255.137  6.091 ms  6.069 ms  6.049 ms
 3  10.64.241.65  10.251 ms 10.16.249.61  6.919 ms 10.16.248.1  11.034 ms
 4  10.16.248.170  11.886 ms 10.16.248.150  6.304 ms 10.16.248.218  5.947 ms
 5  10.16.248.189  6.276 ms 10.16.248.134  6.266 ms 10.16.248.253  6.256 ms
 6  188.170.164.32  18.947 ms  16.526 ms *
 7  192.178.241.181  16.218 ms * 192.178.241.119  17.601 ms
 8  192.178.241.234  21.769 ms * *
 9  * 142.251.237.154  35.099 ms 142.251.49.78  30.975 ms
10  83.169.204.119  16.939 ms  17.826 ms 72.14.232.76  32.535 ms
11  216.239.49.113  31.979 ms 172.253.79.237  33.885 ms 216.239.62.13  31.406 ms
12  * * *
13  192.178.241.146  15.657 ms 192.178.243.132  16.090 ms *
14  192.178.240.239  31.242 ms * *
15  * 216.239.48.224  36.964 ms 108.170.232.251  30.573 ms
16  108.170.233.163  28.881 ms 108.170.233.161  36.066 ms *
17  * * *
18  * * *
19  * * *
20  * * *
21  8.8.8.8  32.160 ms * *
```

Демонстрация traceroute - проблемы на маршруте

```
$ traceroute 10.255.255.255
traceroute to 10.255.255.255 (10.255.255.255), 30 hops max, 60 byte packets
 1  192.168.3.1  6.153 ms  7.963 ms  7.610 ms
 2  10.16.255.148 (10.16.255.148)  9.263 ms  9.257 ms  9.250 ms
 3  10.16.255.148 (10.16.255.148)  9.243 ms !X  9.237 ms !X  9.231 ms !X
```

Утилита `ipconfig`

Назначение и возможности

`ipconfig` (Internet Protocol Configuration) — встроенная сетевая утилита Windows для просмотра и управления базовой сетевой конфигурацией.

Основные функции

- Просмотр IP-адресов, масок подсети, шлюзов
- Отображение MAC-адресов сетевых адаптеров
- Просмотр DNS-серверов и DHCP-статуса
- Обновление и освобождение DHCP-аренды
- Очистка DNS-кэша

Утилита ipconfig

```
Настройка протокола IP для Windows

Адаптер Ethernet Ethernet0:

   Состояние среды. . . . . . . . : Работает
   DNS-суффикс подключения . . . . . : home.local
   Описание. . . . . . . . . . . . : Realtek PCIe GbE Family Controller
   Физический адрес. . . . . . . . : 08-00-27-FC-12-34
   DHCP включен. . . . . . . . . . : Да
   Автонастройка включена. . . . . : Да
   IPv4-адрес. . . . . . . . . . . : 192.168.1.100
   Маска подсети . . . . . . . . . : 255.255.255.0
   Аренда получена. . . . . . . . . : 11 ноября 2025 г. 10:30:45
   Срок аренды истекает. . . . . . : 12 ноября 2025 г. 10:30:45
   Основной шлюз. . . . . . . . . . : 192.168.1.1
   DNS-серверы. . . . . . . . . . . : 8.8.8.8
                                      8.8.4.4

Адаптер беспроводной сети Беспроводная сеть0:

   Состояние среды. . . . . . . . : Медиа отключена
   Описание. . . . . . . . . . . . : Intel(R) Wi-Fi 6 AX200
   Физический адрес. . . . . . . . : 52-54-00-12-34-56
   DHCP включен. . . . . . . . . . : Да
   Автонастройка включена. . . . . : Да
```

Другие сетевые утилиты

Утилита `route` - управление маршрутизацией

Назначение: Просмотр и управление таблицей маршрутизации

Основные команды:

route print

route add 192.168.2.0 mask 255.255.255.0 192.168.1.1

route delete 192.168.2.0

Утилита `arp` - управление ARP-таблицей

Назначение: Просмотр соответствия IP-адресов MAC-адресам в локальной сети

Основные команды:

arp -a

arp -d *

arp -s 192.168.1.99 aa-bb-cc-dd-ee-ff

Утилита `tcpdump`

Назначение

Перехват и анализ всего сетевого трафика, проходящего через интерфейсы

Ключевые возможности

- **Фильтрация** по протоколам, IP-адресам, портам
- **Глубокий анализ** содержимого пакетов
- **Сохранение** трафика в файл для последующего анализа

Области применения

- Диагностика сетевых проблем
- Мониторинг безопасности
- Отладка сетевых приложений

Протокол ARP

ARP (Address Resolution Protocol) - протокол разрешения адресов

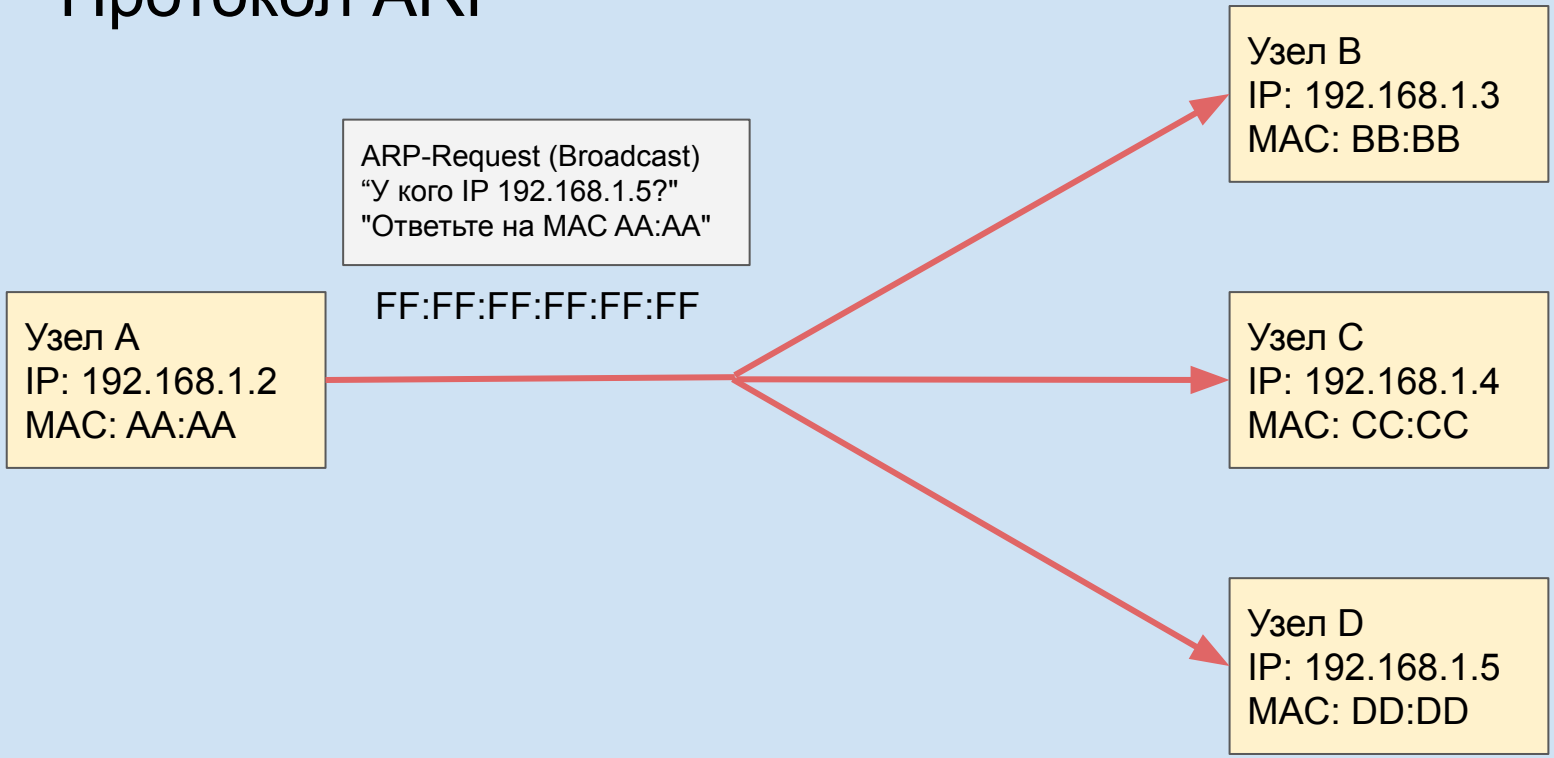
- Сетевое взаимодействие выполняется с использованием IP-адресов
- Данные передаются с помощью технологий канального уровня
- Необходимо средство определения MAC-адреса компьютера по его IP-адресу

Протокол ARP

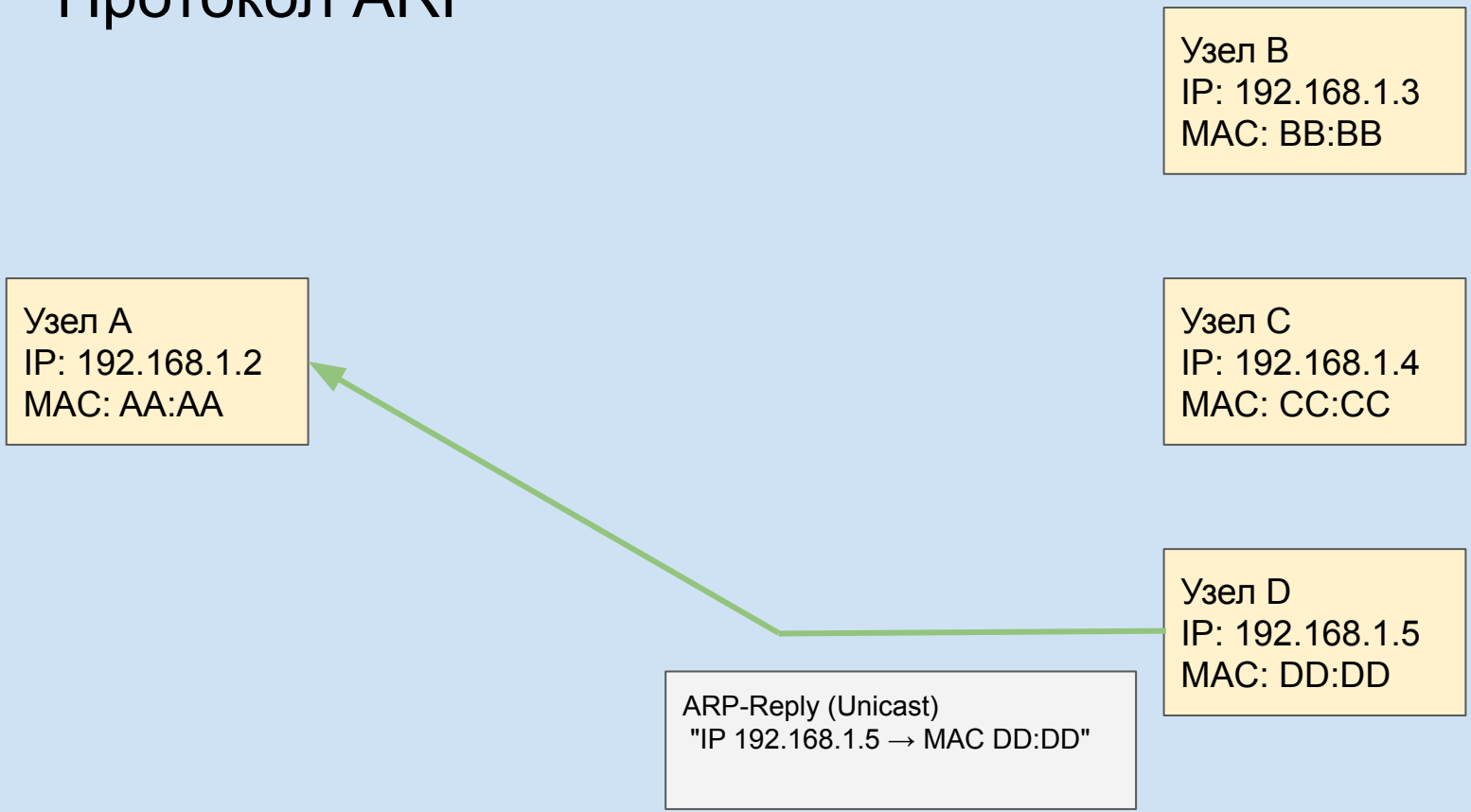
Протокол ARP позволяет автоматически определить MAC-адрес компьютера по его IP-адресу

ARP работает в режиме запрос-ответ

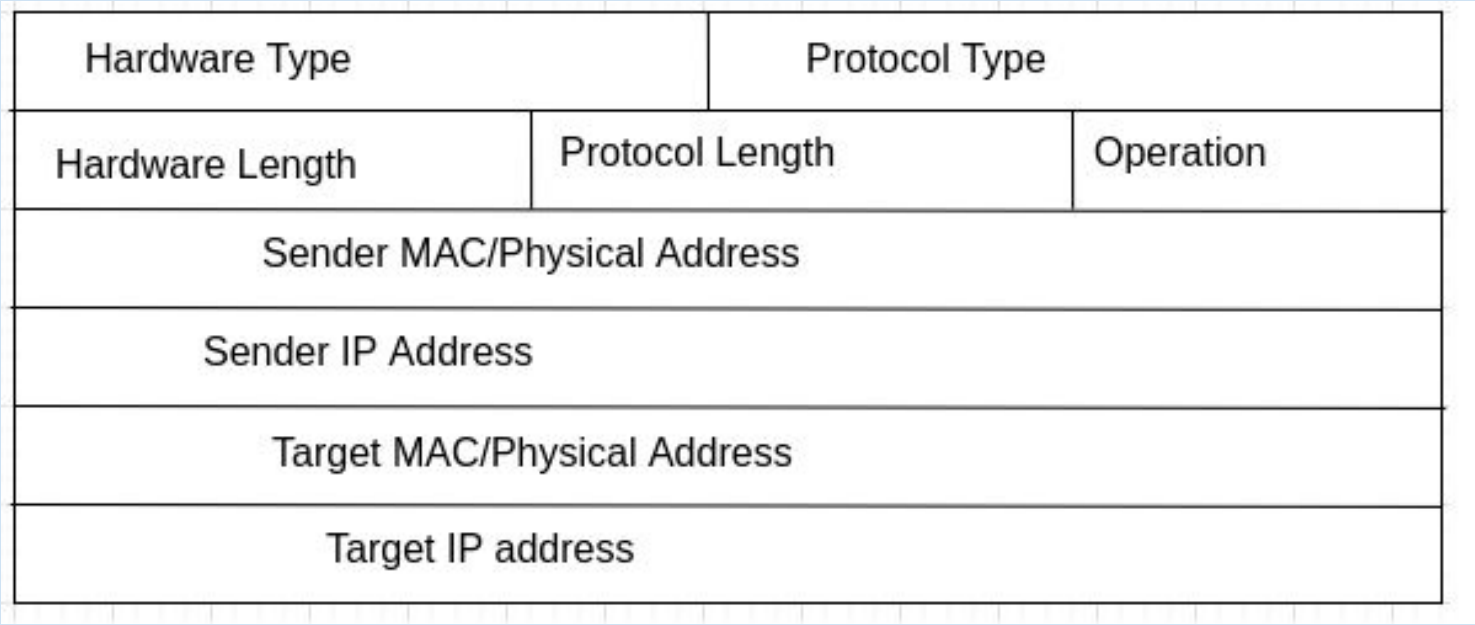
Протокол ARP



Протокол ARP



Формат ARP-заголовка



Формат ARP-заголовка

- **Hardware Type (16 бит):** Тип аппаратного адреса
  - **1** - Ethernet (наиболее распространенный)
  - **17** - HDLC
- **Protocol Type (16 бит):** Тип протокольного адреса
  - **2048** - IPv4
- **Hardware Addr Length (8 бит):** Длина MAC-адреса в байтах
  - **6** - для Ethernet
- **Protocol Addr Length (8 бит):** Длина IP-адреса в байтах
  - **4** - для IPv4
- **Operation (16 бит):** Тип операции
  - **1** - ARP Request
  - **2** - ARP Reply
- **Sender Hardware Addr (6 байт):** MAC-адрес отправителя
- **Sender Protocol Addr (4 байта):** IP-адрес отправителя
- **Target Hardware Addr (6 байт):** MAC-адрес цели
  - В запросе: обычно 00:00:00:00:00:00
  - В ответе: заполняется MAC-адресом целевого узла
- **Target Protocol Addr (4 байта):** IP-адрес цели

ARP-таблица

- Кэш соответствий IP и MAC на каждом узле.
- **Пример команды для просмотра:**
  - **Windows:** `arp -a`
  - **Linux/macOS:** `ip neigh show`
- Записи имеют время жизни.

IP-адрес	Физический адрес	Тип записи
192.168.1.1	aa-bb-cc-dd-ee-ff	динамическая
192.168.1.3	11-22-33-44-55-66	динамическая
192.168.0.1	ab-cd-ef-12-34-56	статическая

Оптимизации ARP

Извлечение информации из ARP-запроса

- Запросы отправляются на широковещательный адрес (его получают все устройства в сети)
- Все компьютеры извлекают и запоминают IP и MAC-адреса отправителя запроса

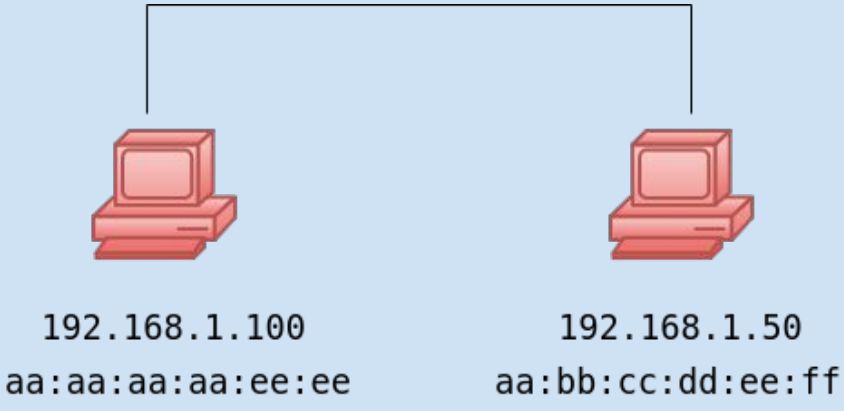
Добровольный ARP-запрос

- Запрос собственного IP-адреса
- Используется при назначении нового IP-адреса для быстрого оповещения всех устройств в сети
- Предотвращение использования одинаковых IP-адресов в сети

Наблюдение за ARP

Подготовка

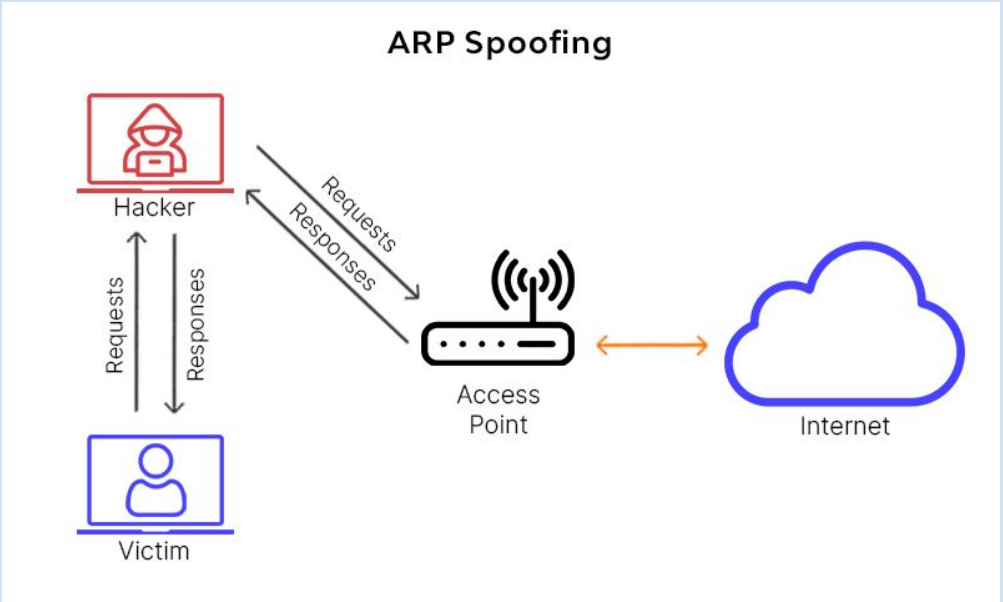
1. Очистить ARP-кэш: `arp -d *`
2. Запустить `ping` на адрес в локальной сети




```
$ tcpdump -i any -n arp

12:34:56.789123 ARP, Request who-has 192.168.1.50 tell 192.168.1.100, length 28
12:34:56.790456 ARP, Reply 192.168.1.50 is-at aa:bb:cc:dd:ee:ff, length 28
```

Уязвимость ARP Spoofing

- **Уязвимость:** Протокол не аутентифицирует отправителей ARP-сообщений.
- **Суть атаки:** Злоумышленник рассылает поддельные ARP-ответы, связывая свой MAC с IP-адресом другого узла (например, шлюза по умолчанию).
- **Результат:** Атака "Человек посередине" (Man-in-the-Middle). Трафик перенаправляется через компьютер злоумышленника. Либо может не передавать трафик вообще (отказ в обслуживании)



Методы защиты от ARP Spoofing

- **Статические ARP-записи:**
  - Ручное добавление критических записей (например, для шлюза) в ARP-таблицу.
  - **Недостаток:** Сложность администрирования, непрактично в больших сетях.
- **DHCP Snooping:**
  - Функция управляемых коммутаторов.
  - Свитч строит таблицу доверенных привязок IP-MAC-Порт, отслеживая DHCP-трафик.
- **Dynamic ARP Inspection (DAI):**
  - Работает вместе с DHCP Snooping.
  - Свитч проверяет все входящие ARP-пакеты на соответствие доверенной таблице.
  - Поддельные ARP-пакеты блокируются.

Протокол RARP (Reverse ARP) и его эволюция

RARP

- **Задача:** Определение своего IP-адреса по известному MAC-адресу.
- **Применение:** Загрузка бездисковых рабочих станций.
- **Недостатки:**
  - Работа на канальном уровне (L2) -> не работает через маршрутизаторы.
  - Передает только IP-адрес (без маски, шлюза, DNS).
  - Требует ручной настройки таблицы MAC -> IP на сервере.

DHCP

Преимущества:

- Работа на прикладном уровне (L7): может работать через relay-агенты.
- Передает полный набор параметров (IP, маска, шлюз, DNS, время аренды).
- Поддержка динамического выделения адресов и механизма аренды.
- Надежность и контроль: Механизм аренды (lease time) позволяет автоматически освобождать неиспользуемые адреса.

Протокол DHCP

DHCP (Dynamic Host Configuration Protocol) - протокол динамической конфигурации хостов

Для работы в сети компьютеру нужен IP-адрес

Методы назначения IP-адресов

- Вручную
- Автоматически

Протокол DHCP

- Позволяет назначать IP-адресе компьютерам в сети автоматически
- Требует создания инфраструктуры (DHCP сервер)
- IP-адреса компьютеров могут меняться

Протокол DHCP

Клиент DHCP

- Компьютер, который получает IP-адрес автоматически

Сервер DHCP

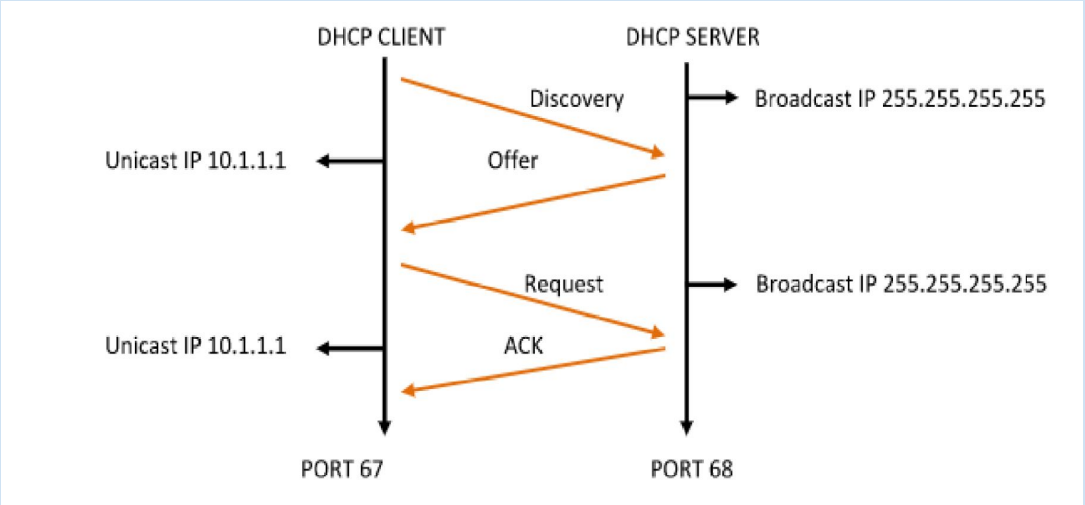
- Компьютер, который обеспечивает назначение IP-адресов
- Ведет таблицу выделенных IP-адресов, чтобы избежать дублирования

Клиент и сервер обмениваются сообщениями DHCP в режиме запрос-ответ

Алгоритм работы DHCP: Процесс DORA

Основные этапы

1. **DISCOVER:** Клиент отправляет широковещательный запрос на поиск DHCP-сервера.
2. **OFFER:** Сервер(ы) отправляет широковещательное предложение с параметрами конфигурации.
3. **REQUEST:** Клиент отправляет широковещательный запрос на выбранное предложение.
4. **ACKNOWLEDGE:** Выбранный сервер подтверждает аренду и передает окончательные параметры.



Протокол DHCP

Сообщение HDCP	Назначение
DISCOVER	Поиск DHCP-сервера
OFFER	Предложение IP-адреса DHCP сервером клиенту
REQUEST	Запрос IP-адреса DHCP-клиентом
ACK	Подтверждение назначения IP-адреса DHCP-клиенту
NACK	Запрет использования запрошенного DHCP-клиентом IP-адреса
RELEASE	Освобождение IP-адреса
INFORM	Запрос и передача дополнительной конфигурационной информации

Назначение адресов в DHCP

Фиксированный

- Выделенный IP-адрес для каждого MAC-адреса

Динамический

- Выделение компьютеру любого IP-адреса из пула адресов

Пул адресов

- Список (диапазон) IP-адресов, которые назначает DHCP-сервер
- DHCP-сервер следит за уникальностью распределения адресов

Механизм обновления аренды (Renewal)

- IP-адрес выдается клиенту на ограниченное время (Lease Time).
- После окончания времени аренды IP-адрес освобождается
- DHCP-клиент может продлить использование IP-адреса

Процесс обновления:

- **T1 (обычно 50% Lease Time):** Клиент пытается обновить аренду у исходного сервера, отправляя **DHCP Request (unicast)**.
- **T2 (обычно 87.5% Lease Time):** Если сервер не ответил, клиент пытается найти любой доступный DHCP-сервер, отправляя **DHCP Request (broadcast)**.

Результат:

- При успехе аренда продлевается.
- При неудаче по истечении Lease Time адрес освобождается.

Прекращение использования адреса

Окончание использование IP-адреса

- Сообщение DHCP Release
- Сервер может назначать освободившийся адрес другому клиенту

Сообщение DHCP Release автоматически отправляются современными ОС при корректном выключении

Отсутствие сообщения DHCP Release

- До окончания срока аренды DHCP-сервер считает IP-адрес занятым
- После окончания аренды адрес освобождается

Конфигурационная информация

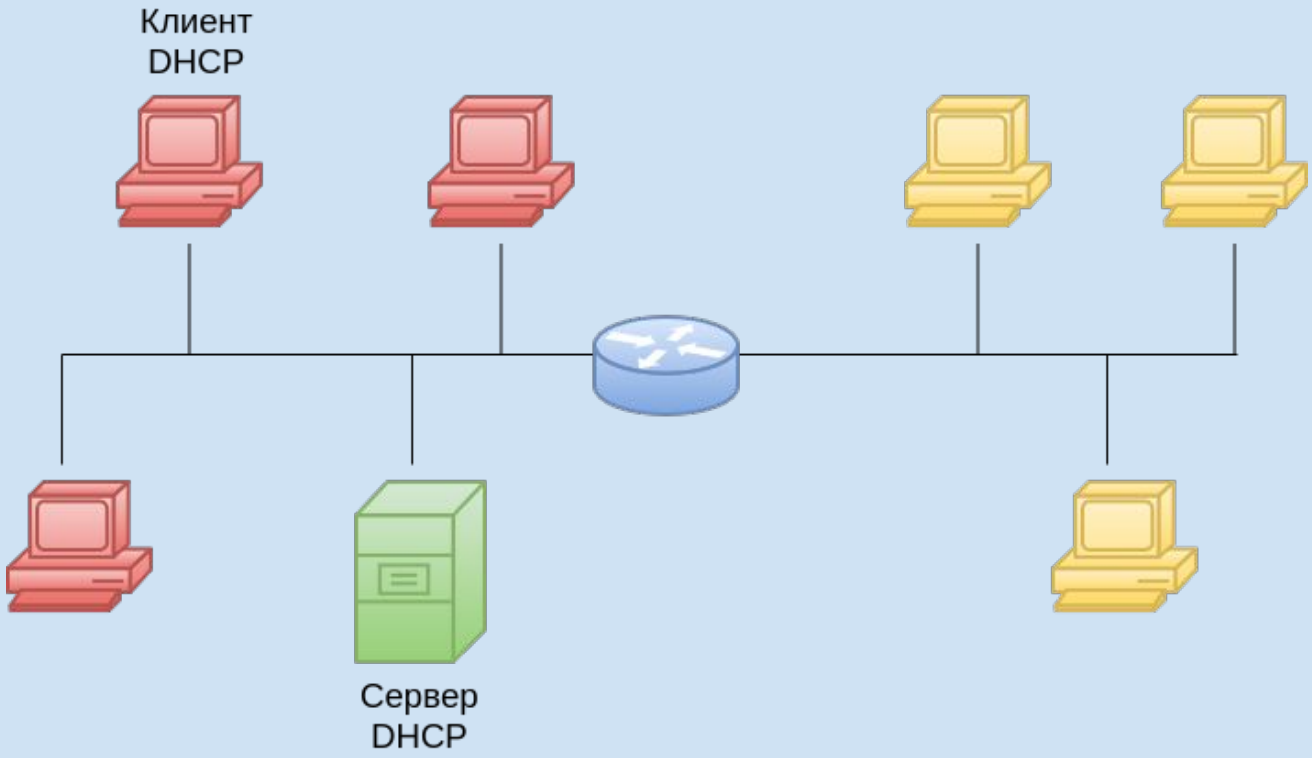
Для работы в сети нужен не только IP-адрес

Конфигурационные параметры передаются DHCP в качестве **опций**

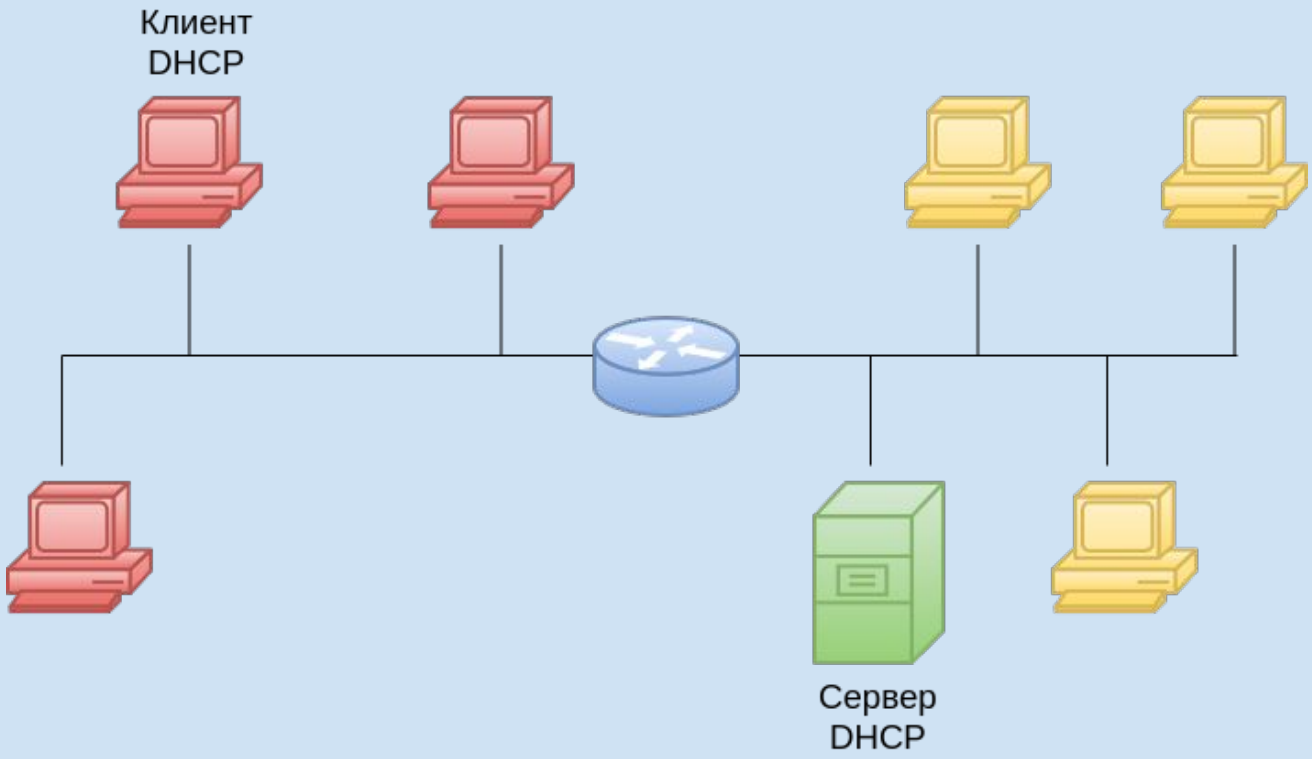
DHCP предоставляет дополнительно:

- Маску подсети
- Маршрутизатор по умолчанию
- Адреса DNS-серверов
- Адреса серверов времени
- и другую информацию

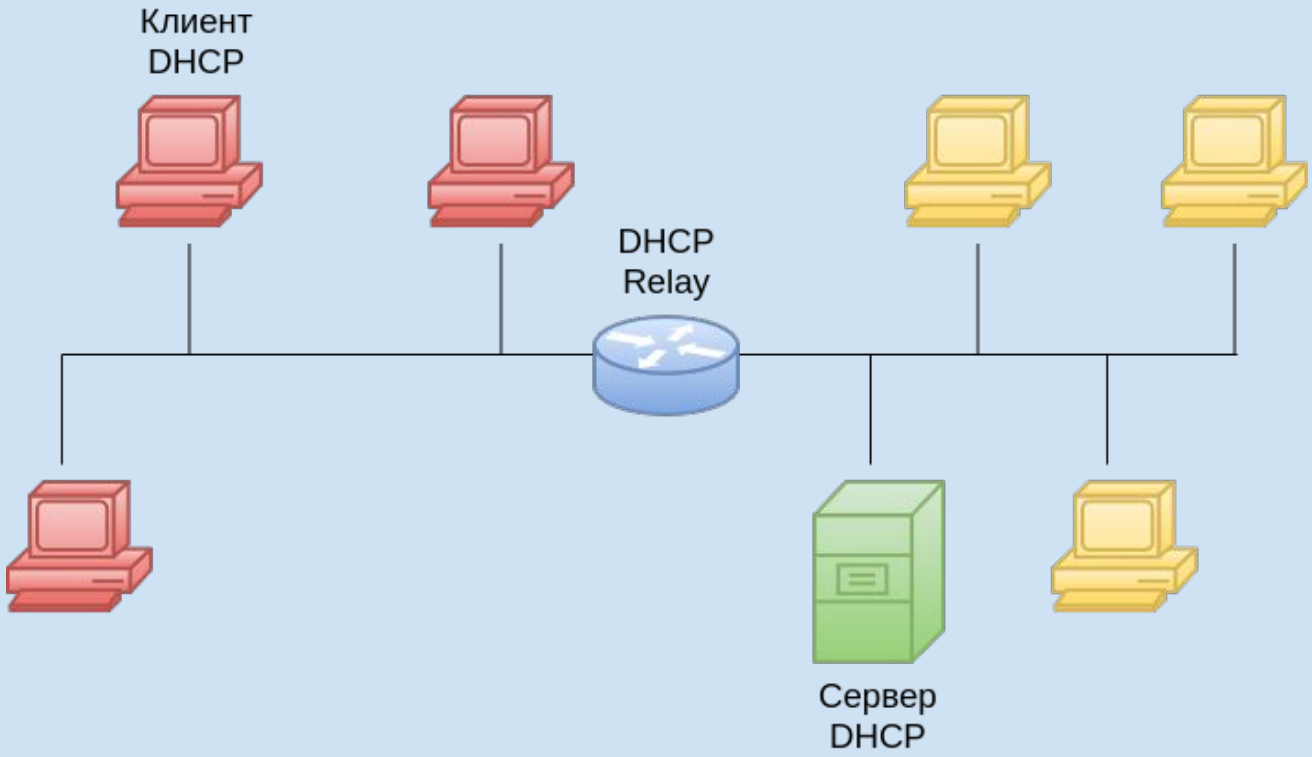
Поиск DHCP-сервера в сети



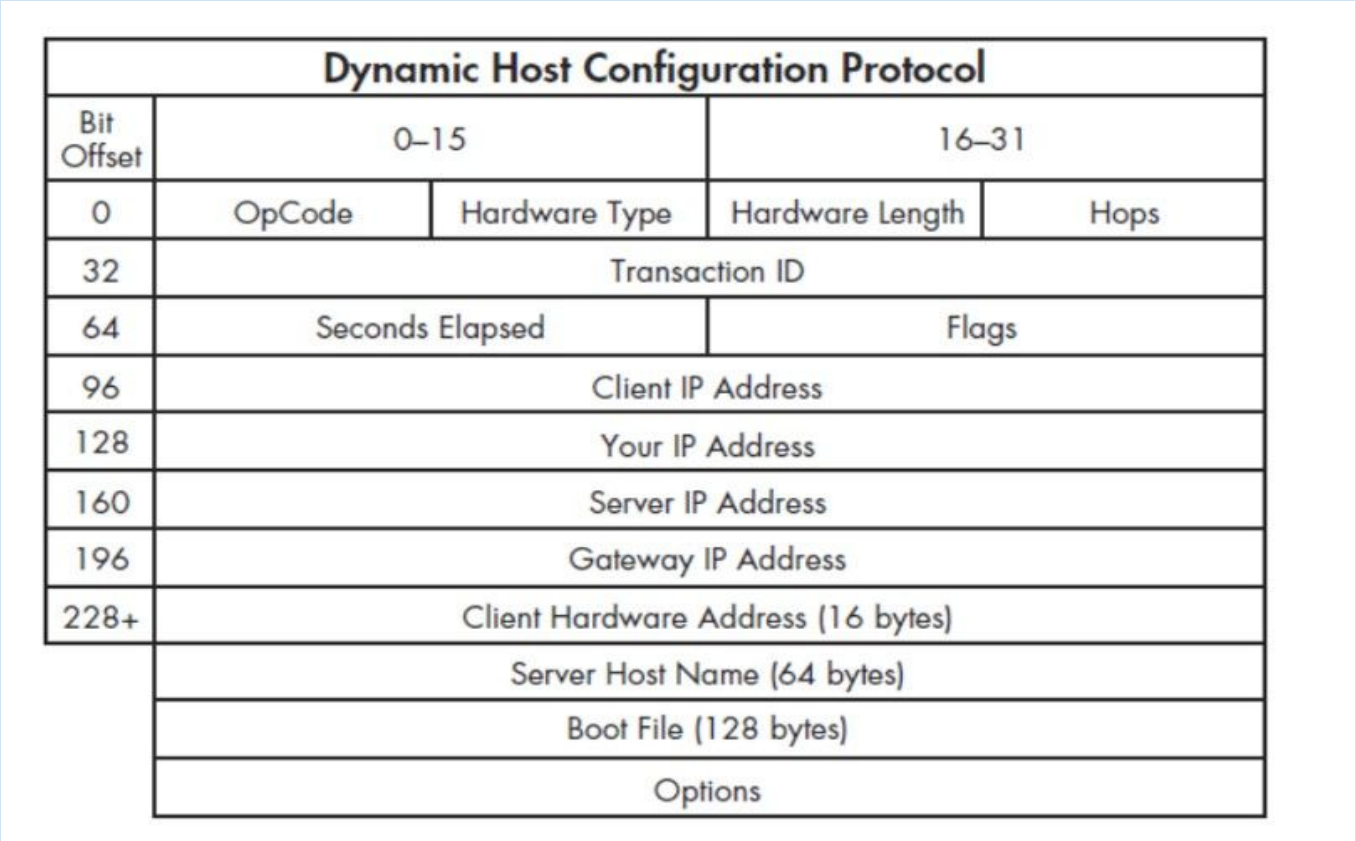
Поиск DHCP-сервера в сети



Поиск DHCP-сервера в сети



Формат DHCP-сообщений



Dynamic Host Configuration Protocol				
Bit Offset	0–15		16–31	
0	OpCode	Hardware Type	Hardware Length	Hops
32	Transaction ID			
64	Seconds Elapsed		Flags	
96	Client IP Address			
128	Your IP Address			
160	Server IP Address			
196	Gateway IP Address			
228+	Client Hardware Address (16 bytes)			
	Server Host Name (64 bytes)			
	Boot File (128 bytes)			
	Options			

Формат DHCP-сообщений

- **OpCode (1 байт):** Тип сообщения (1 = запрос, 2 = ответ).
- **Hardware Type (16 бит):** Тип аппаратного адреса (эквивалент поля в протоколе ARP
  - 1 - Ethernet (наиболее распространенный)
- **Hardware Length (16 бит):** Длина аппаратного адреса в байтах
  - 6 - для Ethernet
- **Hops (1 байт):** Количество ретрансляторов
- **Transaction ID (4 байта):** Идентификатор транзакции. Случайное число, генерируется клиентом для связи запроса и ответа.
- **Seconds elapsed (2 байта):** Время в секундах с момента начала процесса получения/обновления адреса.
- **Flags (2 байта):** Флаги**.** Старший бит: 0 = Unicast, 1 = Broadcast (если клиент не умеет работать с Unicast до получения IP).

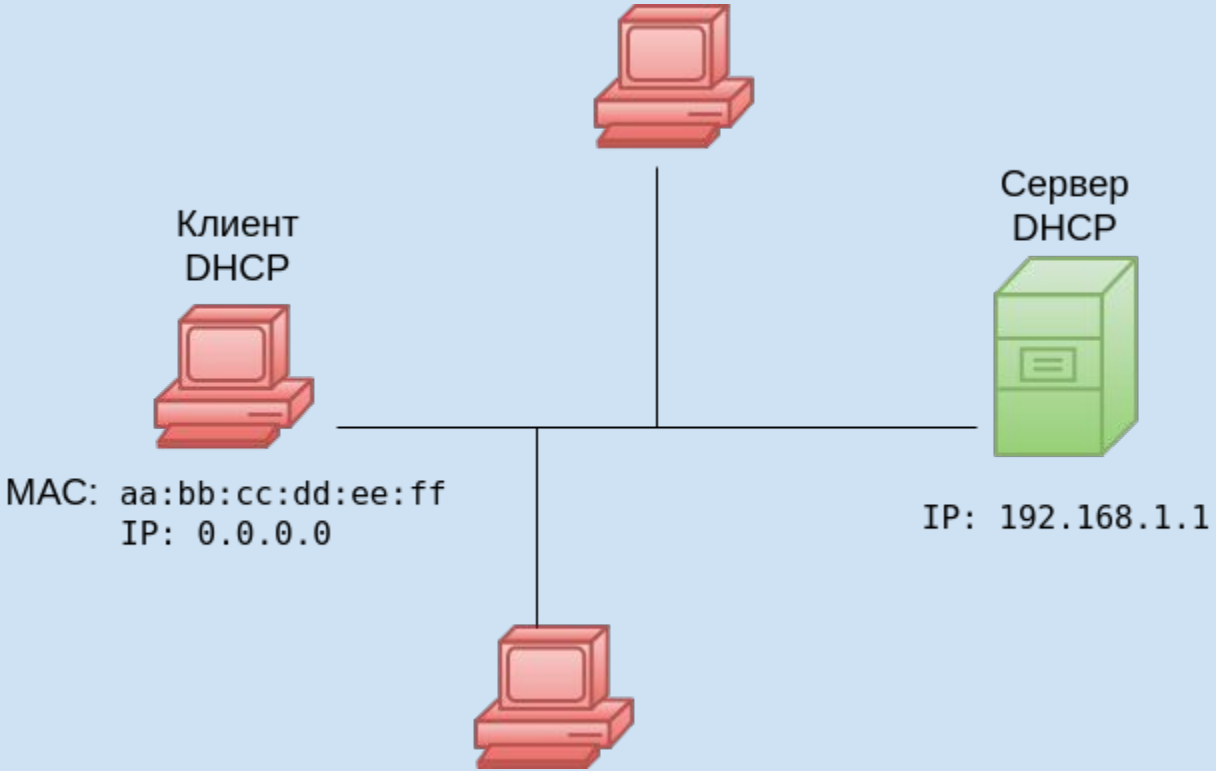
Формат DHCP-сообщений

- **Client IP Address (4 байта):** IP-адрес клиента. Заполняется клиентом, если он его уже знает.
- **Your IP Address (4 байта):** Ваш IP-адрес (предлагаемый/выданный сервером).
- **Server IP Address (4 байта):** IP-адрес сервера.
- **Gateway IP Address (4 байта):** IP-адрес DHCP-ретранслятора. Заполняется ретранслятором, если клиент и сервер в разных подсетях.
- **Client Hardware Address (4 байта):** Аппаратный (MAC) адрес клиента. Для Ethernet первые 6 байт - MAC, остальные 10 - нули.
- **Server Host Name (64 байта):** Необязательное имя сервера в виде строки.
- **Boot File (128 байта):** Необязательное имя файла для бездисковой загрузки.
- **options (переменной длины):** Опции. Формат [Код][Длина][Данные]. Здесь содержится вся специфичная для DHCP информация

Ключевые опции DHCP

Код	Имя	Размер	Описание
53	**DHCP Message Type**	1	**Тип DHCP-сообщения:** 1 = Discover, 2 = Offer, 3 = Request, 5 = ACK, 6 = NAK
1	**Subnet Mask**	4	Маска подсети.
3	**Router**	4+	Адрес(а) маршрутизатора по умолчанию (шлюза).
6	**Domain Name Server**	4+	Адреса DNS-серверов.
51	**IP Address Lease Time**	4	Время аренды IP-адреса в секундах.

Наблюдение за DHCP (DORA)



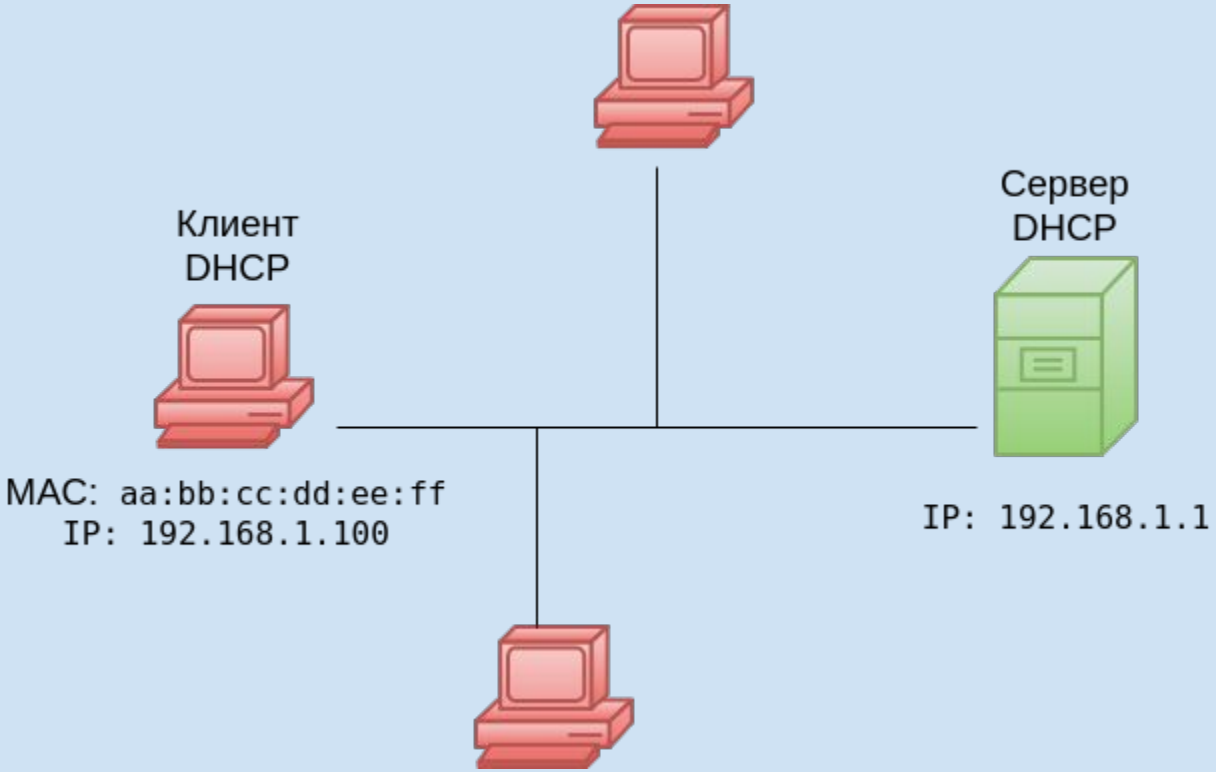
Наблюдение за DHCP (DORA)

Подготовка

1. Освободить текущий IP: `dhclient -r`

```
$ tcpdump -i any -n -vvv port 67 or port 68

14:20:30.123456 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request from
aa:bb:cc:dd:ee:ff
    DHCP-Message Option 53: Discover

14:20:30.134567 IP 192.168.1.1.67 > 192.168.1.100.68: BOOTP/DHCP, Reply
    DHCP-Message Option 53: Offer
    Your-IP 192.168.1.100

14:20:30.145678 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request
    DHCP-Message Option 53: Request

14:20:30.156789 IP 192.168.1.1.67 > 192.168.1.100.68: BOOTP/DHCP, Reply
    DHCP-Message Option 53: ACK
```

Наблюдение за DHCP (DORA)



Уязвимости DHCP

Проблема: Протокол DHCP изначально проектировался для удобства, а не безопасности. Он доверяет сообщениям в локальном сегменте сети, что создает несколько векторов атак.

Основные уязвимости:

- **Отсутствие аутентификации** — DHCP-сервер не проверяет, легитимен ли клиент, и клиент не проверяет, легитимен ли сервер.
- **Широковещательная природа** — злоумышленник в одном L2-сегменте может видеть весь DHCP-трафик.
- **Статус "первый пришел — первый обслужен"** — кто первый ответил на DHCP Discover, того клиент и выбирает.

DHCP Starvation

Суть атаки

- Злоумышленник истощает пул доступных IP-адресов на легитимном DHCP-сервере.

Как работает

1. Атакующий с помощью инструментов массово отправляет DHCP-запросы.
2. Каждый запрос использует случайный MAC-адрес отправителя.
3. Сервер честно выделяет IP-адрес на каждый запрос.
4. Весь пул адресов быстро заканчивается.

Результат

- Новые легитимные устройства не могут получить IP-адрес.
- Сервис отказан (DoS) для обычных пользователей.

Rogue DHCP Server

Суть атаки

- Злоумышленник развертывает в сети свой собственный DHCP-сервер, который выдает клиентам ложные настройки.

Как работает

1. Атакующий запускает DHCP-сервер с вредоносными настройками.
2. Когда легитимный клиент отправляет DHCP Discover, атакующий **отвечает быстрее** легитимного сервера (DHCP Offer).
3. Клиент принимает предложение атакующего.

Выдаваемые ложные настройки

- **Свой IP как шлюз по умолчанию** — весь трафик пользователя идет через компьютер злоумышленника (Man-in-the-Middle).
- **Подконтрольные DNS-серверы** — перенаправление на фишинговые сайты или блокировка ресурсов.

Результат

- Полный контроль над сетевым трафиком жертвы, кража учетных данных, перенаправление на мошеннические сайты.

Комбинированная атака — Starvation + Rogue Server

Используются две предыдущие атаки в комбинации:

1. **DHCP Starvation.** Атакующий истощает пул адресов на легитимном сервере с помощью DHCP Starvation.
2. **Rogue Server.** Когда легитимный сервер уже не может отвечать, атакующий предлагает свои услуги как "единственно доступный" DHCP-сервер.

Результат

- Клиенты, не сумевшие получить адрес у легитимного сервера, с радостью принимают предложение атакующего, так как альтернатив нет.
- Повышается успешность атаки Rogue DHCP Server, так как легитимный сервер "выведен из строя".

Защита от DHCP-атак

1. DHCP Snooping (Функция на управляемых сетевых коммутаторах)

- Коммутатор разделяет порты на **доверенные** (только для серверов) и **недоверенные** (для клиентов)
- Блокирует DHCP-ответы от неавторизованных серверов
- Невозможно подключить нелегитимный DHCP-сервер

2. Dynamic ARP Inspection (DAI)

- Коммутатор с включенным DHCP Snooping автоматически строит таблицу соответствий **`IP -> MAC -> Порт`**.
- DAI проверяет все входящие ARP-ответы.
- Если пришел ARP-ответ, где заявлено, что IP **`X`** имеет MAC **`Y`**, но в таблице от DHCP Snooping этот IP X привязан к другому MAC и порту — пакет блокируется.

3. Port Security (Безопасность портов)

- Ограничивает количество MAC-адресов на порту
- Препятствует DHCP Starvation-атакам с подменой MAC-адресов
- Нельзя исчерпать пул адресов с одного порта

Протокол NAT

NAT (Network Address Translation) - трансляция сетевых адресов

Технология преобразования IP-адресов внутренней сети в IP-адреса внешней сети

Цель создания - преодоление нехватки адресов IPv4

Протокол NAT

Предпосылки появления NAT

- **Исчерпание IPv4-адресов:** 4.3 млрд адресов недостаточно для всех устройств.
- **Рост количества устройств:** Компьютеры, смартфоны, планшеты, IoT-гаджеты.

Внешние и внутренние IP-адреса

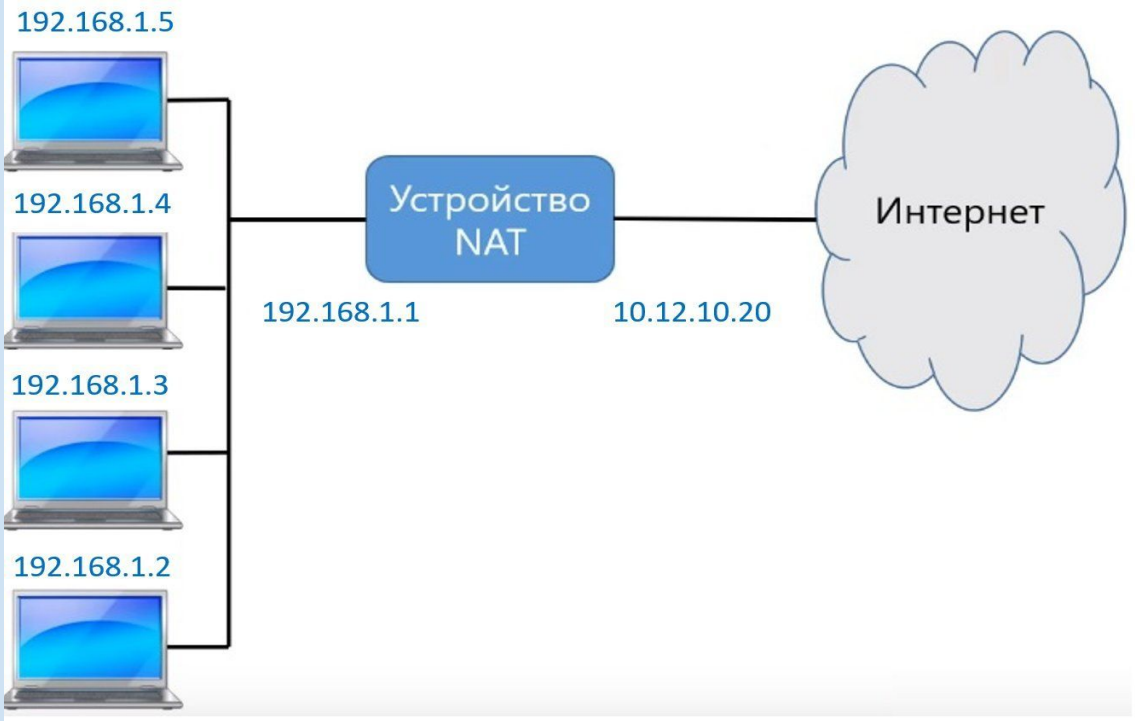
Внешние IP-адреса

- Применяются в сети Интернет
- Должны быть уникальными
- Распределяются ICANN
- Адресов не хватает для всех устройств

Внутренние IP-адреса

- Диапазон частных сетей (RFC 1918):
  - `10.0.0.0/8`
  - `172.16.0.0/12`
  - `192.168.0.0/16`
- Не маршрутизируются в Интернет
- Могут использоваться без обращения в ICANN
- Допускается использование одинаковых адресов в разных сетях

Протокол NAT



Типы NAT по способу выделения адресов

Статический NAT

- Отображение один к одному

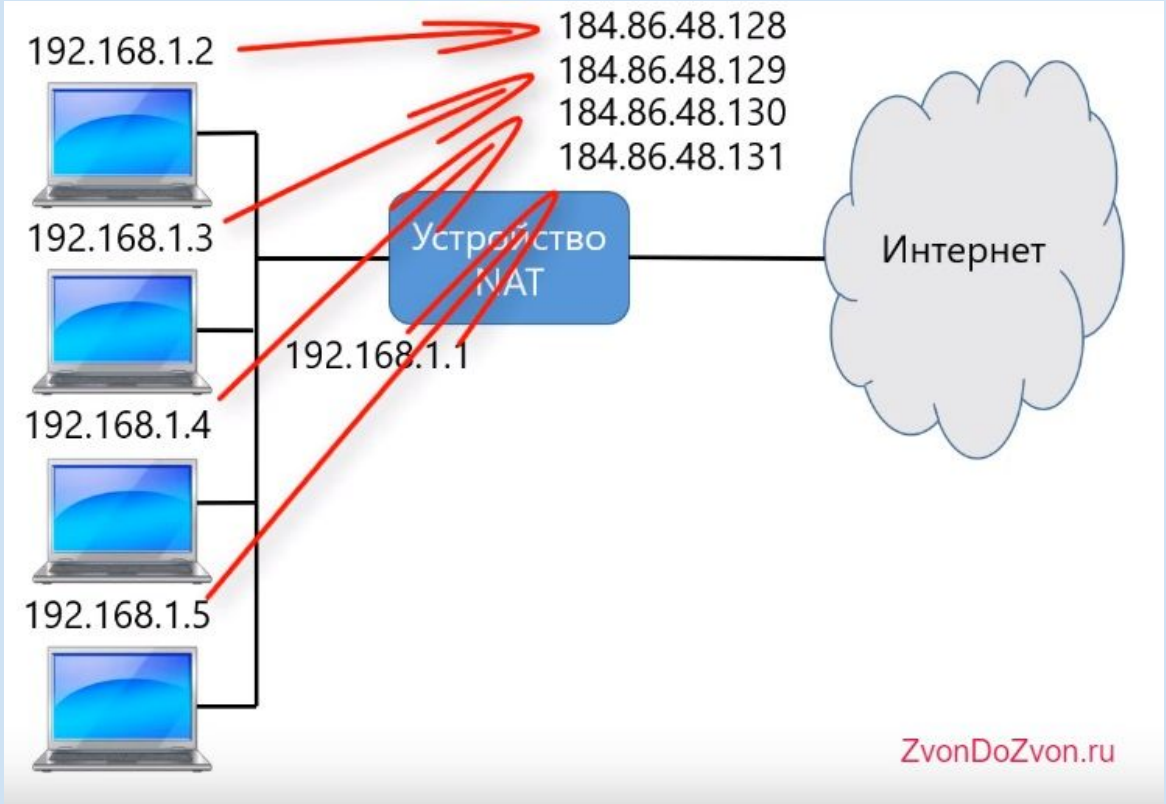
Динамический NAT

- Отображение внутренних адресов на группу внешних адресов

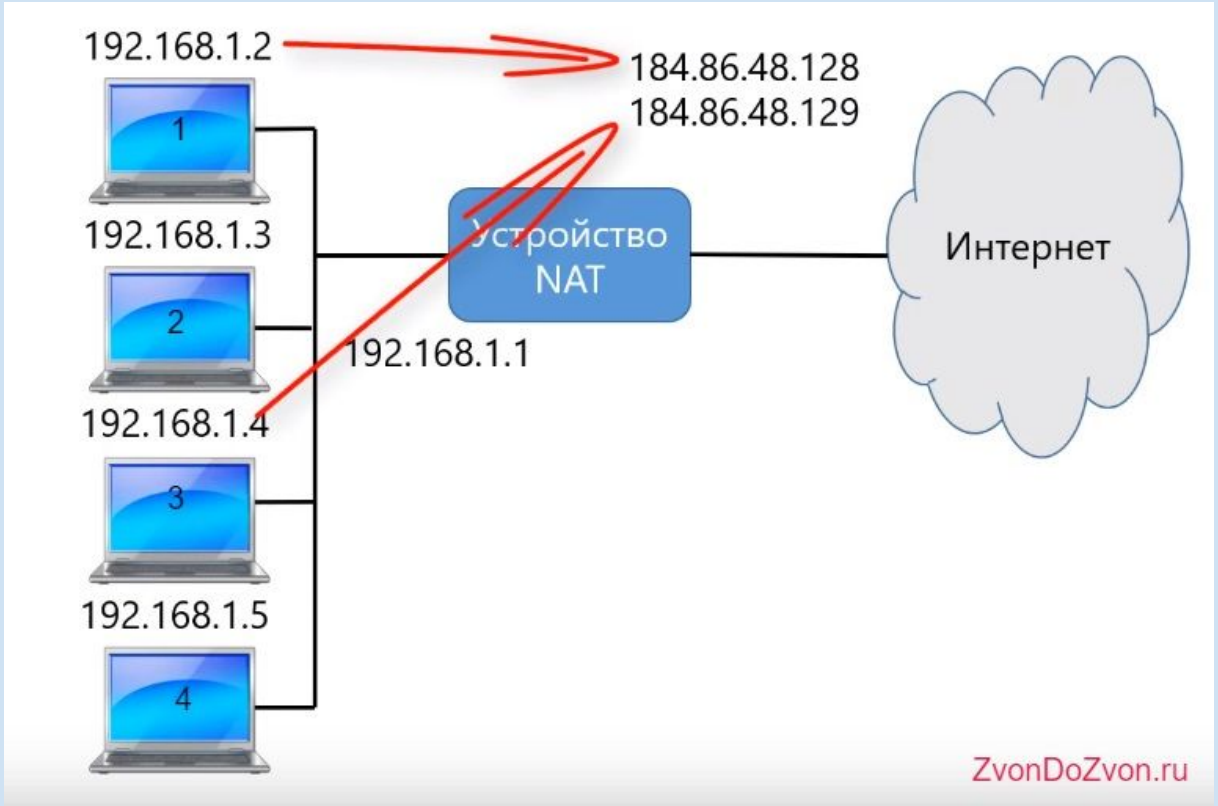
Один ко многим

- Отображение внутренних адресов на один внешний адрес

Статический NAT



Динамический NAT



NAT один ко многим

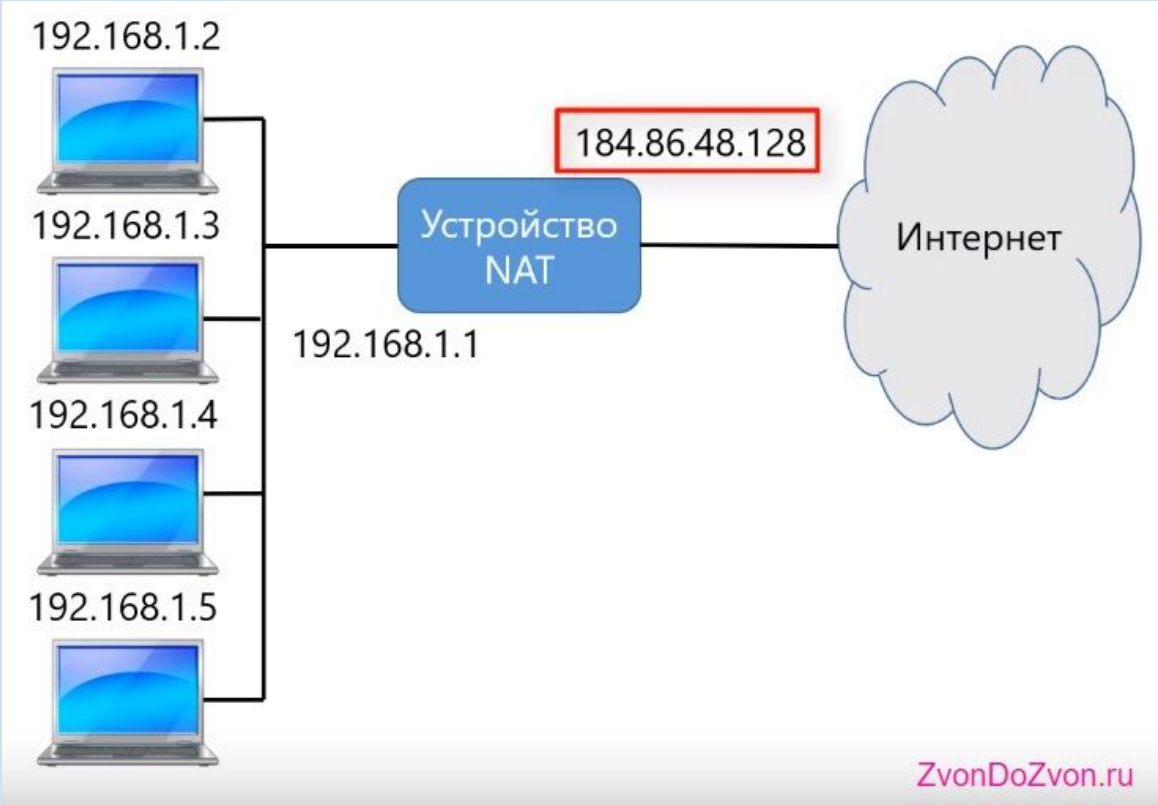


Таблица NAT

Преобразование выполняется с помощью таблицы NAT

Использует комбинацию IP-адрес + порт

Вид таблицы NAT

Внутренний IP	Внутренний порт	Внешний IP	Внешний порт
192.168.1.2	50300	184.86.48.128	49127
192.168.1.3	52001	184.86.48.128	49128
192.168.1.4	49238	184.86.48.128	49129

Типы NAT по направлению трансляции

Source NAT

- Подмена источника
- Используется когда инициатор соединения находится внутри нашей частной сети, а цель — снаружи. Классический выход в интернет.

Destination NAT

- Подмена назначения
- Используется когда инициатор соединения находится снаружи, а цель внутри нашей частной сети. Классический проброс портов (Port Forwarding).

Принцип работы SNAT (Source NAT)

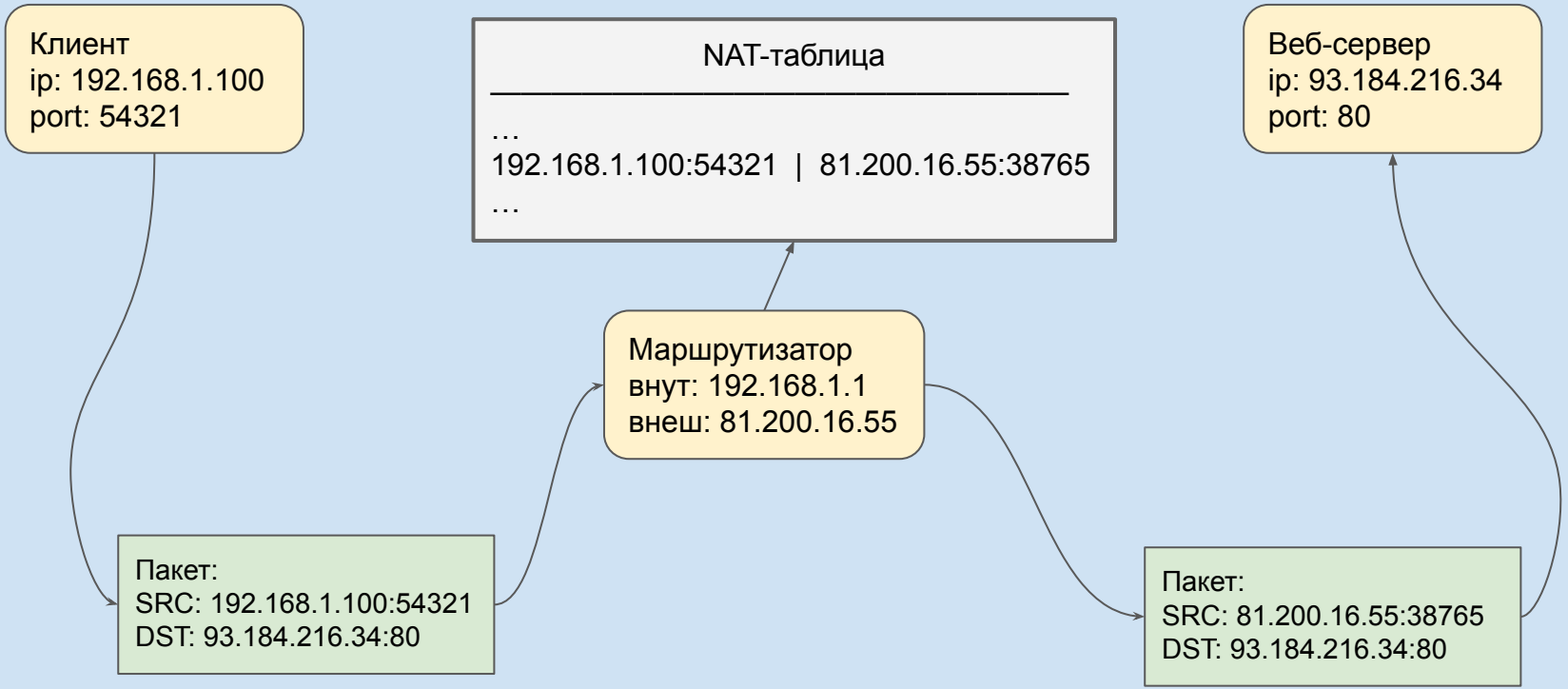
Сценарий

- **Внутренний хост:** `192.168.1.100:54321`
- **Внешний хост (веб-сервер):** `93.184.216.34:80`
- **Публичный IP маршрутизатора:** `81.200.16.55`

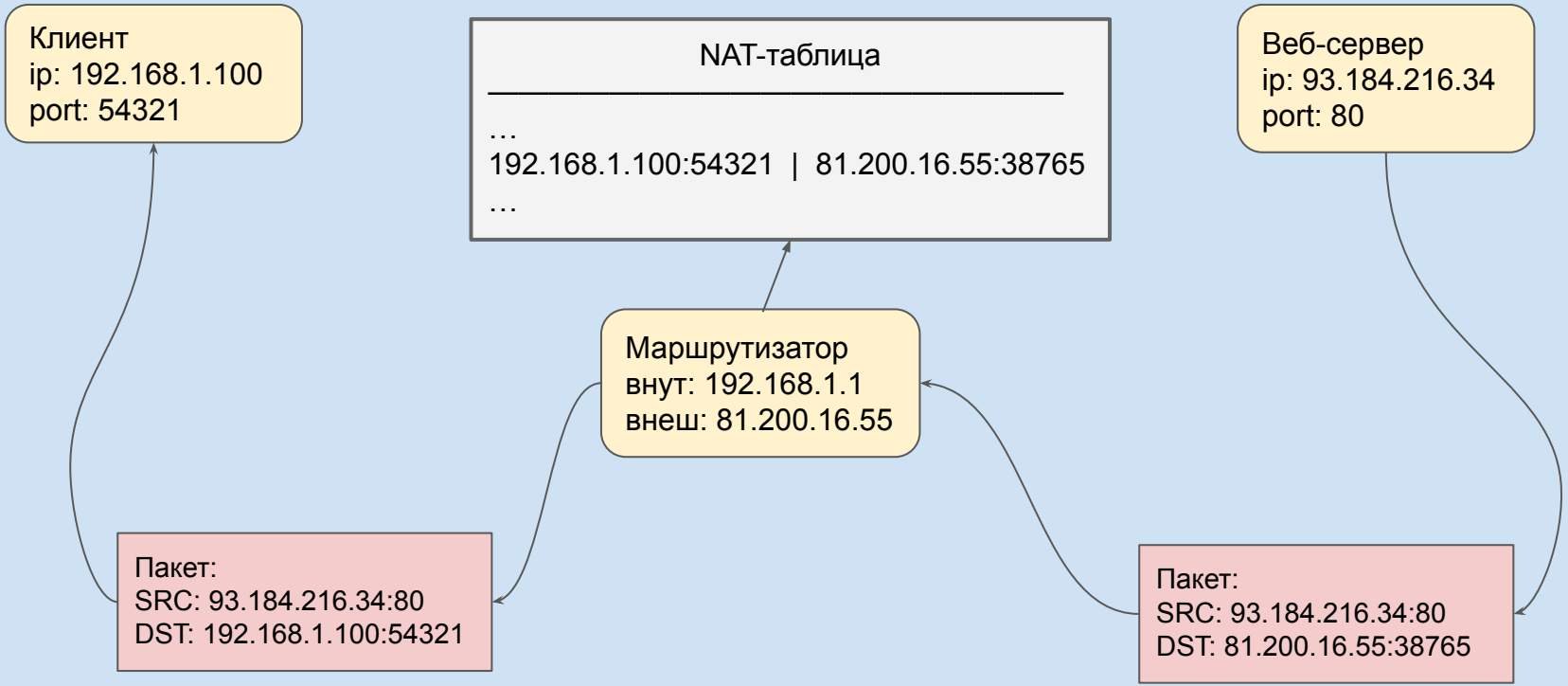
Процесс

1. **Исходный пакет (LAN -> WAN):**
   - Src: `192.168.1.100:54321`
   - Dst: `93.184.216.34:80`
2. **NAT на маршрутизаторе:**
   - Заменяет Source IP:Port на `81.200.16.55:38765`
   - Сохраняет запись в NAT-таблице: `(192.168.1.100:54321 <-> 81.200.16.55:38765)`
3. **Пакет в Интернете:**
   - Src: `81.200.16.55:38765`
   - Dst: `93.184.216.34:80`
4. **Ответный пакет (WAN -> LAN):**
   - Маршрутизатор по NAT-таблице выполняет обратную замену.

Принцип работы SNAT



Принцип работы SNAT



Принцип работы DNAT (Destination NAT)

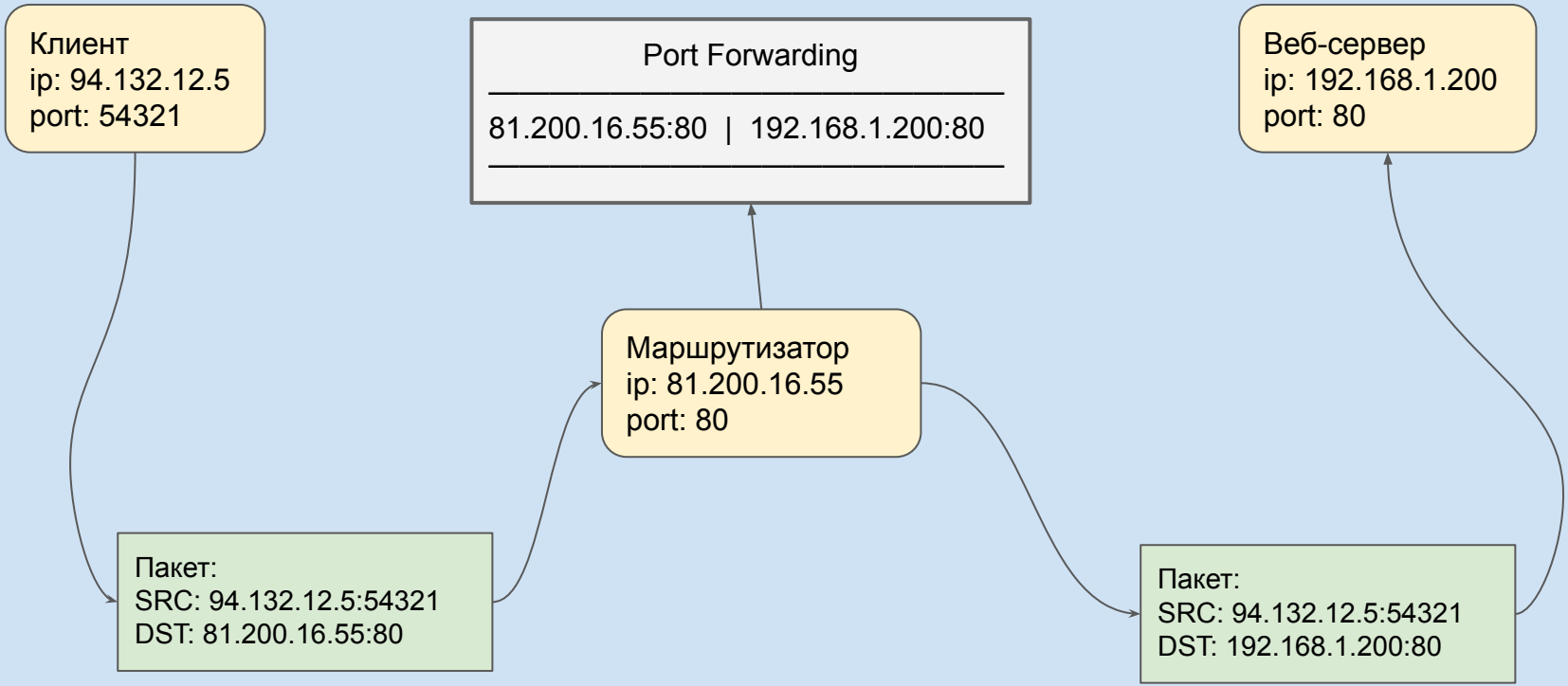
Задача

- **Внутренний хост:** Предоставление доступа извне к серверу в частной сети (веб-сервер, игровой сервер).
- **Пример:** Веб-сервер в LAN: `192.168.1.200:80`

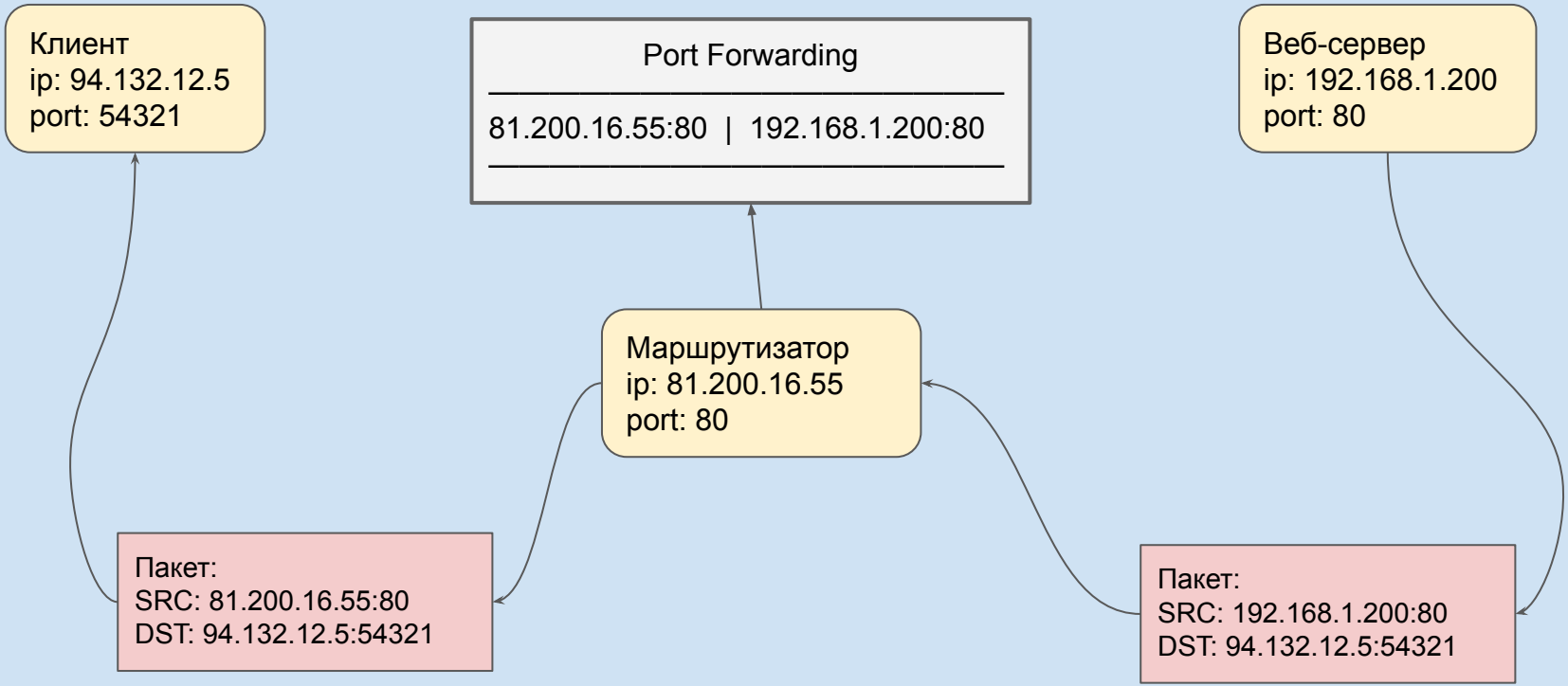
Процесс

1. **Настройка правила на маршрутизаторе (Port Forwarding):**
   - "Все входящие подключения на `81.200.16.55:80` перенаправлять на `192.168.1.200:80`".
2. **Входящий пакет из Интернета:**
   - Src: `client_ip:random_port`
   - Dst: `81.200.16.55:80`
3. **NAT на маршрутизаторе:**
   - Заменяет Destination IP:Port на `192.168.1.200:80`
4. **Пакет поступает на внутренний сервер:**
   - Src: `client_ip:random_port`
   - Dst: `192.168.1.200:80`

Принцип работы DNAT



Принцип работы DNAT



Ограничения NAT

- **Нарушение сквозного принципа (End-to-End Principle):**
  - Прямое соединение между двумя хостами в разных частных сетях невозможно без обходных механизмов.
- **Проблемы для P2P-приложений:**
  - Торренты, VoIP (Skype), онлайн-игры.
  - Требуют сложных технологий для обхода NAT (STUN, TURN, ICE).
- **Усложнение протоколов:**
  - Некоторые протоколы (например, FTP в активном режиме) вкладывают IP-адреса в тело пакета, что требует специальных ALG (Application Level Gateways) на NAT-устройстве.
- **Единая точка отказа:**
  - Выход из строя NAT-маршрутизатора лишает всю внутреннюю сеть выхода в Интернет.
- **Нет единого стандарта NAT:**
  - Много разных вариантов

Необходимость перехода на IPv6

Проблемы IPv4

- **Исчерпание адресов:** 4.3 млрд адресов недостаточно.
- **Сложность и "костыли":** NAT, CIDR усложнили архитектуру.
- **Ограниченная безопасность:** IPsec — опциональное дополнение.

Решение — IPv6

- **Гигантское адресное пространство:** 128 бит, 2^{128} адресов.
- **Упрощение архитектуры:** Отказ от NAT, возврат к сквозной модели.
- **Встроенные механизмы:** IPsec, QoS, автоконфигурация.

IPv6 в моделях OSI и TCP/IP

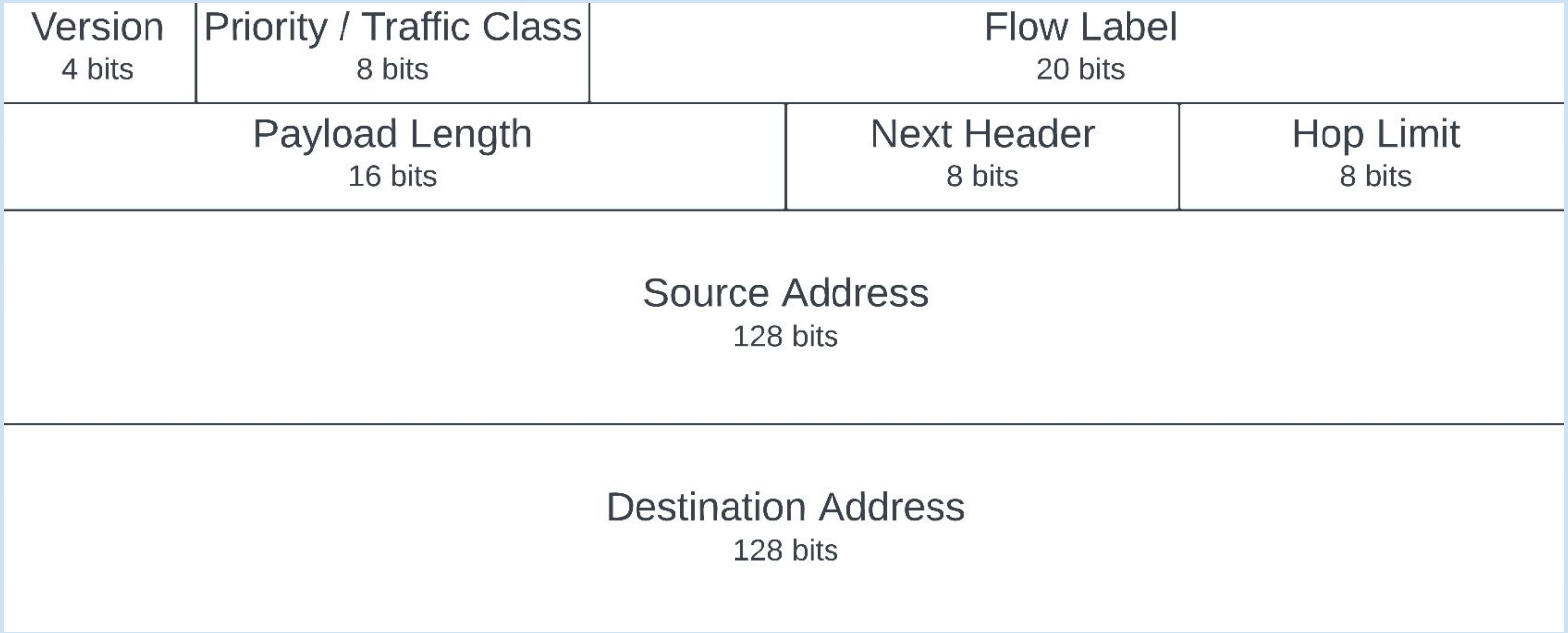
Модель OSI

- Уровень 7: Прикладной
- Уровень 6: Представительский
- Уровень 5: Сеансовый
- Уровень 4: Транспортный
- **-> Уровень 3: Сетевой -> IPv6**
- Уровень 2: Канальный
- Уровень 1: Физический

Модель TCP/IP

- Уровень 4: Прикладной
- Уровень 3: Транспортный
- **-> Уровень 2: Межсетевое взаимодействие -> IPv6**
- Уровень 1: Сетевой интерфейс

Формат IPv6-заголовка



Формат IPv6-заголовка

Ключевые особенности:

- **Упрощенная структура:** 8 полей вместо 13.
- **Отсутствие контрольной суммы:** Расчет целостности возложен на L2 и L4.

Основные поля:

- **Version (4 бита):** = 6.
- **Traffic Class (8 бит):** Приоритет пакета (QoS).
- **Flow Label (20 бит):** Идентификатор потока данных. Одна и та же метка используется для пакетов с одинаковых тпом обслуживания (Traffic Class)
- **Payload Length (16 бит):** Длина полезной нагрузки.
- **Next Header (8 бит):** Используется при наличии дополнительных заголовков.
- **Hop Limit (8 бит):** Аналог IPv4 TTL.
- **Source/Destination Address (128 бит каждый):** IP-адреса.
- **Дополнительные заголовки:** (Необязательно)

Дополнительные заголовки IPv6

- **Hop-by-Hop Options (Параметры от узла к узлу):** Данные, которые должен обработать каждый маршрутизатор на пути следования пакета.
- **Destination Options (Параметры получателя):** Данные, необходимые для обработки пакета только на стороне конечного получателя.
- **Routing Header (Заголовок маршрутизации):** Содержит список узлов, через которые пакет обязательно должен пройти на пути к месту назначения.
- **Fragment Header (Заголовок фрагментации):** Используется для фрагментации пакетов, слишком больших для MTU канала. В IPv6 выполняется только отправителем.
- **Authentication Header (AH, Заголовок аутентификации):** Обеспечивает аутентификацию отправителя, целостность данных и защиту от повторов.
- **Encapsulating Security Payload (ESP, Инкапсулирующая нагрузка безопасности):** Обеспечивает конфиденциальность (шифрование), аутентификацию и целостность данных.

Формат записи IPv6-адресов

IP-адреса:

- IPv4 - 4 байта
- IPv6 - 16 байт

IPv6

- 8 групп по 4 шестнадцатеричных цифры, разделенных двоеточиями.
- **Пример:** `2001:0db8:85a3:0000:0000:8a2e:0370:7334`

Терминология IPv6

IPv4:

- Адрес подсети
- Адрес хоста
- Маска подсети

IPv6:

- Префикс IPv6
- Адрес интерфейса
- Длина префикса

Пример префикса IPv6:

- `2001:0db8:85a3:0000:0000:8a2e:0370:7334/64`

Вычисление префикса IPv6

Длина префикса кратна 16:

- Адрес: `2001:0db8:85a3:ad61:0000:8a2e:0370:7334/64`
- Префикс: `2001:0db8:85a3:ad61:0000:0000:0000:0000`

Длина префикса кратна 4:

- Адрес: `2001:0db8:85a3:ad61:0000:8a2e:0370:7334/52`
- Префикс: `2001:0db8:85a3:a000:0000:0000:0000:0000`

Длина префикса не кратна 4:

- Адрес: `2001:0db8:85a3:ad61:0000:8a2e:0370:7334/54`
- Префикс: `2001:0db8:85a3:ac00:0000:0000:0000:0000`

Правила сокращения IPv6 адресов

Полный адрес IPv6:

- `2a02:06b8:0000:0001:0000:0000:feed:0a11`

Удаление ведущих нулей:

- `2a02:6b8:0:1:0:0:feed:a11`

Пропуск двух и более подряд идущих групп нулей (только единожды):

- `2a02:6b8:0:1::feed:a11`

Неправильные сокращения

Полный адрес IPv6:

- `2a02:06b8:0000:1000:0000:0000:feed:0a11`
- Неправильно: `2a02:6b8:0:1::feed:a11`
- Правильно: `2a02:6b8:0:1000::feed:a11`

Пропуск двух групп нулей:

- `fe80:0000:0000:abcd:0000:0000:0000:ab11`
- Неправильно: `fe80::abcd::ab11`
- Правильно: `fe80:0:0:abcd::ab11`

Типы IPv6-адресов

Unicast

- **Назначение:** Один интерфейс.

Multicast

- **Назначение:** Группа интерфейсов.
- **Особенность:** Полная замена broadcast.
- **Пример:** ff02::1 (все узлы), ff02::2 (все маршрутизаторы).

Anycast

- **Назначение:** Несколько интерфейсов. В отличие от Multicast, сообщение получает только один интерфейс.
- **Доставка:** Ближайшему узлу.
- **Применение:** DNS-серверы, CDN.

Типы Unicast-адресов в IPv6

Глобальные (global unicast address):

- Действуют в интернет
- Распределяются IANA

Локальные (unique local address):

- Не маршрутизируются в интернет
- Можно использовать без обращения в IANA

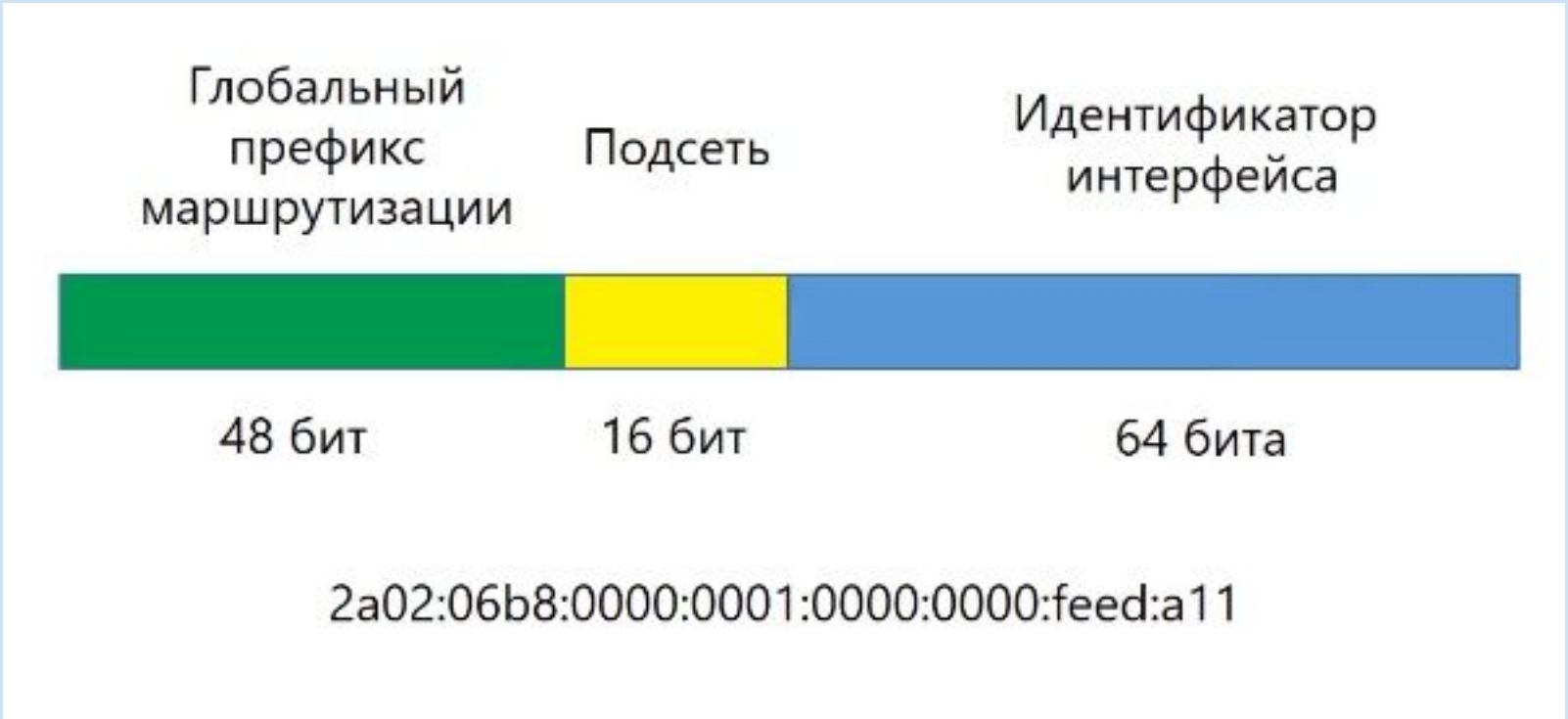
Локальные канала связи (link-local address):

- Действуют в пределах одного сегмента сети (не проходят через маршрутизатор)
- Не маршрутизируются в интернет

Начальные цифры адресов IPv6

Тип адреса	Начальные цифры
Глобальный	Любые, кроме цифр, предназначенных для других типов адресов (обычно 2 или 3)
Локальный	FD
Локальный канала связи	FE80
Групповой	FF

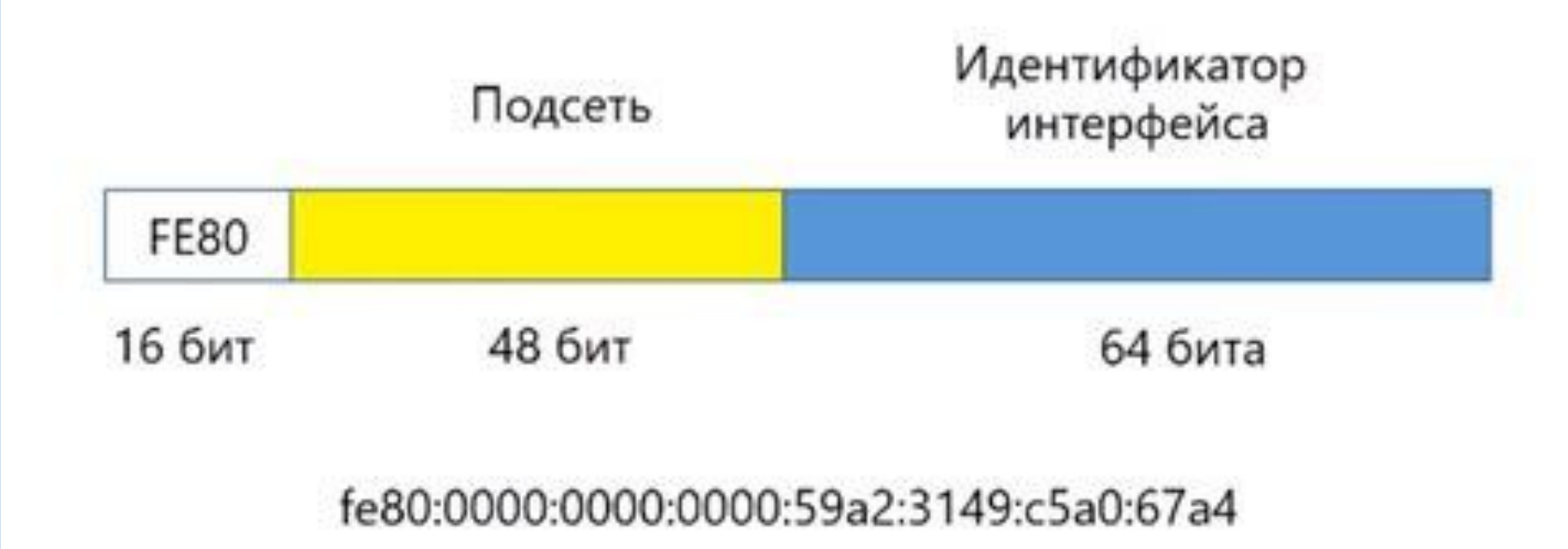
Структура глобального IPv6 адреса



Структура локального IPv6 адреса

Документ RFC 4193 - правила выбора глобального идентификатора

Структура локального адреса канала связи



Специальные IPv6 адреса

::/128 - текущий хост

::/0 - маршрут по-умолчанию

::1/128 - loopback

ff02::1 - все узлы в канале связи

ff02::2 - все маршрутизаторы в канале связи

Способы назначения адресов

Адреса IPv4:

- Вручную
- DHCP

Адреса IPv6:

- Вручную
- DHCPv6
- Stateless Address Auto Configuration (SLAAC, RFC 4862)

SLAAC


Маршрутизатор
Сервер DHCPv6
Префикс IPv6
Шлюз
Адреса
DNS
Хост IPv6
Префикс IPv6
Идентификатор
интерфейса
64 бита
64 бита

Процедура EUI-64

Extended Unique Identifier EUI-64:

- Идентификатор интерфейса на основе MAC-адреса
- Длина 64 бита
- RFC 2373

Процедура получения EUI-64:

- Берем MAC-адрес устройства: e4:a7:a0:46:c3:7d
- Вставляем в середину байты ff:fe: e4:a7:a0:ff:fe:46:c3:7d
- Инвертируем предпоследний бит первого байта: e6:a7:a0:ff:fe:46:c3:7d

Недостатки

- Идентификатор интерфейса всегда постоянный и основан на MAC
- Можно отследить действия пользователя

Другие подходы

Временные адреса

- RFC 4941
- Адрес выбирается случайным образом
- Адреса меняются
- Не все приложения могут работать с адресами, которые могут меняться
- Пример:
  - Сегодня: 2001:db8:1234::a1b2:c3d4:e5f6:7890
  - Завтра: 2001:db8:1234::f5e6:d4c3:b2a1:0987

Stable Privacy

- RFC 7217
- Адрес выбирается случайным образом
- Генерируется одинаковый в одной сети, но разный при переходе в другую сеть
- Пример:
  - В сети А: 2001:db8:1234::5a4b:3c2d:1e0f:9876
  - В сети Б: 2001:db8:5678::f0e1:d2c3:b4a5:6789

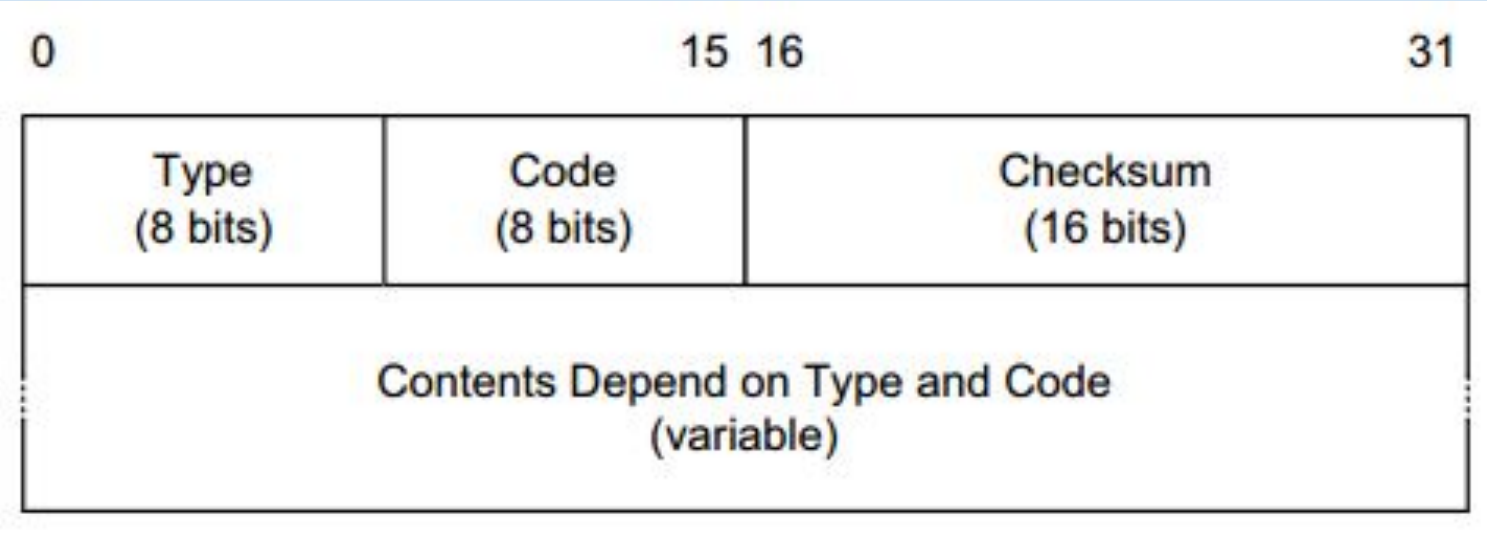
Протокол ICMPv6

Основные отличия от ICMPv4

- **Единый протокол:** Объединяет функции ICMP, IGMP и ARP
- **Обязательная поддержка:** Должен быть реализован во всех стеках IPv6
- **Расширенные функции:** Поддержка NDP, MLD, Path MTU Discovery
- **Next Header = 58**

- **Type (8 бит):** Категория сообщения (ошибка, информационное)
- **Code (8 бит):** Детализация типа сообщения
- **Checksum (16 бит):** Контрольная сумма заголовка ICMPv6 и данных

Протокол ICMPv6



Основные типы сообщений ICMPv6

Тип	Название	Назначение	Категория
1	Destination Unreachable	Сообщения об ошибках доставки	Ошибка
2	Packet Too Big	Обнаружение MTU пути	Ошибка
3	Time Exceeded	Превышение времени жизни	Ошибка
4	Parameter Problem	Ошибки в заголовке пакета	Ошибка
128	Echo Request	Запрос эха	Информационное
129	Echo Reply	Ответ эха	Информационное
133-137	**NDP Messages**	Обнаружение соседей и маршрутизаторов	**NDP**
130-132	**MLD Messages**	Управление групповыми рассылками	**MLD**

NDP (Neighbor Discovery Protocol)

NDP (протокол обнаружения соседей)

- Используется совместно с протоколом IPv6
- RFC 4861

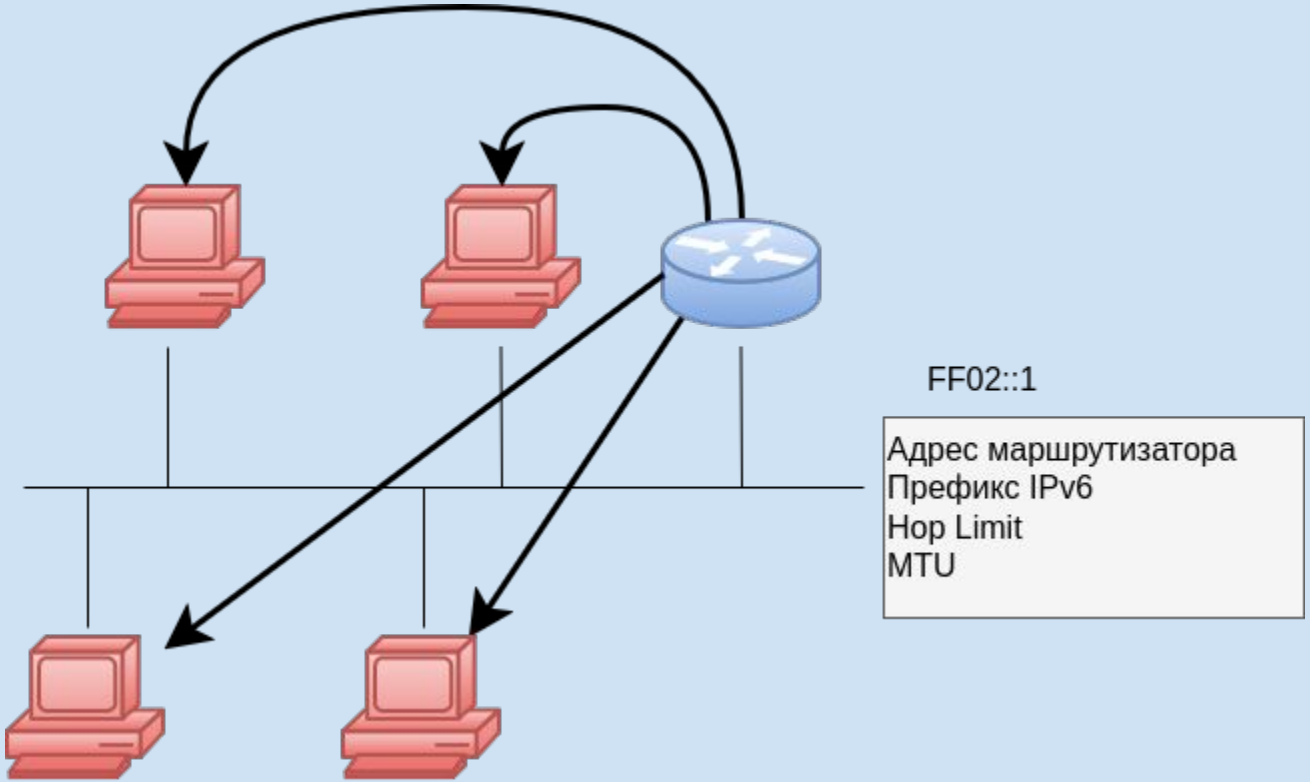
Назначение NDP:

- Определение адреса маршрутизатора и префикса IPv6 (SLAAC)
- Замена ARP для IPv6
- Настройка маршрутизации
- Проверка доступности узлов сети (соседей)
- Определение конфликта IP адресов

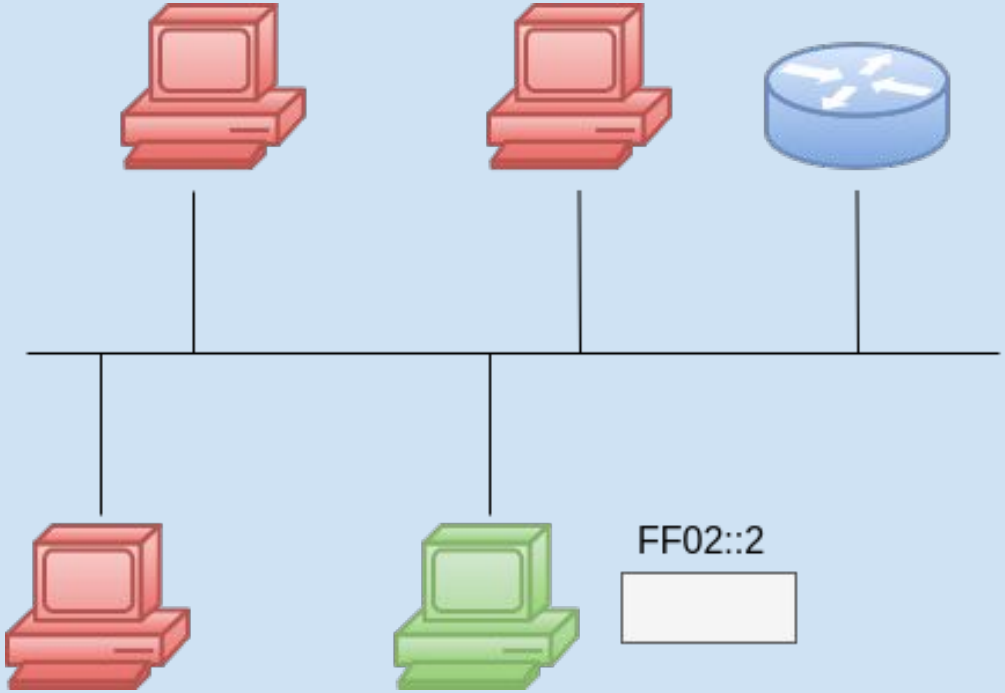
Основные сообщения NDP (ICMPv6)

Тип ICMPv6	Название	Назначение	Аналог в IPv4
133	Router Solicitation (RS)	Поиск маршрутизаторов в сети	ICMP Router Solicitation
134	Router Advertisement (RA)	Рассылка параметров сети маршрутизаторами	ICMP Router Advertisement
135	Neighbor Solicitation (NS)	Поиск MAC-адреса по IPv6	ARP Request
136	Neighbor Advertisement (NA)	Ответ с MAC-адресом	ARP Reply
137	Redirect	Указание оптимального маршрута	ICMP Redirect

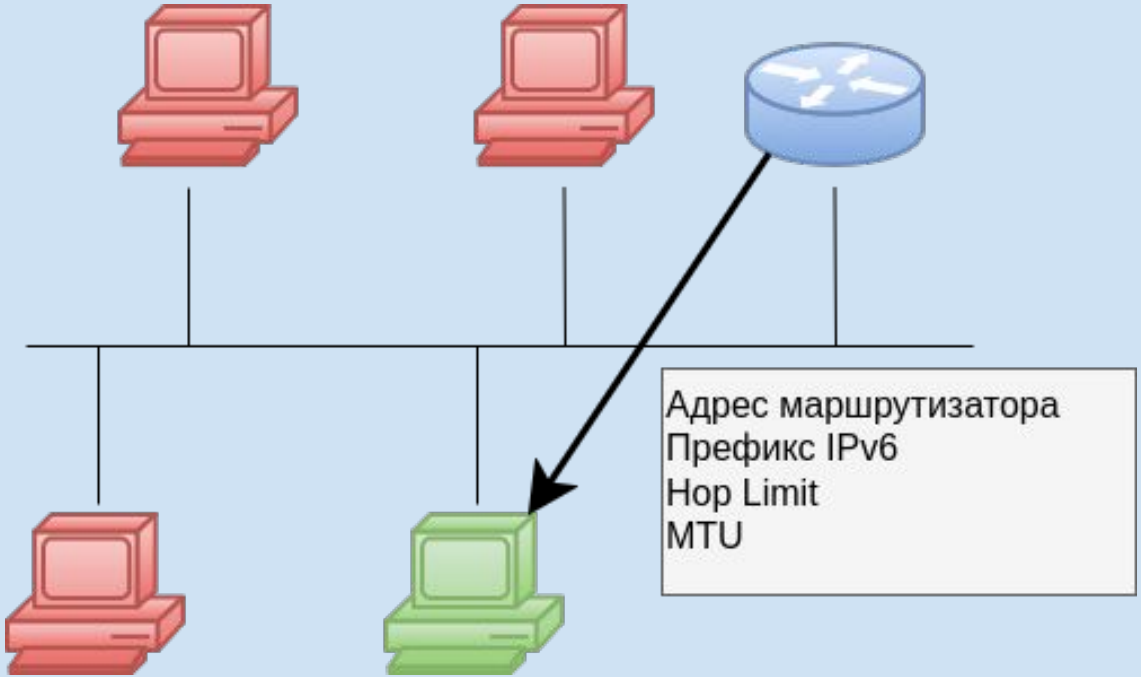
Router Advertisement



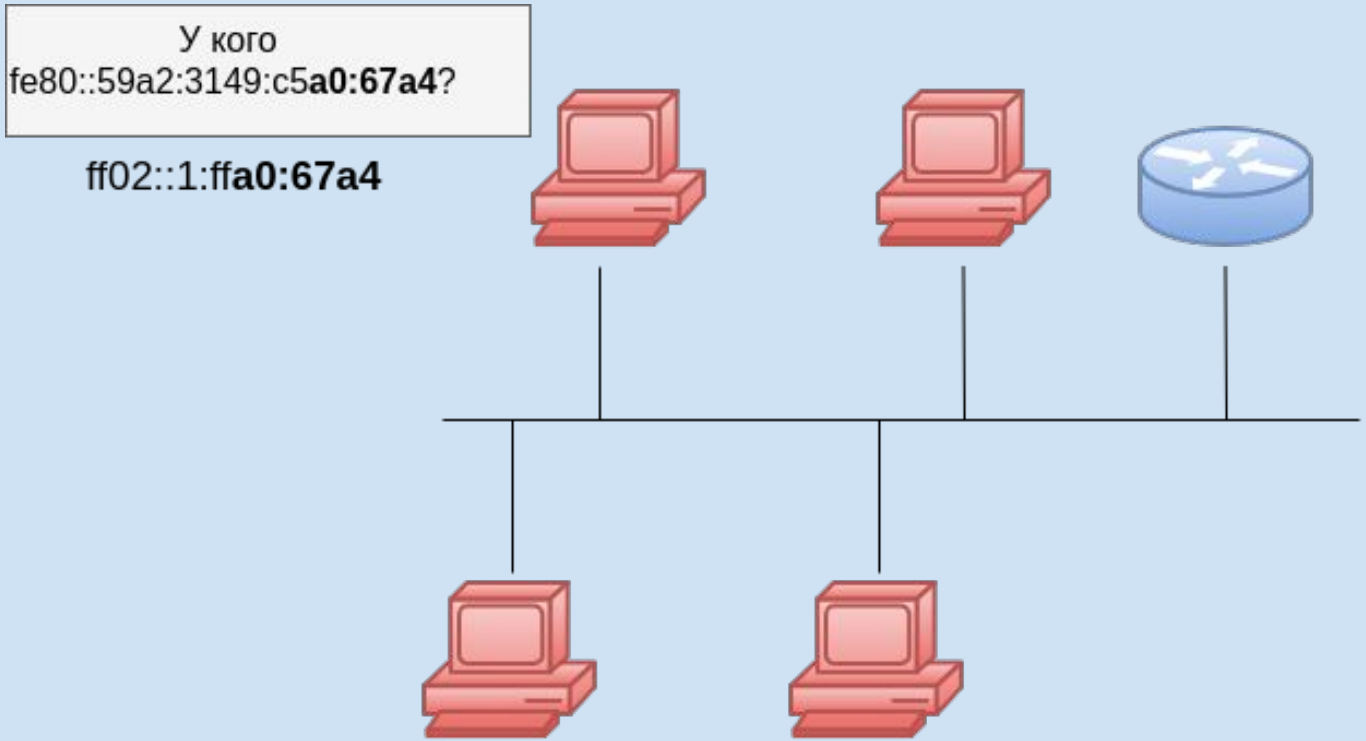
Router Solicitation



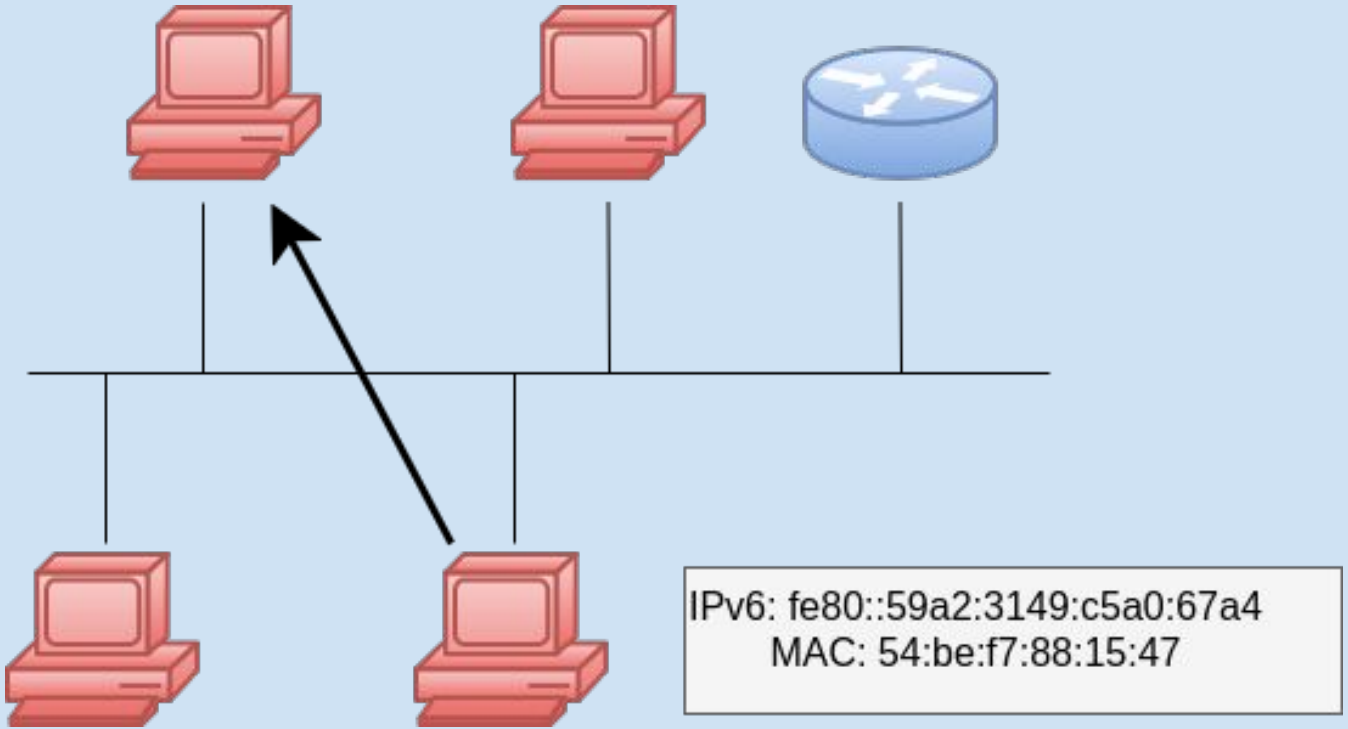
Router Advertisement



Neighbor Solicitation



Neighbor Advertisement



Кэширование MAC-адресов

Протокол ARP

- ARP таблица
- **Windows:** arp -a
- **Linux:** ip neigh show

Протокол NDP

- NDP таблица
- **Windows:** netsh `interface` ipv6 show neighbors
- **Linux:** ip -6 neigh show

Безопасность NDP и утилиты управления

Безопасность

- **Угрозы:**
  - Neighbor Advertisement Spoofing
  - Router Advertisement Spoofing
  - DAD-атаки
- **Защита:**
  - **SEcure Neighbor Discovery (SEND):** Использует криптографию
  - **RA Guard:** Фильтрация RA на свитчах
  - **DHCPv6 Guard:** Защита от поддельных DHCP-серверов

Утилиты

- **Просмотр кэша соседей:**
  - Windows: `netsh interface ipv6 show neighbors`
  - Linux: `ip -6 neigh show`
- **Диагностика:**
  - `ping6` для проверки связности
  - `traceroute6` для трассировки маршрута
- **Управление:**
  - `ip -6 addr` - управление IPv6-адресами
  - `ip -6 route` - управление маршрутами

Тест - Вопрос 1

Вопрос: Хост с IP 10.1.1.100 и маской 255.255.255.0 отправляет пакет на 10.1.2.200. Куда будет отправлен пакет?

Варианты ответов:

a) Напрямую, так как адреса в одной сети.

b) На шлюз, так как адреса в разных сетях.

c) Пакет будет отброшен.

Тест - Вопрос 2

Вопрос: IPv4 пакет с установленным флагом DF (Don't Fragment) приходит на маршрутизатор, но размер пакета превышает MTU следующего интерфейса. Что сделает маршрутизатор?

Варианты ответов:

a) Выполнит фрагментацию, несмотря на флаг
b) Отбросит пакет и отправит ICMP "Fragmentation Needed"
c) Увеличит MTU интерфейса для передачи пакета
d) Отправит пакет частями без фрагментации

Тест - Вопрос 3

Вопрос: В сети с маской `/26` сколько всего доступных для хостов адресов?

Варианты ответов:

a) 254
b) 62
c) 64
d) 126

Тест - Вопрос 4

Вопрос: Клиент в сети 192.168.1.0/24 имеет IP 192.168.1.100. Он отправляет пакет на адрес 192.168.1.255. Что произойдет с этим пакетом?

Варианты ответов:

а) Он будет доставлен только хосту 192.168.1.255

б) Он будет доставлен всем хостам в сети 192.168.1.0/24

в) Он будет доставлен только маршрутизатору

г) Пакет будет отброшен как ошибочный

Тест - Вопрос 5

Вопрос: Маршрутизатор получает пакет с TTL=1. Пакет нужно передать на другой интерфейс. Что произойдет?

Варианты ответов:

a) Пакет будет отправлен, TTL уменьшится до 0

b) Пакет будет отправлен, TTL останется 1

c) Пакет будет отброшен, отправителю отправится ICMP "Time Exceeded"

d) Пакет будет отброшен без уведомления отправителя

Тест - Вопрос 6

Вопрос: DHCP-клиент находится в сети 192.168.1.0/24, а DHCP-сервер - в сети 192.168.2.0/24. Что должно быть настроено для работы DHCP?

Варианты ответов:
a) DHCP Relay Agent на маршрутизаторе
b) Статический маршрут между сетями
c) Настроить сервер в обеих сетях
d) DHCP будет работать без дополнительных настроек

Тест - Вопрос 7

Вопрос: Хост с IP 192.168.1.5/24 пытается отправить пакет на 192.168.2.10/24. Шлюз по умолчанию не настроен. Что произойдет с пакетом?

Варианты ответов:

a) Пакет будет отправлен в широковещательной рассылке
b) Хост отправит ARP-запрос для 192.168.2.10
c) Пакет будет отброшен, и хост вернет ошибку "Network unreachable"
d) Пакет будет отправлен на шлюз по умолчанию провайдера

Тест - Вопрос 8

Вопрос: После успешного завершения DHCP DORA клиент получил IP 192.168.1.100/24 и шлюз 192.168.1.1. Что произойдет, если вручную прописать на этом же клиенте статический IP 192.168.1.200/24 и перезагрузить сетевой интерфейс?

Варианты ответов:
a) Возникнет конфликт IP-адресов, так как DHCP-сервер помнит предыдущую аренду.
b) Клиент будет использовать статический адрес 192.168.1.200, DHCP больше не участвует.
c) Клиент отправит DHCPRELEASE для старого адреса и запросит новый.
d) Клиент будет пытаться использовать оба адреса одновременно.

Тест - Вопрос 9

Вопрос: Маршрутизатор выполняет NAT. Во внутренней сети два компьютера (192.168.1.10 и 192.168.1.20) одновременно устанавливают соединение с одним и тем же внешним веб-сервером 93.184.216.34:80. Как маршрутизатор различит, какому внутреннему хосту принадлежат приходящие ответы?

Варианты ответов:

a) По IP-адресу назначения.
b) По MAC-адресу назначения.
c) По исходному порту во внешнем пакете.
d) По полю TTL в IP-заголовке.

Тест - Вопрос 10

Вопрос: В сети 10.1.0.0/16 нужно создать подсеть минимум на 500 хостов. Какая маска будет оптимальной?

Варианты ответов:

a) 255.255.0.0 (/16)

b) 255.255.254.0 (/23)

c) 255.255.255.0 (/24)

d) 255.255.252.0 (/22)