Отчёт по лабораторной работе №1

*дисциплина: Операционные системы*

Максим Александрович Мишонков

Содержание

# 1 Цель работы

Целью данной лабораторной работы является установка операционной системы Linux (дистрибутив Fedora 36) на виртуальную машину VirtualBox и настройка минимально необходимых параметров для дальнейшей работы с системой.

# 2 Теоретическое введение

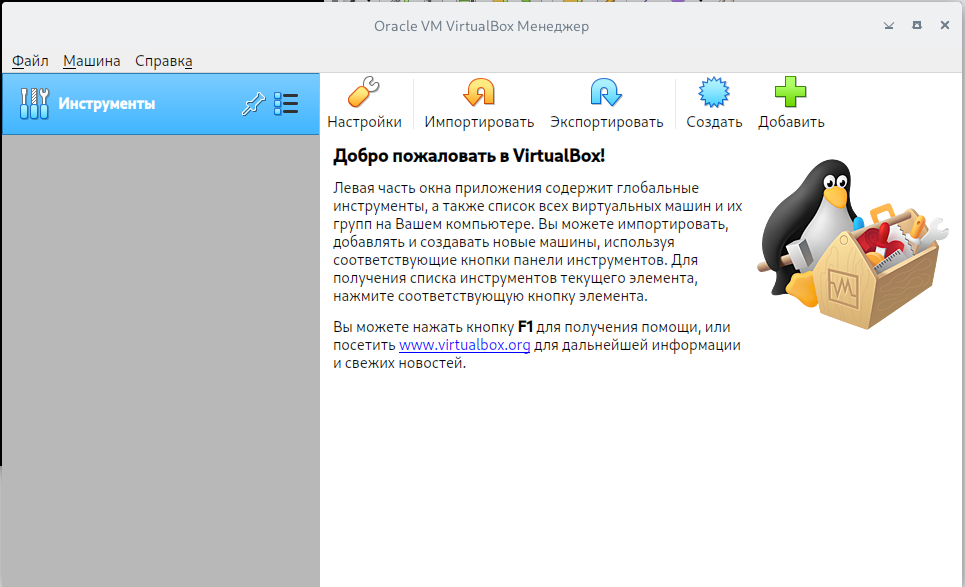
**Операционная система (OC)** - это комплекс взаимосвязанных программ, предназначенных для управления ресурсами компьютера и организации взаимодействия с пользователем.

**GNU Linux** - это семейство переносимых, многозадачных и многопользовательских систем на базе ядра Linux, включающих тот или иной набор утилит и программ проекта GNU, и, возможно, другие компоненты. Как и ядро Linux, системы на его основе, как правило, создаются и распространяются в соответствии с моделью разработки свободного и открытого программного обеспечения.

**Дистрибутив GNU Linux** - это общее определение ОС, использующих ядро Linux и набор библиотек и утилит, выпускаемых в рамках проекта GNU, а также графическую оконную подсистему X Window System. Кроме ядра и, собственно, операционной системы дистрибутивы обычно содержат широкий набор приложений, таких как редакторы документов и таблиц, мультимедийные проигрыватели, системы для работы с базами данных и т.д.

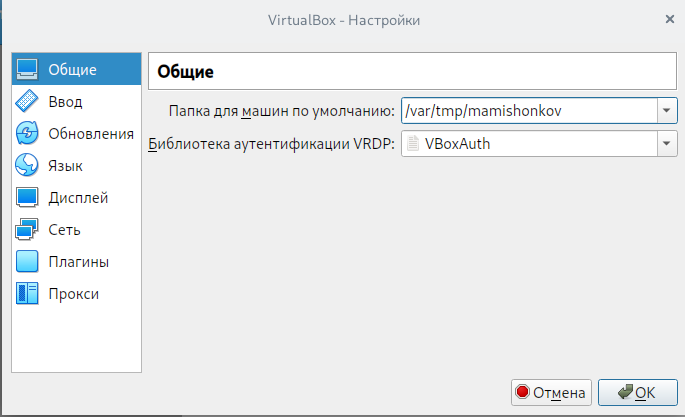
# 3 Выполнение лабораторной работы

1. Запустил виртуальную машину, введя в командной строке VirtualBox &. (рис. [??])



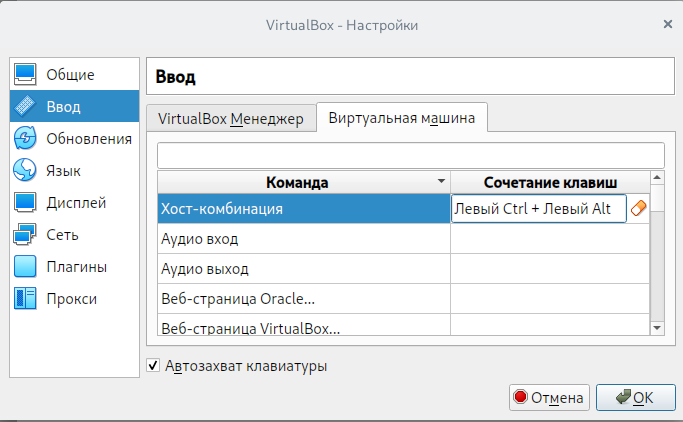
Запуск виртуальной машины

1. В настройках VirtualBox изменил месторасположение каталога для виртуальных машин, указав учётную запись на месте «имя пользователя». (рис. [??])



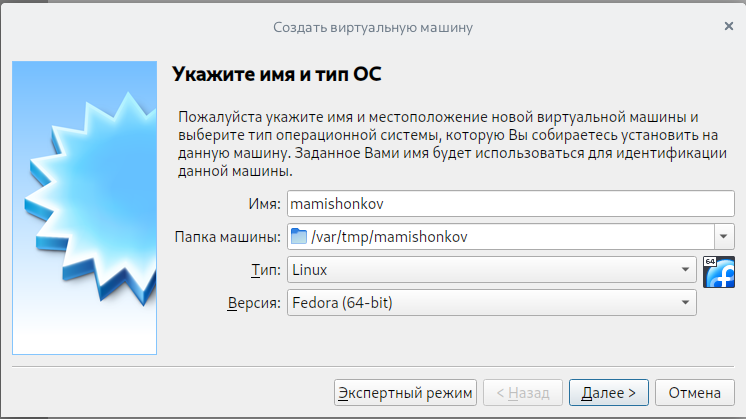
Окно “Свойства VirtualBox”

1. Сменил комбинацию хост-клавиши, использующейся для освобождения курсора мыши, который может захватывать виртуальная машина. (рис. [??])



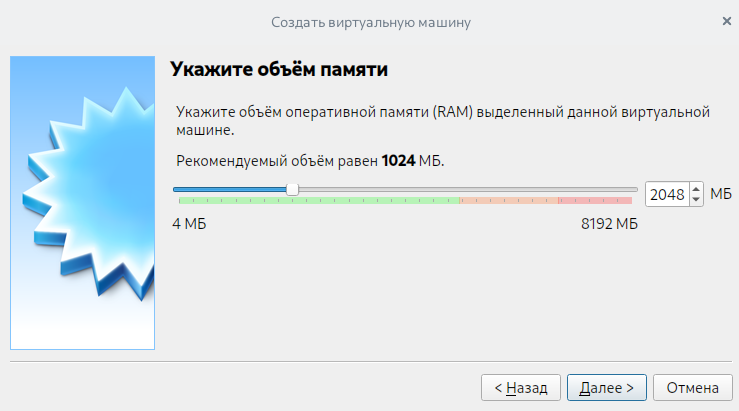
Смена хост-клавиши

1. Создадал новую виртуальную машину, указав имя виртуальной машины (учётная запись) и тип операционной системы (Linux, Fedora (64-bit)). (рис. [??])



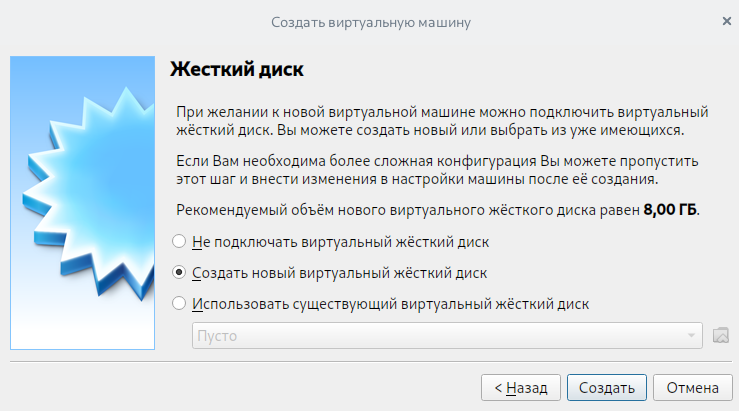
Окно “Имя машины и тип ОС”

1. Указал размер основной памяти виртуальной машины (2048 МБ). (рис. [??])

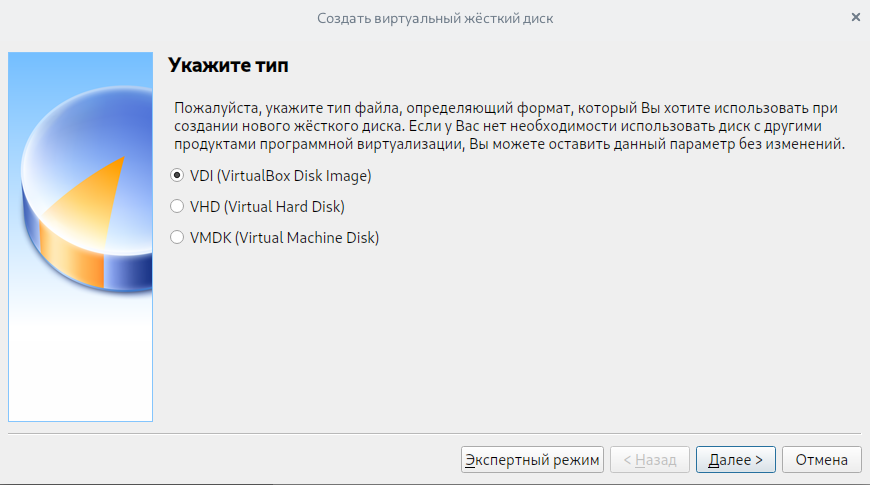


Окно “Размер основной памяти”

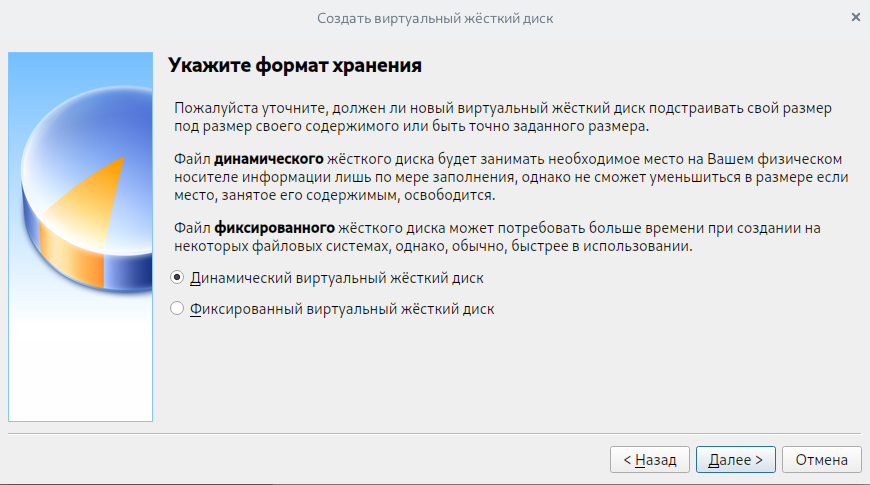
1. Задал конфигурацию жёсткого диска (загрузочный, VDI, динамический виртуальный диск). (рис. [??], [??], [??])



Окно создания жёсткого диска на виртуальной машине

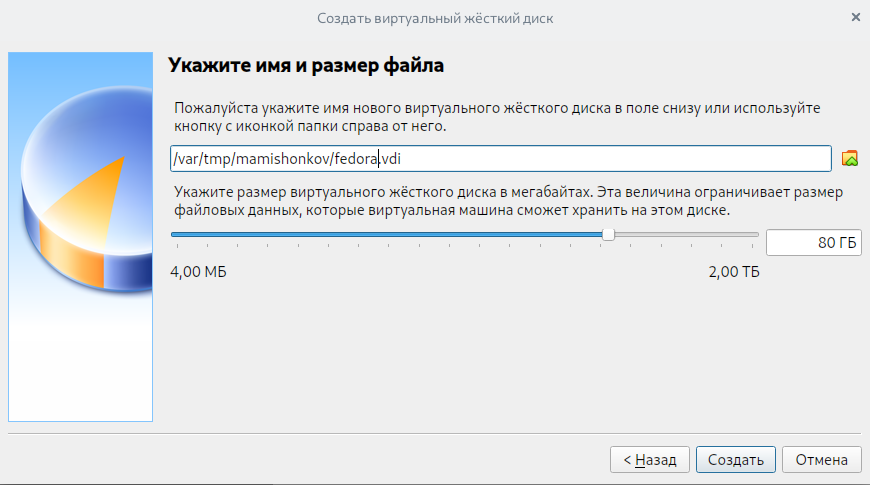


Окно определения типа подключения виртуального жёсткого диска



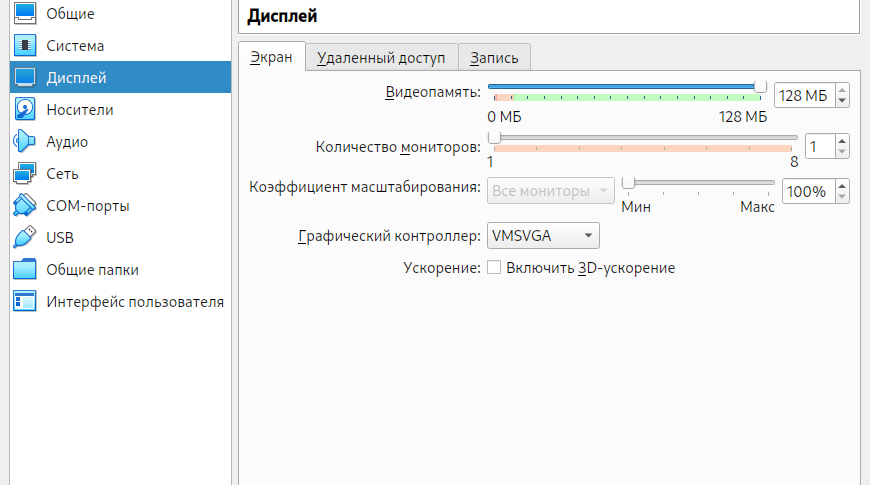
Окно определения формата виртуального жѐсткого диска

1. Задал расположение диска и его размер (80 ГБ). (рис. [??])



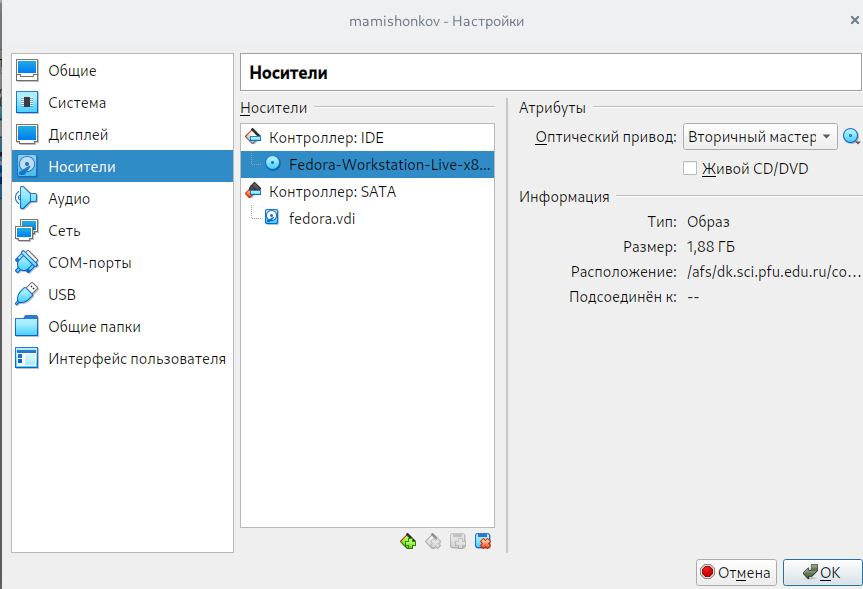
Размер виртуального динамического жёсткого диска и его расположение

1. В настройках виртуальной машины во вкладке «дисплей -> экран» увеличил доступный объём видеопамяти до 128 МБ. (рис. [??])



Настройка виртуальной машины

1. В настройках виртуальной машины добавил новый привод оптических дисков. (рис. [??])

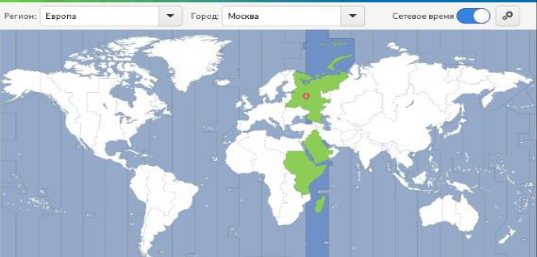


Окно «Носители» виртуальной машины: выбор оптического диска

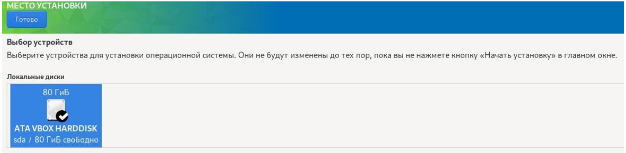
1. Скорректировал настройки системы (раскладку клавиатуры, часовой пояс, место установки). (рис. [??], [??], [??])



Окно выбора языка

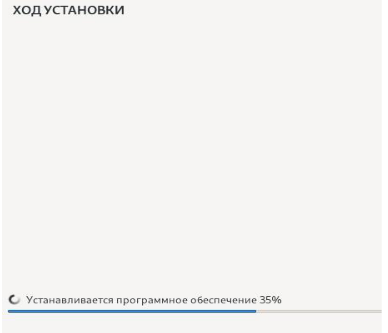


Окно выбора часового пояса



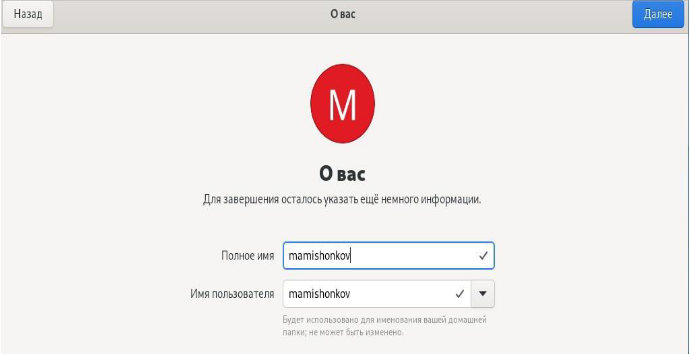
Окно выбора места установки

1. Запустил установку операционной системы. (рис. [??])

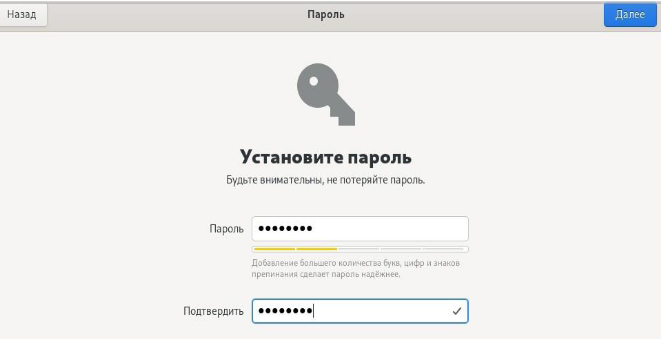


Установка операционной системы

1. Создал имя пользователя, используя свой логин в дисплейном классе, и установил пароль. (рис. [??], [??])

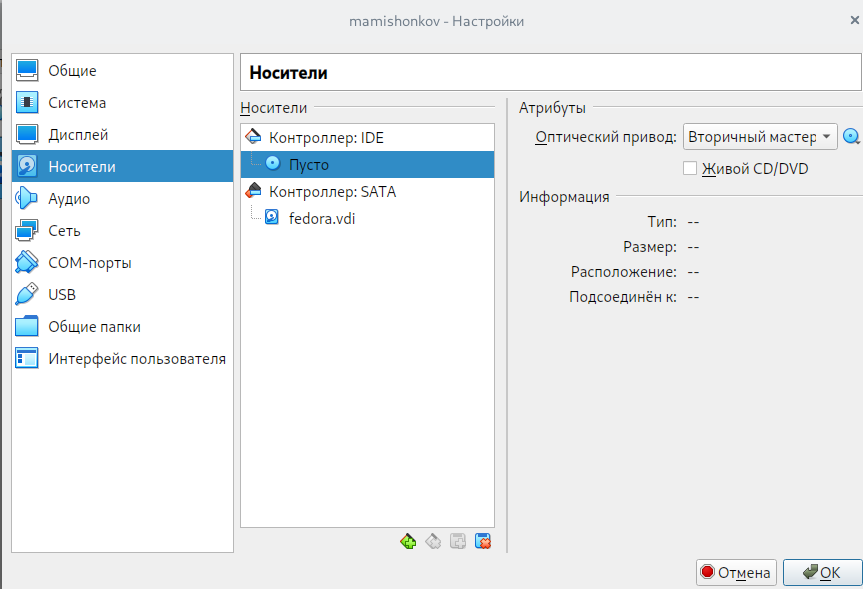


Окно создания имя пользователя



Окно установки пароля

1. Выключил систему и совершил изъятие образа диска из дисковода. (рис. [??])



Извлечение образа диска

1. Вошёл в ОС под заданной мной при установке учётной записью. Нажала комбинацию Win+Enter для запуска терминала.
2. Переключился на роль супер-пользователя и ввёл пароль. (рис. [??])

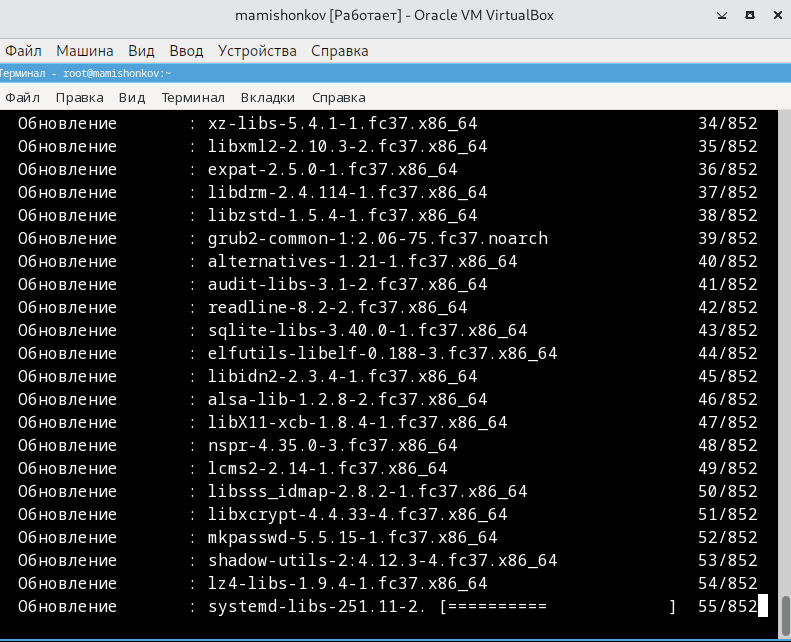
Команда sudo -i

Команда sudo -i

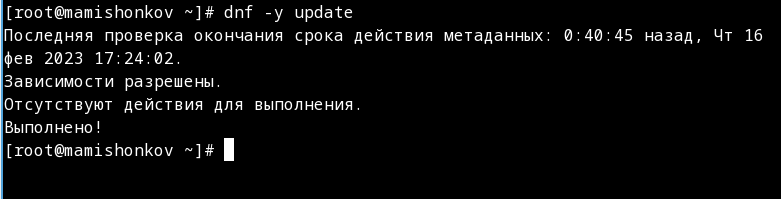
1. Обновил все пакеты. (рис. [??], [??], [??])

Команда dnf -y update

Команда dnf -y update

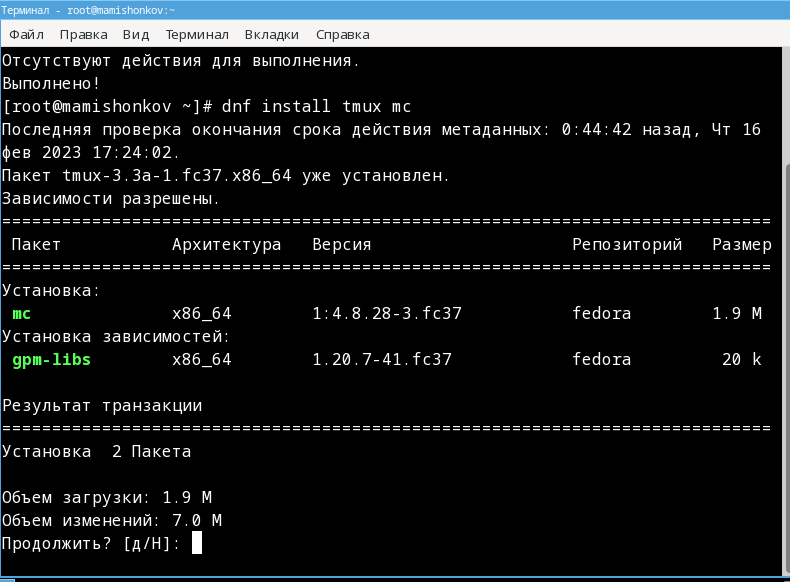


Обновление пакетов, запуск скриптлетов



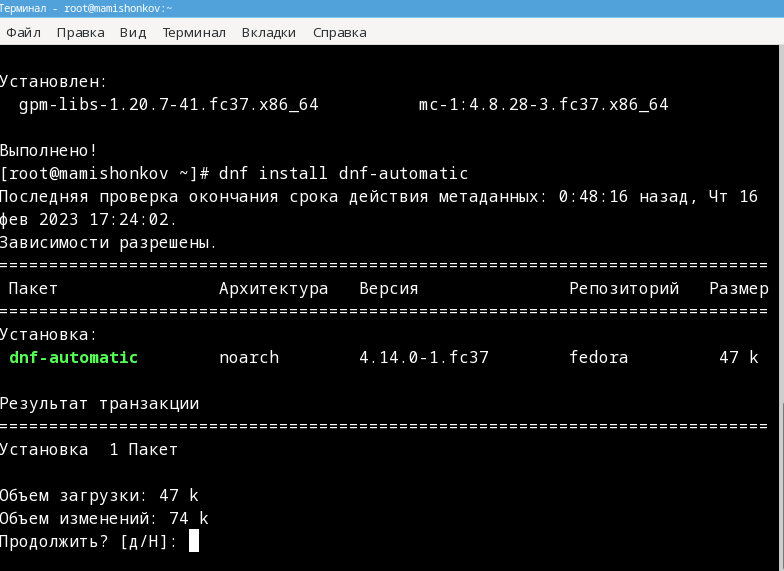
Пакеты обновлены

1. Я использовал программы для удобства работы в консоли. (рис. [??])

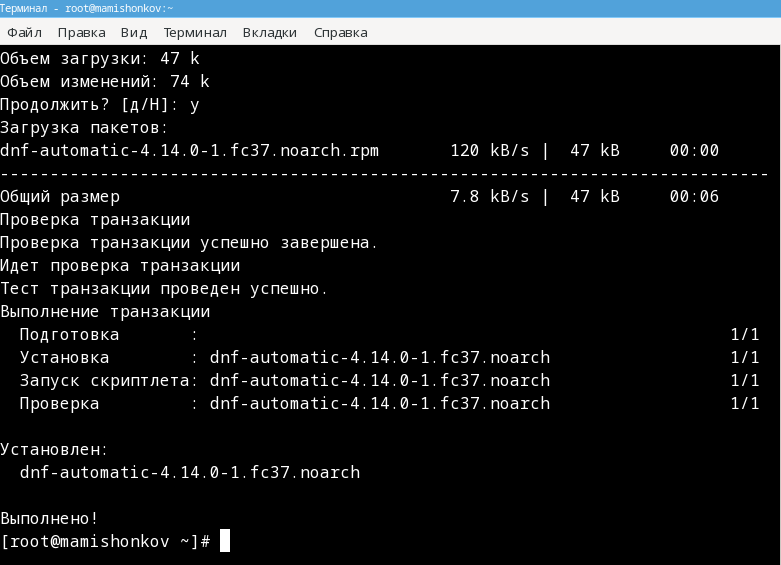


Команда dnf install tmux mc

1. Я использовал автоматическое обновление. (рис. [??], [??])

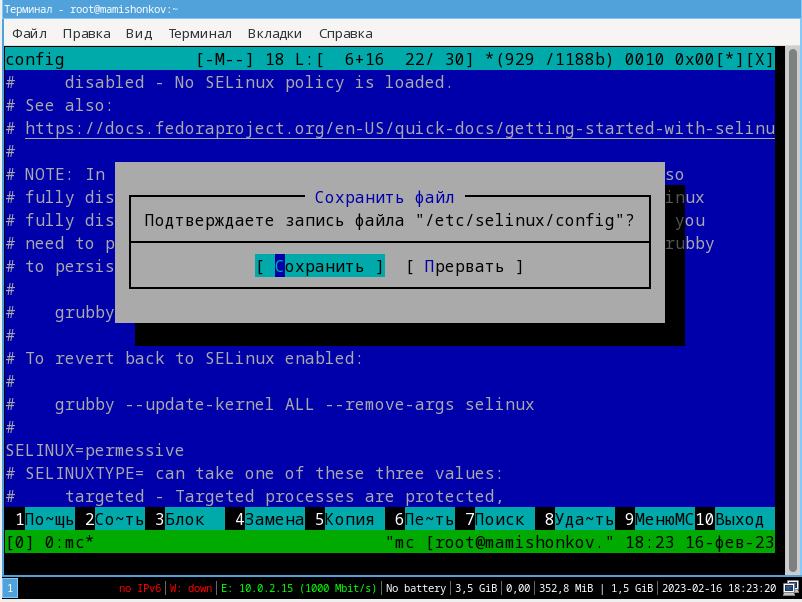


Команда dnf install dnf-automatic



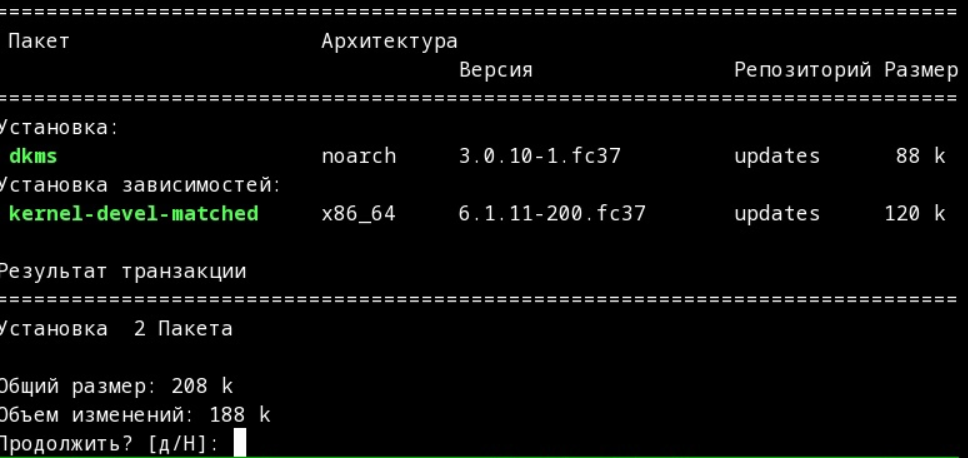
Результат

1. Открыл в mc и в файле /etc/selinux/config заменил значение SELINUX=enforcing на значение SELINUX=permissive. (рис. [??])

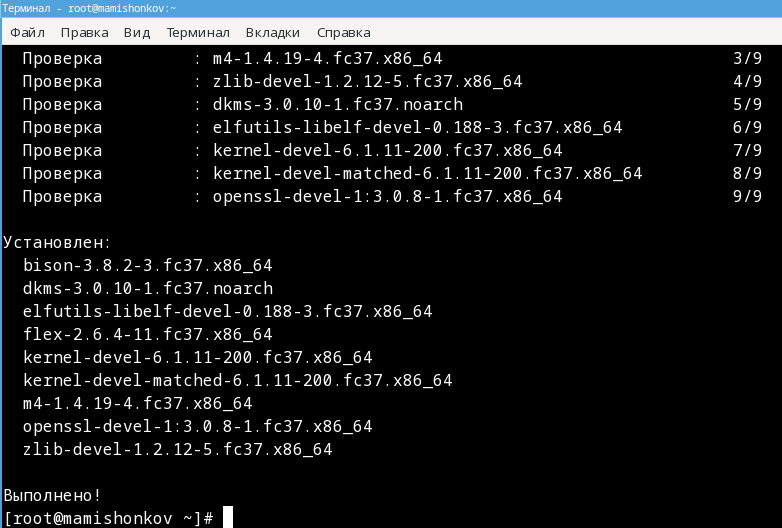


Окно mc

1. Установил пакет DKMS. (рис. [??], [??])

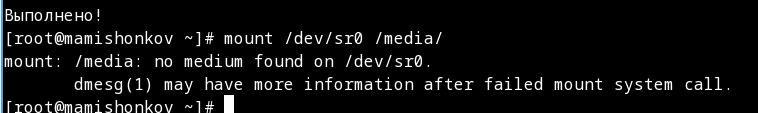


Команда dnf -y install dkms

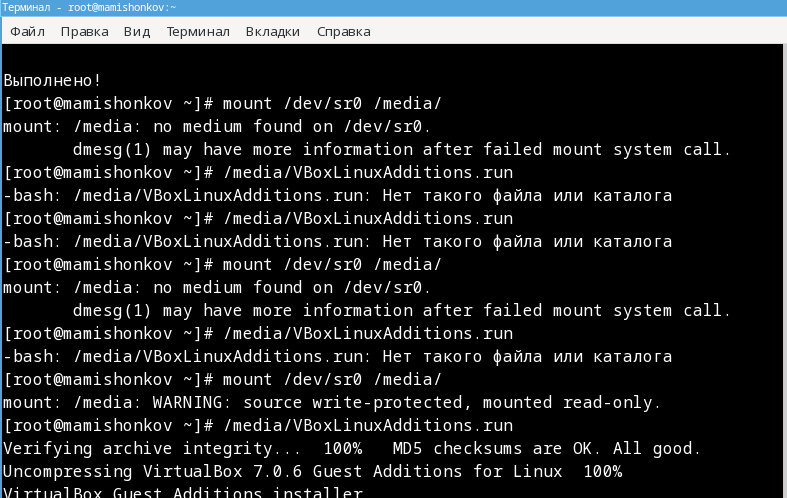


Результат

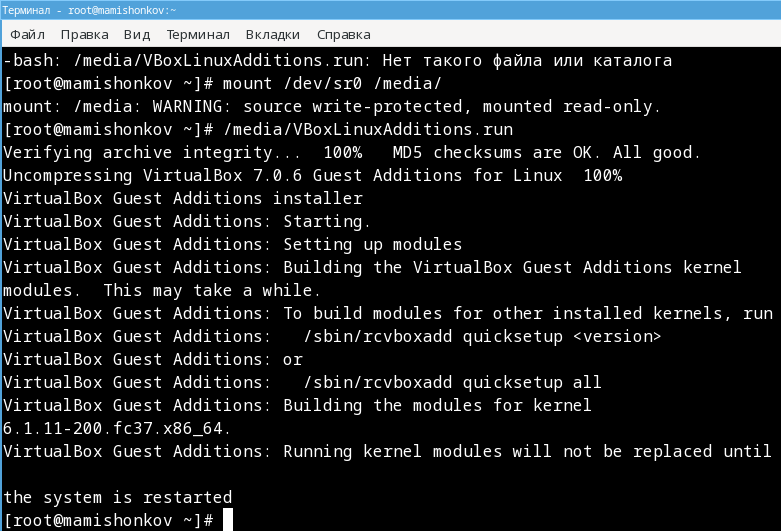
1. В меню виртуальной машины я подключил образ диска дополнений гостевой ОС. Подмонтировал диск. Установил драйвера: /media/VBoxLinuxAdditions.run. Перезагрузил виртуальную машину с помощью команды reboot.(рис. [??])



Команда mount /dev/sr0 /media

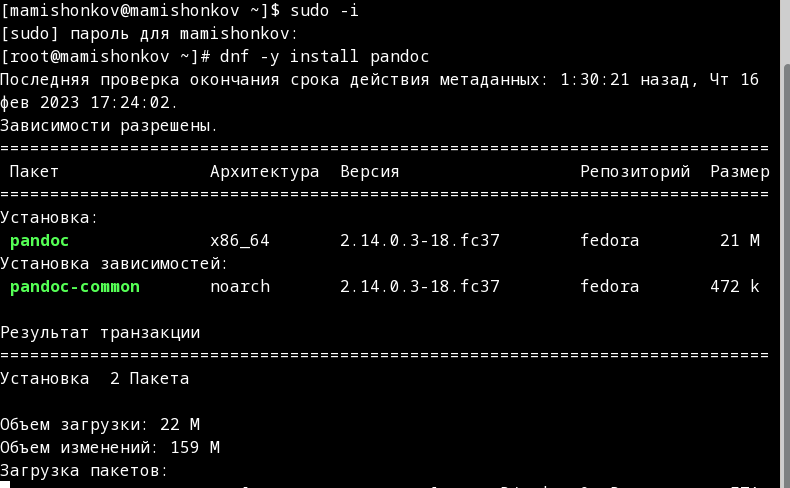


Установка драйверов

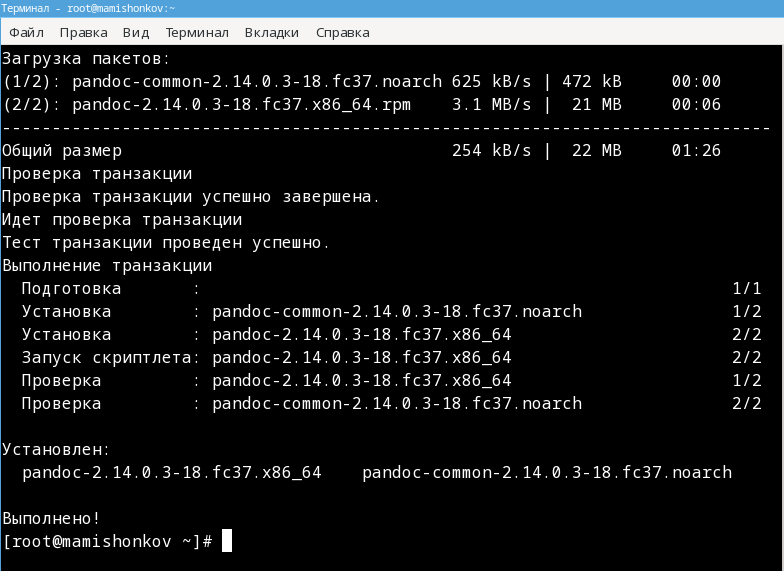


Установка драйверов

1. Нажал комбинацию Win+Enter для запуска терминала. Запустил терминальный мультиплексор tmux. Переключился на роль супер-пользователя: sudo -i. Установил pandoc. (рис. [??], [??])

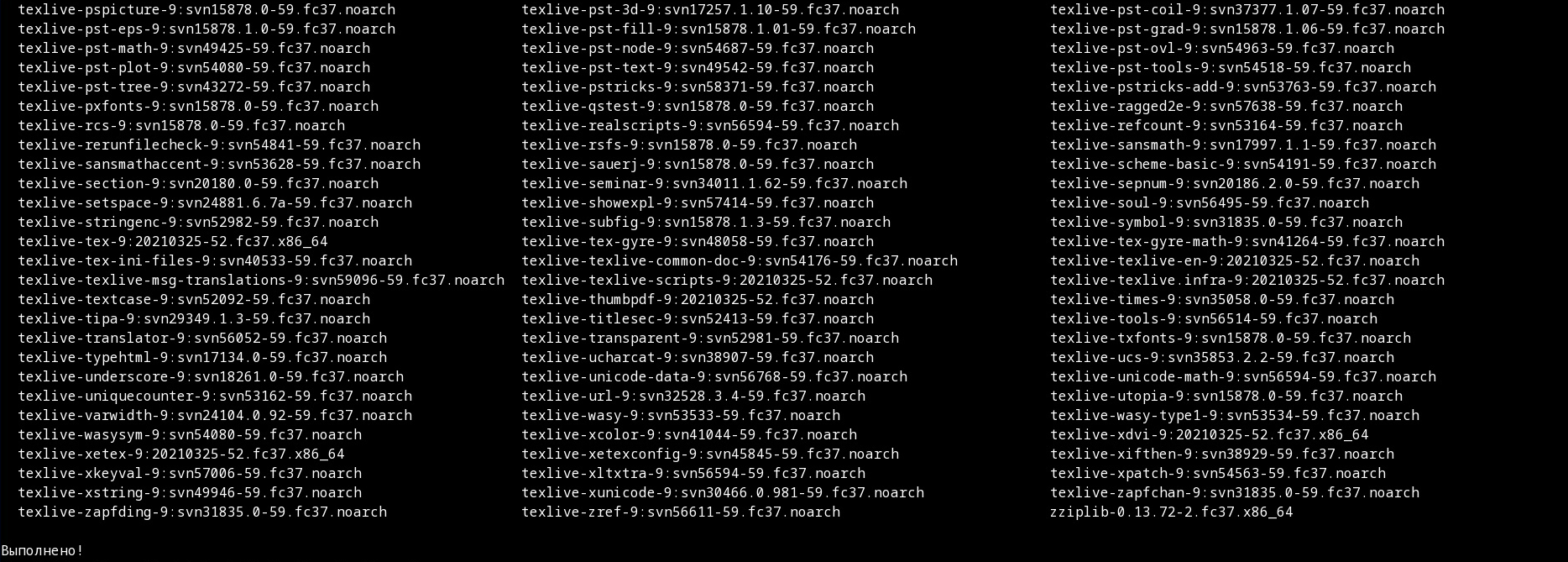


Команда dnf -y install pandoc

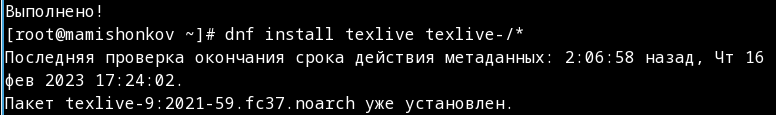


Результат

1. Установил дистрибутив TeXlive. (рис. [??])



Установка texlive texlive-\*



texlive texlive-\* установлен

**Домашнее задание**

1. Версия ядра Linux: 5.2.0-kali2-amd64.
2. Частота процессора: 1995.390 МГц.
3. Модель процессора: Intel(R)Core(TM)i3-5005U CPU @ 2.00GHz.
4. Объём доступной оперативной памяти: 2096628 Кб.
5. Тип обнаруженного гипервизора: Vmware
6. Тип файловой системы корневого раздела: EXT4.

**Ответы на контрольные вопросы**

1. Учётная запись пользователя – это необходимая для системы информация о пользователе, хранящаяся в специальных файлах. Информация используется Linux для аутентификации пользователя и назначения ему прав доступа. Аутентификация – системная процедура, позволяющая Linux определить, какой именно пользователь осуществляет вход. Вся информация о пользователе обычно хранится в файлах /etc/passwd и /etc/group. Учётная запись пользователя содержит: -Имя пользователя (user name) -Идентификационный номер пользователя (UID) -Идентификационный номер группы (GID) -Пароль (password) -Полное имя (full name) -Домашний каталог (home directory) -Начальную оболочку (login shell)
2. Команды терминала: -Для получения справки по команде: man [команда]. Например, команда «man ls» выведет справку о команде «ls». -Для перемещения по файловой системе: cd [путь]. Например, команда «cd newdir» осуществляет переход в каталог newdir. -Для просмотра содержимого каталога: ls [опции] [путь]. Например, команда «ls –a ~/newdir» отобразит имена скрытых файлов в каталоге newdir. -Для определения объёма каталога: du [опция] [путь]. Например, команда «du –k ~/newdir» выведет размер каталога newdir в килобайтах. -Для создания / удаления каталогов / файлов: mkdir [опции] [путь] / rmdir [опции] [путь] / rm [опции] [путь]. Например, команда «mkdir –p ~/newdir1/newdir2» создаст иерархическую цепочку подкаталогов, создав каталоги newdir1 и newdir2; команда «rmdir -v ~/newdir» удалит каталог newdir; команда «rm –r ~/newdir» так же удалит каталог newdir. -Для задания определённых прав на файл / каталог: сhmod [опции] [путь]. Например, команда «сhmod g+r ~/text.txt» даст группе право на чтение файла text.txt. -Для просмотра истории команд: history [опции]. Например, команда «history 5» покажет список последних 5 команд.
3. Файловая система имеет два значения: с одной стороны – это архитектура хранения битов на жестком диске, с другой – это организация каталогов в соответствии с идеологией Unix. Файловая система (англ. «file system») – это архитектура хранения данных в системе, хранение данных в оперативной памяти и доступа к конфигурации ядра. Файловая система устанавливает физическую и логическую структуру файлов, правила их создания и управления ими. В физическом смысле файловая система Linux представляет собой пространство раздела диска, разбитое на блоки фиксированного размера. Их размер кратен размеру сектора: 1024, 2048, 4096 или 8120 байт. Существует несколько типов файловых систем: **XFS** – начало разработки 1993 год, фирма Silicon Graphics, в мае 2000 года предстала в GNU GPL, для пользователей большинства Linux систем стала доступна в 2001-2002 гг. Отличительная черта системы – прекрасная поддержка больших файлов и файловых томов, 8 эксбибайт (8*260 байт) для 64-х битных систем.* ***ReiserFS (Reiser3)*** *– одна из первых журналируемых файловых систем под Linux, разработана Namesys, доступна с 2001 г. Максимальный объём тома для этой системы равен 16 тебибайт (16*240 байт). **Tux2** – известная, но так и не анонсированная публично файловая система. Создатель Дэниэл Филипс (Daniel Phillips). Cистема базируется на алгоритме «Фазового Дерева», который как и журналирование защищает файловую систему от сбоев. Организована как надстройка на ext2.
4. Команда «findmnt» или «findmnt –all» будет отображать все подмонтированные файловые системы или искать файловую систему.
5. Основные сигналы (каждый сигнал имеет свой номер), которые используются для завершения процесса: -SIGINT – самый безобидный сигнал завершения, означает Interrupt. Он отправляется процессу, запущенному из терминала с помощью сочетания клавиш Ctrl+C. Процесс правильно завершает все свои действия и возвращает управление; -SIGQUIT – это еще один сигнал, который отправляется с помощью сочетания клавиш, программе, запущенной в терминале. Он сообщает ей что нужно завершиться и программа может выполнить корректное завершение или проигнорировать сигнал. В отличие отпредыдущего, она генерирует дамп памяти. Сочетание клавиш Ctrl+/; -SIGHUP – сообщает процессу, что соединение с управляющим терминалом разорвано, отправляется, в основном, системой при разрыве соединения с интернетом; -SIGTERM – немедленно завершает процесс, но обрабатывается программой, поэтому позволяет ей завершить дочерние процессы и освободить все ресурсы; SIGKILL – тоже немедленно завершает процесс, но, в отличие от предыдущего варианта, он не передается самому процессу, а обрабатывается ядром. Поэтому ресурсы и дочерние процессы остаются запущенными. Также для передачи сигналов процессам в Linux используется утилита kill, её синтаксис: kill [-сигнал] [pid\_процесса] (PID – уникальный идентификатор процесса). Сигнал представляет собой один из выше перечисленных сигналов для завершения процесса. Перед тем, как выполнить остановку процесса, нужно определить его PID. Для этого используют команды ps и grep. Команда ps предназначена для вывода списка активных процессов в системе и информации о них. Команда grep запускается одновременно с ps (в канале) и будет выполнять поиск по результатам команды ps. Утилита pkill – это оболочка для kill, она ведет себя точно так же, и имеет тот же синтаксис, только в качестве идентификатора процесса ей нужно передать его имя. killall работает аналогично двум предыдущим утилитам. Она тоже принимает имя процесса в качестве параметра и ищет его PID в директории /proc. Но эта утилита обнаружит все процессы с таким именем и завершит их.

# 4 Выводы

В ходе выполнения данной лабораторной работы я установил операционную систему Linux (дистрибутив Fedora 36) на вирутальную машину VIrtualBox и настроил минимально необходимые параметры для дальнейшей работы с ситемой.