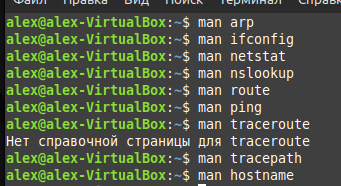
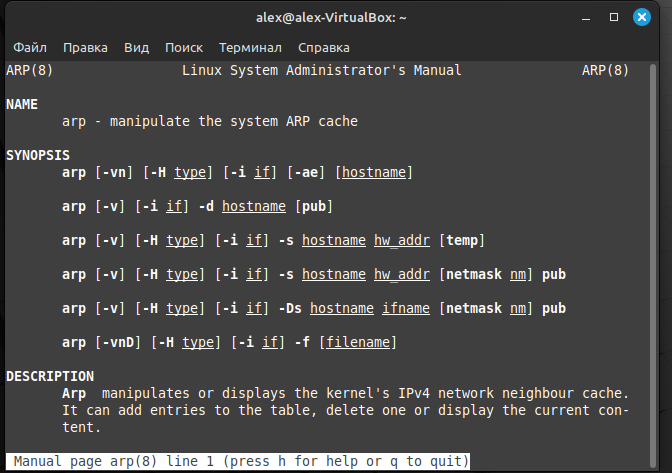
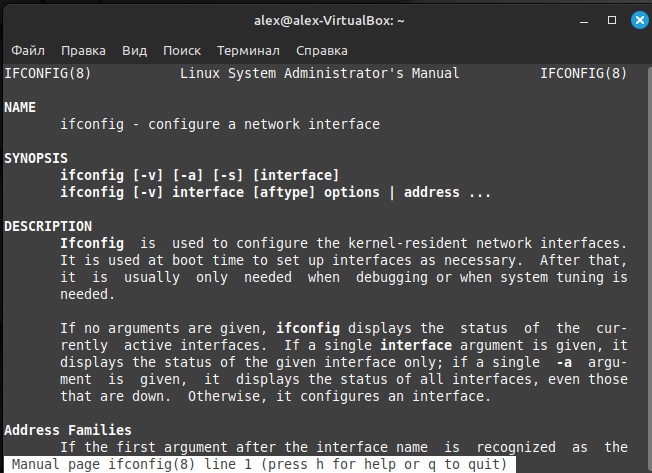
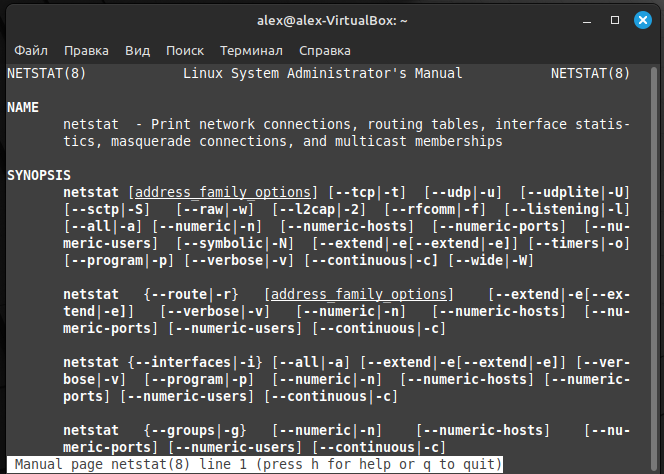
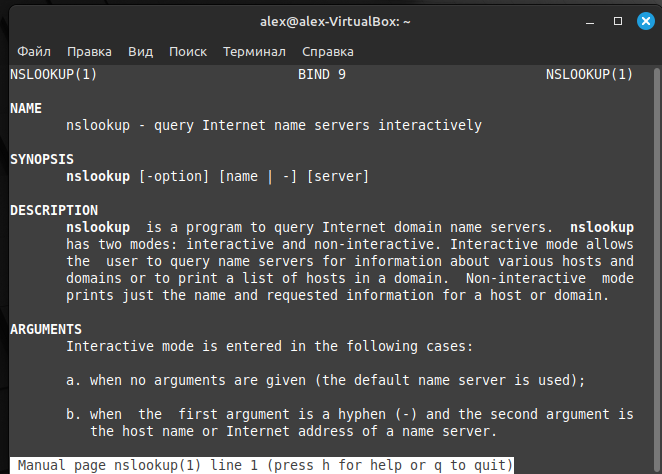
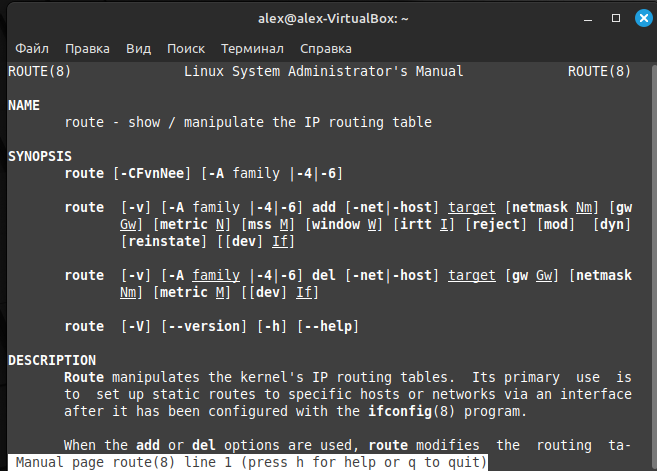
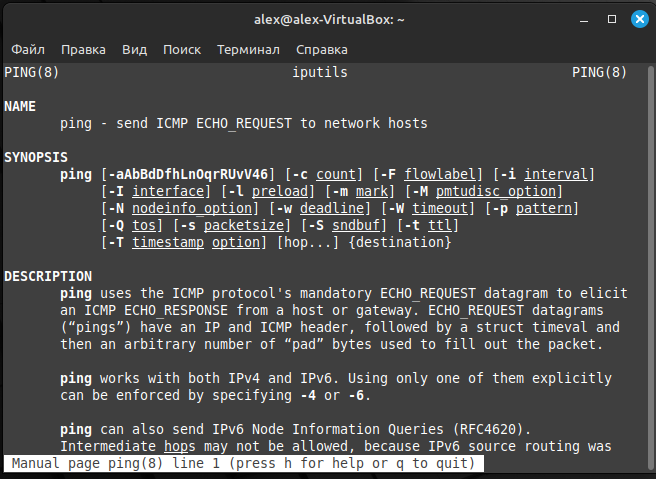
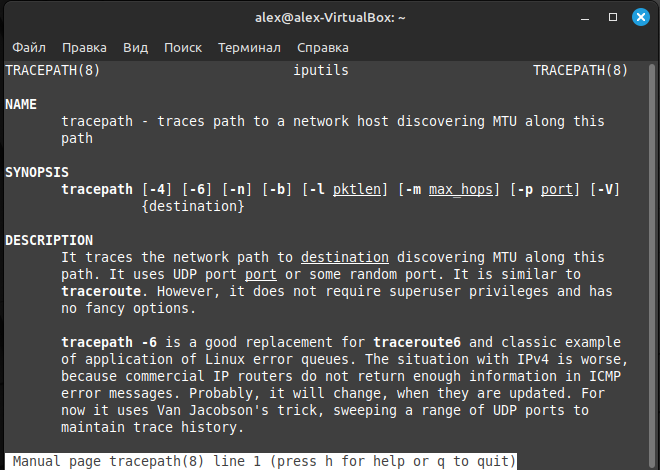
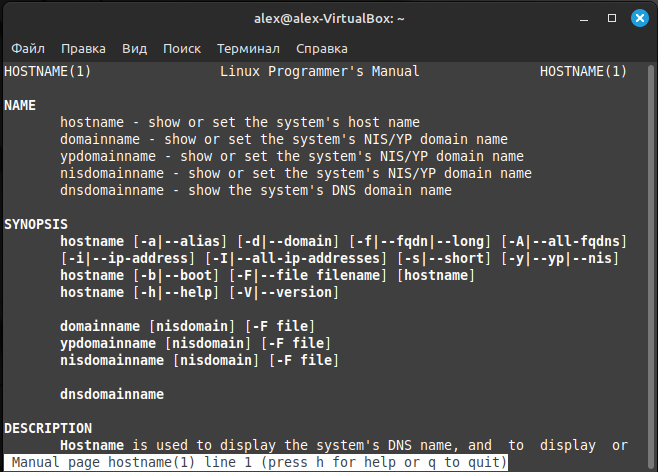
**Задания для выполнения**

1. Выведите на экран справочную информацию по утилитам arp, ifconfig, netstat, nslookup, route, ping, traceroute, hostname. Изучите ключи, используемые при запуске утилит.

Справочную информацию по утилитам выводится командой “man”



В некоторых дистрибутивах Linux команда `traceroute` может быть заменена на `tracert` или `tracepath`, в зависимости от того, какая утилита используется в системе. Названия утилит и их справочных страниц могут различаться в разных дистрибутивах.

Ключи:

1. arp\*

`-a` - отображает кэш ARP.

-`-s` - добавляет статическую запись ARP в кэш.

2. ifconfig (в некоторых дистрибутивах Linux устарел, и лучше использовать `ip`):

`ifconfig <интерфейс> up` - включает сетевой интерфейс.

`ifconfig <интерфейс> down` - выключает сетевой интерфейс.

`ifconfig <интерфейс> <IP-адрес>` - назначает IP-адрес интерфейсу.

3. netstat (также устарел, рекомендуется использовать `ss`):

`-t` - отображает TCP-порты.

`-u` - отображает UDP-порты.

`-n` - отображает числовые адреса вместо разрешения имен.

4. nslookup:

- Выполняет DNS-запросы для определения IP-адресов доменных имен. Ключи могут меняться в зависимости от системы и используемого DNS-клиента.

5. route:

`add` - добавляет маршрут.

`delete` - удаляет маршрут.

`print` - отображает таблицу маршрутизации.

6. ping:

`-c <число>` - отправляет указанное количество запросов.

`-s <размер>` - указывает размер пакета данных.

7. traceroute:

`traceroute <адрес>` - отслеживает маршрут до указанного адреса.

`-I` - использует ICMP-пакеты для трассировки маршрута.

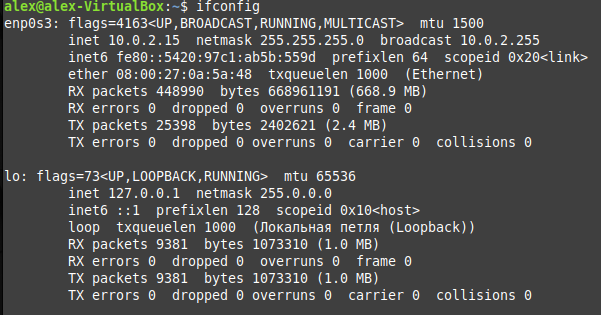
8. hostname:

Показывает текущее имя хоста. Нет ключей для данной утилиты.

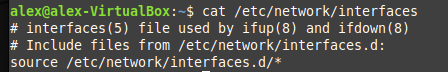
Доступные ключи могут различаться в зависимости от дистрибутива Linux.

1. Определите параметры подключения компьютера к сети, для чего откройте окно сетевых настроек.

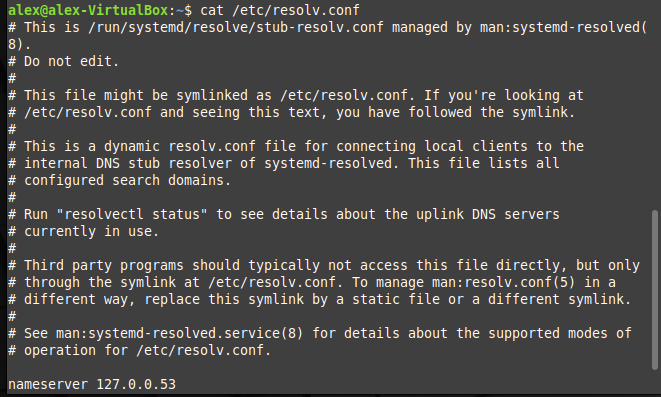
Информация о сетевых интерфейсах, включая IP-адреса и другие параметры подключения



Настройки сети в файлах: **cat /etc/network/interfaces**



Для DNS-серверов: cat /etc/resolv.conf.



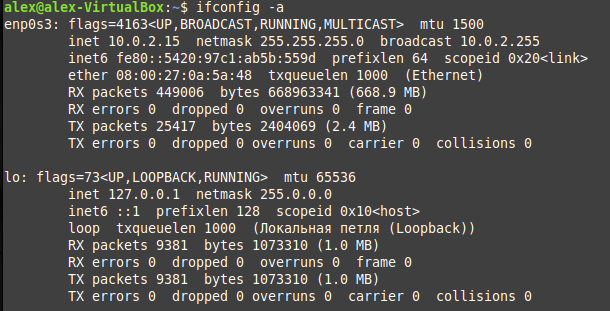
Имя хоста: cat /etc/hostname

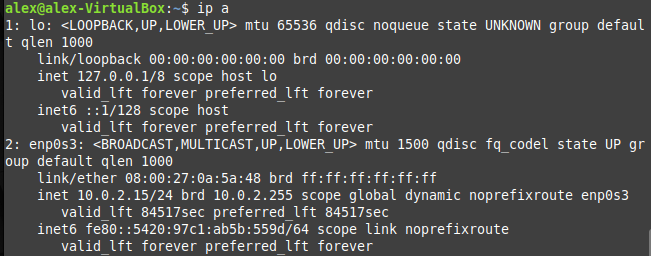


1. Определите те же параметры, используя консольные команды. Посмотрите, сколько сетевых интерфейсов используется в компьютере. Определите среди них основной.

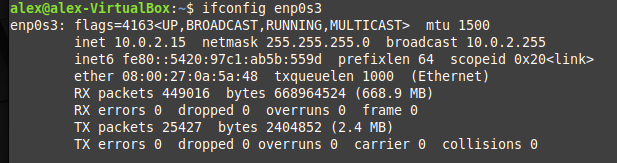
Посмотреть список сетевых интерфейсов: ifconfig -a

Эта команда выведет список всех сетевых интерфейсов на компьютере, включая неактивные.

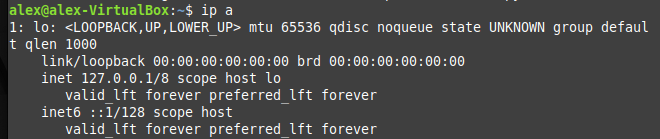




Определить основной сетевой интерфейс: ifconfig enp0s3



1. Определите, является ли IP-адрес компьютера публичным или приватным. Сделайте вывод о типе подключения к Интернет.



IP-адрес "127.0.0.1" принадлежит к диапазону "127.0.0.0/8", и он является адресом петлевого интерфейса (loopback). Этот адрес всегда указывает на текущий компьютер, и он используется для локальных коммуникаций на самом компьютере.

Компьютер использует собственный петлевой интерфейс для локальных коммуникаций и не соединен с внешней сетью посредством этого адреса. Этот адрес не является публичным или приватным IP-адресом, он служит для внутренних целей на самом компьютере.

1. Произведите расчет подсети, к которой принадлежит ваш компьютер.



Для расчета подсети, к которой принадлежит компьютер с IP-адресом 127.0.0.1 и маской 255.0.0.0, давайте выполним следующие шаги:

1. Преобразуем IP-адрес и маску подсети в двоичное представление:

- IP-адрес 127.0.0.1: 01111111.00000000.00000000.00000001

- Маска подсети 255.0.0.0: 11111111.00000000.00000000.00000000

2. Выполним побитовое "И" (AND) между IP-адресом и маской подсети:

01111111.00000000.00000000.00000001 (IP-адрес)

11111111.00000000.00000000.00000000 (Маска подсети)

-------------------------------

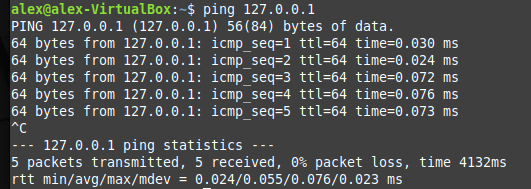
01111111.00000000.00000000.00000000 (Результат)

3. Преобразуем результат обратно в десятичное представление:

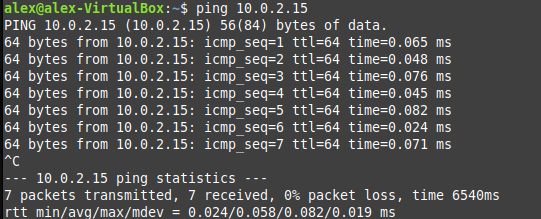
-Результат: 127.0.0.0

Таким образом, компьютер с IP-адресом 127.0.0.1 и маской 255.0.0.0 принадлежит подсети с адресом 127.0.0.0. Однако, важно отметить, что в контексте адреса 127.0.0.1 и маски 255.0.0.0, эта подсеть не используется для сетевых коммуникаций внутри или за пределами компьютера, так как это адрес петлевого интерфейса, предназначенный для локальных коммуникаций на самом компьютере.

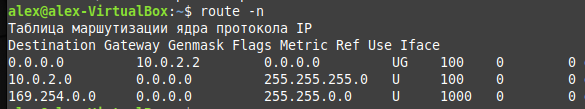
1. Произведите диагностику работоспособности сети при помощи команды ping последовательно проверив доступ к:
2. интерфейсу обратного вызова (127.0.0.1);

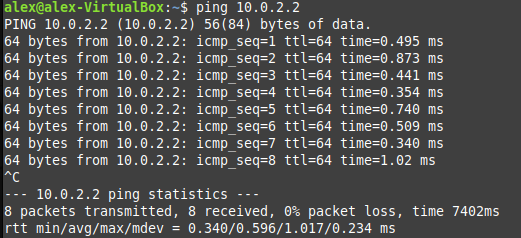


1. IP-адресу самого компьютера;

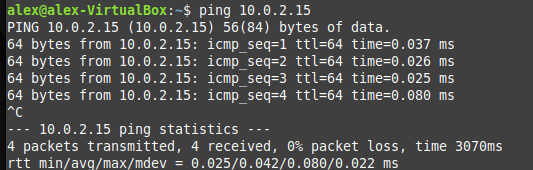


1. IP-адресу шлюза по умолчанию;

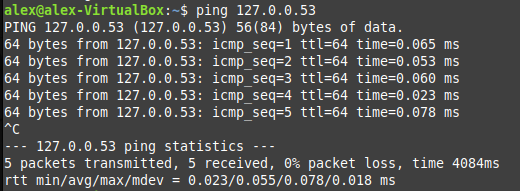




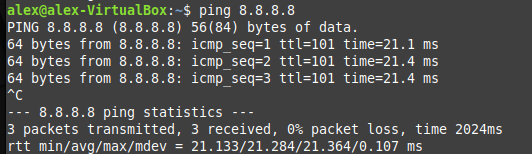
1. IP-адресу другого (включенного) хоста той же подсети;



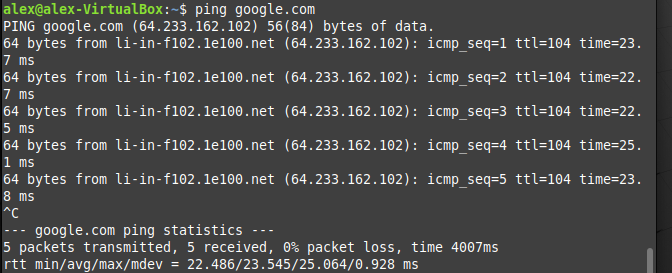
1. IP-адресу DNS-сервера по умолчанию;



1. IP-адресу публичного сервера Интернет (например, 8.8.8.8 - публичный DNS-сервер Гугла);



1. DNS-имени публичного сервера;



1. Сделайте вывод о работоспособности сети и возможных проблемах.

1. Интерфейс обратного вызова (127.0.0.1)

- Если пинг к 127.0.0.1 успешен, это свидетельствует о том, что локальный петлевой интерфейс работает нормально. Проблем с самим компьютером скорее всего нет.

2. IP-адрес самого компьютера

- Если пинг к IP-адресу вашего собственного компьютера успешен, это также указывает на нормальную работоспособность вашего сетевого интерфейса.

3. IP-адрес шлюза по умолчанию

- Успешный пинг к шлюзу по умолчанию (обычно вашему маршрутизатору) означает, что вы можете связаться с устройством, которое соединяет вас со внешней сетью. Отсутствие ответа на пинг может указывать на проблемы с вашим маршрутизатором.

4. IP-адрес другого хоста в той же подсети

- Успешный пинг к другому хосту в вашей локальной сети свидетельствует о работоспособности локальной сети. Неуспешные попытки могут указывать на проблемы с другим устройством или сетевой настройкой.

5. IP-адрес хоста из другой подсети

- Успешный пинг к хосту из другой подсети зависит от наличия правильных маршрутов между подсетями. Если пинг успешен, это означает, что сетевая маршрутизация работает корректно.

6. IP-адрес DNS-сервера по умолчанию

- Успешный пинг к DNS-серверу указывает на доступность сервера для разрешения доменных имен. Если пинг неудачен, это может означать проблемы с DNS-сервером.

7. IP-адрес публичного сервера Интернета

- Успешный пинг к публичному серверу, такому как 8.8.8.8 (DNS-сервер Google), указывает на работоспособность вашего доступа в Интернет. Неудачные попытки могут указывать на проблемы с вашим интернет-подключением.

8. DNS-имя публичного сервера

- Успешный пинг по DNS-имени публичного сервера указывает на работоспособность DNS-разрешения имени. Если пинг по имени неудачен, это может свидетельствовать о проблемах с настройками DNS.

Исходя из результатов пингов и их успешности, вы можете сделать вывод о работоспособности сети и возможных проблемах. Если какие-либо пинги не проходят успешно, это может указывать на проблемы в соответствующих частях сети или настройках вашей системы.

**Дополнительные задания**

1. Выполните основные задания лабораторной в среде Windows и Linux, на реальной и виртуальной машине. Объясните разницу (или сходство) результатов.

Лабораторные работы, проводимые на реальных или виртуальных машинах с операционными системами Linux и Windows, имеют сходства и различия в зависимости от конкретных задач и целей лабораторной работы

Сходства:

1. Основные сетевые концепции: Вне зависимости от операционной системы, лабораторные работы по сетевым темам обычно включают в себя изучение основных сетевых концепций, таких как IP-адресация, маршрутизация, DNS и т.д. Эти концепции справедливы как для Linux, так и для Windows.

2. Использование терминала (командной строки): Как в Linux, так и в Windows, можно использовать командный интерфейс (терминал) для выполнения различных сетевых команд, таких как `ping`, `ifconfig` (в Linux) и `ipconfig` (в Windows), `route`, `traceroute`, и так далее.

3. Работа с сетевыми протоколами: Лабораторные работы могут включать в себя настройку и отладку сетевых протоколов, таких как TCP/IP, DHCP, DNS, и других. Эти протоколы широко используются как в Linux, так и в Windows.

Различия:

1. Операционные системы: Очевидным различием является операционная система, используемая в лабораторной работе. Linux и Windows имеют разные команды, настройки и интерфейсы. Например, в Linux вы используете команду `ifconfig` для настройки сети, в то время как в Windows используется `ipconfig`. Также, в Linux используется оболочка Bash, в то время как в Windows - командная строка CMD или PowerShell.

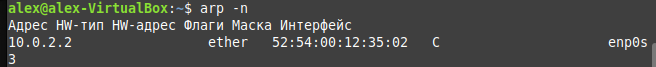
2. Сетевая структура: В зависимости от лабораторной работы, структура сети может различаться. Виртуальные машины могут быть настроены для взаимодействия через разные сетевые адаптеры и маршрутизацию. Разница может возникнуть в настройке сетевых параметров виртуальных машин, которые могут быть разными для Linux и Windows.

3. Инструменты и программное обеспечение: В Linux и Windows существует различное программное обеспечение и инструменты для сетевых задач. Например, в Linux вы можете использовать `iptables` для настройки брандмауэра, а в Windows - `Windows Firewall`. Это также относится к инструментам мониторинга сети и другим утилитам.

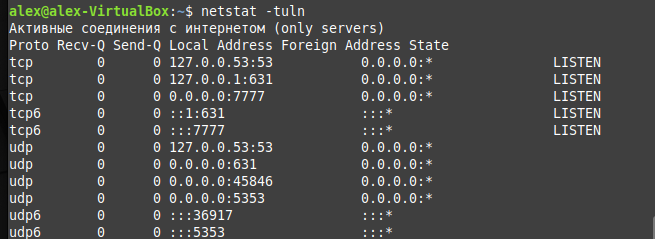
4. Общие сценарии: Часть лабораторных работ может быть уникальной для определенных операционных систем. Например, работа с Active Directory будет специфичной для Windows, в то время как создание скриптов на Bash будет специфичным для Linux.

Итак, сходства и различия лабораторных работ на Linux и Windows зависят от конкретных задач и целей лабораторной работы. Важно учитывать контекст и требования к работе для выбора правильного инструментария и подхода.

1. С помощью утилиты arp просмотрите и выпишите arp-таблицу локального компьютера (несколько записей).



1. Получите список активных TCP-соединений локального компьютера. Поясните полученный результат.



Опции `-tuln` в командах `netstat` и `ss` означают следующее:

1. -t: Эта опция фильтрует вывод, показывая только информацию о TCP-соединениях.

2. -u: Эта опция фильтрует вывод, показывая только информацию о UDP-соединениях.

3. -l: Эта опция означает "листен" и фильтрует вывод, показывая только слушающие (принимающие) порты, на которых ожидают входящих соединений.

4. -n: Эта опция отключает разрешение DNS и выводит IP-адреса в числовом формате вместо попыток разрешения на имена хостов. Это полезно, чтобы ускорить выполнение команды и избежать задержек, связанных с разрешением DNS.

В данном контексте, комбинация `-tuln` означает, что мы просматриваем только активные TCP-соединения, игнорируя UDP, и показываем только слушающие порты (состояние "LISTEN") с IP-адресами и портами в числовом формате.

- `Proto`: Протокол (TCP).

- `Recv-Q` и `Send-Q`: Количество байт в приемной и передающей очереди.

- `Local Address`: Локальный адрес и порт.

- `Foreign Address`: Удаленный адрес и порт.

- `State`: Состояние соединения, такое как "LISTEN" (ожидание входящего соединения) или "ESTABLISHED" (установлено).

**Контрольные вопросы**

1. Отличие концентратора от коммутатора

- Концентратор (hub) - это устройство на физическом уровне, которое просто повторяет входящий сигнал на все порты. Он не различает устройства в сети и не имеет интеллектуальных функций. Все устройства в сети, подключенные к концентратору, видят весь сетевой трафик.

- Коммутатор (switch) - это устройство на уровне канального (data link) и выше. Он различает устройства по MAC-адресам и умеет пересылать трафик только к нужному устройству, что улучшает производительность сети. Коммутатор более интеллектуальный и эффективный по сравнению с концентратором.

2. Отличие маршрутизатора от коммутатора

- Коммутатор (switch) - это устройство на уровне канального (data link) и выше, предназначенное для коммутации кадров внутри локальной сети (LAN). Он работает на уровне Ethernet и используется для соединения устройств внутри одной сети.

- Маршрутизатор (router) - это устройство на сетевом (network) уровне, которое принимает решения о маршрутизации данных между разными сетями. Он имеет возможность передавать данные между разными сетями, как внутри LAN, так и между LAN и внешними сетями, включая интернет. Маршрутизатор обеспечивает маршрутизацию пакетов на основе IP-адресов.

3. Протокол, необходимый для работы с утилитой ping

- Утилита ping использует Internet Control Message Protocol (ICMP) для отправки эхо-запросов (ping-пакетов) и получения ответов. ICMP - это протокол сетевого уровня, который используется для обмена информацией о состоянии сети и для диагностики сетевых проблем. Он обеспечивает проверку доступности устройств и измерение задержек в сети.

4. Можно ли утилитой tracert задать максимальное число ретрансляций?

- В утилите `tracert` (или `traceroute` в Linux) можно задать максимальное число ретрансляций (hops) при помощи параметра `-m` или `--max-hops`. Например, `tracert -m 10` ограничит максимальное число шагов (ретрансляций) до 10 при трассировке маршрута. Это полезно, чтобы ограничить диапазон поиска маршрута и ускорить выполнение команды.

5. Результат, выдаваемый утилитой netstat с параметрами -a -s -r

- `netstat -a` показывает список всех активных сетевых соединений и слушающих портов.

- `netstat -s` выводит статистику сетевых протоколов.

- `netstat -r` отображает таблицу маршрутизации.

Когда вы выполняете `netstat -a -s -r`, вы получите подробную информацию о сетевых соединениях, статистике сетевых протоколов и маршрутной таблице. Результаты будут включать в себя список портов, состояния соединений, количество отправленных и полученных пакетов, а также информацию о маршрутах в вашей системе. Эта информация полезна для мониторинга сетевой активности и диагностики сетевых проблем.