МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ | НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

ICMP

Введение в сетевые технологии

ICMPv4 и ICMPv6 сообщения

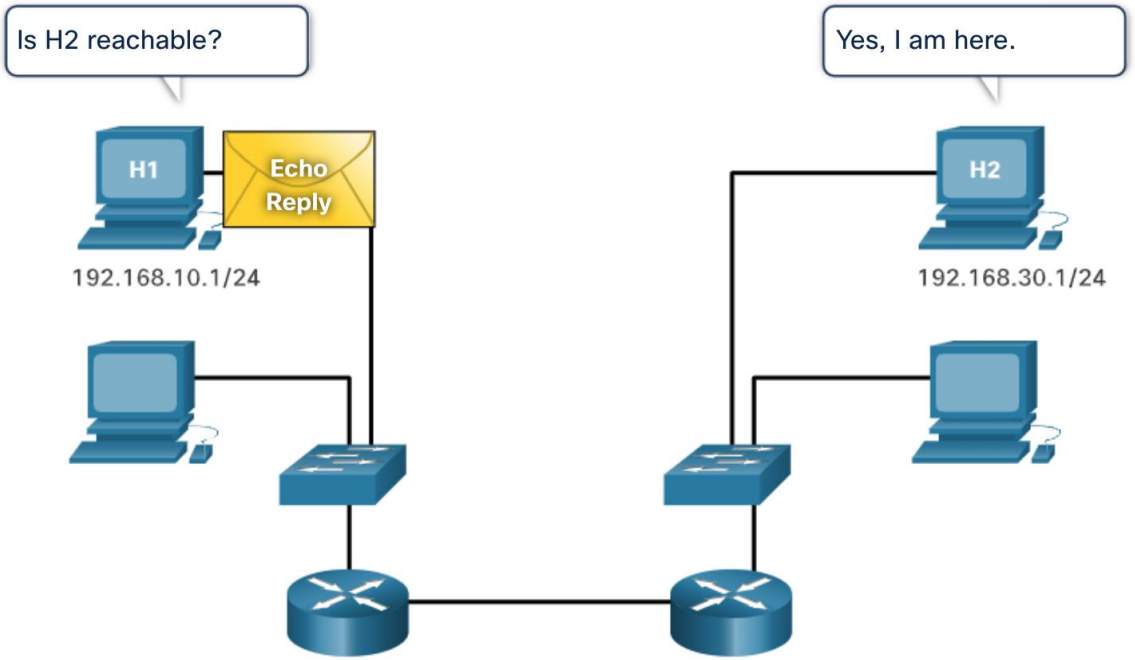
- Протокол ICMP обеспечивает обратную связь по вопросам, связанным с обработкой IP-пакетов при определенных условиях.
- Сообщения ICMP, общие для ICMPv4 и ICMPv6, включают в себя:
  - достижимость узла;
  - назначение узла или сервис недоступны;
  - превышение интервала ожидания.
- Хотя IP не является надёжным протоколом, пакет TCP/IP обеспечивает отправку сообщений в случае возникновения определённых ошибок. Эти сообщения отправляются посредством служб ICMP.

Достижимость узла

Эхо-сообщение ICMP можно использовать для проверки доступности узла в IP-сети.

В примере:

- Локальный узел отправляет эхо-запрос ICMP.
- Если узел доступен, узел назначения отправляет эхо-ответ.



Назначение узла или сервис недоступны

- Сообщение ICMP Destination Unreachable может использоваться для уведомления источника о недоступности места назначения или службы.
- Такое сообщение содержит код, определяющий причину, по которой пакет не может быть доставлен.

Примеры некоторых кодов сообщений о недоступном узле назначения для ICMPv4:

- 0 — сеть недоступна;
- 1 — узел недоступен;
- 2 — протокол недоступен;
- 3 — порт недоступен.

Примеры некоторых кодов назначения недостижимых пакетов для ICMPv6:

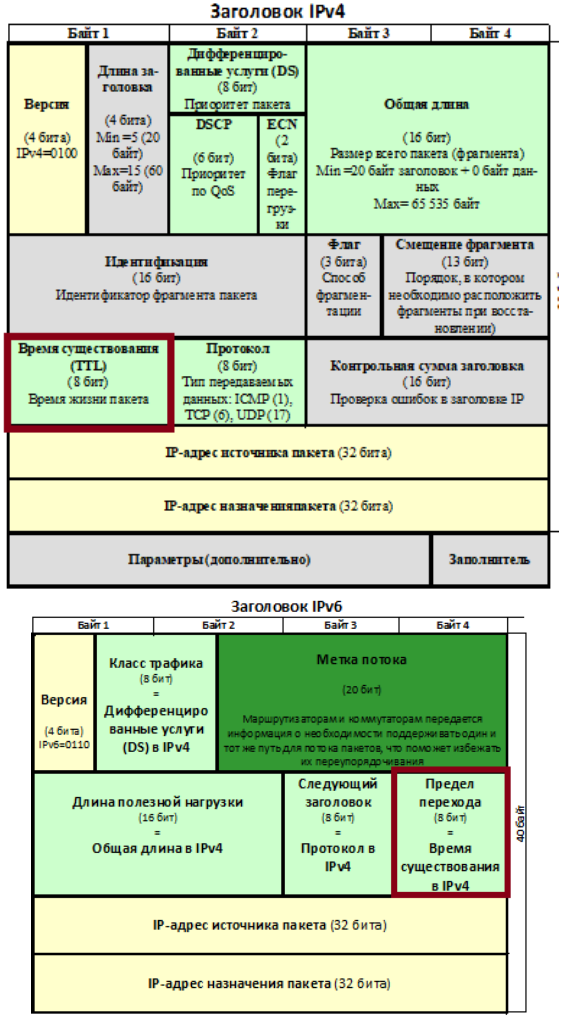
- 0 — нет маршрута до пункта назначения
- 1 — связь с пунктом назначения административно запрещена (например, брандмауэр)
- 2 — за пределами области адреса источника
- 3 — адрес недоступен
- 4 — порт недоступен.

Превышение интервала ожидания

- Когда поле Time to Live (TTL) в пакете IPv4 уменьшается до 0, в ICMPv4 будет отправлено сообщение ICMPv4 Time Exceeded.
- Когда поле «предел переходов» (hop limit) в пакете IPv6 уменьшается до 0, ICMPv6 также отправляет сообщение о превышении времени.

```
Pinging 8.8.8.8 with 32 bytes of data:
Reply from 192.168.1.1: TTL expired in transit.
Reply from 192.168.1.1: TTL expired in transit.
Reply from 192.168.1.1: TTL expired in transit.
Reply from 192.168.1.1: TTL expired in transit.

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

Примечание. Сообщения о превышении времени используются инструментом traceroute.



ICMPv6 сообщения

ICMPv6 имеет новые функции и улучшенные функциональные возможности, отсутствующие в ICMPv4, включая 4 новых протокола в рамках протокола обнаружения соседей (ND или NDP).

Обмен сообщениями между маршрутизатором IPv6 и устройством IPv6, включая динамическое распределение адресов, осуществляется следующим образом:

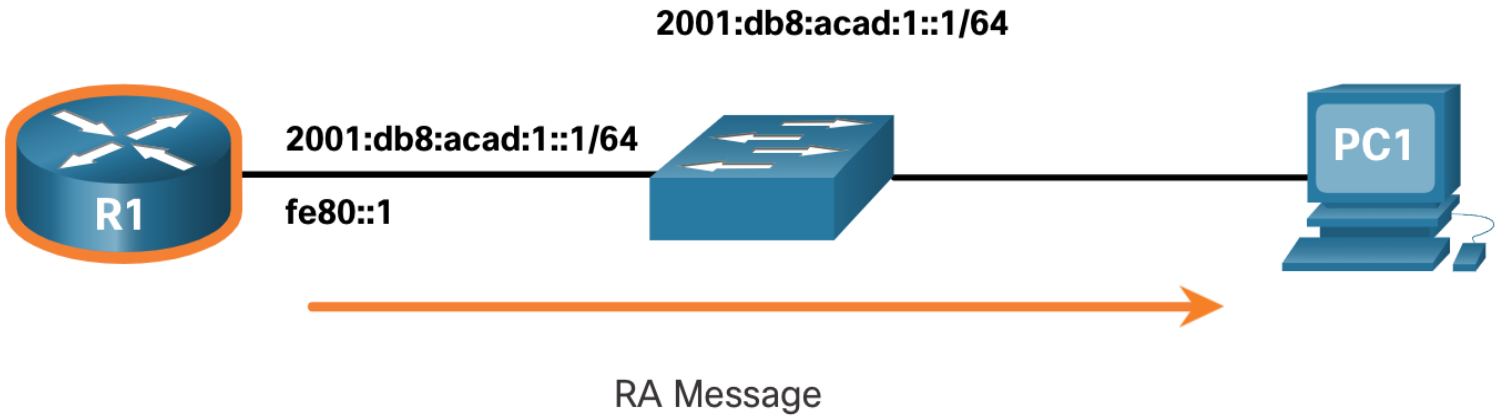
- Сообщение «Запрос к маршрутизатору» (Router Solicitation, RS)
- Сообщение «Ответ маршрутизатора» (Router Advertisement, RA)

Обмен сообщениями между устройствами IPv6, включая обнаружение повторяющихся адресов и разрешение адресов, осуществляется следующим образом:

- Сообщение с запросом поиска соседей (NS)
- Сообщение об объявлении соседних узлов (NA)

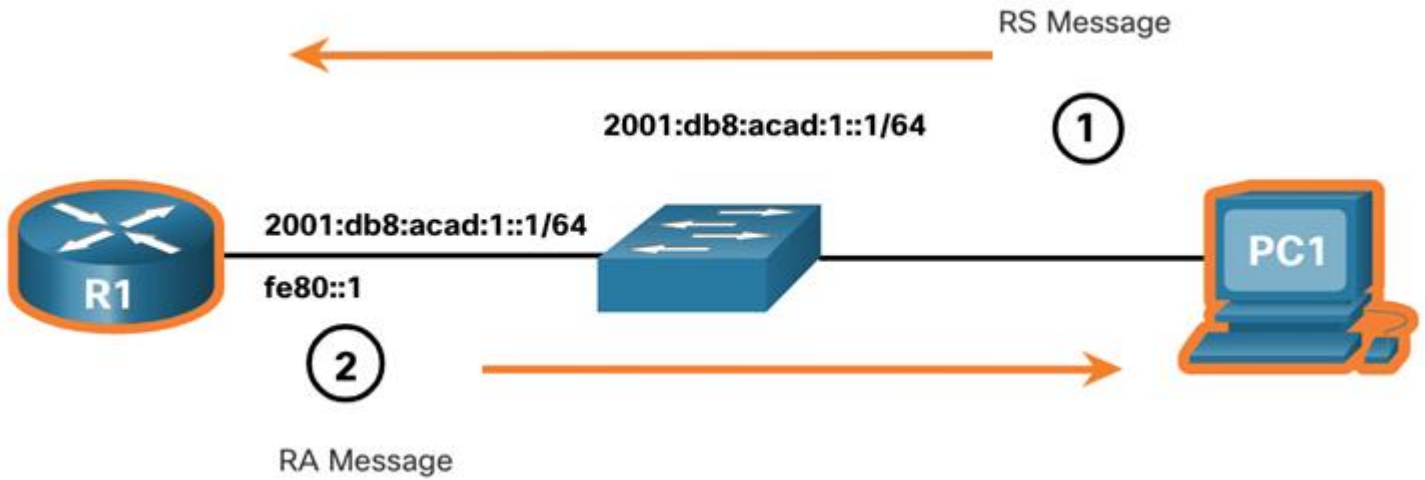
Примечание. ND-протокол ICMPv6 также включает сообщение перенаправления, которое имеет аналогичную с сообщением перенаправления, используемым в ICMPv4, функцию.

ICMPv6 сообщения



- Сообщения RA «Ответ маршрутизатора» отправляются маршрутизаторами с поддержкой IPv6 каждые 200 секунд для предоставления информации об адресации узлам с поддержкой IPv6.
- Сообщение RA «Ответ маршрутизатора» может включать такие данные об адресах для хостов, как префикс, длина префикса, DNS-адрес и доменное имя.
- Узел, использующий SLAAC, установит в качестве своего шлюза по умолчанию локальный адрес канала маршрутизатора, отправившего RA.

ICMPv6 сообщения



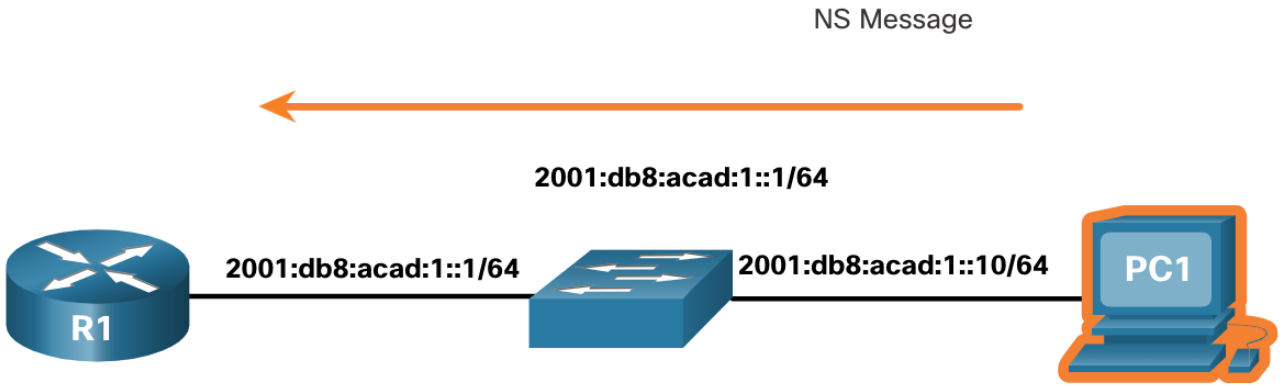
- Маршрутизатор с поддержкой IPv6 также отправит сообщение RA «Ответ маршрутизатора» в ответ на сообщение RS «Запрос к маршрутизатору».
- PC1 отправляет сообщение RS «Запрос к маршрутизатору», чтобы определить, как получать информацию об адресах IPv6 динамически:

  R1 отвечает PC с сообщением RA «Ответ маршрутизатора».

  PC1 отправляет сообщение RS: «Привет, я только что загрузился. Есть ли IPv6 маршрутизатор в сети? Мне нужно знать, как динамически получать информацию об адресах IPv6».

  R1 отвечает сообщением RA. "Привет всем устройствам с поддержкой IPv6. Я R1, и вы можете использовать SLAAC для создания глобального одноадресного адреса IPv6. Префикс: 2001:db8:acad:1::/64. Кстати, используйте мой локальный адрес связи fe80::1 в качестве шлюза по умолчанию"

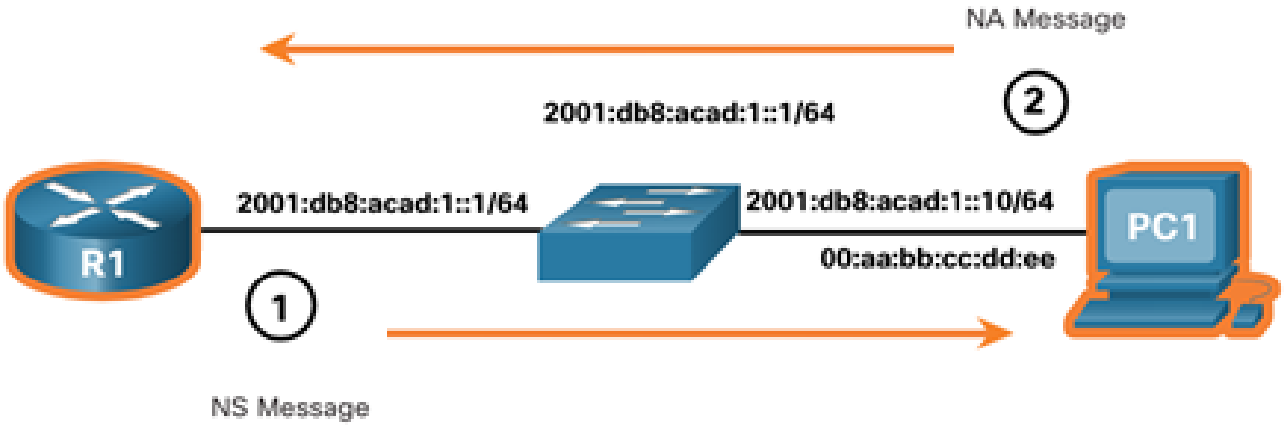
ICMPv6 сообщения



- Устройство, которому назначен глобальный адрес одноадресной рассылки IPv6 или локальной одноадресной рассылки, может выполнять обнаружение дубликатов адресов (DAD), чтобы убедиться, что адрес IPv6 уникален.
- Для проверки уникальности адреса устройство отправляет сообщение NS с собственным IPv6-адресом в качестве целевого.
- Если другое устройство в сети имеет этот адрес, оно ответит сообщением NA, уведомляющим отправляющее устройство о том, что адрес используется.

Примечание. Процесс обнаружения дублирующих адресов не обязателен, однако документ RFC 4861 рекомендует выполнять его для индивидуальных адресов.

ICMPv6 сообщения



- Для того, чтобы определить MAC-адрес назначения, R1 отправляет сообщение NS «Сообщение с запросом поиска соседей» на адрес запрашиваемого узла: R1 отправляет сообщение NS в 2001:db8:acad:1::10 с запросом его MAC-адреса.
- Сообщение включает известный (целевой) IPv6-адрес. Устройство с целевым IPv6-адресом отправляет в ответ сообщение NA «Сообщение об объявлении соседних узлов», содержащее его MAC-адрес Ethernet.

Тестирование и проверка

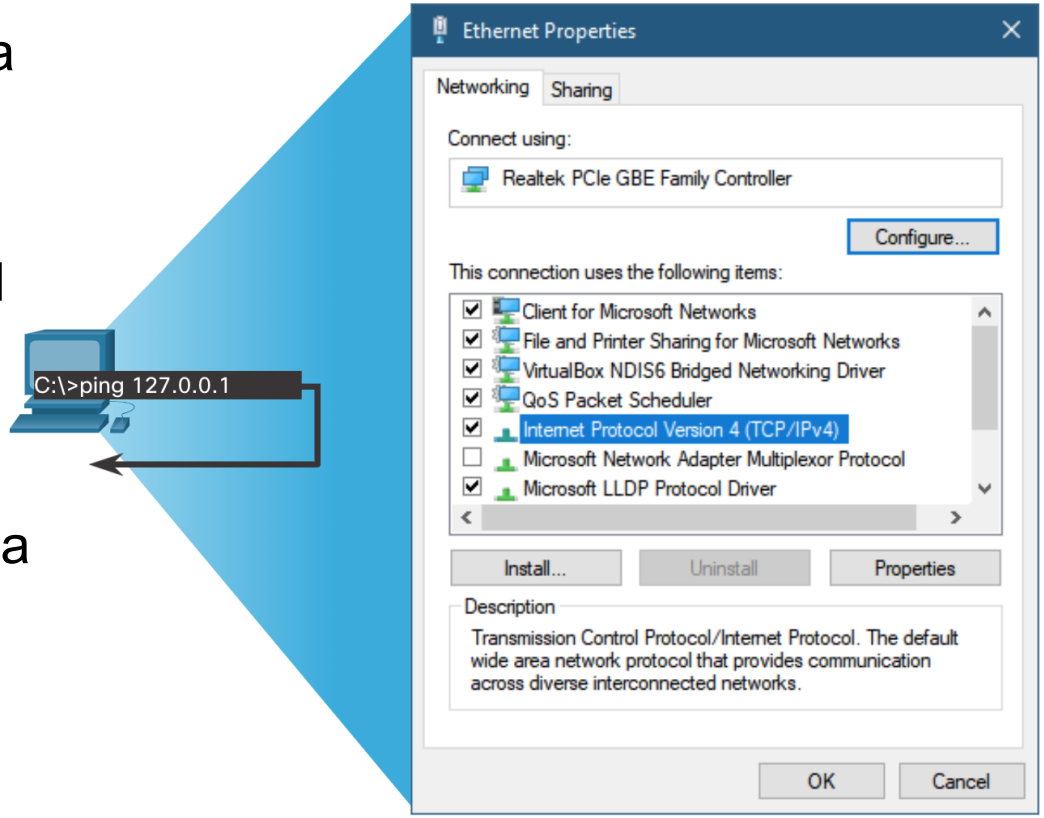
Ping — Тест подключения

- Команда **ping** — это утилита тестирования IPv4 и IPv6, которая использует сообщения эхо-запроса ICMP и эхо-ответа для проверки подключения между узлами и предоставляет сводную информацию, включающую в себя степень успеха и среднее время передачи туда и обратно до места назначения.
- Если в течение этого интервала ответ не получен, команда ping выдает сообщение об отсутствии ответа.
- Обычно для первого эхо-запроса требуется выполнить разрешение адреса (ARP или ND) перед отправкой эхо-запроса ICMP.

```
S1#ping 192.168.20.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.20.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 0/0/1 ms
```

Ping интерфейса loopback

- Ping может использоваться для проверки внутренней конфигурации IPv4 или IPv6 на локальном хосте. Для выполнения этой проверки отправим ping на адрес loopback 127.0.0.1 для IPv4 (::1 для IPv6).
- Ответ от адреса 127.0.0.1 для IPv4 или :: 1 для IPv6 означает, что IP-сеть настроена на хосте правильно.
- Если мы получаем сообщение об ошибке, это означает, что протокол TCP/IP не работает на данном хосте.



Команда Traceroute: тестирование пути

Traceroute (tracert)

- Создаёт список переходов, успешно выполненных на пути
- Если данные достигают места назначения, команда трассировки создаёт список интерфейсов для каждого маршрутизатора на пути между узлами
- Если при передаче данных произошёл сбой на любом из переходов на пути, то адрес последнего маршрутизатора, от которого получен отклик трассировки, может указывать место, где имеется проблема или ограничения, налагаемые системой безопасности
- Предоставляет время прохождения сигнала туда и обратно для каждого перехода на пути и сообщает, когда переход не отправляет отклик

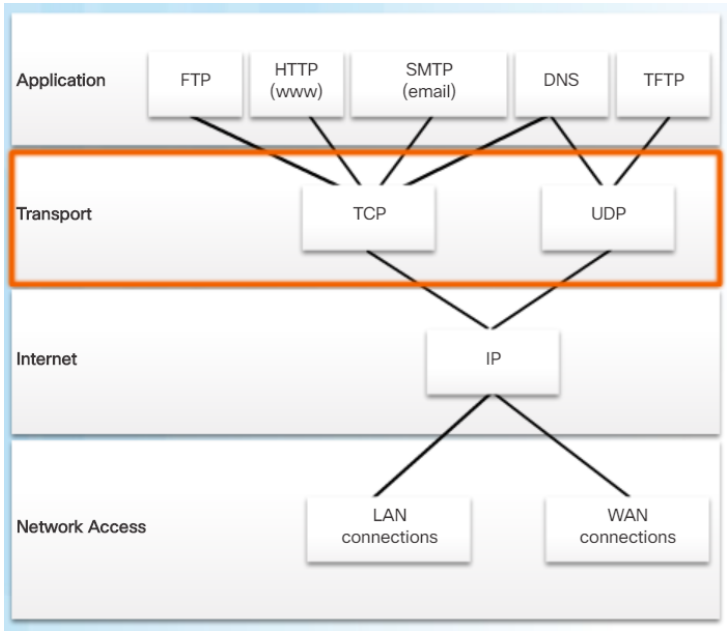
```
R1#traceroute 192.168.40.2
Type escape sequence to abort.
Tracing the route to 192.168.40.2

  1   192.168.10.2    1 msec    0 msec    0 msec
  2   192.168.20.2    2 msec    1 msec    0 msec
  3   192.168.30.2    1 msec    0 msec    0 msec
  4   192.168.40.2    0 msec    0 msec    0 msec
```

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ | НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Транспортный уровень

Введение в сетевые технологии

Роль транспортного уровня



Роль транспортного уровня

Транспортный уровень отвечает за установление временного соединения между двумя приложениями и передачу данных между ними. Для этих целей TCP/IP использует два протокола:

- **протокол управления передачей (TCP)**
- **протокол UDP**

Основные функции протоколов транспортного уровня

- *Отслеживание отдельных сеансов передачи данных между приложениями* на узле-источнике и узле назначения
- *Сегментирование данных* для управления ими, а также для их повторной компоновки в потоки данных приложений на узле назначения
- *Определение соответствующего приложения для каждого потока* обмена данными

Мультиплексирование сеансов связи

Сегментирование данных

- Обеспечивает возможность чередования (мультиплексирования) параллельных потоков передачи данных от большого числа пользователей в пределах одной сети.
- Предоставляет средства как для отправки, так и для получения данных при работе нескольких приложений.
- В каждый сегмент добавляется заголовок для его идентификации.

Надёжность транспортного уровня

Различные приложения предъявляют разные требования к надёжности транспортного уровня

TCP/IP предоставляет два протокола транспортного уровня:
TCP и UDP

Протокол управления передачей (TCP)

- Обеспечивает надёжную доставку всех данных в место назначения.
- Использует подтверждённую доставку и другие процессы, обеспечивающие успешную передачу данных.
- Предъявляет расширенные требования к сети — повышенную нагрузку.

Протокол UDP

- Предоставляет только базовые функции доставки — надёжность не гарантируется.
- Меньшая нагрузка.

TCP или UDP

- Выгода от надежности и нагрузка на сеть, которую она вызывает, находятся между собой в оптимальном соотношении.
- При выборе протокола передачи данных разработчики приложений учитывают требования, предъявляемые приложениями.

Функции протокола TCP

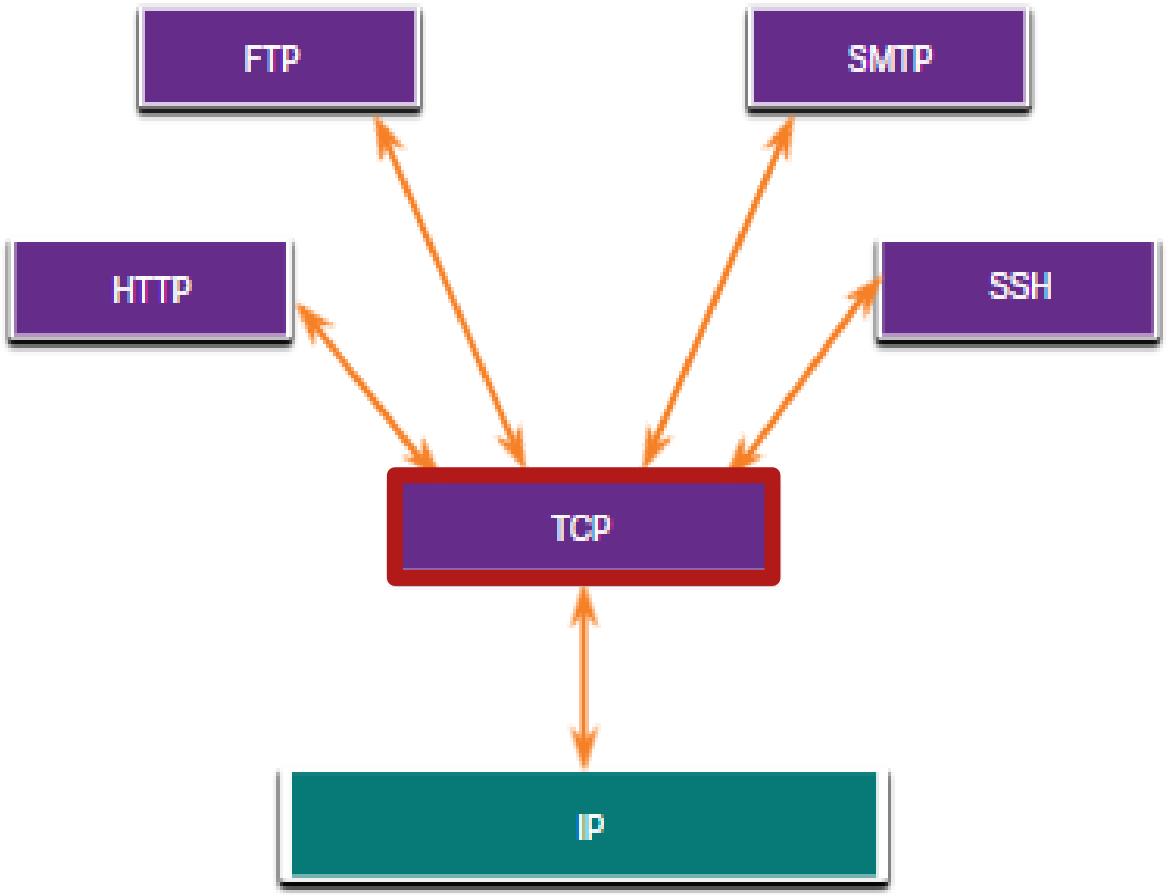
- **Установление сессии** - Перед пересылкой любого трафика протокол с установлением соединения согласовывает и настраивает постоянное соединение (или сеанс) между устройством источника и устройством назначения.
- **Гарантия надежной доставки** – При передаче по сети один из сегментов может быть поврежден или полностью утрачен. TCP - Обеспечивает гарантированную доставку на узел назначения всех без исключения сегментов данных, отправленных источником
- **Обеспечение доставки в нужном порядке** - Поскольку в сетях могут использоваться несколько маршрутов с разными скоростями передачи информации, в процессе доставки данных их порядок может измениться.
- **Управление потоком передачи данных** - Ресурсы сетевых узлов, такие как память или вычислительные мощности, ограничены. Когда протокол TCP получает информацию о том, что эти ресурсы используются слишком активно, он может потребовать от отправляющего приложения снизить скорость потока данных.

Обзор протокола TCP

Заголовок протокола TCP

<table>
<tr><th colspan="4">Сегмент TCP (20 байт)</th></tr>
<tr><th>Бит (0)</th><th colspan="2">Бит(15)</th><th>Бит(16)</th><th>Бит(31)</th></tr>
<tr><td colspan="3">Порт источника (16) идентификация исходного приложения по номеру порта</td><td colspan="2">Порт назначения (16) идентификация приложения назначения по номеру порта</td></tr>
<tr><td colspan="5">Порядковый номер (32) для пересборки данных</td></tr>
<tr><td colspan="5">Номер подтверждения (32) для указания того, что данные получены и ожидается следующий байт от источника</td></tr>
<tr><td>Длина заголовка (4) - «смещение данных», указывает длину заголовка сегмента TCP</td><td>Зарезервировано (6)</td><td>Управляющие биты (6) двоичные коды или флаги, которые указывают назначение и функцию сегмента TCP</td><td colspan="2">Окно (16) для указания количества байтов, которые могут быть приняты</td></tr>
<tr><td colspan="3">Контрольная сумма (16) для проверки ошибок заголовка и данных датаграммы</td><td colspan="2">Срочность (16) для указания срочности содержащихся данных</td></tr>
<tr><td colspan="5">Опции (0 или 32, если имеются)</td></tr>
<tr><td colspan="5">Данные уровня приложений (переменный размер)</td></tr>
</table>

Приложения, использующие протокол TCP



Функции протокола UDP

Протокол UDP

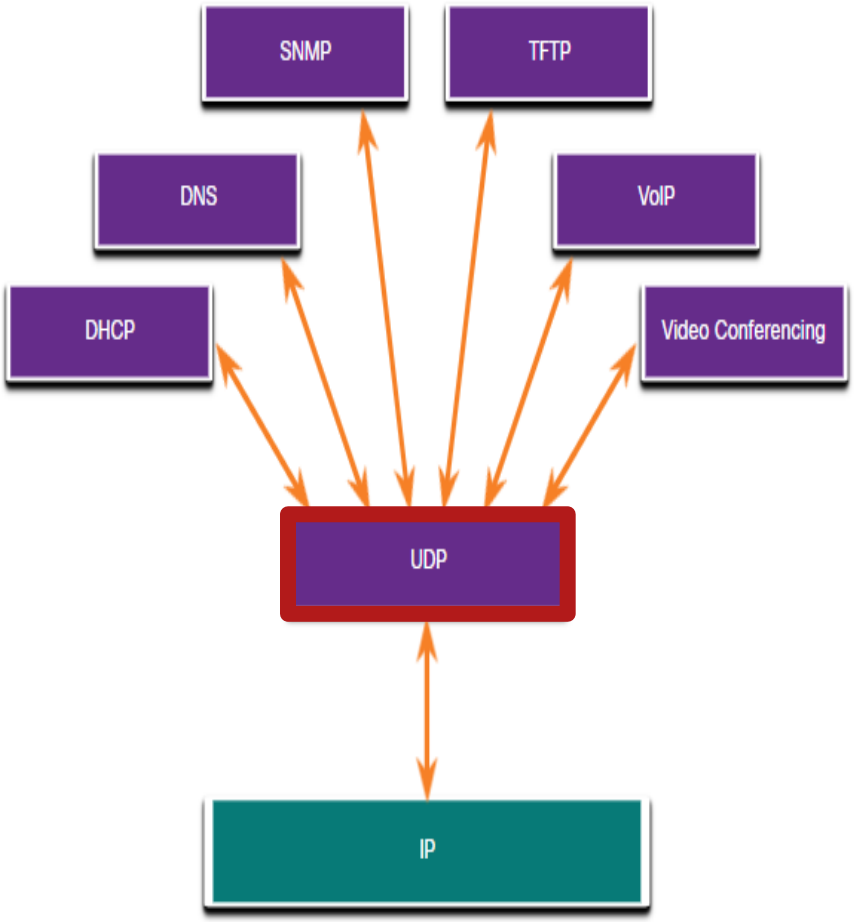
- Без установления соединения
- Ненадёжная доставка
- Без упорядоченного восстановления данных
- Без контроля потока
- Протокол без определенного состояния

Заголовок протокола UDP

Поле заголовка UDP	Описание
Порт источника	16-битное поле, используемое для идентификации исходного приложения по номеру порта.
Порт назначения	16-битное поле, используемое для идентификации приложения назначения по номеру порта.
Длина	16-битное поле, указывающее длину заголовка датаграммы UDP.
Контрольная сумма	16-битное поле, используемое для проверки ошибок заголовка и данных датаграммы.

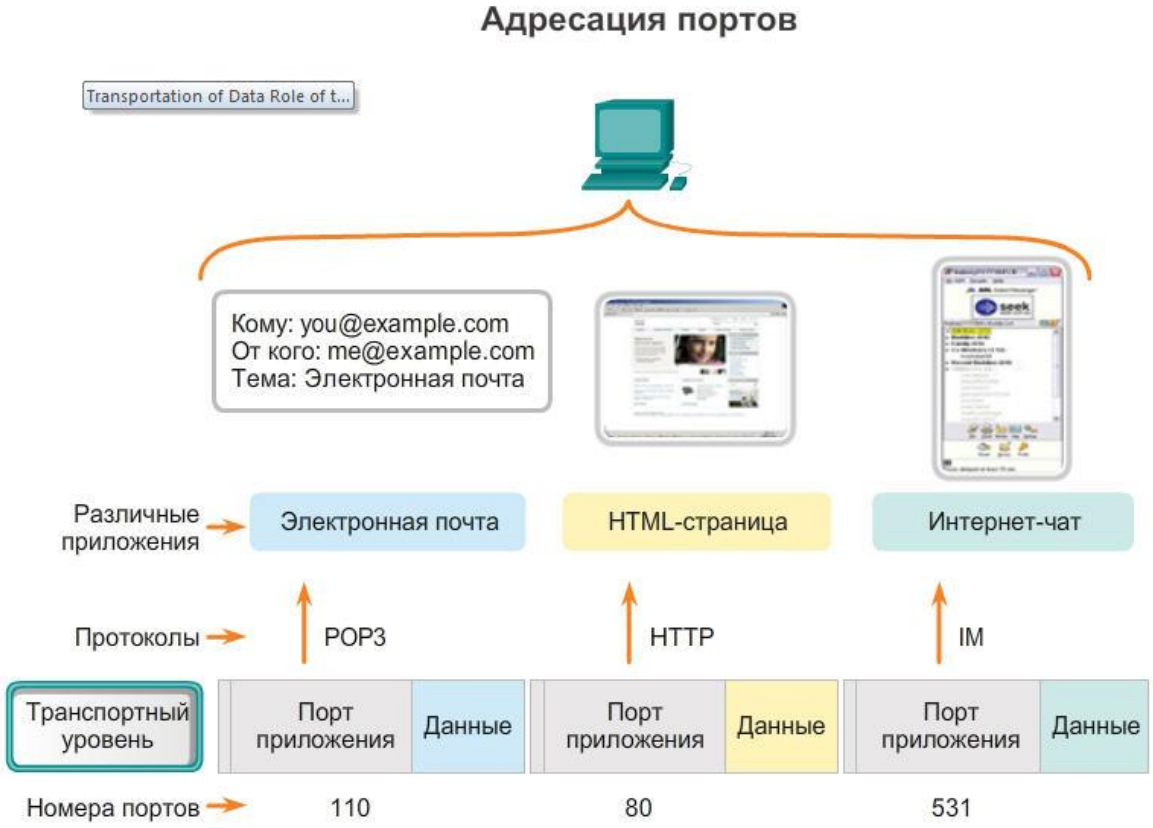
Приложения, использующие протокол UDP

- Мультимедийные приложения и передача видео в режиме реального времени. Такие приложения допускают небольшие потери данных, но не допускают задержки (либо минимальные). Например, VoIP и потоковое видео.
- Простые приложения запросов и ответов. Приложения с операциями, где хост отправляет запрос и может получить или не получить ответ. Например, DNS и DHCP.
- Приложения, самостоятельно обеспечивающие надежность передачи данных, — ненаправленный обмен данными, при котором управление потоком, обнаружение ошибок, отправка подтверждений и восстановление после сбоев не требуются или выполняются самим приложением. Например, SNMP и TFTP.



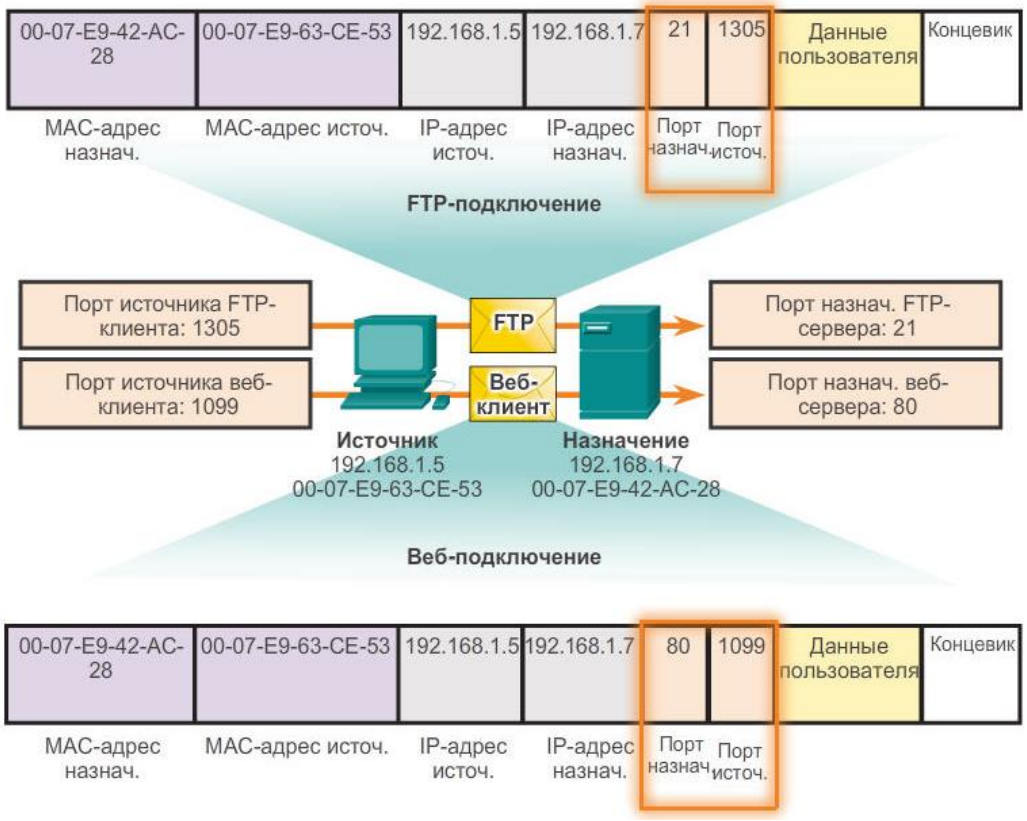
Разделение нескольких каналов обмена данными

В протоколах TCP и UDP используются номера портов, чтобы различать приложения.



Данные различных приложений направляются соответствующему приложению, так как каждому приложению присваивается уникальный номер порта.

Адресация портов в TCP и UDP



Сочетание номера порта транспортного уровня и IP-адреса узла сетевого уровня однозначно идентифицирует конкретный процесс приложения на конкретном физическом узле. Такая совокупность параметров называется ***сокетом***. Пара сокетов, состоящая из IP-адресов и номеров портов источника и адресата, также уникальна и идентифицирует конкретную операцию обмена данными между двумя узлами.

Сокет веб-клиента может иметь следующий вид, где 1099 — это номер порта источника: 192.168.1.5:1099
Сокет веб-сервера может иметь следующий вид: 192.168.1.7:80
Вместе эти два сокета образуют следующую пару: 192.168.1.5:1099, 192.168.1.7:80

Номера портов

Адресация портов в TCP и UDP

Номера портов

Диапазон номеров портов	Группа портов
от 0 до 1023	Общеизвестные порты
1024–49151	Зарегистрированные порты
49152–65535	Частные и/или динамические порты

Условные обозначения

Зарегистрированные порты TCP:
1863 MSN Messenger
2000 Cisco SCCP (VoIP)
8008 Alternate HTTP
8080 Alternate HTTP

Общеизвестные порты TCP:
21 FTP
23 Telnet
25 SMTP
80 HTTP
143 IMAP
194 Internet Relay Chat (IRC)
443 Secure HTTP (HTTPS)

Условные обозначения

Зарегистрированные порты UDP:
1812 Протокол аутентификации RADIUS
5004 RTP (Протокол передачи голоса и видео)
5040 SIP (VoIP)

Общеизвестные порты UDP:
69 TFTP
520 RIP

Условные обозначения

Зарегистрированные общие порты TCP/UDP:
1433 MS SQL
2948 WAP (MMS)

Известные общие порты TCP/UDP:
53 DNS
161 SNMP
531 AOL Instant Messenger, IRC

Адресация портов в TCP и UDP

Netstat

- Используется для проверки открытых соединений по TCP, которые работают на сетевом узле

```
 C:\> netstat
Активные соединения
Proto Local Address          Foreign Address          State
TCP   192.168.1.124:3126     192.168.0.2:netbios-ssn ESTABLISHED
TCP   192.168.1.124:3158     207.138.126.152:http    ESTABLISHED
TCP   192.168.1.124:3159     207.138.126.169:http    ESTABLISHED
TCP   192.168.1.124:3160     207.138.126.169:http    ESTABLISHED
TCP   192.168.1.124:3161     sc.msn.com:http         ESTABLISHED
TCP   192.168.1.124:3166     www.cisco.com:http      ESTABLISHED
```

Процессы TCP-сервера

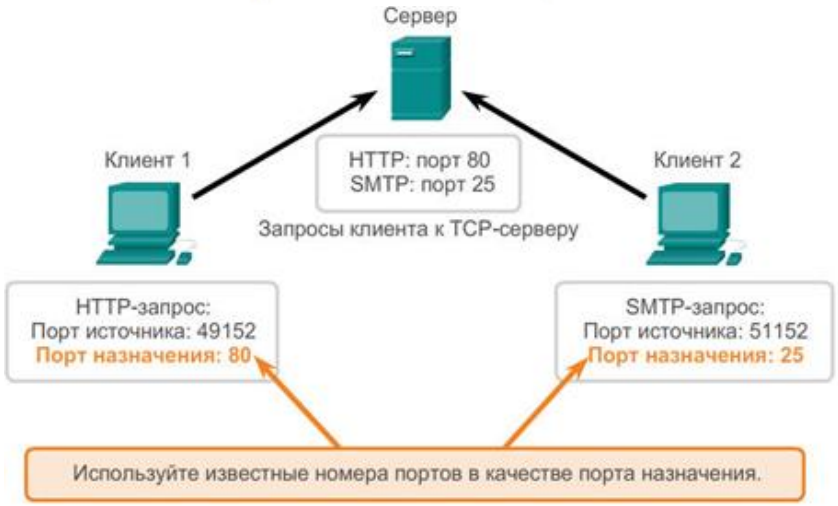
Каждый процесс приложения, работающий на сервере, использует номер порта

- **Не допускается** использование двумя различными службами **на одном и том же** сервере **одного и того же порта** с **одинаковым** протоколом транспортного уровня.
- Активное серверное приложение, которому присвоен какой-либо определенный порт, считается открытым, что означает, что транспортный уровень может принимать и обрабатывать сегменты, направляемые на этот порт.
- Любой входящий запрос, который адресован правильному сокету, будет принят, а данные будут переданы приложению сервера.

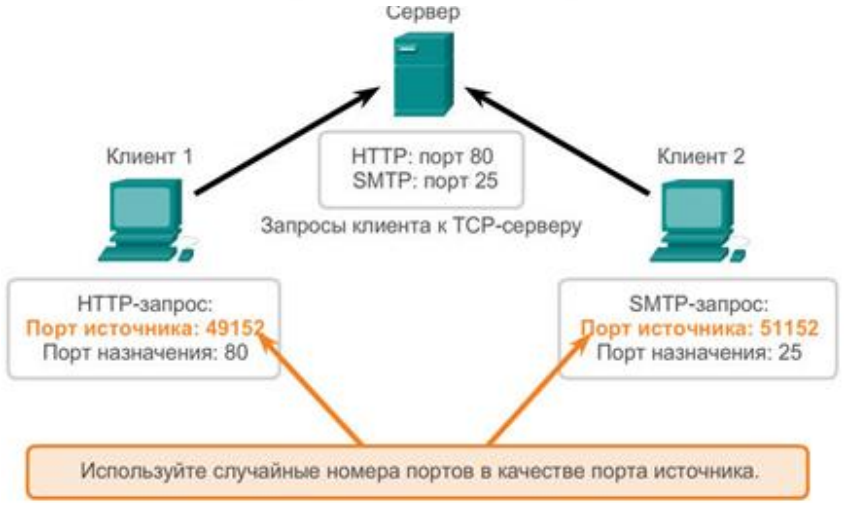
Обмен данными по протоколу TCP

Процессы TCP-сервера

Порты назначения запроса



Порты источника запроса



Порты назначения ответа

Порты источника ответа

HTTP-запрос:
Порт источника: 49152
Порт назначения: 80

SMTP-запрос:
Порт источника: 51152
Порт назначения: 25

HTTP-запрос:
Порт источника: 49152
Порт назначения: 80

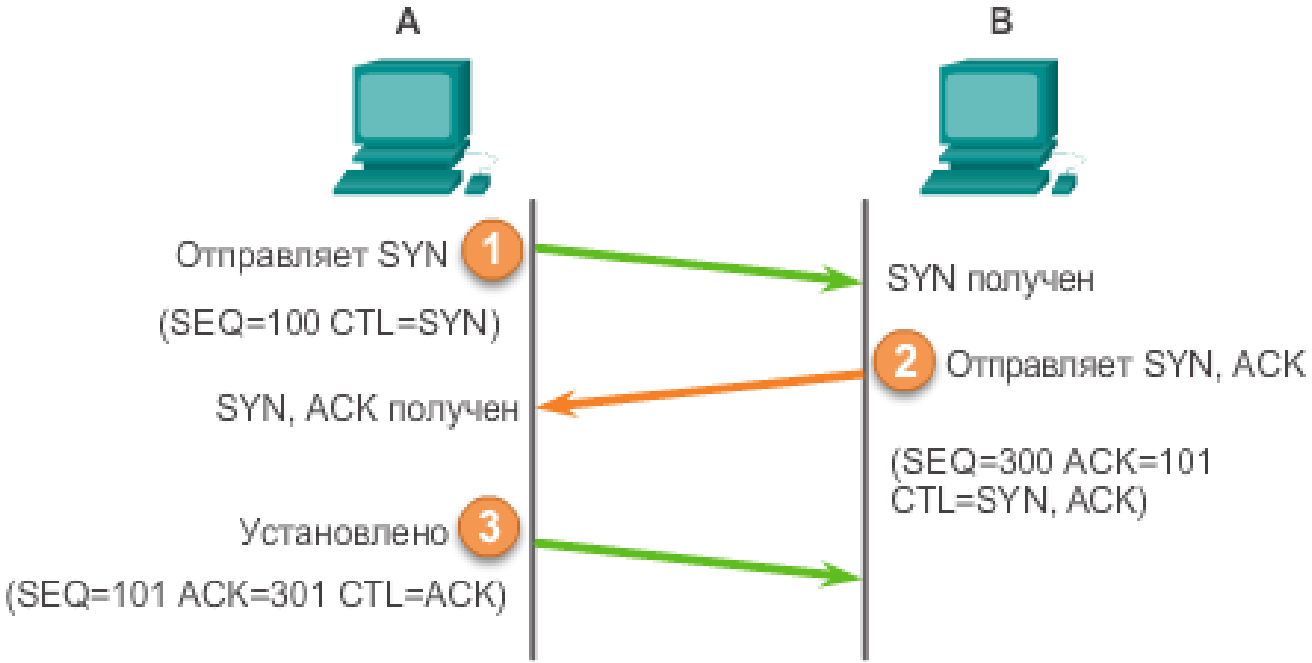
SMTP-запрос:
Порт источника: 51152
Порт назначения: 25

Установление TCP-соединения

Шаг 1. Инициирующий клиент запрашивает сеанс связи «клиент-сервер» с сервером.

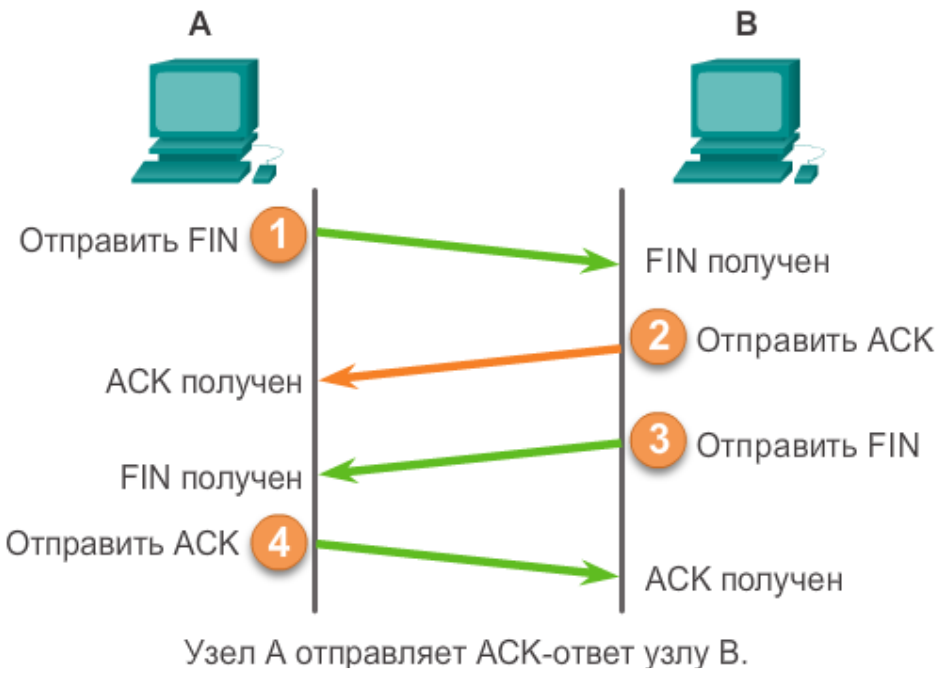
Шаг 2. Сервер подтверждает сеанс обмена данными «клиент-сервер» и запрашивает сеанс обмена данными «сервер-клиент».

Шаг 3. Инициирующий клиент подтверждает сеанс связи «сервер-клиент».



Прекращение TCP-соединения

- Шаг 1. Когда у клиента больше нет данных для отправки в потоке, он отправляет сегмент с установленным флагом FIN.
- Шаг 2. Сервер отправляет подтверждение ACK, чтобы подтвердить получение FIN для завершения сеанса связи «клиент-сервер».
- Шаг 3. Сервер отправляет FIN клиенту, чтобы завершить сеанс «сервер-клиент».
- Шаг 4. Клиент отправляет в ответ сегмент ACK для подтверждения получения сегмента FIN от сервера.



Узел A отправляет ACK-ответ узлу B.

Анализ трехстороннего квитирования TCP

Функции процесса трехстороннего рукопожатия:

- Определяет, присутствует ли в сети устройство назначения.
- Проверяет, имеется ли на устройстве назначения активная служба и принимает ли она запросы на номер порта назначения, который инициирующий клиент планирует использовать.
- Информирует устройство назначения, что клиент источника планирует установить сеанс связи на этом номере порта.

По завершении обмена данными все сеансы закрываются, а соединение прерывается. Механизмы подключения и осуществления сеанса связи включают в себя функции TCP, обеспечивающие надежность.

Анализ трехстороннего квитирования TCP

Шесть контрольных битовых флагов выглядят следующим образом:

URG - флаг «Указатель важности»

ACK - Флаг подтверждения, используемый при установке соединения и завершении сеанса

PSH - флаг "Push"

RST- Флаг RST используется для сброса соединения при возникновении ошибки или в случае превышения времени ожидания.

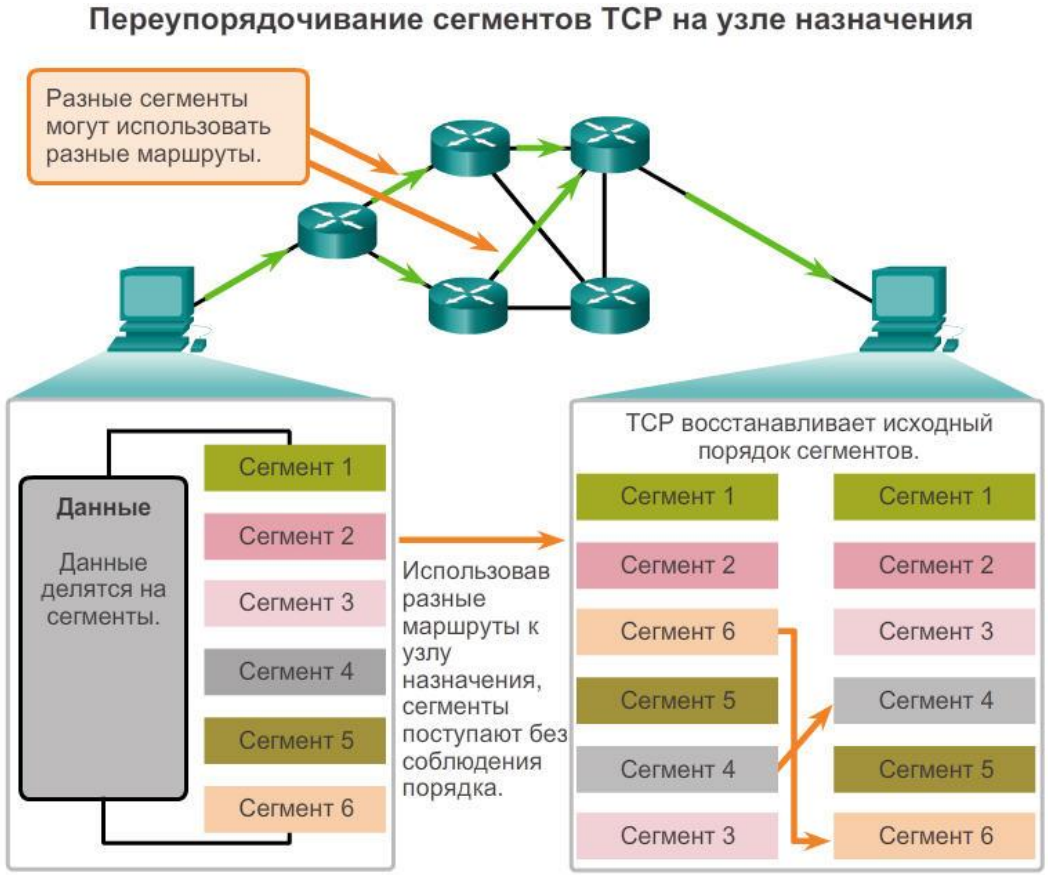
SYN - Синхронизировать порядковые номера, используемые при установке соединения

FIN - больше нет данных от отправителя и используется при завершении сеанса

Сегмент TCP (20 байт)			
Бит (0)		**Бит(15)** **Бит(16)**	**Бит(31)**
Порт источника (16) идентификация исходного приложения по номеру порта			**Порт назначения (16)** идентификация приложения назначения по номеру порта
Порядковый номер (32) для пересборки данных			
Номер подтверждения (32) для указания того, что данные получены и ожидается следующий байт от источника			
Длина заголовка (4) - «смещение данных», указывает длину заголовка сегмента TCP	**Зарезервировано (6)**	**Управляющие биты (6)** двоичные коды или флаги, которые указывают назначение и функцию сегмента TCP	**Окно (16)** для указания количества байтов, которые могут быть приняты
Контрольная сумма (16) для проверки ошибок заголовка и данных датаграммы			**Срочность (16)** для указания срочности содержащихся данных
Опции (0 или 32, если имеются)			
Данные уровня приложений (переменный размер)			

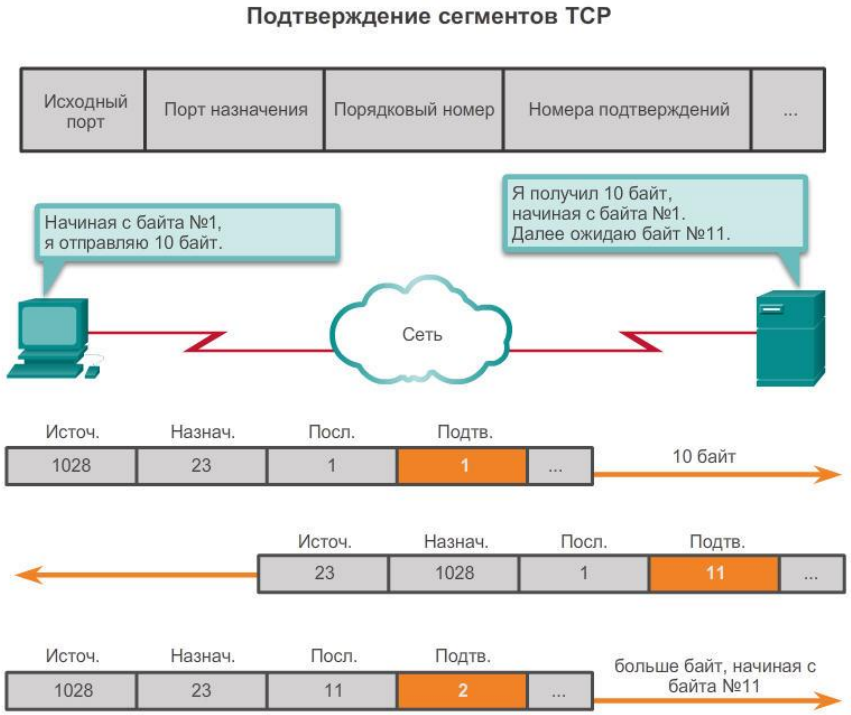
Надёжность TCP — упорядоченная доставка

Номера последовательности, используемые для того, чтобы оставлять сегменты в исходном порядке



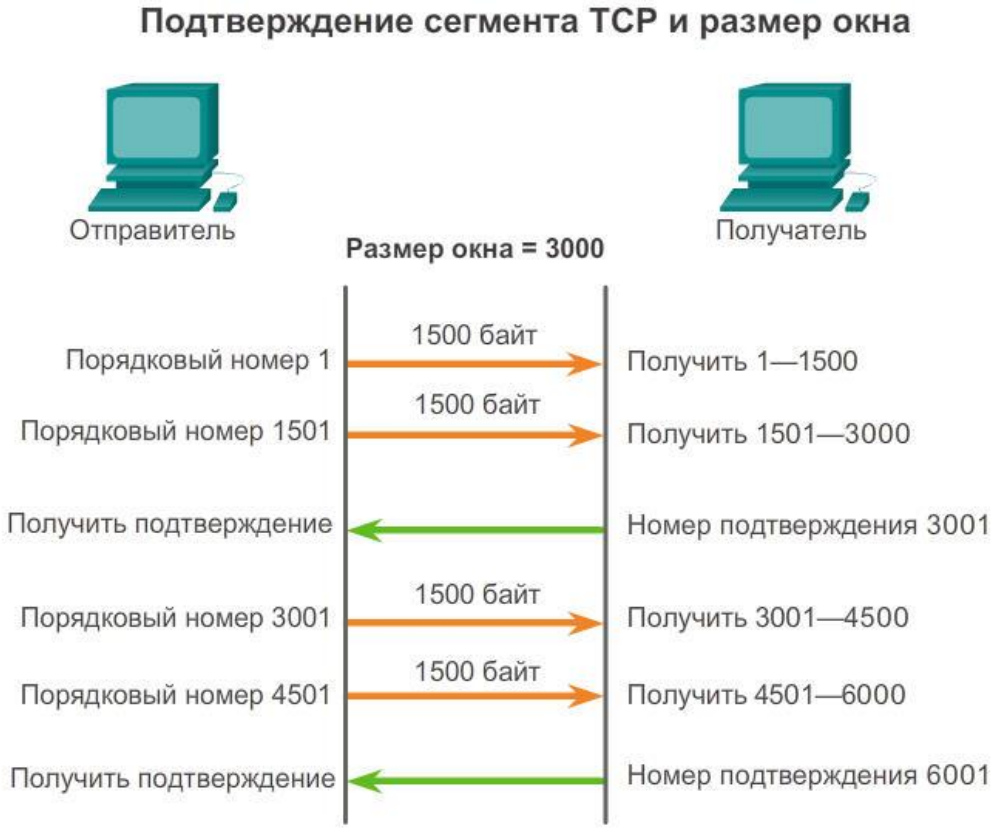
Надёжность TCP — подтверждение и размер окна

Для подтверждения получения используются одновременно номер последовательности и номер подтверждения.



Размер окна — объём данных, которые источник может передать до обязательного получения подтверждения.

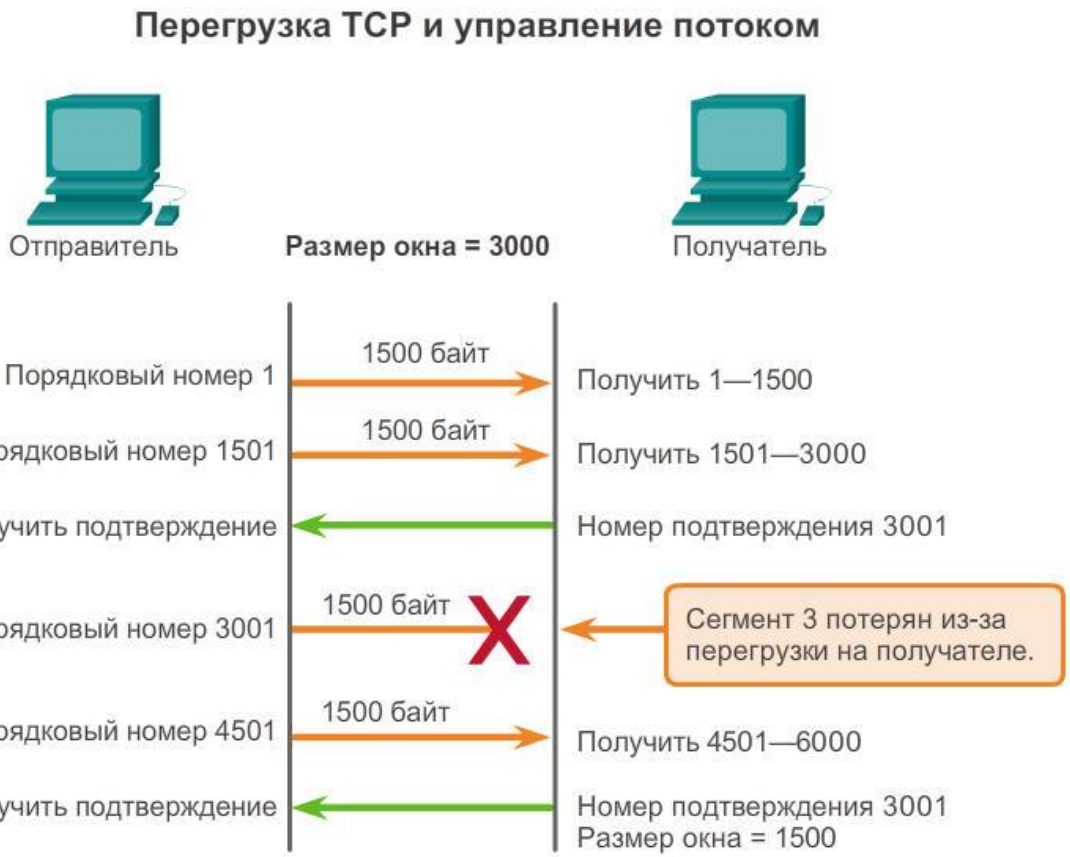
Размер окна и подтверждения



С помощью **размера окна** определяется количество байт, отправленных до того, как ожидается подтверждение.

Номер **подтверждения** — это номер следующего ожидаемого байта.

Управление потоком TCP — предотвращение заторов



Если из-за перегрузки сегменты будут потеряны, получатель подтвердит последний полученный последовательный сегмент, предоставив ответ в свёрнутом окне.

После периода передачи данных без потерь или ограничения ресурсов получатель начинает увеличивать поле окна, что уменьшает нагрузку на сеть, поскольку требуется отправка меньшего количества подтверждений. Размер окна продолжает увеличиваться до тех пор, пока не начнутся потери данных, в результате чего размер окна будет уменьшаться.

Низкая нагрузка UDP или надёжность

Протокол UDP

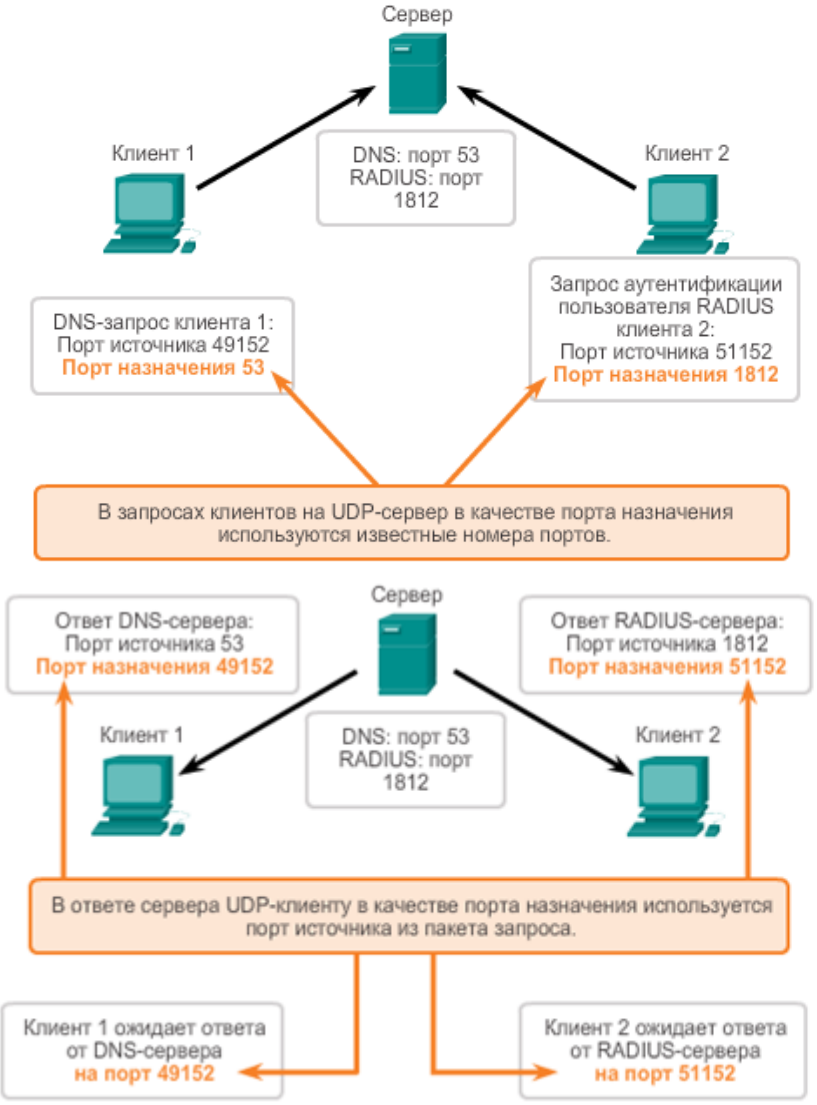
- Простой протокол, предоставляющий базовые функции транспортного уровня
- Не устанавливает соединение перед отправкой данных
- Обеспечивает передачу данных с меньшими накладными расходами, т.к. он имеет небольшой заголовок датаграммы и не обменивается управляющим трафиком
- Используется приложениями, которые допускают потерю незначительного объёма данных
- Используется приложениями, для которых не допускаются задержки

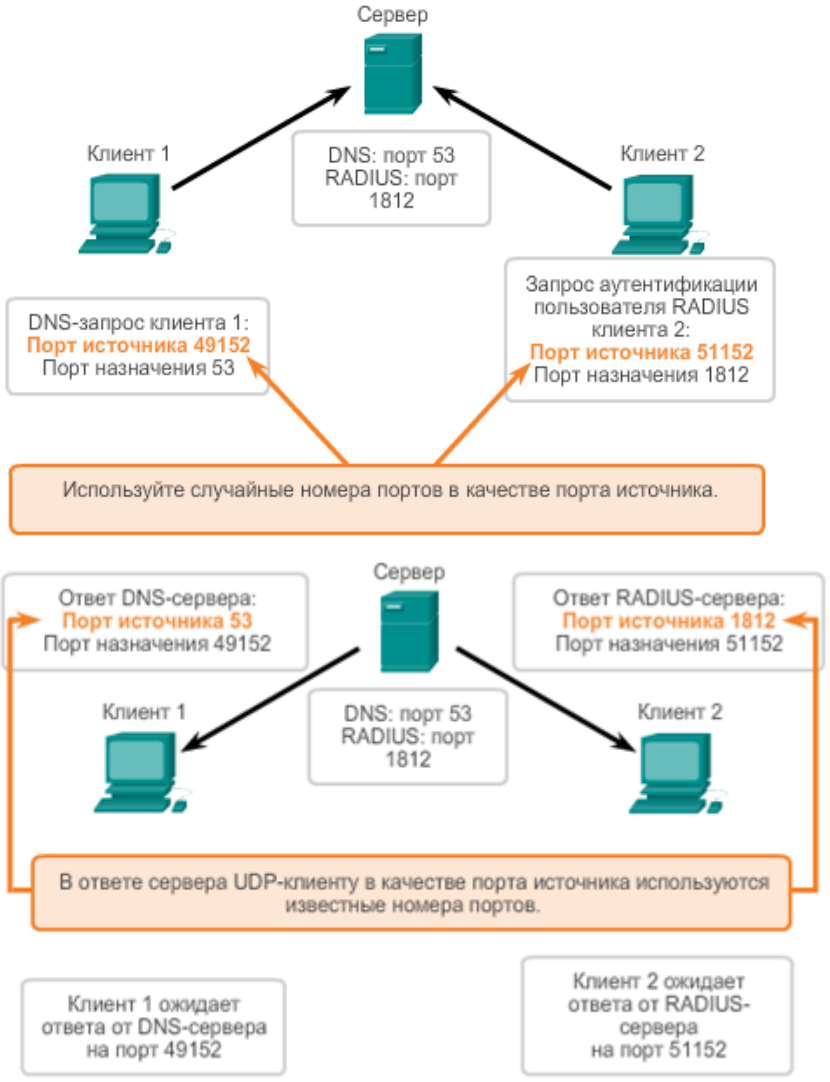
Обмен данными по UDP

Разбор датаграммы

Обмен данными по UDP

Процессы и запросы UDP-сервера





МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ | НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Уровень приложений

Введение в сетевые технологии

Уровень приложений

DNS	**Протокол преобразования имён интернет-доменов в IP адреса**
HTTP	**Протокол передачи гипертекста**
SMTP	**Простой протокол передачи эл. почты** для передачи сообщений и вложений электронной почты
POP3	**Почтовый протокол** используется почтовыми клиентами для получения электронной почты с удалённого сервера
DHCP	**Протокол динамической конфигурации узла** для назначения узлу IP-адреса, маски подсети, шлюза по умолчанию и DNS-сервера
FTP	**Протокол передачи файлов** для интерактивной передачи файлов между системами
IMAP	**Протокол доступа к сообщениям в Интернете**
TFTP	**Простой протокол передачи файлов** для активной передачи файлов без установления соединения

Уровень представления и сеансовый уровень

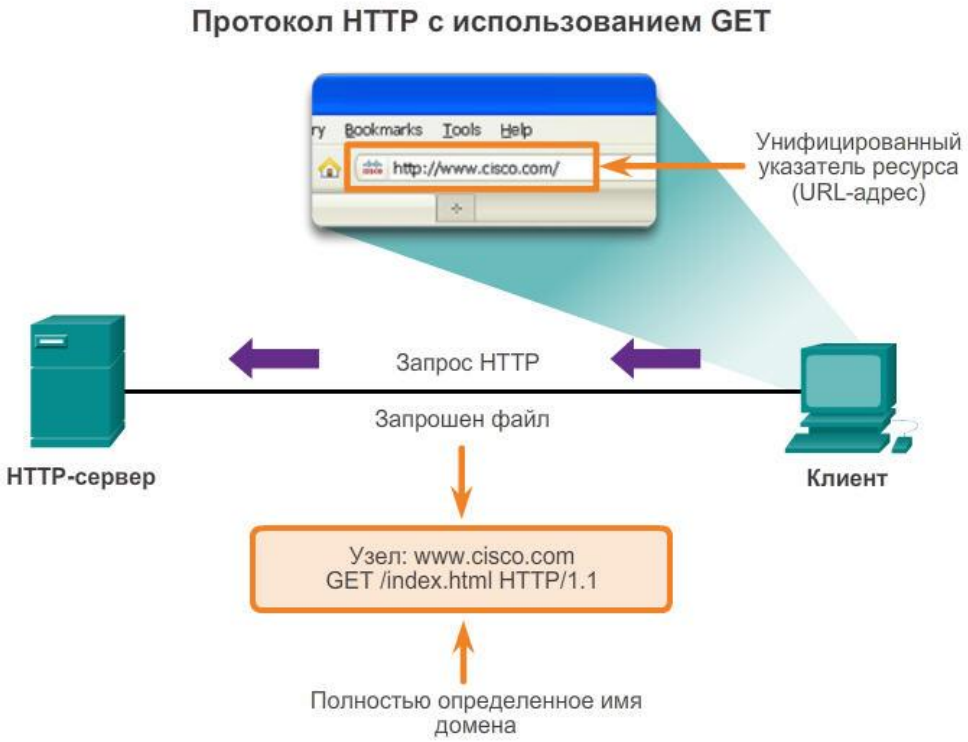
На ***уровне представления*** задействованы три основные функции:

- **кодирование и преобразование** данных уровня приложений;
- **сжатие** данных;
- **шифрование** данных для передачи и их расшифровка после получения по адресу назначения.

Сеансовый уровень

- Функции сеансового уровня создают и обеспечивают диалоги между исходными и конечными приложениями
- Сеансовый уровень обрабатывает обмен данными для запуска диалогов, поддержания их активности и перезапуска сеансов

Протокол передачи гипертекста HTTP и язык разметки HTML



Шаг 1. Сначала браузер интерпретирует три части URL-адреса:

1. **http** (протокол или схема)
2. www.cisco.com (имя сервера)
3. **index.html** (имя конкретного запрашиваемого файла)

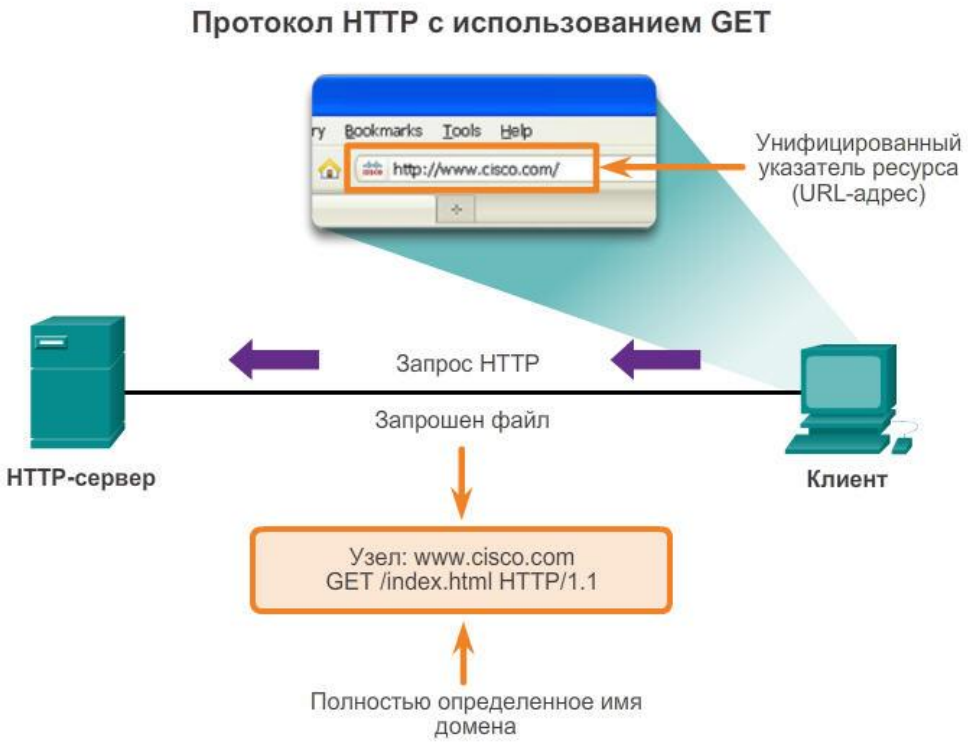
Шаг 2. Браузер сверяется с сервером имён и выполняет преобразование имени www.cisco.com в числовой адрес

В соответствии с требованиями протокола HTTP на сервер отправляется запрос **GET** и запрашивается файл **index.html**

Шаг 3. В ответ на запрос сервер отправляет в браузер HTML-код для этой веб-страницы.

Шаг 4. Браузер декодирует HTML-код и форматирует страницу в окне браузера.

Протоколы передачи гипертекста HTTP и HTTPS



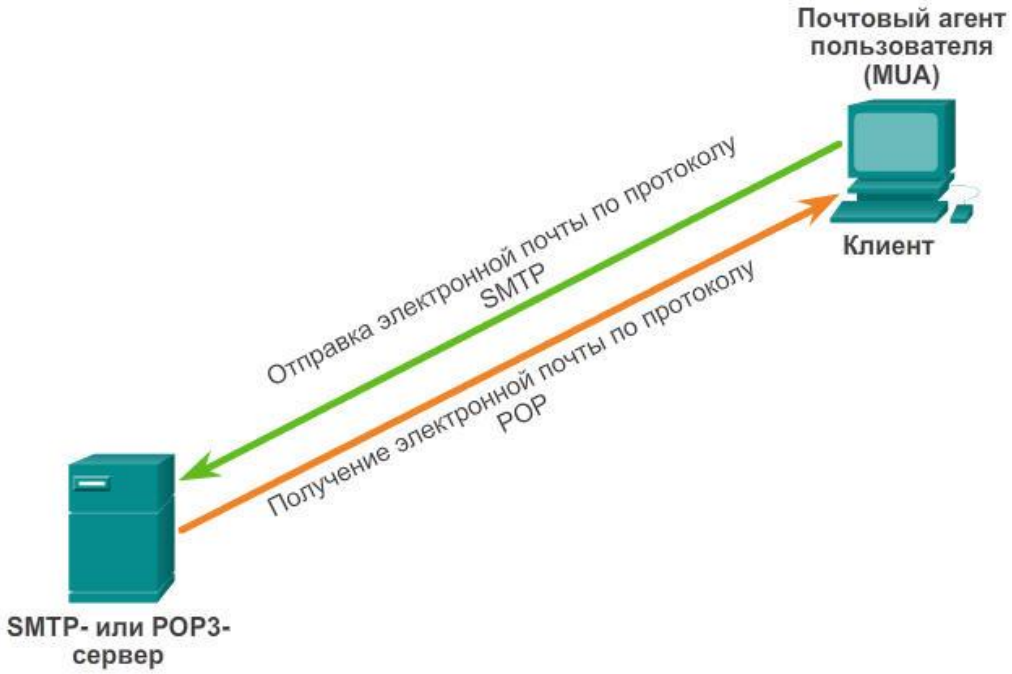
Примечание: HTTP не является безопасным протоколом. Для безопасных сообщений, отправляемых через Интернет, следует использовать HTTPS.

HTTP - это протокол запроса/ответа, который определяет типы сообщений, используемые для этой связи.

Три основных типа сообщений: GET, POST и PUT:

- **GET** - Это запрос данных клиентом. Клиент (веб-обозреватель) отправляет сообщение GET веб-серверу, чтобы запросить HTML-страницы.
- **POST** - Отправляет на веб-сервер файлы данных. Например, если пользователь вводит данные в форму, которая встроена в веб-страницу, веб-серверу отправляется сообщение **POST**.
- **PUT** - Выгружает на веб-сервер ресурсы и контент, например изображения. Например, если пользователь пытается отправить файл или изображение на веб-сайт, клиент отправляет серверу сообщение PUT с вложенным файлом или изображением.

Протоколы электронной почты SMTP, POP и IMAP

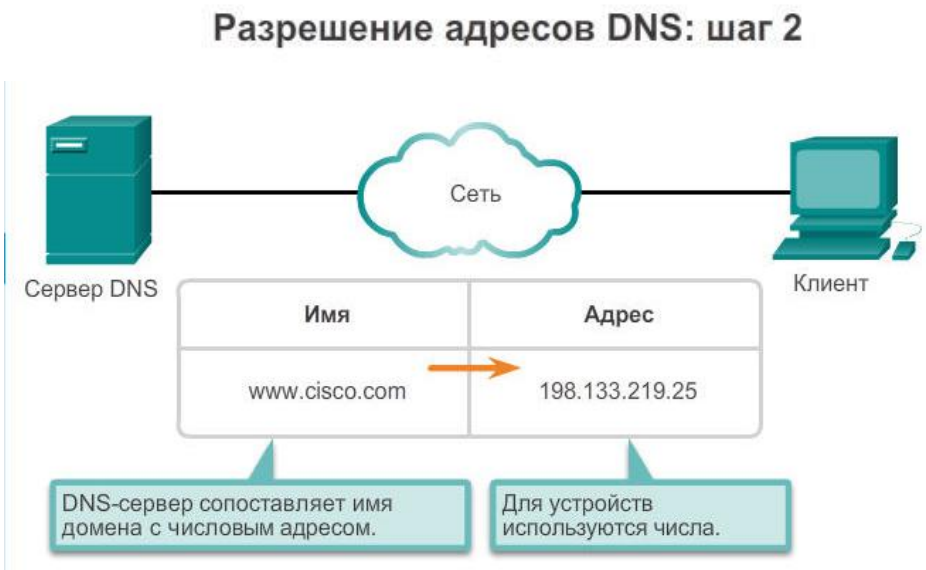


Клиент отправляет сообщения электронной почты на сервер по протоколу SMTP и получает сообщения по протоколу POP3.

- Как правило, почтовым клиентом для этих протоколов служит почтовый агент (**MUA**), который предоставляет функции обоих протоколов в рамках одного приложения
- **SMTP**: отправка сообщений с клиента на сервер. Используется порт 25.
- **POP**: получение сообщение от почтового сервера и удаление его с сервера. Используется порт 110.
- **IMAP**: протокол доступа к сообщениям в Интернете. Его отличие от POP состоит в том, что при подключении пользователя к серверу с поддержкой IMAP в клиентское приложение загружаются только копии сообщений. Исходные сообщения остаются на сервере до тех пор, пока они не будут удалены вручную.

Служба доменных имён

Протокол DNS служит для преобразования читаемых имён, используемых для ссылки на сетевые ресурсы



Предоставление служб IP-адресации

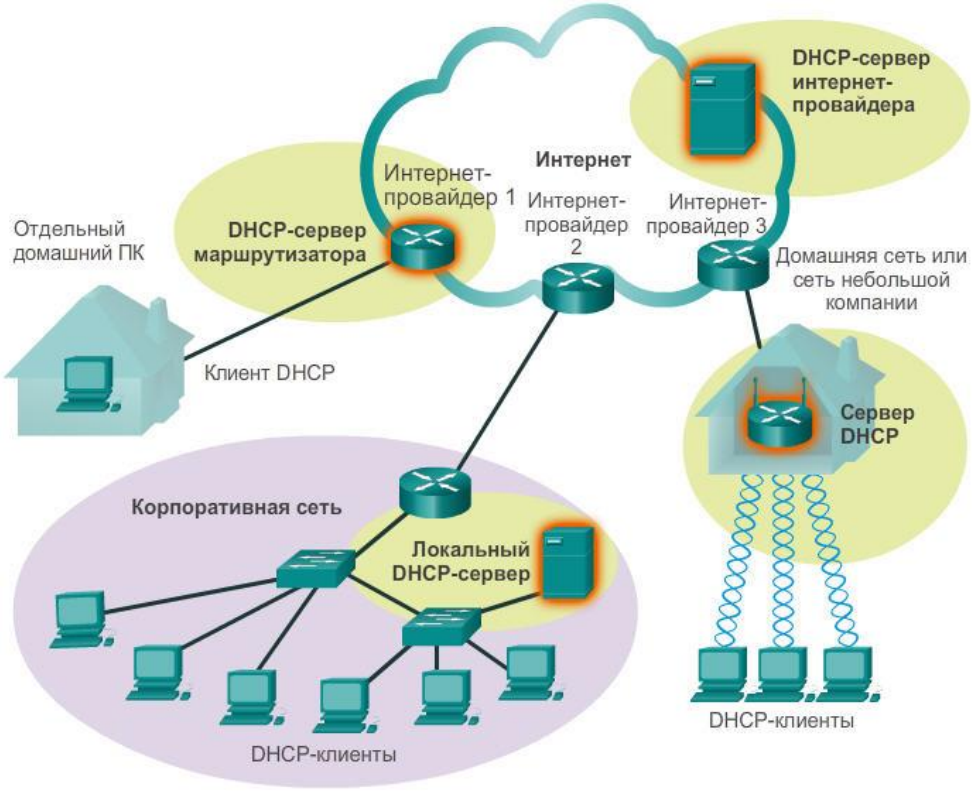
Команда nslookup

- Утилита операционной системы **nslookup** позволяет пользователям вручную отправлять запросы серверам имён на преобразование определённого имени узла
- Эту утилиту можно использовать для устранения неполадок при преобразовании имён и для проверки текущего статуса серверов имён

```
C:\Users> nslookup
Default Server:  dns-sj.cisco.com
Address:  171.70.168.183
> www.cisco.com
Server:  dns-sj.cisco.com
Address:  171.70.168.183
Name:    origin-www.cisco.com
Addresses:  2001:420:1101:1::a
          173.37.145.84
Aliases:  www.cisco.com
> cisco.netacad.net
Server:  dns-sj.cisco.com
Address:  171.70.168.183
Name:    cisco.netacad.net
Address:  72.163.6.223
>
```

Протокол динамической конфигурации узла DHCP

- DHCP позволяет узлу динамически получать IP-адрес
- DHCP-серверу отправляется запрос адреса, после чего сервер выбирает адрес из настроенного диапазона адресов (т. н. «пул») и передаёт его в «аренду» узлу на указанный период
- DHCP используется узлами общего назначения (например оконечные пользовательские устройства), а статическая адресация используется для сетевых устройств (например шлюзы, коммутаторы, серверы и принтеры)



Протокол передачи файлов FTP

- FTP позволяет передавать данные между клиентом и сервером
- В целях успешной передачи данных для FTP требуется два соединения между клиентом с сервером: одно для команд и ответов, другое — для передачи собственно данных.

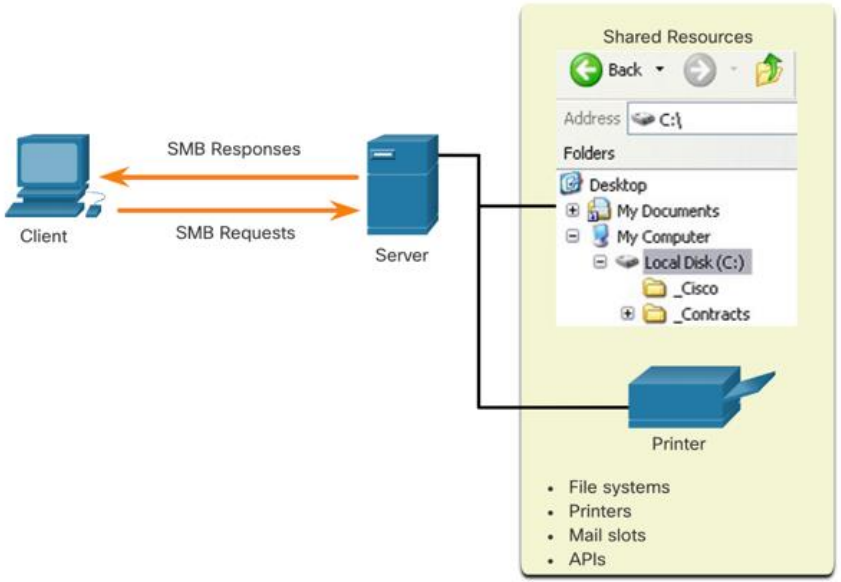
Протокол обмена файлами SMB

Протокол SMB — это клиент-серверный протокол обмена файлами. Серверы могут предоставлять свои ресурсы клиентам в сети.

Ниже приведены три функции сообщений SMB:

- Осуществлять запуск, аутентификацию и завершение сеансов
- Управлять доступом к файлам и принтерам
- Разрешать приложению отправлять сообщения на другое устройство и принимать их.

В отличие от обмена файлами по протоколу FTP, клиенты устанавливают долговременное подключение к серверам. После установки соединения пользователь может получить доступ к ресурсам на сервере аналогично доступу к ресурсам на локальном хосте.

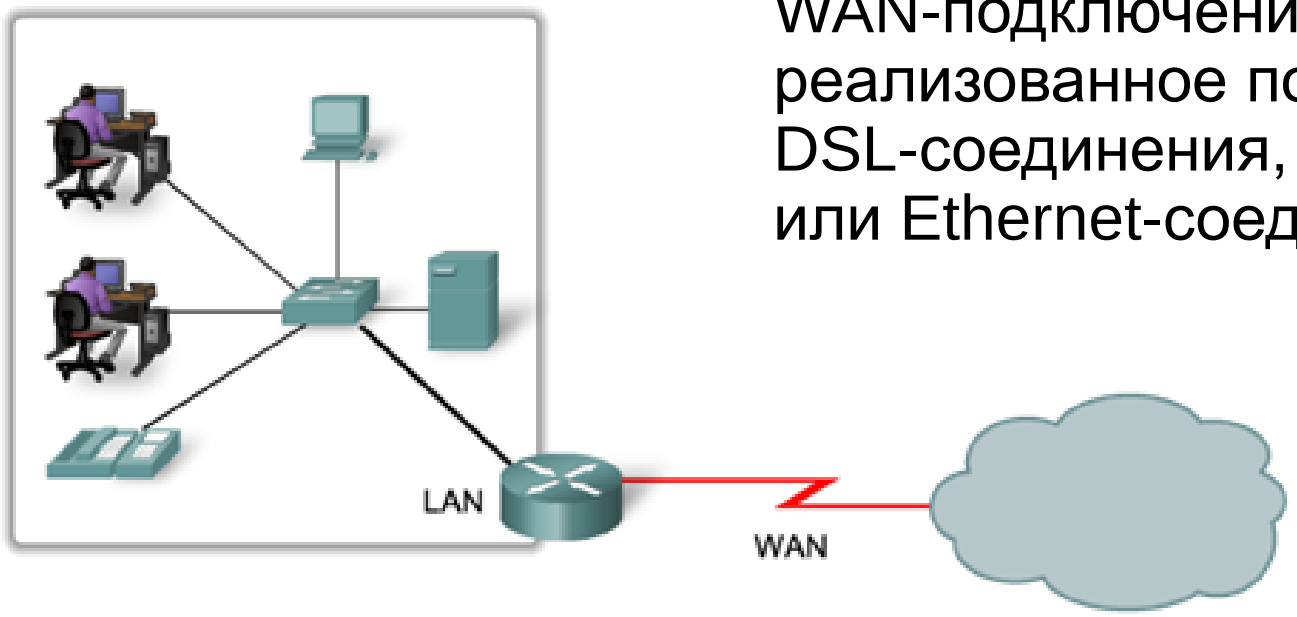


Организация небольшой сети

Введение в сетевые технологии

Топологии небольших сетей

- Типичная топология небольшой сети



- Небольшой сети обычно предусматривается одно WAN-подключение, реализованное посредством DSL-соединения, кабеля или Ethernet-соединения.

Выбор устройств для небольшой сети

- Факторы, которые следует учитывать при выборе промежуточных устройств

Объём затрат

Количество портов

Скорость

Возможности расширения/модульность

Возможности управления

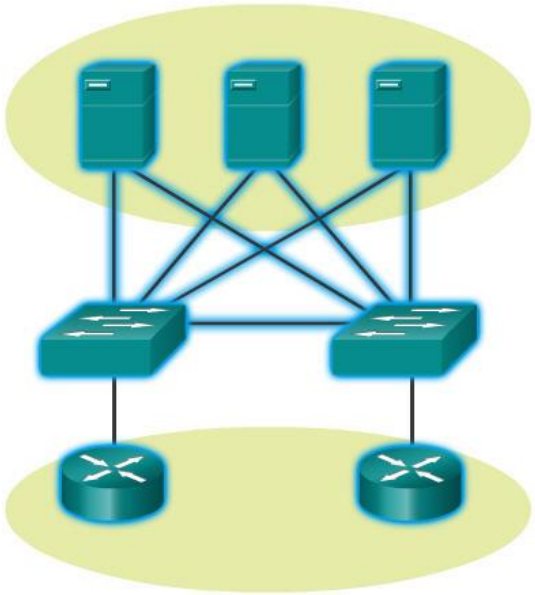
Адресация в небольших сетях

- Схему IP-адресации необходимо планировать, документировать и обслуживать с учётом типа устройств, получающих адрес.
- Примеры устройств, которые будут включены в проект IP-сети:
  - оконечные пользовательские устройства;
  - серверы и периферийные устройства;
  - узлы с доступом через Интернет;
  - промежуточные устройства.
- Спланированные схемы IP-адресации позволяют администратору:
  - отслеживать устройства и устранять неполадки;
  - контролировать доступ к ресурсам.

Резервирование в небольших сетях

- Резервирование позволяет устранить единые точки отказа.
- Также при этом повышается надёжность сети.

Резервирование для серверной фермы



Резервирование может быть достигнуто путем:

- установки дублирующего оборудования;
- предоставления дублирующих сетевых каналов для критически важных областей.

Рекомендации по проектированию небольших сетей

- В проект сети необходимо включить следующие пункты:

  обеспечение защиты файловых и почтовых серверов, имеющих центральное местоположение;

  защита расположения посредством физических и логических мер безопасности;

  резервирование в рамках серверной фермы;

  настройка резервных путей к серверам.

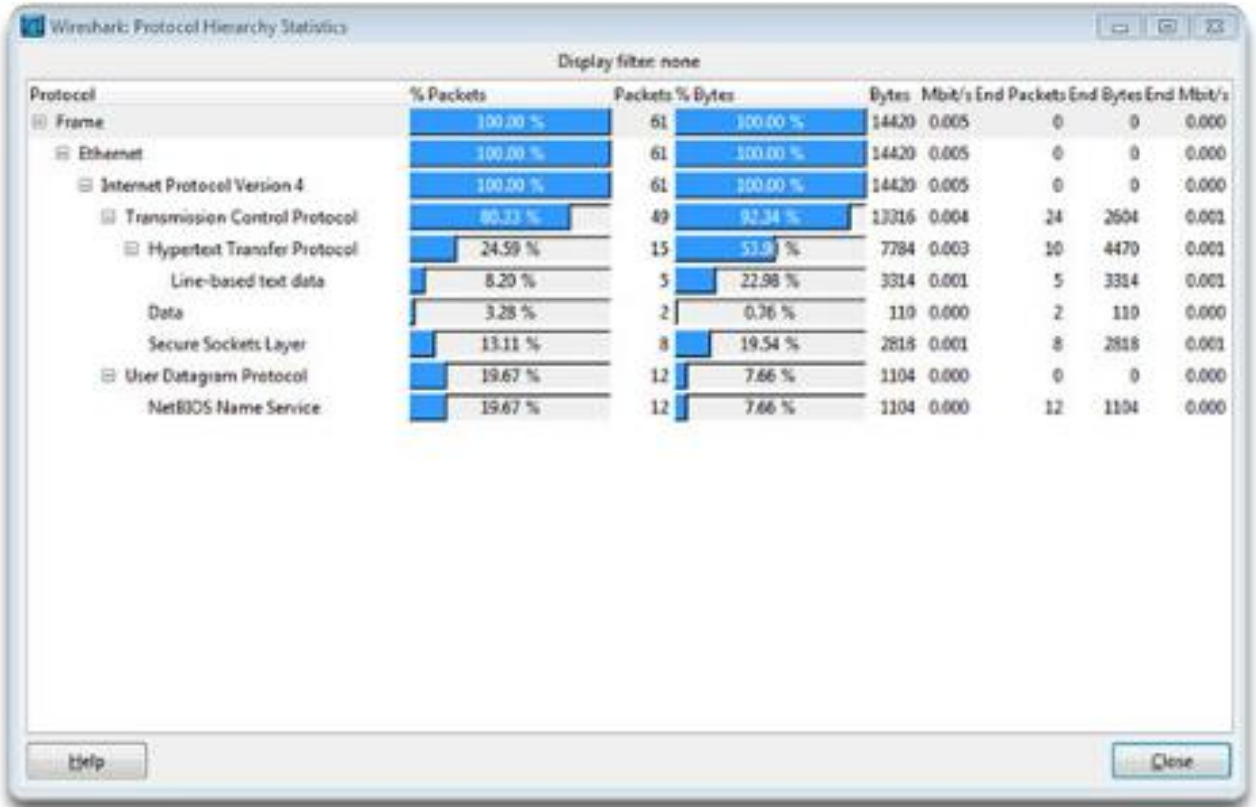
Масштабирование небольших сетей

Важные рекомендации при расширении до более крупных сетей:

- документация (физическая и логическая топология);
- опись устройств (список устройств, которые используют сеть или являются её частью);
- бюджет (детализированный бюджет на ИТ, включая годовой бюджет на закупку оборудования на финансовый год);
- анализ трафика (необходимо задокументировать протоколы, приложения и службы, а также соответствующие требования к трафику).

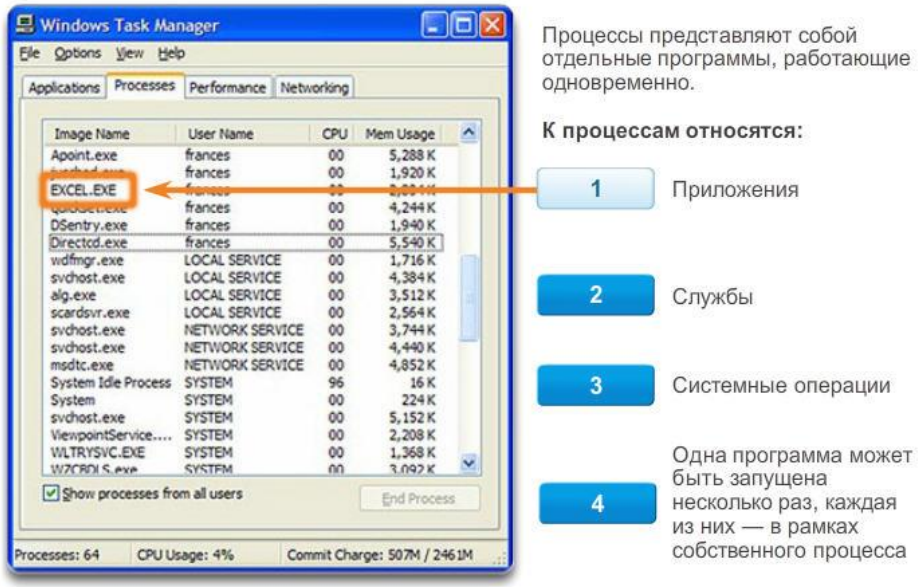
Анализ протоколов в небольшой сети

Сведения, собранные посредством анализа протоколов, можно использовать при принятии решений относительно более эффективного управления трафиком.



Непрерывное развитие требований к протоколам

- Сетевые администраторы могут получить «снимки» схемы потребления сотрудниками ИТ-приложений.
- Снимки позволяют отслеживать потребление ресурсов и требования к потокам трафика.
- Снимки предоставляют информацию о требуемых изменениях сети.



Методы поиска и устранения неполадок

Основные подходы к поиску и устранению неполадок

Шаг	Описание
Шаг 1. Определение неполадки	•Первый шаг в процессе поиска и устранения неполадок. •На этом этапе можно использовать различные методы, в том числе, можно расспросить пользователя, что может оказаться очень полезным.
Шаг 2. Формирование предположений о причинах неполадки	•После выявления проблемы попробуйте установить теорию вероятных причин. •Обычно на этом этапе выявляется несколько возможных причин неполадки.
Шаг 3. Проверка предположений с целью определения причины	•Проверьте свои предположения о вероятных причинах неполадки. •Вы можете попытаться устранить неполадку, применив быструю процедуру. •Если с помощью быстрой процедуры не удается устранить неполадку, следует продолжить поиск точной причины.
Шаг 4. Разработка плана действий решения проблем и его реализация.	Установив точную причину неполадки, разработайте план действий для ее устранения и выполните его.
Шаг 5. Полная проверка функционального состояния сети и принятие профилактических мер	•После устранения неполадки выполните полную проверку функционального состояния системы •При необходимости примите профилактические меры во избежание повторения проблемы в будущем.
Шаг 6. Документирование полученных данных, принятых мер и результатов	•На последнем этапе процедуры поиска и устранения неполадок выполняется документирование полученных данных, выполненных действий и результатов. •Эта информация очень важна для использования в будущем.

Методы поиска и устранения неполадок

Команда debug

- Команда **debug** в IOS позволяет отобразить сообщения в реальном времени для анализа.
- Все команды **debug** вводятся в привилегированном режиме EXEC. Cisco IOS позволяет ограничить выходные данные команды **debug** и включать в них только нужные функции или подфункции. Поэтому команду **debug** следует использовать только для устранения конкретной проблемы.
  - Чтобы отобразить краткое описание всех параметров команды отладки, введите в командной строке команду **debug?** в привилегированном режиме EXEC.
  - Для выключения функции отладки используйте ключевое слово **no debug.**
  - Кроме того, можно ввести команду **undebug** в привилегированном режиме EXEC.
  - Чтобы выключить все активные команды debug, используйте команду **undebug all.**
- Некоторые команды **отладки**, генерируют большое количество выходных данных и задействуют значительную часть системных ресурсов. Маршрутизатор может оказаться настолько занят отображением **сообщений**, что у него не останется вычислительной мощности для выполнения своих основных сетевых функций или даже для восприятия команд отключения отладки.

Методы поиска и устранения неполадок

Команда terminal monitor

- **Debug** и некоторые другие выходные сообщения IOS не отображаются автоматически на удаленных соединениях. Это связано с тем, что сообщения журнала не отображаются на линиях vty.
- Чтобы включить отображение сообщений журнала на терминале (виртуальной консоли), используйте команду **terminal monitor** в привилегированном режиме EXEC. Чтобы отключить отображение сообщений журнала на терминале, используйте команду **terminal no monitor** в привилегированном режиме EXEC.

```
R1#  terminal monitor
R1# ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
*Aug 20 16:03:49.735: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.737: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.738: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.740: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.741: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
R1# no debug ip icmp
ICMP packet debugging is off
R1#
```

Сценарии поиска и устранения неполадок

Вопросы работы и несоответствия дуплекса

- Чтобы обеспечить оптимальную производительность и отсутствие задержек канала связи, два подключенных сетевых интерфейса Ethernet должны работать в одном режиме дуплекса.
- Функция автосогласования Ethernet облегчает настройку, минимизирует проблемы и максимизирует производительность соединения между двумя соединенными каналами Ethernet. Сначала подключенные устройства объявляют поддерживаемые ими возможности, а затем на обоих концах соединения выбирается режим, который обеспечивает наибольшую производительность.
- При несовпадении дуплексных режимов передача данных будет выполняться, однако производительность канала связи будет очень низкой.
- Несовпадения дуплексов обычно вызваны неправильной конфигурацией интерфейса или, в редких случаях, неудачным автосогласованием. Проблему несовпадения дуплексных режимов довольно сложно обнаружить, поскольку обмен данными между устройствами продолжается.

Сценарии поиска и устранения неполадок

Проблемы IP-адресации на устройствах IOS

- Двумя основными причинами неверного назначения IPv4-адресов являются ошибки назначения адресов вручную и неполадки, связанные с протоколом DHCP.
- Часто сетевым администраторам приходится вручную назначать IP-адреса таким устройствам, как серверы и маршрутизаторы. Если во время назначения допущена ошибка, то велика вероятность того, что при связи с устройством возникнет проблема.
- Команды **show ip interface** и **show ip interface brief** на устройстве IOS позволяют проверить, какие IPv4-адреса назначены сетевым интерфейсам. Например, выполнение команды **show ip interface brief** будет проверять состояние интерфейса на R1.

```
R1# show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   209.165.200.225 YES manual up                    up
GigabitEthernet0/0/1   192.168.10.1    YES manual up                    up
Serial0/1/0            unassigned      NO  unset  down                  down
Serial0/1/1            unassigned      NO  unset  down                  down
GigabitEthernet0       unassigned      YES unset  administratively down down
R1#
```

Проблемы IP-адресации на оконечных устройствах

- Если устройству под управлением Windows не удается связаться с сервером DHCP, Windows автоматически назначает ему адрес из диапазона 169.254.0.0/16. Эта функция называется автоматической частной IP-адресации (APIPA).
- Чаще всего компьютер с адресом из диапазона 169.254.0.0/16 не сможет связаться с другими устройствами в сети, потому что эти устройства, скорее всего, не будут принадлежать сети 169.254.0.0/16.
  - Примечание. Другие операционные системы, такие как Linux и OS X, не используют APIPA.
- Если устройству не удается подключиться к серверу DHCP, сервер не может назначить IPv4-адрес для конкретной сети и устройство не сможет установить соединение.
- Используйте команду **ipconfig** для проверки IP-адреса, назначенного компьютеру с ОС Windows.

Неполадки, связанные со шлюзом по умолчанию

- Шлюзом по умолчанию для оконечного устройства является ближайшее сетевое устройство, которое способно пересылать трафик в другие сети. Если для устройства указан неверный или несуществующий адрес шлюза по умолчанию, оно не сможет связываться с устройствами в удаленных сетях.
- Так же, как и проблемы с IPv4-адресами, проблемы со шлюзом по умолчанию могут быть вызваны неверной настройкой (при назначении вручную) или неполадками DHCP (если используется автоматическое назначение).
- Используйте команду **ipconfig** для проверки шлюза по умолчанию на компьютере с ОС Windows.
- Выполните команду **show ip route** на маршрутизаторе для отображения таблицы маршрутизации и убедитесь, что настроен шлюз по умолчанию, также называемый маршрутом по умолчанию. Этот маршрут используется в случаях, когда адрес назначения пакета не соответствует другим маршрутам, указанным в таблице маршрутизации.

Поиск и устранение неполадок, связанных с DNS

- Многие пользователи ошибочно связывают работу интернет-канала с доступностью службы DNS.
- Адреса сервера DNS могут быть заданы вручную или назначены автоматически.
- Компании и организации часто разворачивают свои собственные серверы DNS, но для преобразования имен можно использовать любой доступный сервер DNS.
- Чтобы узнать, какой сервер DNS используется на компьютере с Windows, выполните команду **ipconfig /all**.
- Еще одним полезным инструментом для устранения неполадок DNS на ПК является команда **nslookup**. С помощью команды **nslookup** пользователь может вручную отправлять запросы DNS, а затем анализировать полученные ответы DNS.