**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Мегафакультет трансляционных информационных технологий

Факультет информационных технологий и программирования

**Лабораторная работа №1**

**Консольные утилиты настройки сетевых компонентов Linux**

Выполнила студент группы №**М33091**

**Зыонг Тхи Хуэ Линь и Исрат**

Проверил

**Демин Александр Дмитриевич**

САНКТ-ПЕТЕРБУРГ

2024

Часть 1. Linux

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описаниеИзображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана

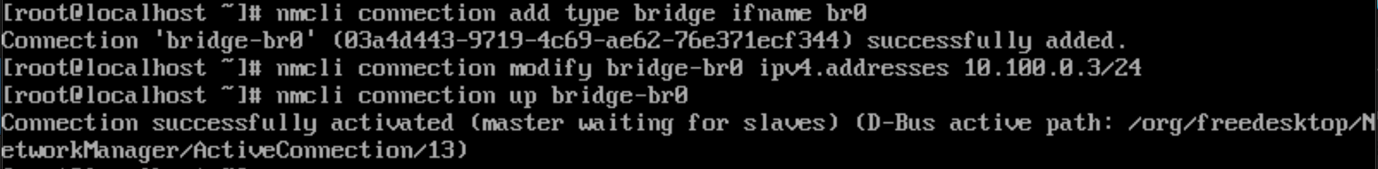
Автоматически созданное описание

Result

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Часть 2.



Изображение выглядит как текст, снимок экрана, черно-белый, черный

Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Часть 3

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, алгебра

Автоматически созданное описание

Text, table

Description automatically generated

Изображение выглядит как текст, снимок экрана

Автоматически созданное описаниеЧасть 4

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**Цель работы:** получить практические навыки по конфигурированию сетевых

интерфейсов (на примере протокола IPv4) в операционных системах Linux,

ознакомится с утилитами командной строки, освоить современные сетевые

менеджеры Linux.

**Вопросы и задания:**

1. **Как с помощью команды ip:**
2. **назначить новый IPv4 адрес?**

**ip addr add [IP adress] dev [interface]**

OR

**sudo ip addr add [IP adress] dev [interface]**

**Note:** К сожалению, все эти настройки будут потеряны после перезагрузки системы.

1. **назначить новый MAC адрес?**

First run: **ip link show**

Это будет список ваших network devices. Найдите тот, который вы хотите изменить.

Next, run: **sudo ip link set dev [your device here] down**

Then: **sudo ip link set dev [your device here] address [your new mac address]**

Finally: **sudo ip link set dev [your device here] up**

1. **назначить новый gateway?**

Если у вас установлено более одного «шлюза» по умолчанию, вы столкнетесь с конфликтами соединений. Удалите существующий шлюз по умолчанию, если вы собираетесь его изменить.:

**sudo route delete default gw [IP adress] [interface]**

Then:

**sudo ip route add default via [IP adress] dev [interface]**

1. **вывести информацию arp кэше?**

**ip neigh show**

1. **очистить arp кэш?**

**sudo ip neigh flush all**

1. **включить интерфейс?**

**sudo ip link set dev [интерфейс] up**

1. **выключить интерфейс?**

**sudo ip link set dev [интерфейс] down**

1. **Как с помощью nmcli назначить на интерфейс статический IP адрес, маску и настроить default gateway?**

Просмотр всех подключений connections: **nmcli connection show**

Затем назначьте статический IP address, mask, and default gateway to the interface , используя следующую команду:

**nmcli connection modify [network\_connection\_name] ipv4.addresses [IP-adress] ipv4.gateway [default-gateway]**

Перезапустите сетевое соединение, чтобы применить изменения с помощью команды:

**nmcli connection down [interface] && nmcli connection up**

**[interface]**

1. **Как с помощью netplan назначить на интерфейс статический IP адрес, маску и настроить default gateway?**

Открыть the Netplan configuration file для редактирования. Этот файл обычно находится в **/etc/netplan/** directory and the **.yaml** extension. Например,

**sudo nano /etc/netplan/01-netcfg.yaml**

Внутри файла добавьте информацию о сетевом интерфейсе. Например eth0:

**network: version: 2 renderer: networkd ethernets: eth0: addresses: [192.168.0.2/24] gateway4:192.168.0.1**

Сохраните файл и сделайте: **sudo netplan apply** для применения новой конфигурации.

1. **Какие режимы bonding стандартно существую в Linux? Опишите их назначение, возможности по отказоустойчивости и необходимость поддержки со стороны оборудования.**

**Round-robin**: пакеты отправляются по порядку на каждый интерфейс в bonding-группе, обеспечивая балансировку нагрузки на каждый интерфейс.

**active backup**: используется только один интерфейс, остальные интерфейсы резервные. В случае отказа активного интерфейса переключение на резервный происходит автоматически

**balance-xor**: использует исключающее ИЛИ (XOR) для балансировки нагрузки. Вычисляется хеш-функция на основе MAC-адреса источника и назначения исходящего трафика. Затем пакеты распределяются между интерфейсами в соответствии с хешфункцией

**broadcast**: пакеты отправляются на все интерфейсы в bonding-группе, независимо от того, какой интерфейс используется для получения пакетов.

**802.3ad (Dynamic link aggregation):** агрегирует несколько интерфейсов в логический интерфейс, с возможностью динамического изменения состава группы. Поддержка 802.3ad требуется со стороны коммутатора

**balance-tlb (Adaptive transmit load balancing):** балансирует исходящий трафик между интерфейсами, используя информацию о состоянии интерфейса и маршрутах. Не требует поддержки со стороны коммутатора

**balance-alb (Adaptive load balancing):** балансирует как входящий, так и исходящий трафик между интерфейсами, используя информацию о состоянии интерфейса и маршрутах. Не требует поддержки со стороны коммутатора.

1. **Какие существуют и чем отличаются режимы работы адаптера**

**(duplex)?**

Адаптеры (сетевые карты) имеют два режима работы duplex: полудуплекс (half-duplex) и полный дуплекс (full-duplex).

Полудуплексный режим означает, что передача данных может осуществляться только в одном направлении за раз. Например, если один компьютер отправляет данные на другой, то в этот момент другой компьютер не может отправлять данные обратно. Вместо этого он должен ждать, пока первый компьютер закончит передачу данных.

Полный дуплексный режим позволяет одновременную передачу и прием данных в обоих направлениях. То есть, два компьютера могут одновременно отправлять и принимать данные друг от друга, без необходимости ждать окончания передачи в одном направлении, чтобы начать передачу в другом.

Выбор режима работы зависит от типа сетевых устройств и используемых сетевых технологий. Некоторые сетевые технологии поддерживают только полудуплексный режим, а другие - только полный дуплекс. В случае использования полудуплексного режима возможны коллизии, когда два устройства одновременно пытаются передавать данные, что может привести к потере данных или повторной передаче. Поэтому, если сетевое устройство и технология позволяют, лучше использовать полный дуплексный режим.

1. **Какой, по-вашему, практический смысл в возможности назначения нескольких IP адресов на один интерфейс?**

Назначение нескольких IP-адресов на один интерфейс может иметь несколько практических применений:

+ Разделение сетевых сервисов: один и тот же физический сервер может предоставлять несколько сетевых сервисов, таких как веб-сервер, почтовый сервер, DNS-сервер и т. д. Каждый сервис может использовать свой собственный IP-адрес, чтобы обеспечить логическое разделение между сервисами и облегчить их управление.

+ Мультихоминг: настройка нескольких IP-адресов на одном интерфейсе может использоваться для обеспечения отказоустойчивости и балансировки нагрузки в сети. Например, если один из интерфейсов сетевого устройства выходит из строя, трафик может быть перенаправлен на другой интерфейс с помощью нескольких IP-адресов.

+ Виртуализация: настройка нескольких IP-адресов на виртуальной машине может использоваться для разделения виртуальных сетевых сервисов и управления ими.

+ Обслуживание нескольких сетей: настройка нескольких IP-адресов на сетевом интерфейсе может использоваться для обслуживания нескольких сетей, например, когда сеть нужно разделить на несколько подсетей для обеспечения безопасности или оптимизации трафика.

В целом, назначение нескольких IP-адресов на один интерфейс позволяет повысить гибкость и эффективность управления сетью, а также обеспечить отказоустойчивость и балансировку нагрузки.

1. **Какой, по-вашему, практический смысл в возможности создания виртуальных интерфейсов?**

Создание виртуальных интерфейсов может иметь несколько практических применений:

+ Виртуализация: виртуальные интерфейсы могут использоваться для виртуализации сетевых ресурсов, таких как виртуальные машины. Каждая виртуальная машина может иметь свой собственный виртуальный интерфейс, который связан с реальным сетевым интерфейсом хоста, чтобы обеспечить доступ к сети.

+ Разделение сетевых сервисов: виртуальные интерфейсы могут использоваться для разделения сетевых сервисов на одном физическом сервере. Например, на одном сервере может быть установлен веб-сервер и почтовый сервер, и каждый из них может использовать свой собственный виртуальный интерфейс для доступа к сети.

+ Обслуживание нескольких сетей: виртуальные интерфейсы могут использоваться для обслуживания нескольких сетей на одном физическом интерфейсе. Например, на одном сетевом интерфейсе может быть настроено несколько виртуальных интерфейсов, каждый из которых связан с отдельной подсетью.

+ Тестирование сети: виртуальные интерфейсы могут использоваться для тестирования сети. Например, можно создать виртуальный интерфейс для эмуляции трафика в сети, чтобы проверить ее производительность и отказоустойчивость.

В целом, создание виртуальных интерфейсов позволяет повысить гибкость и эффективность управления сетью, а также обеспечить логическое разделение между сетевыми ресурсами и управлять ими отдельно друг от друга.