МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение   
высшего образования

«КРЫМСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ им. В. И. ВЕРНАДСКОГО»

ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ

Кафедра компьютерной инженерии и моделирования

**ОТЧЕТ ПО ПРАКТИЧЕСКОМУ ЗАДАНИЮ №3**

**«Виртуализация. Эмуляция. Гипервизоры.»**

Практическая работа

по дисциплине «Современные технологии программирования»

студента 1 курса группы ПИ-б-о-231(2)

Аметов Кемран Ленверович

направления подготовки 09.03.04 «Программная инженерия»

|  |  |  |
| --- | --- | --- |
|  |  |  |

Симферополь, 2024

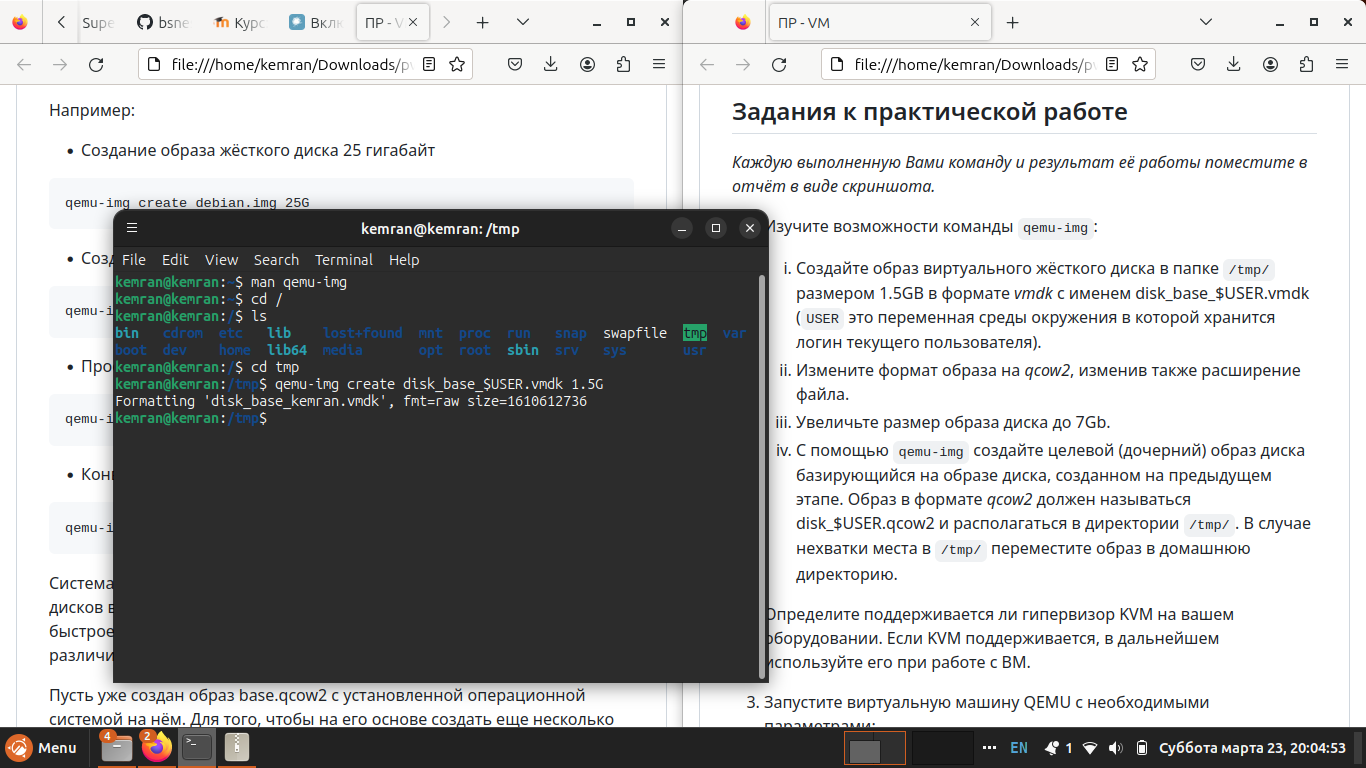
**Цель:**

изучить основы работы гипервизоров, получить навыки работы с гипервизором KVM ядра Linux, а также подготовить базу для выполнения следующих лабораторных и практических занятий.

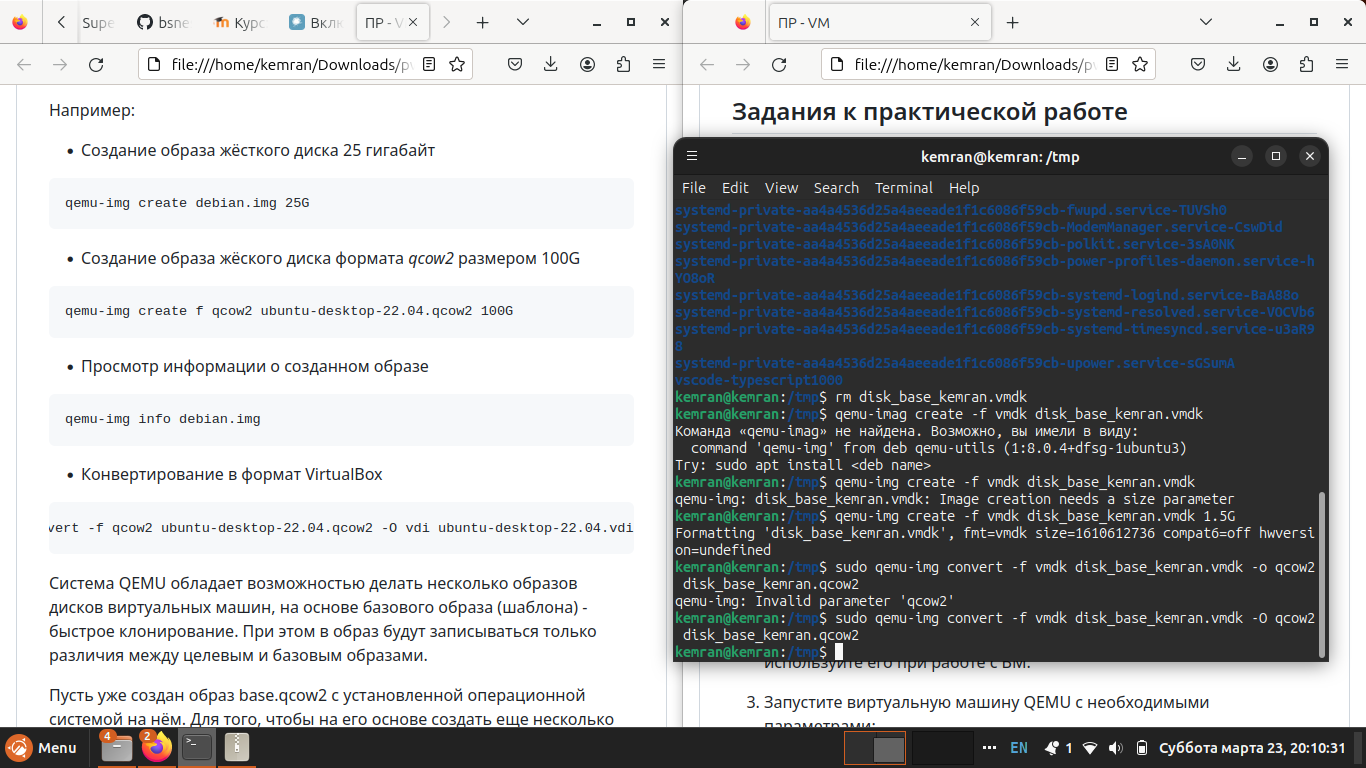
**Ход выполнения задания.**

Изучите возможности команды qemu-img:

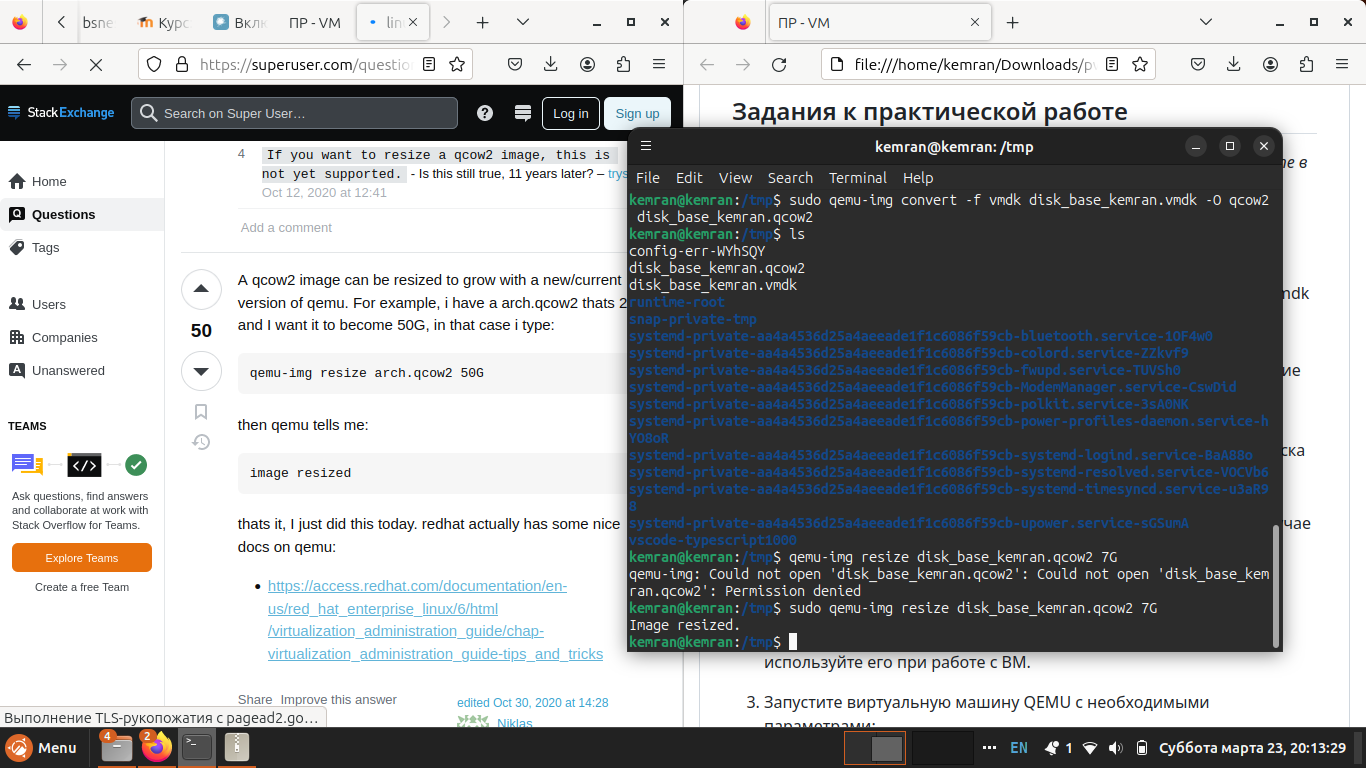
* 1. Создайте образ виртуального жёсткого диска в папке /tmp/ размером 1.5GB в формате vmdk с именем disk\_base\_$USER.vmdk (USER это переменная среды окружения в которой хранится логин текущего пользователя).



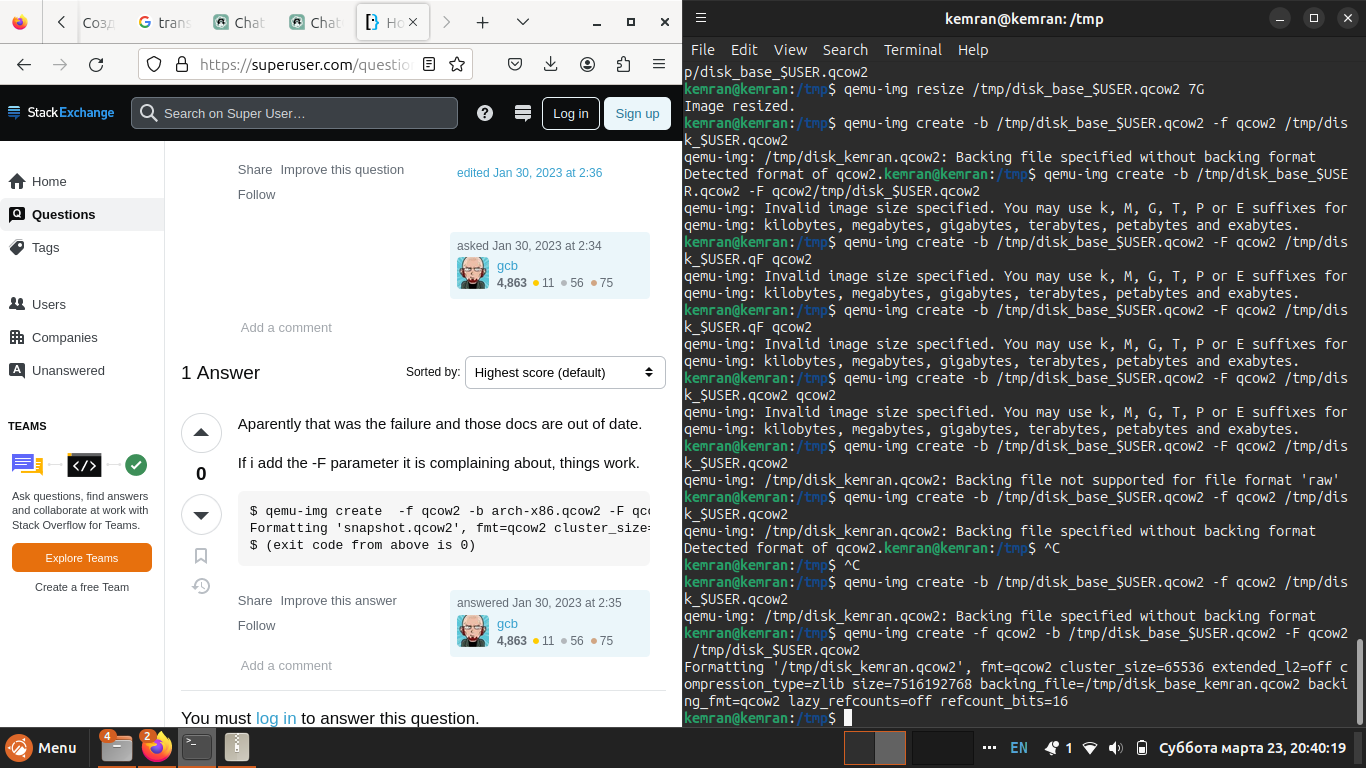
* 1. Измените формат образа на qcow2, изменив также расширение файла.



* 1. Увеличьте размер образа диска до 7Gb.



* 1. С помощью qemu-img создайте целевой (дочерний) образ диска базирующийся на образе диска, созданном на предыдущем этапе. Образ в формате qcow2 должен называться disk\_$USER.qcow2 и располагаться в директории /tmp/. В случае нехватки места в /tmp/ переместите образ в домашнюю директорию.

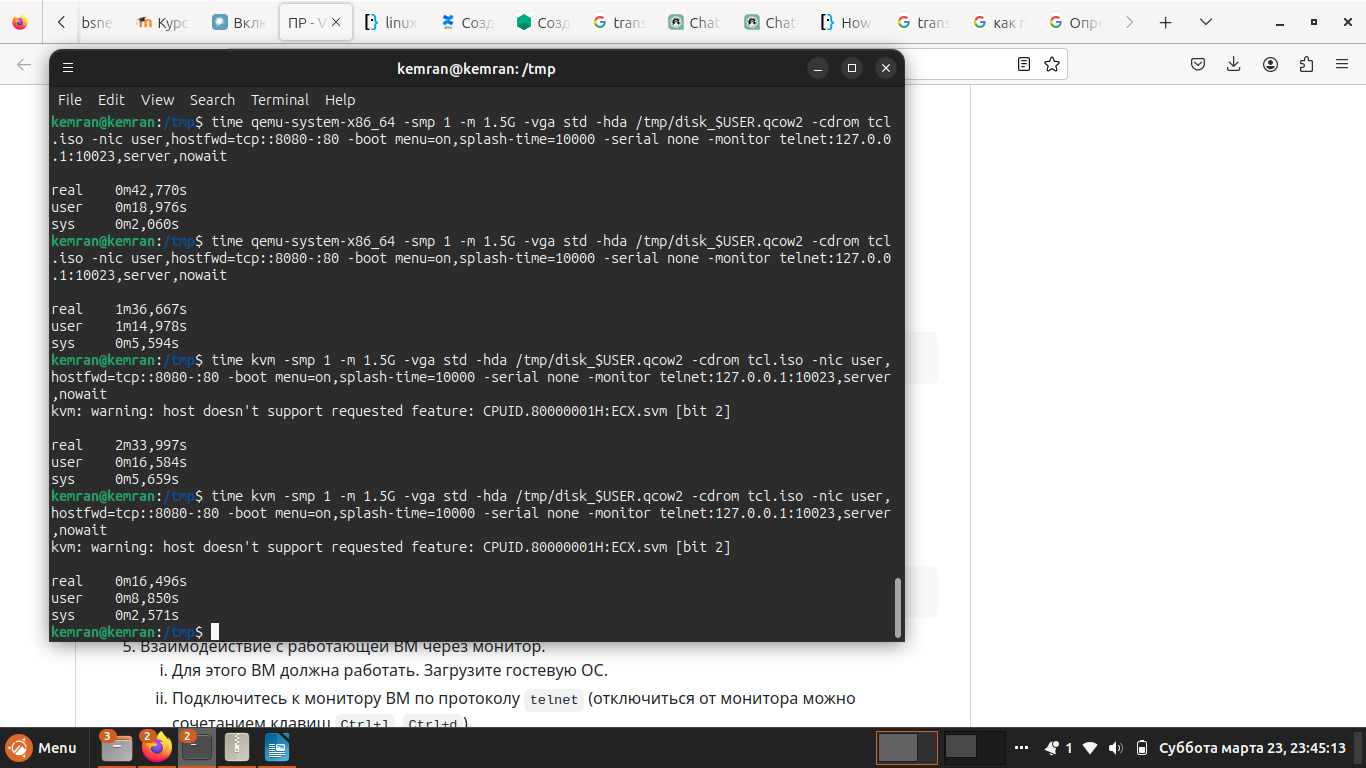


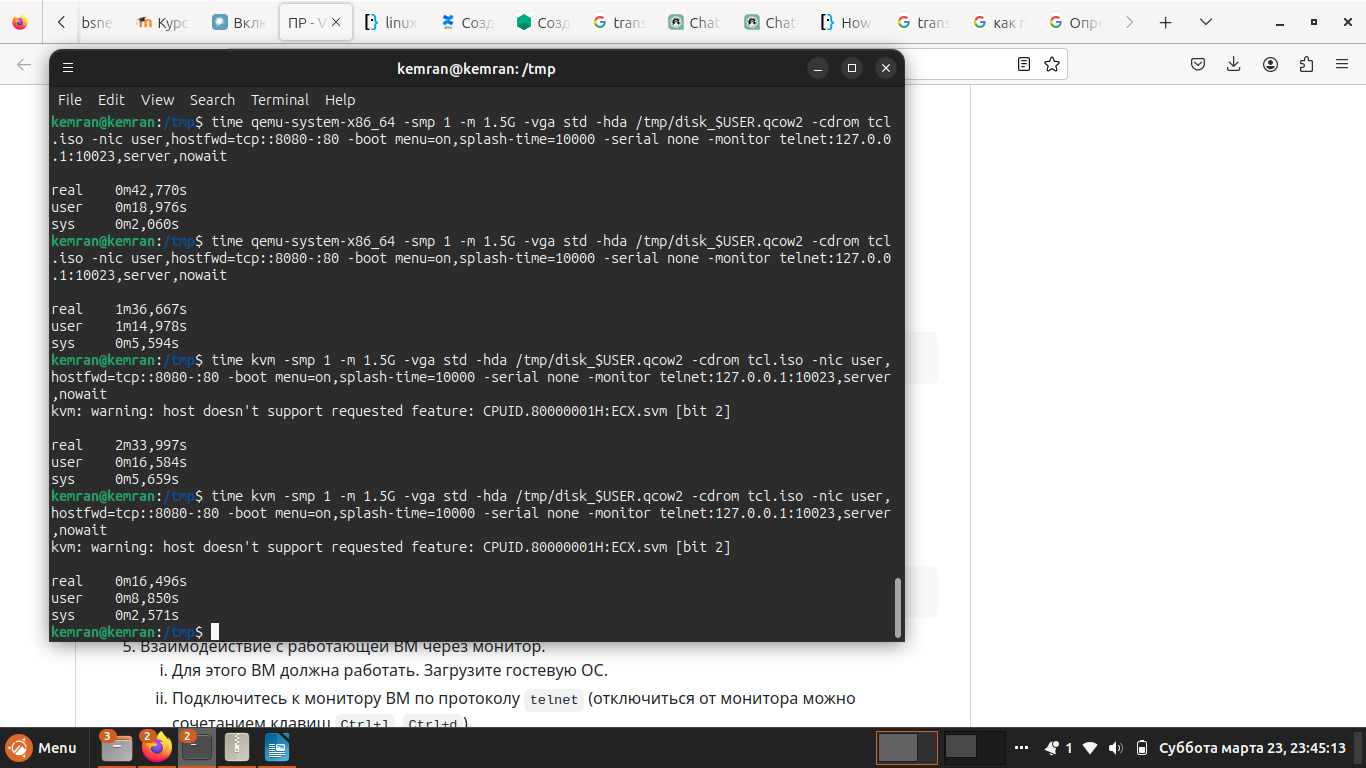
1. Определите поддерживается ли гипервизор KVM на вашем оборудовании. Если KVM поддерживается, в дальнейшем используйте его при работе с ВМ.
2. Запустите виртуальную машину QEMU с необходимыми параметрами:
   1. Количество процессоров: 1
   2. Оперативная память: 1.5G
   3. Тип эмулируемой видеокарты: std
   4. Образ жёсткого диска образ: уже созданный образ (целевой) Вами на предыдущем этапе лабораторной работы
   5. Файл CD-ROM: [tcl.iso](http://tinycorelinux.net/15.x/x86/release/Core-current.iso)
   6. Сеть: пользовательская
   7. Проброс портов: порт хост-компьютера = 8080 на порт виртуальной машины = 80
   8. Включите отображение меню выбора устройства для загрузки
   9. Таймаут отображения меню: 10 секунд
   10. Дополнительные опции:

-serial none -monitor telnet:127.0.0.1:10023,server,nowait

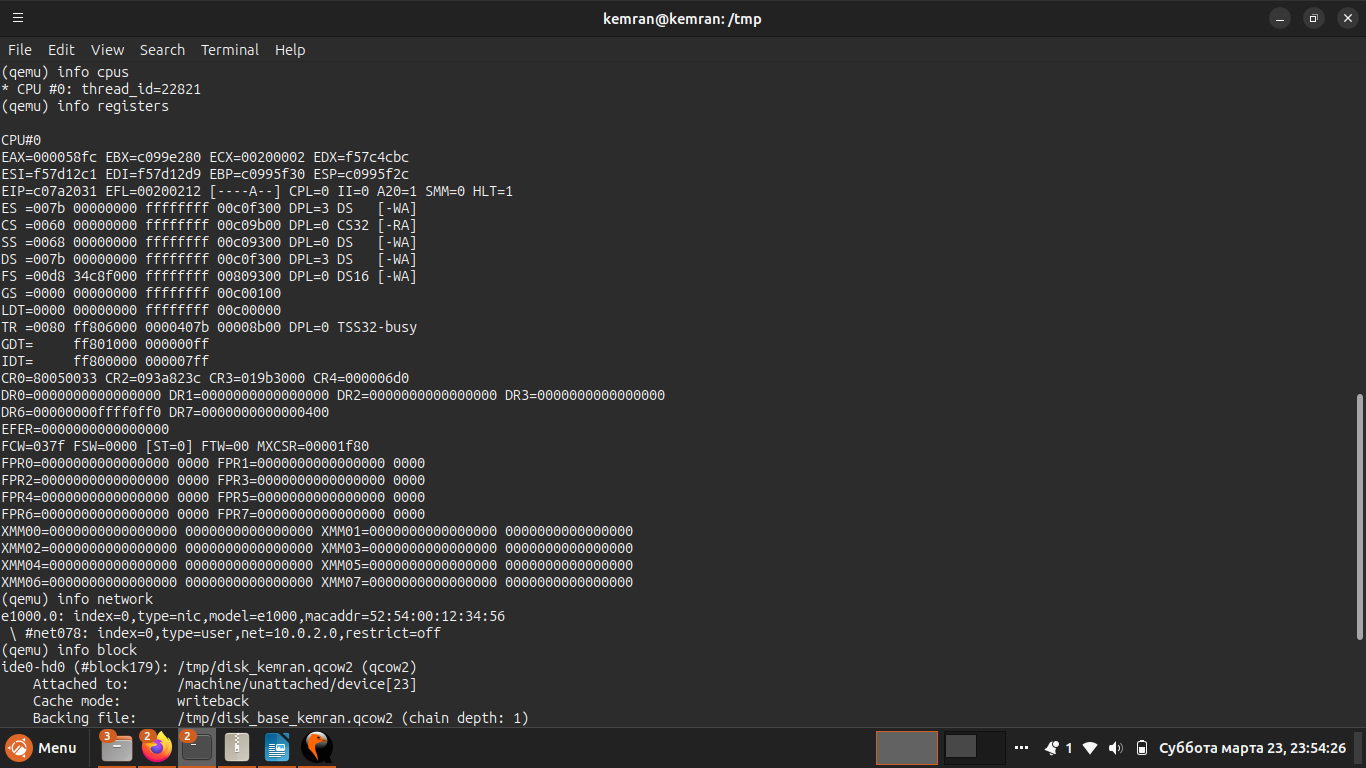
1. Если на вашем оборудовании поддерживается гипервизор KVM, то:
   * Произведите измерение времени полной загрузки системы в режиме эмуляции и в режиме виртуализации.
   * Произведите измерение времени исполнения следующей команды в режиме эмуляции и в режиме виртуализации (измерить точное время исполнения можно с помощью команды time).

ssh-keygen -M generate -O bits=512 /tmp/1.txt

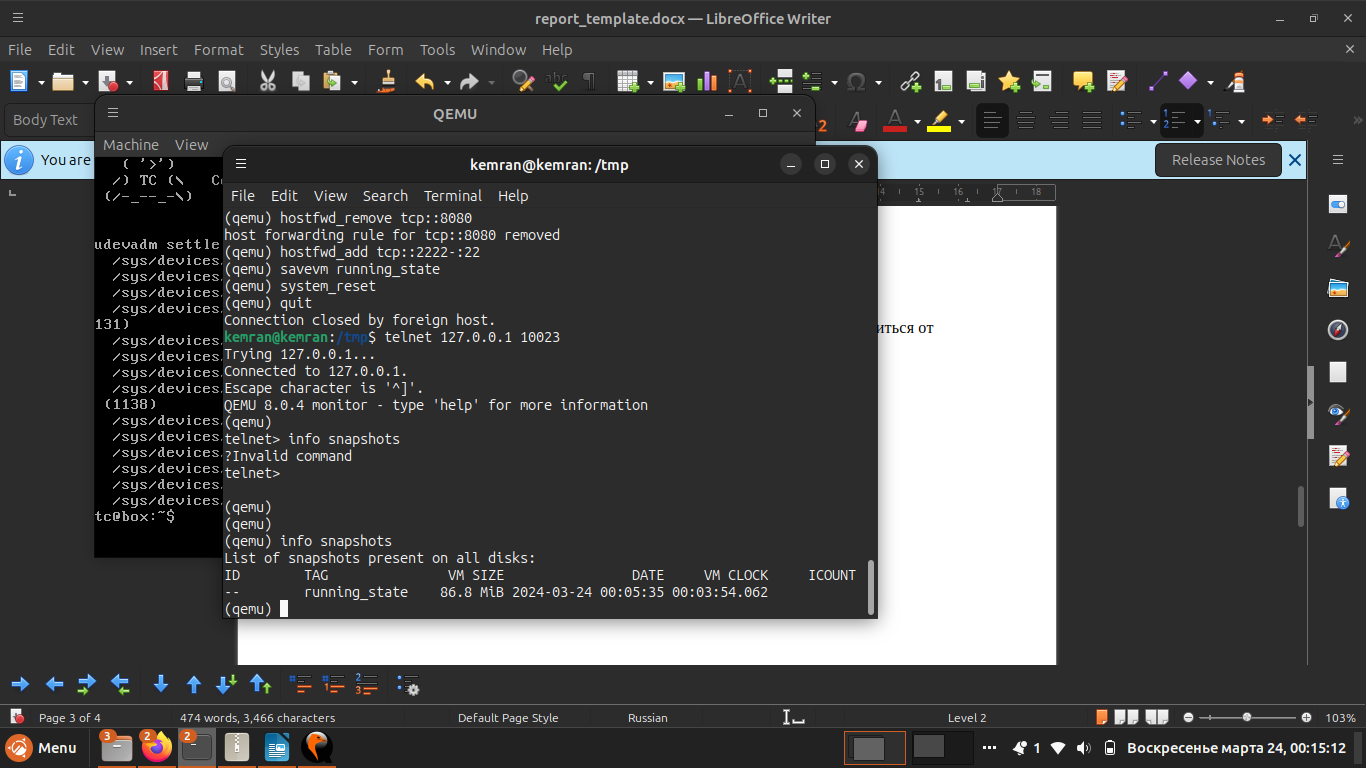


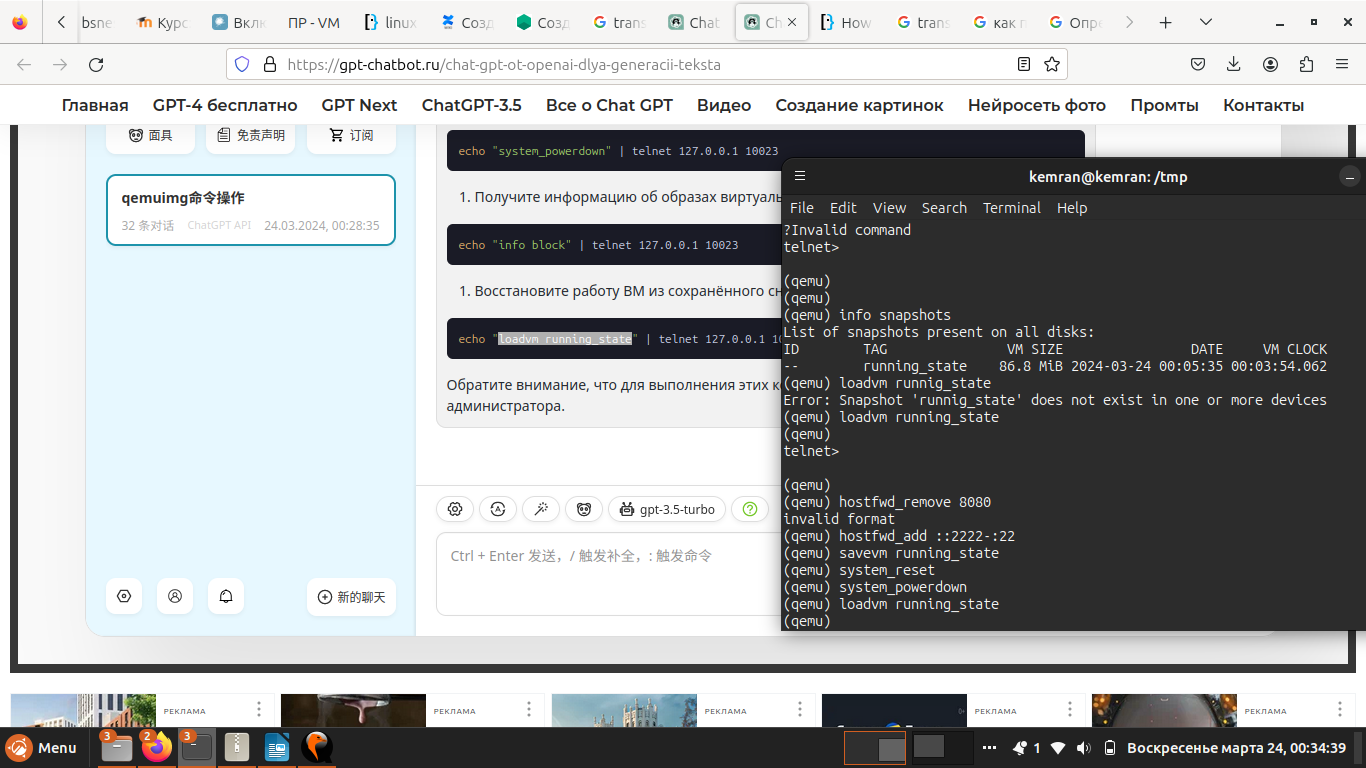


1. Взаимодействие с работающей ВМ через монитор.
   1. Для этого ВМ должна работать. Загрузите гостевую ОС.
   2. Подключитесь к монитору ВМ по протоколу telnet (отключиться от монитора можно сочетанием клавиш Ctrl+], Ctrl+d).
   3. Получите информацию о:
      * процессорах
      * регистрах процессоров
      * сети
      * блочных устройствах



* 1. Удалите существующий проброс портов:
     + порт хост-компьютера = 8080 на порт виртуальной машины = 80.
  2. Добавьте новый проброс портов к виртуальной машине:
     + порт хост-компьютера = 2222 на порт виртуальной машины = 22.
  3. Выполните сохранение текущего состояния ВМ с тегом "running\_state".
  4. Перезагрузите виртуальную систему (аналог нажатия кнопки "Reset" на системном блоке).
  5. Принудительно завершите работу ВМ (равноценно отключению питания для обычного компьютера).
  6. Получите информацию об образах виртуальной машины, которые вы создавали и использовали во время работы ВМ. Какой объём они занимают в данный момент? Какие снимки состояния в них хранятся?
  7. Восстановите работу ВМ из сохранённого снимка состояния. Обратите внимание, что работа ВМ мгновенно должна продолжиться в точности с момента сохранения.





**Ответы на вопросы.**

1. KVM (Kernel-based Virtual Machine) - это гипервизор, встроенный в ядро Linux. Он позволяет запускать несколько виртуальных машин на одном физическом сервере.

2. QEMU (Quick Emulator) - это программное обеспечение для эмуляции и виртуализации. Оно может эмулировать аппаратное обеспечение, такое как процессоры и устройства ввода-вывода, а также виртуализировать гостевые операционные системы.

3. KVM можно использовать, если ваш процессор поддерживает аппаратную виртуализацию (Intel VT-x или AMD-V), и если вы используете 64-битную версию Linux. Кроме того, ваша операционная система должна быть ядром Linux 2.6.20 или более поздней версии.

4. Монитор QEMU - это интерфейс командной строки, который позволяет управлять виртуальной машиной, запущенной в QEMU. Он может использоваться для управления процессором, памятью, устройствами ввода-вывода и другими аспектами виртуализации.

5. Некоторые основные команды монитора QEMU включают `info` (для получения информации о состоянии виртуальной машины), `cont` (для продолжения выполнения виртуальной машины), `stop` (для остановки виртуальной машины), `savevm` (для сохранения состояния виртуальной машины) и `loadvm` (для загрузки сохраненного состояния виртуальной машины).

6. Эмуляция - это процесс, при котором программное обеспечение эмулирует аппаратное обеспечение, такое как процессоры и устройства ввода-вывода. Это позволяет запускать программное обеспечение, разработанное для одной архитектуры, на аппаратуре с другой архитектурой.

7. Виртуализация - это процесс, при котором гипервизор позволяет нескольким виртуальным машинам работать на одном физическом сервере. Каждая виртуальная машина имеет свою собственную операционную систему и приложения, и они могут работать независимо друг от друга.

**Вывод:**

Изучил основы гипервизоров, получил навыки работы с KVM-гипервизором ядра Linux, а также подготовил основу для последующих лабораторных и практических занятий.