МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ
Колледж Пятигорского института (филиал) СКФУ

ОП.07 «Компьютерные сети»
МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ЛАБОРАТОРНЫХ РАБОТ

Специальность СПО 09.02.11 Разработка и управление программным обеспечением

Пятигорск 2026

Методические указания для лабораторных работ по дисциплине «Компьютерные сети» составлены в соответствии с требованиями ФГОС СПО к подготовке выпуска для получения квалификации. Предназначены для студентов, обучающихся по специальности 09.02.11 Разработка и управление программным обеспечением.

Лабораторная работа №1

Тема: Монтаж кабельной системы и работа в симуляторе сети
Цель работы: Получение навыков физического монтажа кабеля (витая пара) и начальных навыков работы в сетевом симуляторе.

1. Краткие теоретические сведения

Кабель «витая пара» (Twisted Pair) — вид кабеля связи, представляет собой одну или несколько пар изолированных проводников, скрученных между собой с небольшим числом витков на единицу длины. Скручивание производится с целью повышения помехозащищенности кабеля.

Для соединения устройств с помощью витой пары используется разъем **8P8C** (ошибочно называемый RJ-45). Существует два основных стандарта обжима: **T568A** и **T568B**. Для соединения разных типов устройств (компьютер-коммутатор) используется **прямой** кабель (оба конца обжаты по одному стандарту). Для соединения однотипных устройств (компьютер-компьютер) используется **перекрестный** кабель (кроссовер), где одна сторона обжата по стандарту T568A, другая — по T568B.

Для построения компьютерных сетей применяются линии связи, использующие различную физическую среду. В качестве физической среды в коммуникациях используются металлы (в основном медь), сверхпрозрач- ное стекло (кварц) или пластик и эфир. Физическая среда передачи данных может представлять собой кабель "витая пара", коаксиальные кабель, волоконно-оптический кабель и окружающее пространство.

Линии связи, или линии передачи, данных - это промежуточная аппаратура и физическая среда, по которой передаются информационные сигналы (данные).

В одной линии связи можно образовать несколько каналов связи (виртуальных или логических каналов), например, путем частотного или временного разделения каналов. Канал связи - это средство односторонней передачи данных. Если линия связи монопольно используется каналом связи, то в этом случае линию связи называют каналом связи.

Канал передачи данных - это средства двухстороннего обмена данными, которые включают в себя линии связи и аппаратуру передачи (приема) данных. Каналы передачи данных связывают между собой источники информации и приемники информации.

В качестве линий связи могут быть применены коаксиальный кабель или кабель "витая пара".

Витая пара (англ. twisted pair) — вид кабеля связи, представляет собой одну или несколько пар изолированных проводников, скрученных между собой

(с небольшим числом витков на единицу длины), покрытых пластиковой оболочкой. Свивание проводников производится с целью повышения связи проводников одной пары (электромагнитная помеха одинаково влияет на оба провода пары) и последующего уменьшения электро- магнитных помех от внешних источников, а также взаимных наводок при передаче дифференциальных сигналов. Для снижения связи отдельных пар кабеля (периодического сближения проводников различных пар) в кабелях UTP категории 5 и выше провода пары свиваются с различным шагом. Витая пара - один из компонентов современных структурированных кабельных систем. Используется в телекоммуникациях и компьютерных сетях в качестве сетевого носителя во многих технологиях, таких как Ethernet, ARCNet и Token ring. В настоящее время благодаря своей дешевизне и лёгкости в установке является самым распространённым решением для построения локальных сетей.

Кабель подключается к сетевым устройствам при помощи соединителя 8P8C (RJ45 или RJ-45), немного большего, чем телефонный со- единитель RJ11.

В зависимости от наличия защиты — электрически заземлённой медной оплетки или алюминиевой фольги вокруг скрученных пар - определяют разновидности данной технологии.

Незащищенная витая пара

Неэкранированная витая пара (UTP — Unscreened twisted pair)

— экранирование полностью отсутствует;

Фольгированная витая пара (FTP — Foiled twisted pair) — также известна как S/UTP [1] присутствует один общий внешний экран;

Фольгированная экранированная витая пара (SFTP — Shielded Foiled twisted pair) — отличается от FTP наличием дополнительного внешнего экрана из медной оплетки;

Виды защищенной витой пары:

Стандартная (STP — Shielded twisted pair) присутствет экран для каждой пары;

Экранированная витая пара (S/STP — Screened shielded twisted pair) отличается от STP наличием дополнительного общего внешнего экрана.

Экранирование обеспечивает лучшую защиту от электромагнитных наводок как внешних, так и внутренних, и т. д. Экран по всей длине соединен с неизолированным дренажным проводом, который объединяет экран в случае разделения на секции при излишнем изгибе или растяжении кабеля. В зависимости от структуры проводников кабель применяется одно- и

многожильный. В первом случае каждый провод состоит из одной медной жилы, а во втором — из нескольких.

Одножильный кабель не предполагает прямых контактов с подключаемой периферией. То есть, как правило, его применяют для прокладки в коробах, стенах и так далее с последующим оконечива нием розетками. Связано это с тем, что медные жилы довольно толстые и при частых изгибах быстро ломаются. Однако для «врезания» в разъемы панелей розеток такие жилы подходят как нельзя лучше.

В свою очередь, многожильный кабель плохо переносит «врезание» в разъёмы панелей розеток (тонкие жилы разрезаются), но замечательно ведет себя при изгибах и скручиваниях. Кроме того, многожильный провод обладает большим затуханием сигнала. Поэтому многожильный кабель используют в основном для изготовления патчкордов (PatchCord), соединяющих периферию с розетками.

Конструкция кабеля

Кабель обычно состоит из четырёх пар. Проводники в парах изготовлены из монолитной медной проволоки толщиной 0,5 - 0,65 мм. Кроме метрической, применяется система AWG, в которой эти величины составляют 24 или 22 соответственно. Толщина изоляции около

0,2 мм, материал обычно поливинилхлорид (английское сокращение PVC), для более качественных образцов 5-й категории - полипропилен (PP), полиэтилен (PE). Особенно высококачественные кабели имеют изоляцию из вспененного (ячеистого) полиэтилена, который обеспечивает низкие диэлектрические потери, или тефлона, обеспечивающего уникальный рабочий диапазон температур.

Также внутри кабеля встречается так называемая «разрывная нить» (обычно капрон), которая используется для облегчения разделки внешней оболочки: при вытягивании она делает на оболочке продольный разрез, который открывает доступ к кабельному сердечнику, гарантированно не повреждая изоляцию проводников.

Внешняя оболочка имеет толщину 0,5 - 0,6 мм и обычно изготавливается из привычного поливинилхлорида с добавлением мела, который повышает хрупкость. Это необходимо для точного облома по месту надреза лезвием отрезного инструмента. Кроме этого начинают применяться так называемые «молодые полимеры», которые не поддерживают горения и не выделяют при нагреве галогенов (такие кабели маркируются как LSZH - Low Smoke Zero Halogen и обычно имеют яркую окраску внешней оболочки).

Самый распространенный цвет оболочки - серый. Оранжевая окраска, как правило, указывает на негорючий материал оболочки, который позволяет

прокладывать линии в закрытых областях. В общем случае цвета не обозначают особых свойств, но их применение позволяет легко отличать коммуникации с разным функциональным назначением как при монтаже, так и обслуживании.

Отдельно нужно отметить маркировку. Кроме данных о производителе и типе кабеля она обязательно включает в себя метровые или футовые метки.

Форма внешней оболочки также может быть различна. Чаще других применяется самая простая - круглая. Только для прокладки под половым покрытием по очевидной причине используется плоский кабель.

Кабели для наружной прокладки обязательно имеют влагостойкую оболочку из полиэтилена, которая наносится (как правило) вторым слоем поверх обычной, поливинилхлоридной. Кроме этого возможны заполнение пустот в кабеле водоотталкивающим гелем и бронирование с помощью гофрированной ленты или стальной проволоки.

Категории кабеля

Существует несколько категорий кабеля „витая пара”, которые нумеруются от CAT1 до CAT7 и определяют эффективный пропускаемый частотный диапазон. Кабель более высокой категории обычно содержит больше пар проводов и каждая пара имеет больше витков на единицу длины. Категории неэкранированной витой пары описываются в стандарте EIA/TIA 568 (Американский стандарт проводки в коммерческих зданиях).

CAT1 (полоса частот 0.1 МГц) - телефонный кабель, всего одна пара (в России применялся кабель без скруток — «лапша», у него характеристики не хуже, но больше влияние помех). В США использовался ранее только в «скрученном» виде. Применяется только для передачи голоса или данных при помощи модема.

CAT2 (полоса частот 1 МГц) - старый тип кабеля, 2 пары проводников, поддерживал передачу данных на скоростях до 4 Мбит/с, использовался в сетях token ring и ARCNet. Сейчас иногда встречается в телефонных сетях.

CAT3 (полоса частот 16 МГц) - 4-парный кабель, использовался при построении локальных сетей 10BASE-T и token ring, поддерживает скорость передачи данных до 10 или 100 Мбит/с по технологии 100BASE-T4. В отличие от предыдущих двух отвечает требованиям стандарта IEEE

802.3. Также до сих пор встречается в телефонных сетях.

CAT4 (полоса частот 20 МГц). Кабель состоит из 4 скрученных пар, использовался в сетях token ring, 10BASE-T, 100BASE-T4, скорость передачи данных не превышает 16 Мбит/с по одной паре, сейчас не используется.

CAT5 (полоса частот 100 МГц) - 4- парный кабель, это и есть то, что обычно называют кабель «витая пара» (рис. 8.19). Благодаря высокой скорости передачи до 100 Мбит/с при использовании 2 пар и до 1000 Мбит/с при использовании 4 пар, является самым распространённым сетевым носителем, использующимся в компьютерных сетях до сих пор. При прокладке новых сетей пользуются несколько усовершенствованным кабелем CAT5e (полоса частот 125 МГц), который лучше пропускает высокочастотные сигналы. Ограничение на длину кабеля между устройствами (компьютер-свитч, свитч- компьютер, свитч-свитч) 100 м. Ограничение хаб-хаб 5 м.

CAT6 (полоса частот 250 МГц) применяется в сетях Fast Ethernet и Gigabit Ethernet, состоит из 4 пар проводников и способен передавать данные на скорости до 1000 Мбит/с. Добавлен в стандарт в июне 2002 года. Существует категория CAT6a, в которой увеличена частота пропускаемого сигнала до 500 МГц. По данным IEEE, в 70 % установленных сетей в 2004 году применялся кабель категории CAT6.

CAT7. Спецификация на данный тип кабеля пока не утверждена, скорость передачи данных до 100 Гбит/с, частота пропускаемого сигнала до 600-700 МГц. Кабель этой категории экранирован. Седьмая категория витой пары не UTP, а S/FTP (Screened Fully shielded Twisted Pair). Благодаря двойному экрану длина кабеля может превышать 100 м.

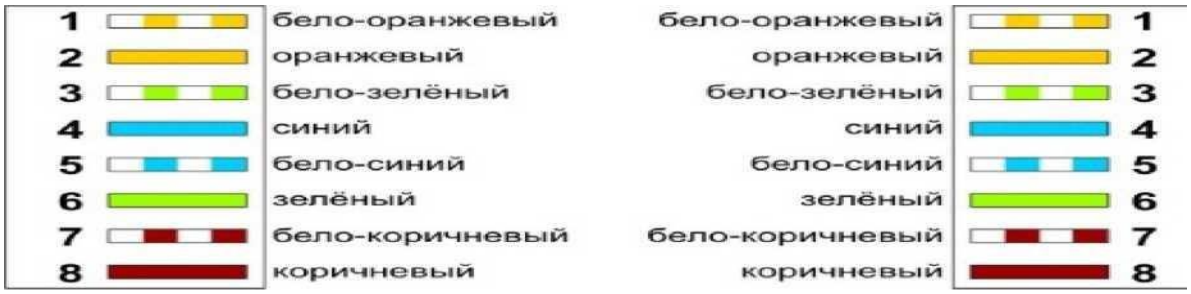

Рис. 1. Схемы обжимки кабеля: а - прямой кабель

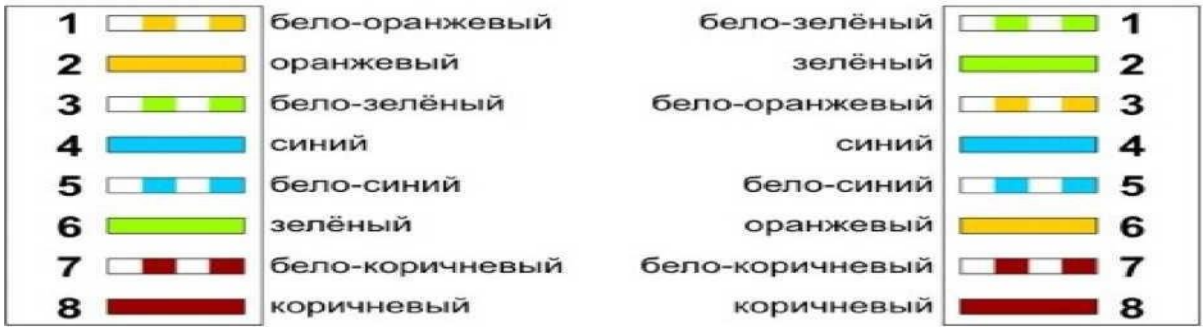

Рис. 2. Схемы обжимки кабеля: б - перекрёстный (кросс-овер) кабель

2. Задание к лабораторной работе

Работа с физическим кабелем.

1.1. Ознакомиться с инструментом для обжима (кримпером), кабелем и коннекторами.
1.2. Используя схему T568B, обжать один конец кабеля.
1.3. Используя схему T568B, обжать второй конец кабеля (прямой кабель).
1.4. Используя схему T568A, обжать второй конец другого отрезка кабеля (кросс-кабель).
1.5. Протестировать оба кабеля с помощью кабельного тестера. Зафиксировать результат.

3. Контрольные вопросы

1. Для чего скручивают провода в витой паре?
2. В чем разница между прямым и кроссовым кабелем?
3. Каково назначение коммутатора в локальной сети?
4. Что такое IP-адрес и маска подсети?
5. Для чего нужна команда ping?

Лабораторная работа №2

Тема: Установка и настройка протокола TCP/IP в ОС. Настройка статического и динамического (DHCP) IP-адреса.

Цель работы: Изучить способы диагностики настроек стека протоколов TCP/ IP; получить сведения о настройке TCP/IP для работы с DHCP сервером.

Ход выполнения работы:

1. Ознакомиться с теоретической частью.
2. Выполнить задания.
3. Ответить на контрольные вопросы.
4. Оформить отчет.

Теоретическая часть

На концептуальной модели взаимодействия открытых систем OSI основан стек протоколов ***TCP/IP****(Transmission Control Protocol - протокол управления передачей / Internet Protocol – Интернет-протокол)*, который предоставляет ряд стандартов для связи компьютеров и сетей.

Стек протоколов TCP/IP – промышленный стандарт, который позволяет организовать сеть масштаба предприятия и связывать компьютеры, работающие под управлением различных операционных систем.

Применение стека протоколов TCP/IP дает следующие преимущества:

- поддерживается почти всеми операционными системами; почти все большие сети основаны на TCP/IP;
- технология позволяет соединить разнородные системы;
- надежная, расширяемая интегрированная среда на основе модели «клиент — сервер»;
- получение доступа к ресурсам сети Интернет.

Каждый узел ***TCP/IP*** идентифицирован своим логическим IP-адресом, который идентифицирует положение компьютера в сети почти таким же способом, как номер дома идентифицирует дом на улице.

Реализация ***TCP/IP*** позволяет узлу ***TCP/IP*** использовать статический IP-адрес или получить IP-адрес автоматически с помощью ***DHCP-сервера*** *(Dynamic Host Configuration Protocol - протокол динамической конфигурации хоста).*

Для простых сетевых конфигураций, основанных на локальных сетях (*LAN, Local Area Network*), он поддерживает автоматическое назначение IP-адресов.

По умолчанию компьютеры клиентов, работающие под управлением ОС **Windows** или **Linux**, получают информацию о настройке протокола ***TCP/IP*** автоматически от службы ***DHCP***.

Однако даже в том случае, если в сети доступен ***DHCP-сервер***, необходимо назначить статический IP-адрес для отдельных компьютеров в сети. Например, компьютеры с запущенной службой ***DHCP*** не могут быть клиентами ***DHCP***, поэтому они должны иметь статический IP-адрес.

Если служба ***DHCP*** недоступна, можно настроить ***TCP/IP*** для использования статического IP-адреса.

Для каждой платы сетевого адаптера в компьютере, которая использует ***TCP/IP***, можно установить IP-адрес, маску подсети и шлюз по умолчанию.

Ниже описаны параметры, которые используются при настройке статического адреса ***TCP/IP***.

Параметр	Описание
IP-адрес	Логический 32-битный адрес, который идентифицирует TCP/IP узел. Каждой плате сетевого адаптера в компьютере с запущенным протоколом TCP/IP необходим уникальный IP-адрес, такой, как 192.168.0.108. Каждый адрес имеет две части: ID сети, который идентифицирует все узлы в одной физической сети и ID узла, который идентифицирует узел в сети. В этом примере ID сети — 192.168.0, и ID узла — 108.

Маска подсети	Подсети делят большую сеть на множество физических сетей, соединенных маршрутизаторами. Маска подсети закрывает часть IP-адреса так, чтобы TCP/IP мог отличать ID сети от ID узла. При соединении узлов TCP/IP, маска подсети определяет, где находится узел получателя: в локальной или удаленной сети. Для связи в локальной сети компьютеры должны иметь одинаковую маску подсети.
Шлюз по умолчанию	Промежуточное устройство в локальной сети, на котором хранятся сетевые идентификаторы других сетей предприятия или Интернета. TCP/IP посылает пакеты в удаленную сеть через шлюз по умолчанию (если никакой другой маршрут не настроен), который затем пересылает пакеты другим шлюзам, пока пакет не достигнет шлюза, связанного с указанным адресатом.

Если сервер с запущенной службой ***DHCP*** доступен в сети, он автоматически предоставляет информацию о параметрах ***TCP/IP*** клиентам ***DHCP***.

Настройка и диагностика сети в ОС Linux

Для работы с сетевыми протоколами TCP/IP в Linux достаточно наличие только петлевого интерфейса, но если необходимо объединить хосты между собой, естественно, необходимо наличие сетевого интерфейса, каналов передачи данных (например витая пара), возможно, какого-либо сетевого оборудования. Так же, необходимо наличие установленных утилит для настройки сети (/sbin/ifconfig, /sbin/route и др.), обычно поставляемые в пакете net-tools. Так же необходимо наличие конфигурационных файлов для сети (например /etc/hosts) и поддержку сети ядром Linux.

Файлы настроек сети в Linux (конфигурационные файлы)

В целом, вся работа Linux основана на процессе init, который рождается при загрузке ОС и плодит своих потомков, которые в свою очередь и выполняют всю необходимую работу. Вся загрузка Linux основана на скриптах bash, в которых прописана вся последовательность запуска мелких утилит с различными параметрами, которые последовательно запускаются/останавливаются при запуске/остановке системы. Аналогично запускается и сетевая подсистема Linux.

/etc/init.d/networking

Это скрипт, отвечающий за инициализацию сети.

/etc/hosts

Данный файл хранит перечень IP адресов и соответствующих им (адресам) имен хостов. Исторически, данный файл использовался вместо службы DNS. В настоящее время, файл так же может использоваться вместо службы DNS, но только при условии, что в вашей сети количество машин измеряется в единицах, а не в десятках или сотнях, потому что в таком случае, придется контролировать

корректность данного файла на каждой машине.

/etc/networks

Данный файл хранит имена и адреса локальной и других сетей. При использовании данного файла, сетями можно управлять по имени. Например добавить маршрут не route add 192.168.1.12, а route add home-network.

/etc/nsswitch.conf

Файл определяет порядок поиска имени хоста/сети.

/etc/resolv.conf

Данный фал определяет параметры механизма преобразования сетевых имен в IP адреса. Простым языком, определяет настройки DNS.

Настройка сети

Чтобы быть уверенным в работоспособности команды в любом дистрибутиве, необходимо пользоваться двумя основными командами. Это **ifconfig** и **route**. Первая команда (**ifconfig**) отвечает за настройку сетевых интерфейсов (ip, маска, шлюз), вторая (**route**) - настройка маршрутизации. Следует заметить, что выполнение данных команд без отключения стандартного скрипта запуска сетевой подсистемы внесет изменения только до первой перезагрузки/перезапуска сетевой службы, т.к. скрипт /etc/init.d/networking при очередном запуске перечитает указанные выше конфиги и применит старые настройки. Соответственно, выход для постоянной установки настроек - либо команда **ifconfig** с соответствующими параметрами - вписать в rc.local, либо поправить руками соответствующие конфиги сетевых интерфейсов.

Так же, если выполняется команда **ifconfig** с недостающими параметрами (например только IP адрес), то остальные дополняются автоматически (например бродкаст адрес добавляется по умолчанию с хостовым адресом, оканчивающимся на 255 и маска подсети по умолчанию берется 255.255.255.0).

Маршрутизация для имеющихся интерфейсов всегда поднимается автоматически силами ядра. Вернее сказать, прямые марштуры в сеть, в которую смотрит поднятый интерфейс формируются автоматически, силами ядра. Поле gateway (шлюз) для таких записей показывает адрес выходного интерфейса или *. Если есть необходимость организовать свои маршруты, то необходимо воспользоваться командой **route**. Данной командой можно добавлять и удалять маршруты, но опять же, это поможет только до перезапуска /etc/init.d/networking (или другого скрипта, отвечающего за вашу сеть). Чтобы маршруты добавлялись автоматом, необходимо так же, как и с ifconfig - добавить команды добавления маршрутов в rc.local, либо поправить руками соответствующие конфиги сетевых интерфейсов (например в Deb - /etc/network/options).

Диагностика сети Linux

Существует большое количество инструментов диагностики сети в Linux, зачастую, они очень похожи на утилиты от Microsoft.

ping

Работа этой утилиты заключается в отправке т.н. пакетов ICMP удаленному серверу, который будет указан в параметрах команды, сервер возвращает отправленные команды, а **ping** подсчитывает время требуемое отправленному

пакету, чтобы дойти до сервера и вернуться.

```
[root@proxy ~]# ping ya.ru
PING ya.ru (87.250.251.3) 56(84) bytes of data.
64 bytes from www.yandex.ru (87.250.251.3): icmp_seq=1 ttl=57 time=42.7 ms
64 bytes from www.yandex.ru (87.250.251.3): icmp_seq=2 ttl=57 time=43.2 ms
64 bytes from www.yandex.ru (87.250.251.3): icmp_seq=3 ttl=57 time=42.5 ms
64 bytes from www.yandex.ru (87.250.251.3): icmp_seq=4 ttl=57 time=42.5 ms
64 bytes from www.yandex.ru (87.250.251.3): icmp_seq=5 ttl=57 time=41.9 ms
^C
--- ya.ru ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4012ms
rtt min/avg/max/mdev = 41.922/42.588/43.255/0.500 ms
```

Как видно, из приведенного примера, **ping** выводит нам кучу полезной информации. Прежде всего, мы выяснили, что можем установить соединение с хостом ya.ru (иногда говорят, что «хост ya.ru нам доступен»). Во-вторых, мы видим, что DNS работает корректно, потому что «пингуемое» имя было корректно преобразовано в IP адрес (ping ya.ru (87.250.251.3)). Далее, в поле icmp_seq= указана нумерация отправляемых пакетов. Каждому отправляемому пакету последовательно присваивается номер и если в данной нумерации будут «провалы», то это нам расскажет о том, что соединение с «пингуемым» неустойчиво, а так же может означать, что сервер, которому шлют пакеты перегружен. По значению time= мы видим, сколько времени пакет путешествовал до 87.250.251.3 и обратно. Остановить работу утилиты **ping** можно клавишами Ctrl+C.

Так же, утилита **ping** интересна тем, что может позволить увидеть, где именно возникли неполадки. Допустим, утилита **ping** выводит сообщение network not reachable (сеть недоступна), либо другое аналогичное сообщение. Это, скорее всего, говорит о некорректной настройке вашей системы. В таком случае, можно послать пакеты по IP-адресу провайдера, чтобы понять, в каком месте возникает проблема (между локальным ПК или «дальше»). Если Вы подключены к интернету через маршрутизатор, то можно послать пакеты по его IP. Соответственно, если проблема проявиться уже на этом этапе, это говорит, о неправильном конфигурировании локальной системы, либо о повреждении кабеля, если маршрутизатор отзывается, а сервер провайдера нет, то проблема - в канале связи провайдера и т.д. Наконец, если неудачей завершилось преобразовании имени в IP, то можно проверить связь по IP, если ответы будут приходить корректно, то можно догадаться, что проблема в DNS.

Следует отметить, что данная утилита не всегда надежный инструмент для диагностики. Удаленный сервер может блокировать ответы на ICMP запросы.

traceroute

Простым языком, команда называется трассировка маршрута. Как можно понять из названия - данная утилита покажет по какому маршруту шли пакеты до хоста. Утилита **traceroute** несколько похожа на **ping**, но отображает больше интересной информации. Пример:

```
[root@proxy ~]# traceroute ya.ru

traceroute to ya.ru (213.180.204.3), 30 hops max, 60 byte packets
```

```
1 243-083-free.kubtelecom.ru (213.132.83.243) 6.408 ms 6.306 ms 6.193 ms
2 065-064-free.kubtelecom.ru (213.132.64.65) 2.761 ms 5.787 ms 5.777 ms
3 lgw.kubtelecom.ru (213.132.75.54) 5.713 ms 5.701 ms 5.636 ms
4 KubTelecom-lgw.Krasnodar.gldn.net (194.186.6.177) 81.430 ms 81.581 ms 81.687 ms
5 cat26.Moscow.gldn.net (194.186.10.118) 47.789 ms 47.888 ms 48.011 ms
6 213.33.201.230 (213.33.201.230) 43.322 ms 41.783 ms 41.106 ms
7 carmine-red-vlan602.yandex.net (87.250.242.206) 41.199 ms 42.578 ms 42.610 ms
8 www.yandex.ru (213.180.204.3) 43.185 ms 42.126 ms 42.679 ms
```

Как видно, можно проследить маршрут от маршрутизатора провайдера 243-083-free.kubtelecom.ru (213.132.83.243) (Юг России) до конечного хоста в www.yandex.ru (213.180.204.3) в Москве.

dig

Данная утилита посылает запросы серверам DNS и возвращает информацию о заданном домене.

Консольные команды управления сетевым интерфейсом.

ifconfig [опции] – стандартная команда Unix управления сетевым интерфейсом. В Mandriva Linux управляющие возможности этой команды ограничены.

ifup <устройство> - консольная команда включения сетевого интерфейса.

ifdown <устройство> - консольная команда выключения сетевого интерфейса.

ping <ip адрес> - команда проверки функционирования сети с использованием icmp пакета. Для выхода из режима проверки сети необходимо нажать CTRL+**C.**

arp [опции] – Команда печати таблицы памяти MAC адресов компьютера.

Задания на лабораторную работу

6. **Задание 1. Проверьте работоспособность стека протоколов TCP/IP в ОС Windows.**

1. Запустите виртуальную машину и загрузите ОС Windows.
2. Запустите консоль ***(Пуск/Программы/Стандартные/Командная строка)***.
3. В командной строке введите **ipconfig /all**.
4. Используя приведенную ниже информацию, создайте в своей папке текстовый документ со следующими данными:
   - Имя компьютера;
   - Основной DNS-суффикс;
   - Описание DNS-суффикса для подключения;
   - Физический адрес;
   - DHCP включен;

- Автоконфигурация включена;
- IP-адрес автоконфигурации;
- Маска подсети;
- Шлюз по умолчанию.

5. Убедитесь в работоспособности стека ***TCP/IP***, отправив эхо-запросы на IP-адреса. Для этого воспользуйтесь командой **ping**:
   - отправьте эхо-запросы на локальный адрес компьютера (*loopback*) **ping 127.0.0.1** (на экране должны появиться сообщения о полученном ответе от узла 127.0.0.1);
   - отправьте эхо-запрос по другому IP-адресу, например **192.168.10.1**.

7.

8. Задание 2. Настройте стек протоколов *TCP/IP* для использования статического IP-адреса.

1. Откройте окно **Сетевые подключения *(Пуск/Панель управления/Сетевые подключения)***.
2. Вызовите **свойства подключения по локальной сети**. Для этого можно воспользоваться контекстным меню.
3. В появившемся диалоговом окне на вкладке **Общие** откройте свойства **Протокол Интернета TCP/IP**.
4. Щелкните переключатель *Использовать следующий IP-адрес* и введите в соответствующие поля данные:**IP_адрес; Маску подсети; Основной шлюз; Предпочитаемый DNS**.
5. Примените параметры кнопкой ***ОК***.
6. Закройте окно свойств подключения кнопкой ***ОК*** (если потребуется, то согласитесь на перезагрузку компьютера).
7. Проверьте работоспособность стека протоколов ***TCP/IP***.

9.

10. Задание 3. Настройте TCP/IP для автоматического получения IP-адреса.

1. Откройте окно **Сетевые подключения**.
2. Вызовите свойства **Подключения по локальной сети**.
3. Откройте свойства **Протокол Интернета TCP/IP**.
4. Установите переключатель *Получить IP-адрес автоматически*.
5. Закройте диалоговое окно **Свойства: Протокол Интернета TCP/IP** кнопкой ***ОК***.
6. Примените параметры кнопкой ***ОК***.
7. Проверьте настройку стека протоколов ***TCP/IP***.
8. Получите другой адрес для своего компьютера. Для этого:
   - запустите консоль (командную строку);
   - введите команду для сброса назначенных адресов - **ipconfig /release**;
   - введите команду для получения нового адреса **ipconfig /renew**;
9. Проверьте работоспособность стека протоколов ***TCP/IP***.

Контрольные вопросы

1. Что означает понятие «статический IP-адрес».

2. Как происходит автоматическое получение IP-адреса.
3. Назначение маски подсети.
4. Как выполнить тестирование протокола TCP/IP.
5. Что такое интерфейс **lo**? Какой у него адрес?
6. Почему **ping** до вашей сетевой платы идет быстрее чем до другого компьютера класса?
7.

Лабораторная работа №3

Тема: Расчет IP-подсетей (VLSM)

Цель работы: Получить практические навыки деления IP-сети на подсети разного размера (VLSM).

Теоретическая часть

IP-адрес – это уникальный числовой адрес, однозначно идентифицирующий узел, группу узлов или сеть. IP-адрес имеет длину 4 байта и обычно записывается в виде четырех чисел (так называемых «октетов»), разделенных точками, каждое из которых может принимать значения в диапазоне от 0 до 255, например:

128.10.2.30 - традиционная десятичная форма представления адреса,

10000000 00001010 00000010 00011110 - двоичная форма представления этого же адреса.

Адрес состоит из двух логических частей - номера сети и номера узла в сети. Какая часть адреса относится к номеру сети, а какая к номеру узла, определяется значениями первых битов адреса:

- Если адрес начинается с 0, то сеть относят к классу A, и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса A имеют номера в диапазоне от 1 до 126. (Номер 0 не используется, а номер 127 зарезервирован для специальных целей). В сетях класса A количество узлов должно быть больше 2^{16}, но не превышать 2^{24}.
- Если первые два бита адреса равны 10, то сеть относится к классу B и является сетью средних размеров с числом узлов 2^8 - 2^{16}. В сетях класса B под адрес сети и под адрес узла отводится по 16 битов, то есть по 2 байта.
- Если адрес начинается с последовательности 110, то это сеть класса C с числом узлов не больше 2^8. Под адрес сети отводится 24 бита, а под адрес узла - 8 битов.
- Если адрес начинается с последовательности 1110, то он является адресом класса D и обозначает особый, групповой адрес - multicast. Если в пакете в качестве адреса назначения указан адрес класса D, то такой пакет должны получить все узлы, которым присвоен данный адрес.

- Если адрес начинается с последовательности 11110, то это адрес класса E, он зарезервирован для будущих применений.

В таблице приведены диапазоны номеров сетей, соответствующих каждому классу сетей.

Класс	Наименьший адрес	Наибольший адрес
A	1.0.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

В протоколе IP существует несколько соглашений об особой интерпретации IP-адресов:

- если IP-адрес состоит только из двоичных нулей, то он обозначает адрес того узла, который сгенерировал этот пакет;

```
0 0 0 0 .....
0 0 0 0
```

- если в поле номера сети стоят 0, то по умолчанию считается, что этот узел принадлежит той же самой сети, что и узел, который отправил пакет;

```
0  0  0  0  .....0
Номер узла
```

- если все двоичные разряды IP-адреса равны 1, то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета. Такая рассылка называется ограниченным широковещательным сообщением (limited broadcast);

```
1        1        1        1
.....1 1
```

- если в поле адреса назначения стоят сплошные 1, то пакет, имеющий такой адрес рассылается всем узлам сети с заданным номером. Такая рассылка называется широковещательным сообщением (broadcast);

Номер сети
1111.....11

- адрес 127.0.0.1 зарезервирован для организации обратной связи при тестировании работы программного обеспечения узла без реальной отправки пакета по сети. Этот адрес имеет название loopback.

Уже упоминавшаяся форма группового IP-адреса - multicast - означает, что данный пакет должен быть доставлен сразу нескольким узлам, которые образуют группу с номером, указанным в поле адреса. Узлы сами идентифицируют себя, то есть определяют, к какой из групп они относятся. Один и тот же узел может входить в несколько групп. Такие сообщения в отличие от широковещательных называются мультивещательными. Групповой адрес не делится на поля номера сети и узла и обрабатывается маршрутизатором особым образом.

Ход работы:

1. Ознакомиться с теоретическими сведениями по теме. Особенно внимательно изучить материал, относящийся к IP-адресации.

Для выполнения задания 2 необходимо выполнить следующие действия:

1. Перевести каждое число IP-адреса в двоичную форму.
2. По первым битам IP-адреса определить класс сети.
3. В соответствии с классом определить маску сети по умолчанию.
4. Выписать только те биты IP-адреса, которые соответствуют единичным битам в маске сети. Представить эти биты в точечной нотации. Это будет номер сети.
5. Выписать те биты IP-адреса, которые соответствуют нулевым битам в маске сети. Представить их в точечной нотации. Это будет номер хоста.
6. В двоичном представлении IP-адреса биты, соответствующие номеру хоста, заменить единицами. Представить получившийся адрес в точечной нотации. Это будет широковещательный адрес.

***Задание* 1**

1. Ознакомьтесь с теоретическими сведениями .
2. Ответьте на следующие вопросы.

IP-адрес - это уникальный числовой адрес, однозначно идентифицирующий узел, группу узлов или сеть.

Если адрес начинается с 0, то сеть относят к классу A

Если первые два бита адреса равны 10, то сеть относится к классу B

Если адрес начинается с последовательности 110, то это сеть классу C

Если адрес начинается с последовательности 1110, то он D

Если адрес начинается с последовательности 11110, то он E

***Задание* 2**

1. **Для IP-адреса, указанного в индивидуальном задании, считая, что маска сети задана по умолчанию, определите** 67.192.44.89/12

67 = 0100011

192 = 11000000

44 = 00101100

89 = 01011001

01000011.11000000. 00101100. 01011001

11111111.11110000.00000000.00000000

1. Класс сети; A
2. Число сетей; 27-2
3. Маску сети по умолчанию; 255.0.0.0
4. Номер сети; 67.0.0.0
5. Номер хоста; 0.192.44.89
6. Широковещательный адрес. 67.255.255.255

2. Используя маску, указанную в индивидуальном задании, определите

1. Маску сети (в десятичной нотации); 11111111. 00000000.00000000.00000000
2. Номер сети (в десятичной нотации); 0100011. 00000000.00000000.00000000
3. Номер хоста (в десятичной нотации); 00000000. 11000000. 00101100. 01011001
4. Широковещательный адрес; 0100011.111111.111111.111111
5. Число хостов.

<u>Пример выполнения задания 2.</u>

Пусть IP-адрес 64.10.20.30

Переводим числа в двоичный формат:

$64_{10}=01000000_2$

$10_{10}=00001010_2$

$20_{10}=00010100_2$

$30_{10}=00011110_2$

Записываем двоичную форму представления IP-адреса:

01000000.00001010.00010100.00011110

Первые биты адреса – 01, значит, это сеть класса A.

Маска сети по умолчанию: 255.0.0.0

Записываем в двоичной форме маску сети и IP-адрес:

Маска: 11111111. 00000000.00000000.00000000

IP-адрес: 01000000. 00001010.00010100.00011110

Эти биты соответствуют номеру сети

А эти биты соответствуют номеру хоста

Значит, номер сети - 01000000_2 или 64_{10}

номер хоста - $00001010.00010100.00011110_2$ или $10.20.30_{10}$

Заменяем в IP-адресе номер хоста единицами, получим широковещательный адрес $01000000.111111.111111.111111_2$ или 64.255.255.255

Следовательно:

IP-адрес	64.10.20.30
Класс сети	A
Маска сети	255.0.0.0
Номер сети	64.0.0.0
Номер хоста	0.10.20.30
Широковещательный адрес	64.255.255.255
Число сетей 2^7-2	

При выполнении задания 3 необходимо вначале определить маску сети. Маска содержит столько единичных битов, сколько указано в числе после дробной черты. Остальные вычисления выполняются подобно заданию 2.

Контрольные вопросы для защиты лабораторной работы

- Что такое протокол? это набор правил, описывающих метод передачи информации по сети. Понятие протокола является исключительно важным для компьютерных сетей.
- Назовите уровни модели протоколов модели OSI и назначение протоколов каждого уровня.
- Назовите уровни стека протоколов TCP/IP и назначение протоколов каждого уровня. Стек протоколов TCP/IP является протокольной основой
- Приведите примеры протоколов, входящих в стек TCP/IP.
- Что такое IP-адрес? это уникальный числовой адрес, однозначно идентифицирующий узел, группу узлов или.

- Каковы правила назначения IP-адресов? IP-адрес имеет длину 4 байта и обычно записывается в виде четырех чисел (так называемых «октетов»), разделенных точками, каждое из которых может принимать значения в диапазоне от 0 до 255
- Как проанализировать IP-адрес? Адрес состоит из двух логических частей - номера сети и номера узла в сети. Какая часть адреса относится к номеру сети, а какая к номеру узла, определяется значениями первых битов адреса:

Варианты индивидуальных заданий

Таблица 2

Номер варианта	IP-адрес к заданию 2	IP-адрес к заданию 3
1.	192.168.72.33	192.168.72.33/20
1.	190.172.55.40	190.172.55.40/25
1.	123.232.14.72	123.232.14.72/18
1.	196.232.66.54	196.232.66.54/25
1.	193.123.55.67	193.123.55.67/26
1.	191.172.55.42	191.172.55.42/27
1.	178.66.57.18	178.66.57.18/20
1.	10.0.0.20	10.0.0.20/12
1.	67.192.44.89	67.192.44.89/12
1.	128.34.67.11	128.34.67.11/18
1.	193.34.126.44	193.34.126.44/26
1.	156.32.11.93	156.32.11.93/23
1.	167.168.169.170	167.168.169.17/20
1.	145.44.11.77	145.44.11.77/22
1.	132.45.171.99	132.45.171.99/25
1.	198.164.55.55	198.164.55.55/26
1.	192.77.121.144	192.77.121.144/25
1.	12.13.14.15	12.13.14.15/18
1.	44.57.62.39	44.57.62.39/18
1.	152.15.66.5	152.15.66.5/26
1.	132.45.171.99	132.45.171.99/27
1.	198.164.155. 5	198.164.155.5/26
1.	192.77.11.44	192.77.11.44/29
1.	12.130.140.150	12.130.140.150/17

1.	44.57.162.31	44.57.162.31/18
1.	152.154.66.65	152.154.66.65/20
1.	152.15.66.17	152.15.66.17/22
1.	132.45.171.88	132.45.171.88/21

Сетевая маска	Инверсия	Префикс	Используется	Размер
0.0.0.0	255.255.255.255	/0	4,294,967,294	весь интернет
128.0.0.0	127.255.255.255	/1	2,147,483,646	128 классов ‘a’
192.0.0.0	63.255.255.255	/2	1,073,741,822	64 класса ‘a’
224.0.0.0	31.255.255.255	/3	536,870,910	32 класса ‘a’
240.0.0.0	15.255.255.255	/4	268,435,454	16 классов ‘a’
248.0.0.0	7.255.255.255	/5	134,217,726	8 классов ‘a’
252.0.0.0	3.255.255.255	/6	67,108,862	4 класса ‘a’
254.0.0.0	1.255.255.255	/7	33,554,430	2 класса ‘a’
255.0.0.0	0.255.255.255	/8	16,777,214	1 класс ‘a’
255.128.0.0	0.127.255.255	/9	8,388,606	128 классов ‘b’
255.192.0.0	0.63.255.255	/10	4,194,302	64 класса ‘b’
255.224.0.0	0.31.255.255	/11	2,097,150	32 класса ‘b’
255.240.0.0	0.15.255.255	/12	1,048,574	16 классов ‘b’
255.248.0.0	0.7.255.255	/13	524,286	8 классов ‘b’
255.252.0.0	0.3.255.255	/14	262,142	4 класса ‘b’
255.254.0.0	0.1.255.255	/15	131,07	2 класса ‘b’
255.255.0.0	0.0.255.255	/16	65,534	1 класс ‘b’
255.255.128.0	0.0.127.255	/17	32,766	128 классов ‘c’
255.255.192.0	0.0.63.255	/18	16,382	64 класса ‘c’
255.255.224.0	0.0.31.255	/19	8,19	32 класса ‘c’
255.255.240.0	0.0.15.255	/20	4,094	16 классов ‘c’
255.255.248.0	0.0.7.255	/21	2,046	8 классов ‘c’
255.255.252.0	0.0.3.255	/22	1,022	4 класса ‘c’
255.255.254.0	0.0.1.255	/23	510	2 классов ‘c’
255.255.255.0	0.0.0.255	/24	254	1 класс ‘c’
255.255.255.128	0.0.0.127	/25	126	128 хостов
255.255.255.192	0.0.0.63	/26	62	64 хоста
255.255.255.224	0.0.0.31	/27	30	32 хоста
255.255.255.240	0.0.0.15	/28	14	16 хостов
255.255.255.248	0.0.0.7	/29	6	8 хостов
255.255.255.252	0.0.0.3	/30	2	4 хоста
255.255.255.254	0.0.0.1	/31	0	2 хоста
255.255.255.255	0.0.0.0	/32	1	1 хост

Лабораторная работа №4

Тема: Расчет и проектирование VLAN-сетей

Цель работы: Научиться проектировать схему адресации для VLAN и рассчитывать параметры сетей в текстовом редакторе.

Теоретическая часть

Беспроводная точка доступа (англ. Wireless Access Point, WAP) — устройство для объединения компьютеров в единую беспроводную сеть [1].

Виртуальные локальные сети (VLAN) позволяют логически разделить одну физическую сеть на несколько изолированных сегментов. При проектировании VLAN необходимо решить две задачи:

1. **Логическая схема** — определить, какие устройства и отделы должны быть в каких VLAN.
2. **IP-адресация** — выделить для каждого VLAN отдельную подсеть и рассчитать адреса.

Правила выделения подсетей:

- Каждый VLAN получает свою IP-подсеть.
- Для экономии адресов используется метод VLSM (маски переменной длины).
- Адрес сети — первый адрес в подсети.
- Широковещательный адрес — последний адрес в подсети.
- Адреса хостов — все между ними.

Задание к лабораторной работе

Часть 1. Проектирование логической схемы

Предприятие имеет следующую структуру отделов и требования к VLAN:

Отдел	Количество компьютеров	Требования безопасности
Бухгалтерия	28	Полная изоляция от других отделов
Отдел кадров	15	Изоляция, доступ только к серверу отдела
IT-отдел	8	Доступ к оборудованию всех отделов
Гости (Wi-Fi)	до 50	Только доступ в Интернет, без доступа к внутренним ресурсам
Серверная ферма	5 серверов	Доступ из всех отделов (кроме гостевой сети)

Задание 1.1. Предложите логическую схему разделения на VLAN. Заполните таблицу:

№ VLAN	Название	Отдел/назначение	Требуется ли доступ в другие VLAN?
10	VLAN_Buh	Бухгалтерия	Нет
20	...	...	...
30	...	...	...
40	...	...	...
50	...	...	...

Часть 2. Расчет IP-адресации

Предприятию выделена сеть **172.16.0.0/16**. Необходимо распределить подсети под каждый VLAN с учетом количества хостов, максимально экономя адресное пространство.

Задание 2.1. Для каждого VLAN рассчитайте:

- маску подсети (в десятичном и CIDR-формате);
- адрес сети;
- первый и последний доступный адрес для хостов;
- широковещательный адрес.

Пример расчета для VLAN 10 (бухгалтерия, 28 хостов):

Параметр	Расчет	Значение
Требуется адресов	28 хостов + адрес сети + broadcast = минимум 30 адресов	30
Ближайшая степень двойки	2^5 = 32	32 адреса
Маска	32 - 5 = 27 бит	/27 (255.255.255.224)
Адрес сети	172.16.0.0/27	172.16.0.0
Первый хост	сеть + 1	172.16.0.1
Последний хост	broadcast - 1	172.16.0.30

Параметр	Расчет	Значение
Broadcast	следующий адрес перед новой сетью	172.16.0.31

Задание 2.2. Заполните таблицу для всех VLAN. Учитывайте, что подсети не должны пересекаться.

VLAN	Отдел	Кол-во хостов	Требуется адресов	Маска (/x)	Адрес сети	Первый хост	Последний хост	Broadcast
10	Бухгалтерия	28	30	/27	172.16.0.0	172.16.0.1	172.16.0.30	172.16.0.31
20	Отдел кадров	15	17	...	...	...	...	...
30	IT-отдел	8	...	...	...	...	...	...
40	Гости	50	...	...	...	...	...	...
50	Серверы	5	...	...	...	...	...	...

Часть 3. Проверка расчетов

Задание 3.1. Проверьте, не пересекаются ли подсети. Для этого выпишите диапазоны адресов каждой подсети и убедитесь, что они не накладываются друг на друга.

Задание 3.2. Используя онлайн-калькулятор IP-адресов (например, https://ip-calculator.ru или любой другой доступный), проверьте правильность расчетов для трех любых VLAN.

3. Содержание отчета

Отчет должен содержать:

1. Логическую схему VLAN (можно нарисовать от руки и сфотографировать, либо создать в любом графическом редакторе).
2. Таблицу с расчетами IP-подсетей для всех VLAN.
3. Результаты проверки через онлайн-калькулятор (скриншоты или скопированные результаты).
4. Выводы по работе.

4. Контрольные вопросы

1. Для чего нужны VLAN в компьютерных сетях?
2. Что такое маска подсети и как она связана с количеством хостов?
3. Почему при расчете подсети к количеству хостов нужно прибавлять 2?
4. Как определить, не пересекаются ли две подсети?
5. Что такое VLSM и зачем он нужен?

Лабораторная работа №5

Тема: Анализ таблиц маршрутизации и диагностика сетевых соединений

Цель: изучить таблицы маршрутизации на реальном компьютере и освоить методы диагностики сетевых проблем с использованием встроенных утилит ОС.

Теоретическая часть

Таблица маршрутизации — это набор правил, которые определяют, куда отправлять сетевые пакеты. На каждом компьютере с ОС Windows/Linux есть своя таблица маршрутизации. Основные параметры записи:

- **Сеть назначения (Destination)** — адрес целевой сети.
- **Маска (Netmask)** — определяет размер сети назначения.
- **Шлюз (Gateway)** — адрес, на который нужно отправить пакет для достижения целевой сети.
- **Интерфейс (Interface)** — сетевой адаптер, через который отправляется пакет.
- **Метрика (Metric)** — стоимость маршрута (чем меньше, тем приоритетнее).

Основные утилиты диагностики (встроенные в ОС):

Команда	Назначение	Пример
ipconfig (Windows) ifconfig или ip addr (Linux)	Просмотр IP-конфигурации	ipconfig /all
route print (Windows) route -n (Linux)	Просмотр таблицы маршрутизации	route print
ping	Проверка доступности узла	ping 8.8.8.8

Команда	Назначение	Пример
tracert (Windows) traceroute (Linux)	Трассировка маршрута до узла	tracert ya.ru
pathping (Windows)	Комбинированная трассировка с анализом потерь	pathping ya.ru
nslookup	Проверка DNS-разрешения имен	nslookup google.com
netstat	Просмотр сетевых соединений и портов	netstat -an
arp -a	Просмотр ARP-таблицы (соответствие IP и MAC-адресов)	arp -a

2. Задание к лабораторной работе

Часть 1. Исследование конфигурации сети

Задание 1.1. Откройте командную строку (cmd) от имени администратора и выполните команду ipconfig /all.

Задание 1.2. Зафиксируйте в отчете следующие параметры:

- IP-адрес вашего компьютера;
- маска подсети;
- основной шлюз;
- адреса DNS-серверов;
- физический адрес (MAC-адрес) сетевого адаптера;
- включен ли DHCP.

Задание 1.3. Выполните команду route print и изучите таблицу маршрутизации. Найдите и запишите в отчет:

- маршрут по умолчанию (Destination 0.0.0.0);
- маршрут к вашей локальной сети (сеть, в которой находится ваш компьютер);
- маршрут обратной петли (127.0.0.0).

Часть 2. Диагностика соединений

Задание 2.1. Проверьте связь с локальным шлюзом:

text

ping [адрес_вашего_шлюза]

Запишите время отклика и количество потерянных пакетов.

Задание 2.2. Проверьте связь с удаленным сервером (например, DNS Google):

text

ping 8.8.8.8 -n 10

Сравните результаты с пингом до локального шлюза.

Задание 2.3. Выполните трассировку до любого сайта (например, ya.ru):

text

tracert ya.ru

Запишите количество переходов (хопов) и адреса промежуточных маршрутизаторов.

Задание 2.4. (Для Windows) Выполните команду pathping ya.ru и дождитесь завершения (может занять несколько минут). Проанализируйте, на каких участках есть потери пакетов.

Часть 3. Исследование DNS и портов

Задание 3.1. Определите IP-адреса для доменных имен:

text

nslookup yandex.ru
nslookup google.com
nslookup vk.com

Задание 3.2. Просмотрите активные сетевые соединения вашего компьютера:

text

netstat -an

Найдите соединения в состоянии ESTABLISHED (установлено). Каким программам они могут принадлежать?

Задание 3.3. Просмотрите ARP-таблицу:

text

arp -a

Какие устройства находятся в вашей локальной сети (по MAC-адресам)?

Часть 4. Анализ сетевых проблем (ситуационные задачи)

Задание 4.1. Проанализируйте ситуацию и предложите порядок диагностики:

Ситуация А: Пользователь жалуется, что не может зайти на сайт mail.ru, при этом WhatsApp и Telegram работают.

Ситуация Б: Компьютер получил IP-адрес 169.254.x.x. Интернета нет.

Ситуация В: Пинг до шлюза идет, но до 8.8.8.8 не идет.

Для каждой ситуации опишите:

- возможные причины;
- какие команды и в каком порядке нужно выполнить для диагностики;
- ожидаемый результат.

3. Содержание отчета

Отчет должен содержать:

1. Результаты выполнения всех команд (скриншоты или скопированный текст).
2. Таблицу с параметрами сети из ipconfig /all.
3. Анализ таблицы маршрутизации.
4. Результаты трассировки до двух разных сайтов.
5. Решение ситуационных задач с подробным описанием диагностики.

4. Контрольные вопросы

1. Что такое шлюз по умолчанию и как его узнать?
2. В чем разница между командами ping и tracert?
3. Для чего нужна команда nslookup?
4. Что означает IP-адрес 169.254.x.x?
5. Как по таблице маршрутизации определить, через какой интерфейс пойдет пакет к определенному адресу?
6. Что показывает команда netstat -an?

Лабораторная работа №6

Тема: Сетевые возможности Docker и контейнеризация.

Цель: Изучить теоретические основы контейнеризации и сетевых взаимодействий в Docker без установки ПО, используя документацию и аналитические материалы.

Теоретическая часть

Сетевые возможности Docker:

- **bridge** — виртуальный сетевой мост (docker0), создается по умолчанию. Контейнеры на одном хосте могут взаимодействовать через этот мост.
- **host** — контейнер использует сеть хоста напрямую (без изоляции).
- **none** — полная сетевая изоляция.
- **overlay** — для связи контейнеров на разных хостах (в режиме Swarm).

Проброс портов (port mapping): позволяет сделать сервис внутри контейнера доступным извне. Например, команда -p 8080:80 делает порт 80 контейнера доступным на порту 8080 хоста.

2. Задание к лабораторной работе

Часть 1. Изучение архитектуры Docker (по документации)

Задание 1.1. Используя открытые источники (официальная документация Docker на русском, habr.com, статьи на cyberleninka.ru), изучите и опишите в отчете:

- Что такое контейнер и чем он отличается от виртуальной машины? (заполните таблицу)

Характеристика	Виртуальная машина	Контейнер
Наличие ОС	...	...
Время запуска	...	...
Изоляция	...	...
Ресурсы	...	...

- Что такое образ Docker и слой (layer)?
- Как работает изоляция контейнеров (namespace, cgroups)?

Часть 2. Сетевые драйверы Docker

Задание 2.1. Изучите типы сетевых драйверов в Docker. Заполните таблицу:

Драйвер	Назначение	Когда применяется	Особенности
bridge	...	...	...
host	...	...	...
overlay	...	...	...
macvlan	...	...	...
none	...	...	...

Задание 2.2. Ответьте на вопросы:

1. Как контейнеры в одной bridge-сети обращаются друг к другу?
2. Что такое CNM (Container Network Model)?
3. Как работает DNS-resolution между контейнерами в пользовательской сети?

Часть 3. Анализ схем сетевого взаимодействия

Задание 3.1. Для каждой из предложенных схем определите, какие настройки Docker необходимо применить:

Схема 1: Веб-сервер и база данных

- Веб-сервер (Nginx) должен быть доступен извне на порту 80.
- База данных (PostgreSQL) должна быть доступна только для веб-сервера, но не из внешней сети.
- Оба контейнера должны "видеть" друг друга по именам.

Схема 2: Микросервисное приложение

- Три микросервиса (Auth, Users, Orders) работают в одной сети.
- API Gateway должен принимать запросы извне на порту 443 и направлять их к нужным микросервисам.
- Все микросервисы пишут логи в общий стек ELK.

Задание 3.2. Для каждой схемы опишите:

- какие сетевые драйверы будут использованы;
- нужен ли проброс портов и для каких сервисов;
- как будет обеспечиваться изоляция.

Часть 4. Решение задач

Задача 1. Запущен контейнер с веб-сервером командой:

text

```
docker run -d --name web1 -p 8080:80 nginx
```

Вопросы:

- По какому адресу будет доступен веб-сервер с хоста?
- По какому адресу будет доступен веб-сервер с другого компьютера в той же сети?
- Что произойдет, если на хосте порт 8080 уже занят?

Задача 2. Создана пользовательская сеть:

text

```
docker network create my-app-net
```

Затем запущены два контейнера:

text

```
docker run -d --name cont1 --network my-app-net alpine sleep 3600
docker run -d --name cont2 --network my-app-net alpine sleep 3600
```

Вопросы:

- Может ли cont1 обратиться к cont2 по IP-адресу?
- Может ли cont1 обратиться к cont2 по имени cont2?
- Что нужно сделать, чтобы обратиться к cont2 по имени из контейнера, не входящего в эту сеть?

Задача 3. Контейнер запущен с драйвером сети host:

text

```
docker run -d --name web-host --network host nginx
```

Вопросы:

- По какому адресу будет доступен веб-сервер?
- В чем преимущества и недостатки такого подхода?

3. Содержание отчета

Отчет должен содержать:

1. Таблицу сравнения контейнеров и виртуальных машин.
2. Таблицу сетевых драйверов Docker.
3. Разбор двух схем сетевого взаимодействия (схема 1 и схема 2).
4. Решения трех задач с подробными пояснениями.
5. Список использованных источников (ссылки на документацию, статьи).

4. Контрольные вопросы

1. В чем принципиальное отличие контейнерной виртуализации от аппаратной?
2. Для чего нужен проброс портов в Docker?
3. Как контейнеры в одной пользовательской сети находят друг друга?
4. Какие риски безопасности связаны с использованием драйвера host?
5. Что такое оркестрация контейнеров и какие задачи она решает?
6. Какие российские аналоги Docker существуют? (назовите не менее двух)

Список рекомендуемой литературы

Основная литература:

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. — СПб.: Питер, 2020.
2. Таненбаум Э., Уэзеролл Д. Компьютерные сети. — СПб.: Питер, 2019.
3. Официальная документация Docker (перевод на русский) — https://docs.docker.com/get-started/
4. Сравнение российских аналогов Docker: DEPO Container Platform, Deckhouse, Shturval

Дополнительная литература:

1. Хабрахабр, тег "Docker" — https://habr.com/ru/hub/docker/
2. Что такое контейнеризация? Объясняем простыми словами — https://practicum.yandex.ru/blog/chto-takoe-konteinerizaciya/