

Отчёт по лабораторной работе №1

Установка ОС Linux

Седохин Даниил Алексеевич

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	44

Список иллюстраций

4.1	Создание новой виртуальной машиныю. Задаём имя и обаз ISO из файлов лабораторной работы	8
4.2	Выделение памяти и рабочих процессоров	9
4.3	Создание виртуального жёсткого диска на 80ГБ	9
4.4	Характеристики новой виртуальной машины	10
4.5	Запуск виртуальной машины	11
4.6	Установка ОС	12
4.7	Выбор места установки ОС	13
4.8	Создание аккаунта администратора	14
4.9	Создание пользователя	15
4.10	Установка Fedora	16
4.11	Отключение оптического диска	16
4.12	Войдем в ОС	17
4.13	Переключение на роль супер-пользователя	18
4.14	Обновление пакетов	19
4.15	Установка программ	21
4.16	Установка ПО	22
4.17	Запуск таймера	23
4.18	Отключение систему безопасности SELinux	24
4.19	Запуск терминального мультплексора и переключение на роль супер-пользователя	25
4.20	Установка средств разработки	26
4.21	Установка пакета DKMS	28
4.22	Подключение образа диска дополнений гостевой ОС	29
4.23	Монтаж диска	29
4.24	Установка драйверов	30
4.25	Создание конфигурационного файла	31
4.26	Редактирование конфигурационного файла	31
4.27	Супер-пользователь	32
4.28	Редактирование конфигурационного файла	32
4.29	Установка имени хоста	33
4.30	Подключение разделяемой папки	33
4.31	Установка pandoc	34
4.32	Установка pandoc-crossref	35
4.33	Переход в папку загрузки	36
4.34	Распаковка архивов	36
4.35	Установка TeXLive	37

4.36	Просмотр вывода команды	39
4.37	Использование команды grep	41

1 Цель работы

Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

2 Задание

Суть задания заключается в установке ОС Linux с дистрибутивом Fedora и настройка ОС.

3 Теоретическое введение

Выполнение работы возможно как в дисплейном классе факультета физико-математических и естественных наук РУДН, так и дома. Описание выполнения работы приведено для дисплейного класса со следующими характеристиками техники:

Intel Core i3-550 3.2 GHz, 4 GB оперативной памяти, 80 GB свободного места на жёстком
ОС Linux Gentoo (<http://www.gentoo.ru/>);
VirtualBox версии 7.0 или новее.

Для установки в виртуальную машину используется дистрибутив Linux Fedora (<https://getfedora.org>), вариант с менеджером окон sway (<https://fedoraproject.org/spins/sway/>).
При выполнении лабораторной работы на своей технике вам необходимо скачать необходимый образ операционной системы (<https://fedoraproject.org/spins/sway/download>).
В дисплейных классах можно воспользоваться образом в каталоге [/afs/dk.sci.pfu.edu.ru/commo](https://afs.dk.sci.pfu.edu.ru/commo)

4 Выполнение лабораторной работы

- 1) Первой задачей будет настройка VirtualBox для дальнейшей установки новой ОС Linux с дистрибутивом Fedora. (рис. 4.1).

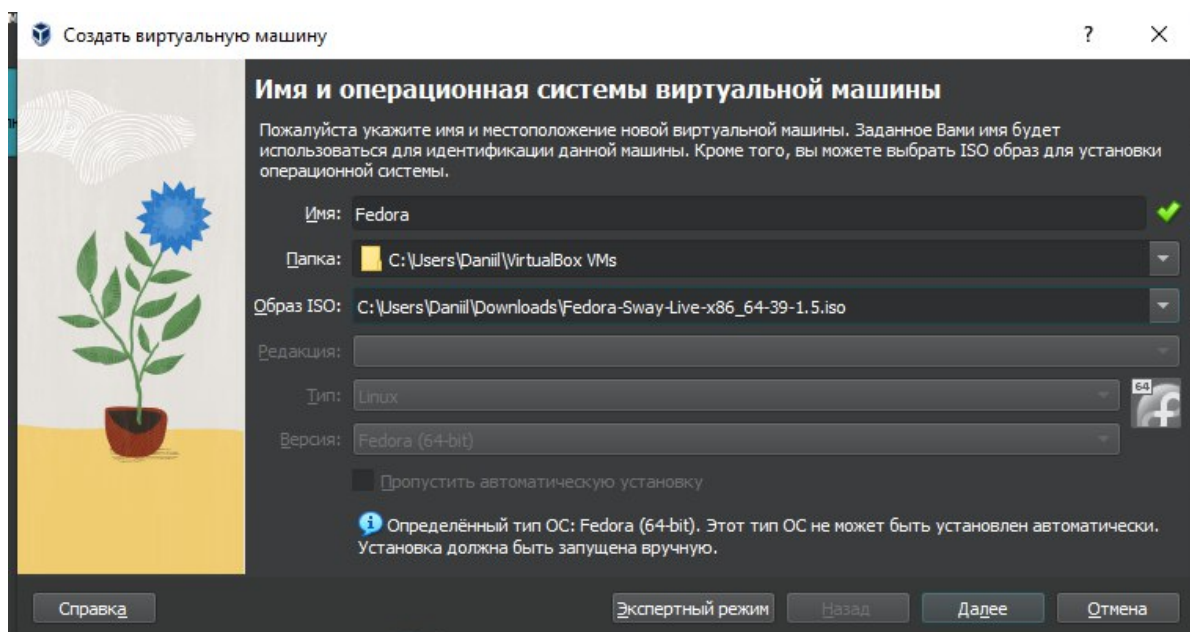


Рис. 4.1: Создание новой виртуальной машины. Задаём имя и образ ISO из файлов лабораторной работы

- 2) Выделим необходимое количество памяти и процессоров. (рис. 4.2).

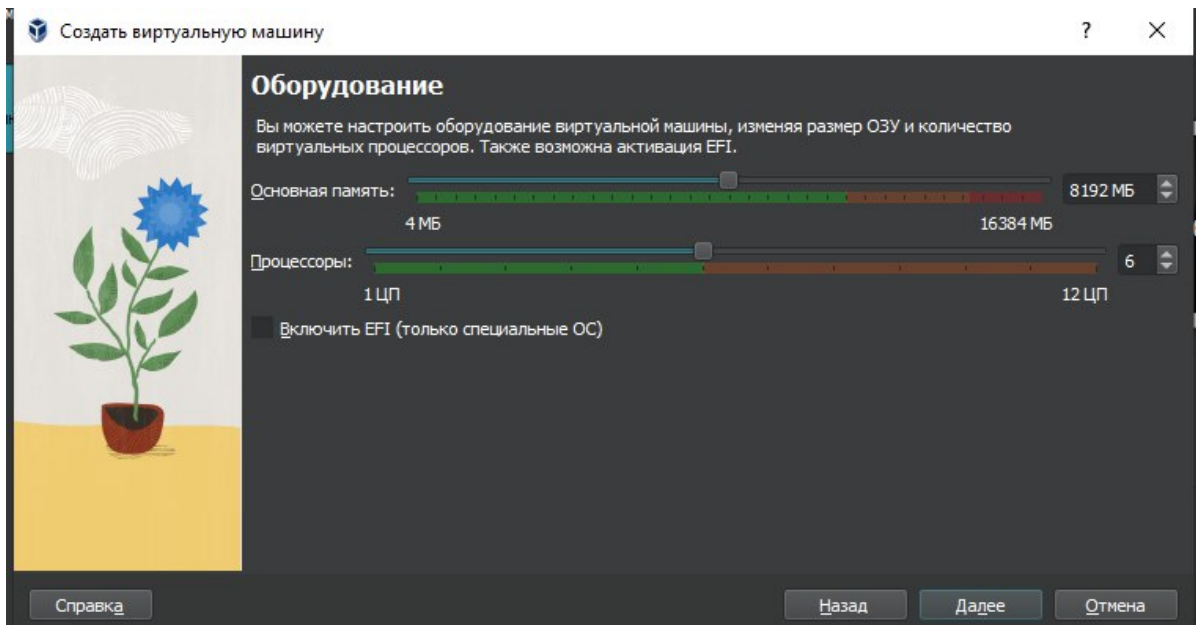


Рис. 4.2: Выделение памяти и рабочих процессоров

3) Создадим виртуальный жёсткий диск на 80ГБ. (рис. 4.3).

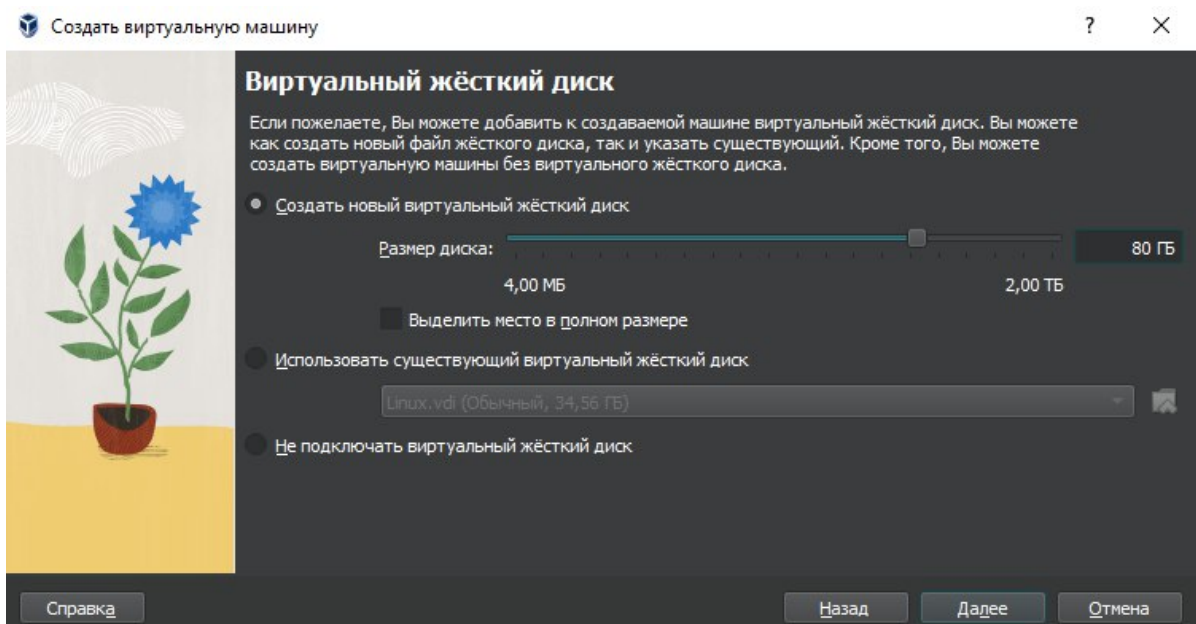


Рис. 4.3: Создание виртуального жёсткого диска на 80ГБ

4) Проверим характеристики новой виртуальной машины. (рис. 4.4)

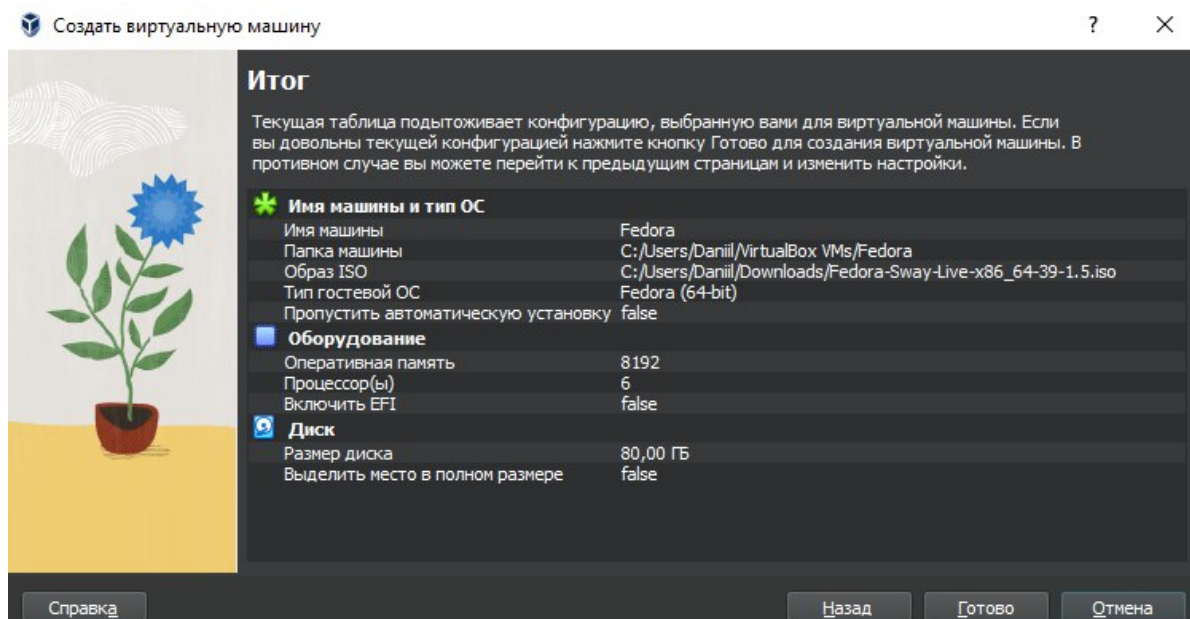


Рис. 4.4: Характеристики новой виртуальной машины

5) Запустим новую виртуальную машину. (рис. 4.5).

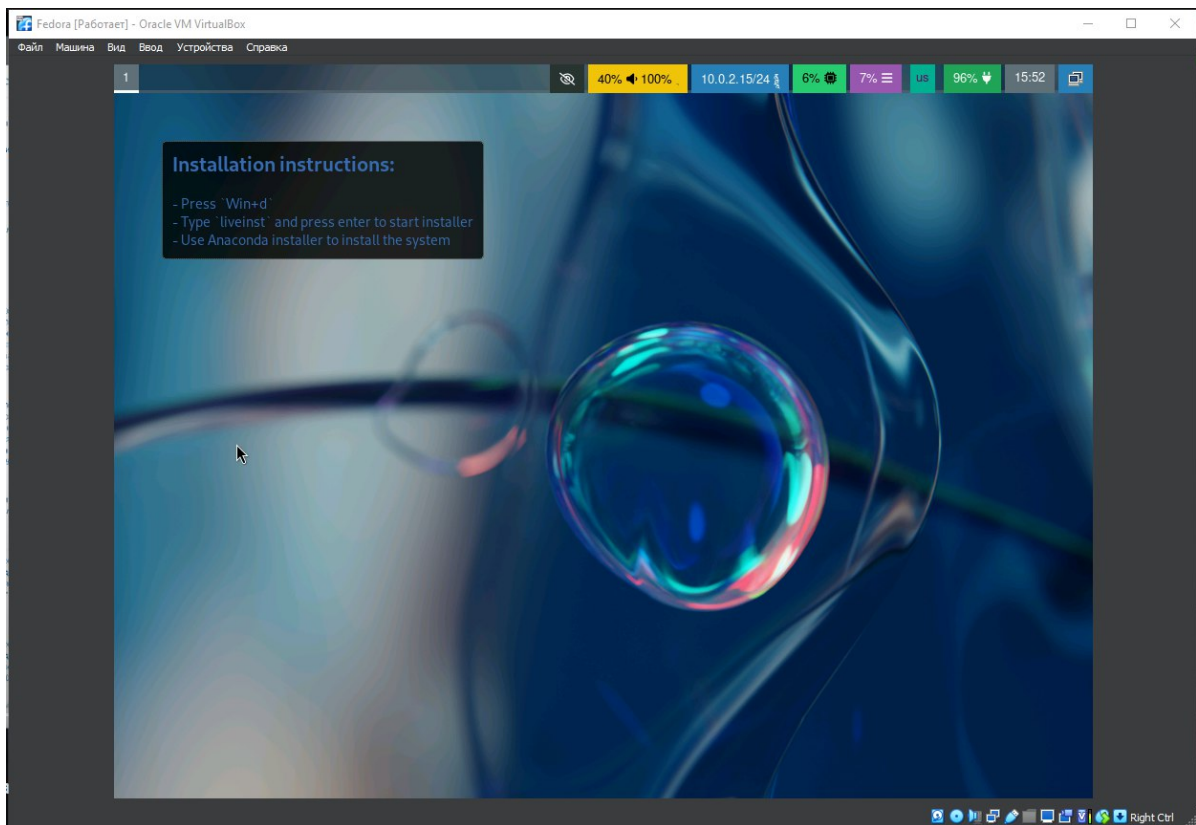


Рис. 4.5: Запуск виртуальной машины

- 6) Нажмем комбинацию Win+Enter для запуска терминала. В терминале запустим liveinst. (рис. 4.6).

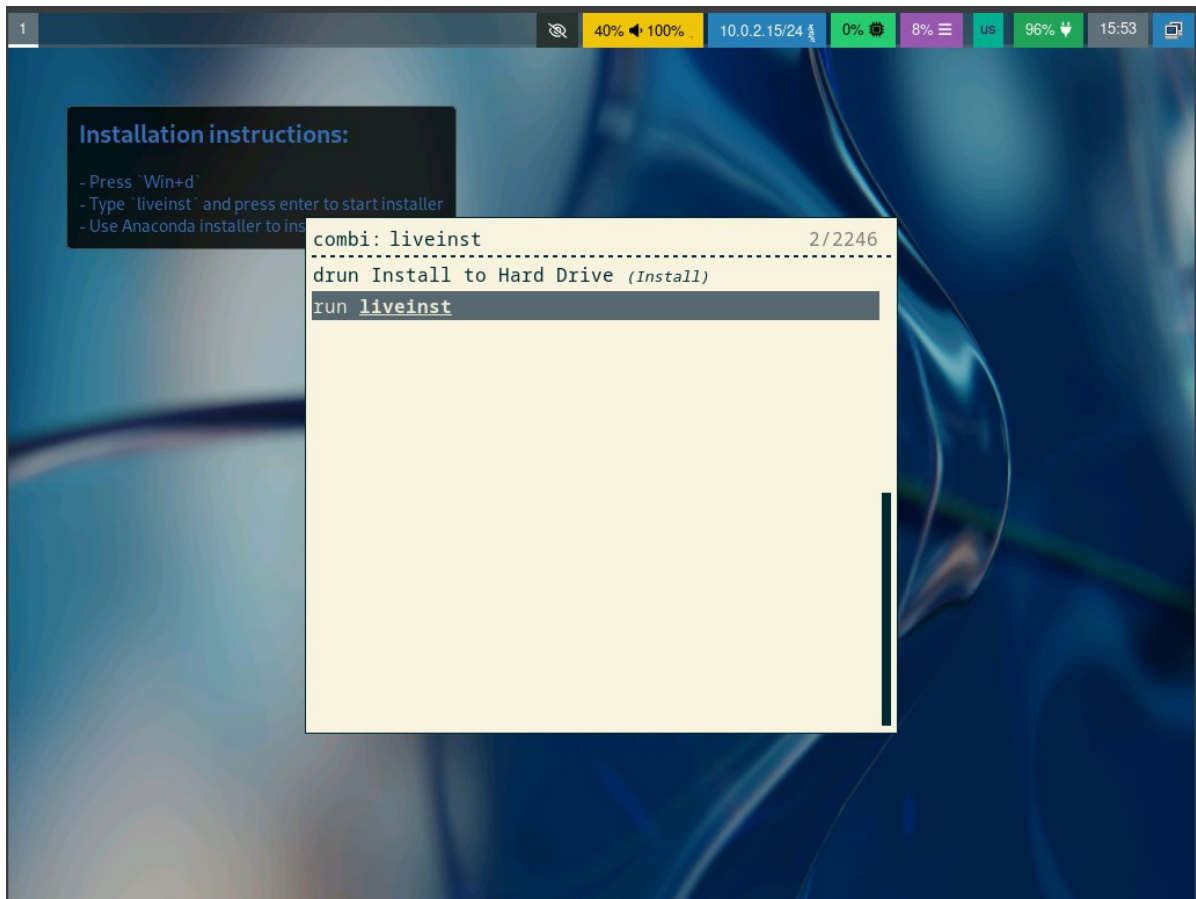


Рис. 4.6: Установка ОС

7) Выберем место установки ОС. (рис. 4.7).

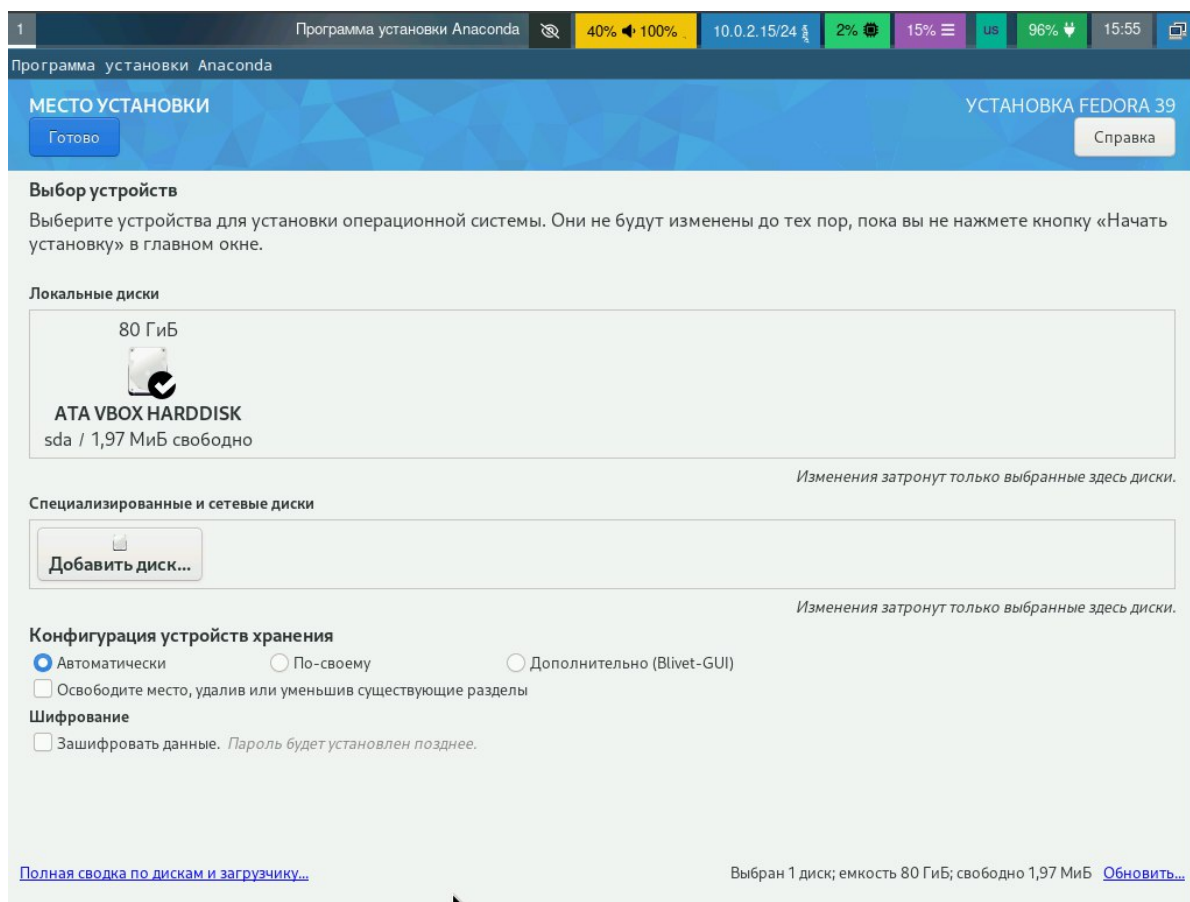


Рис. 4.7: Выбор места установки ОС

8) Создадим аккаунт администратора. (рис. 4.8).

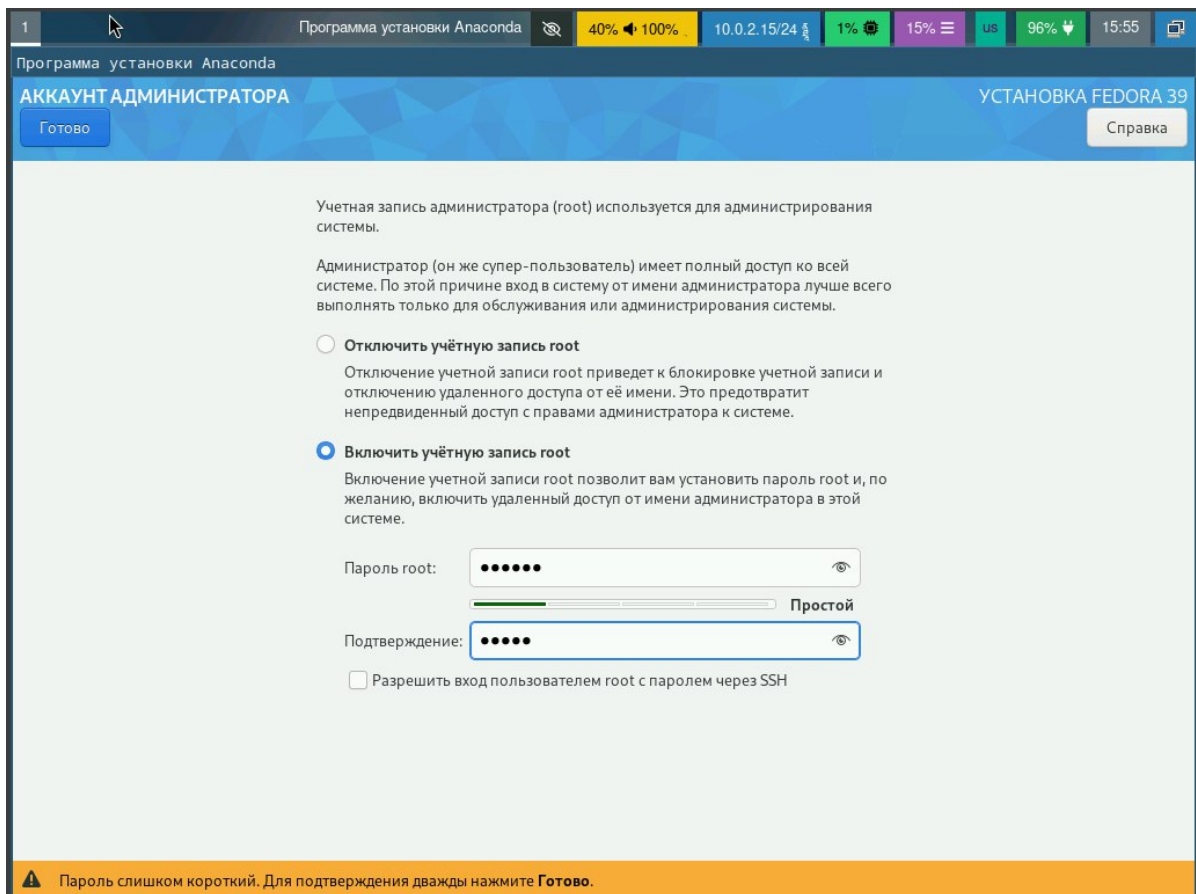


Рис. 4.8: Создание аккаунта администратора

9) Создадим пользователя. (рис. 4.9).

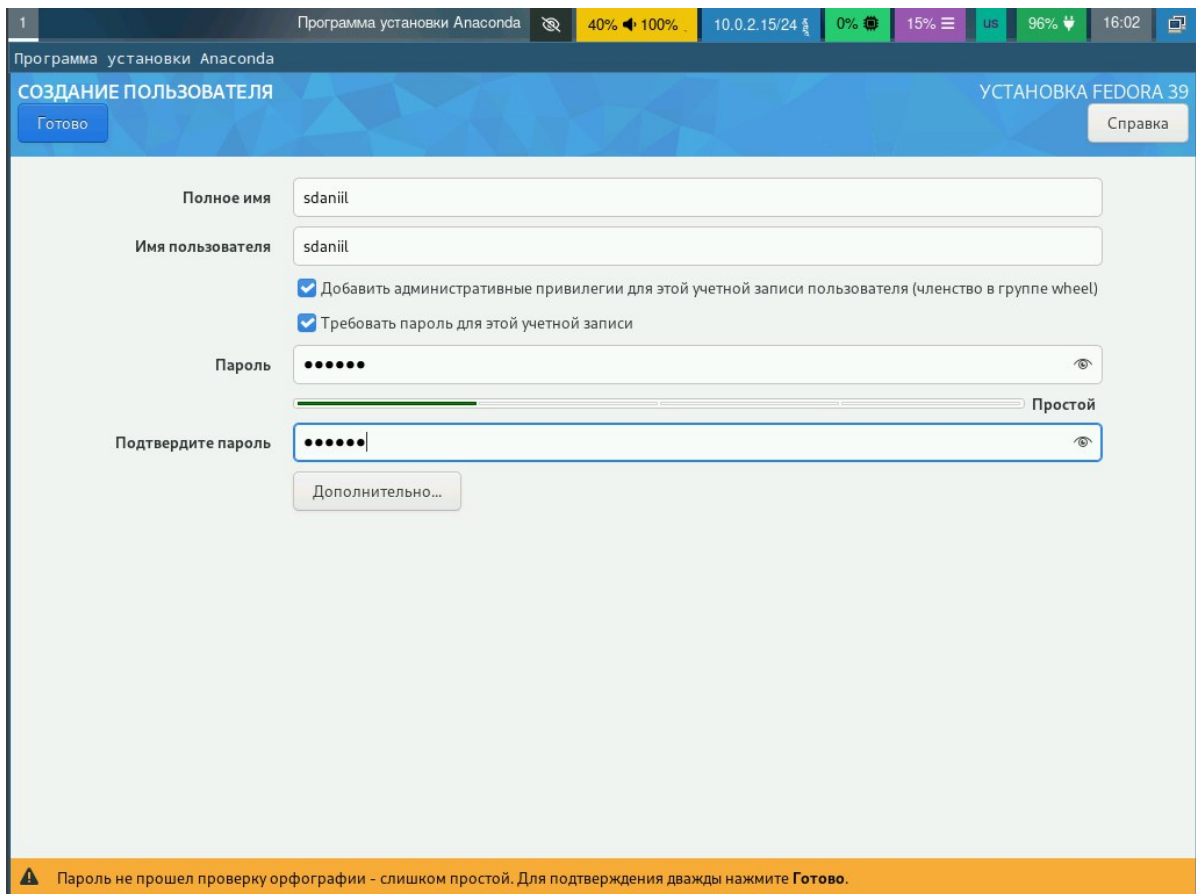


Рис. 4.9: Создание пользователя

10) После чего начнем установку. (рис. 4.10).

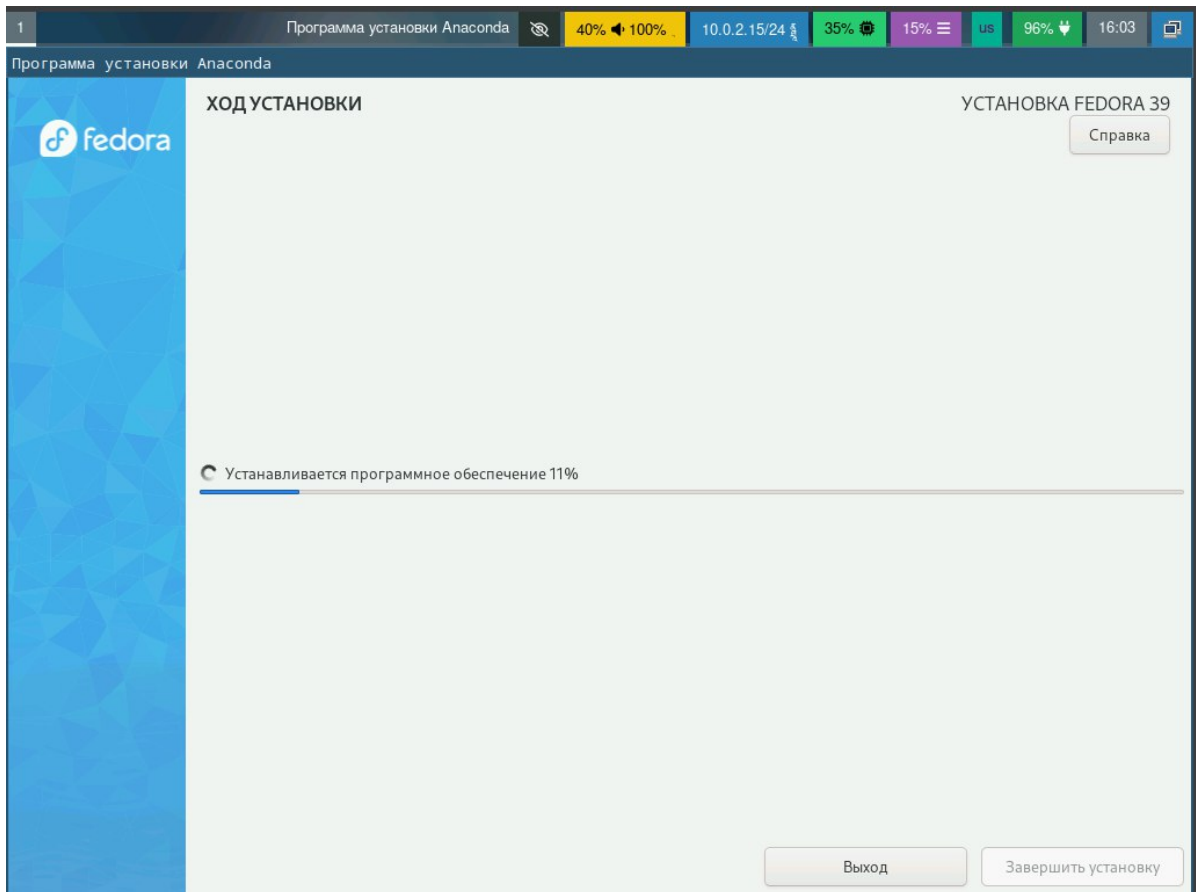


Рис. 4.10: Установка Fedora

11) После установки отключим оптический диск и перезапустим виртуальную машину. (рис. 4.11).

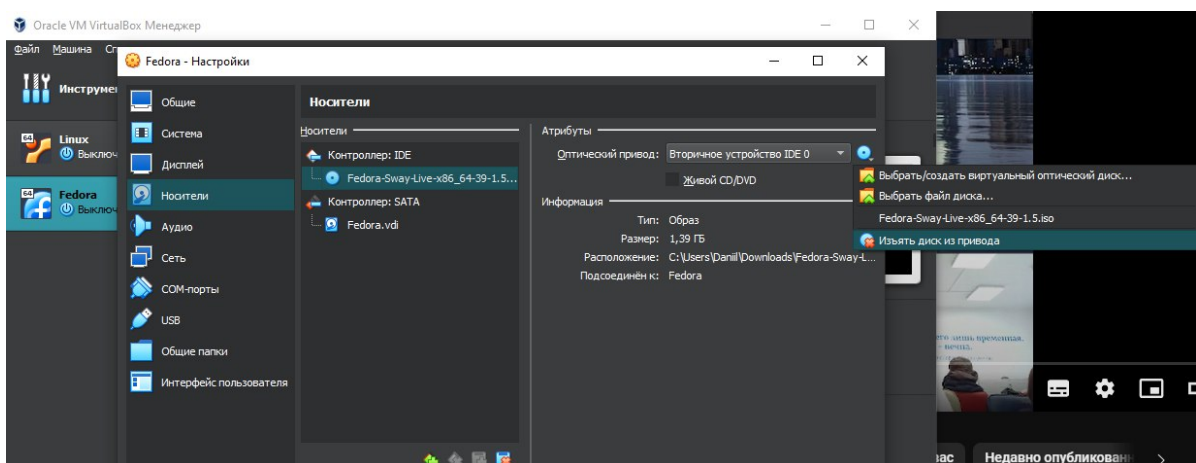


Рис. 4.11: Отключение оптического диска

12) Войдем в ОС под заданной при установке учётной записью. (рис. 4.12).

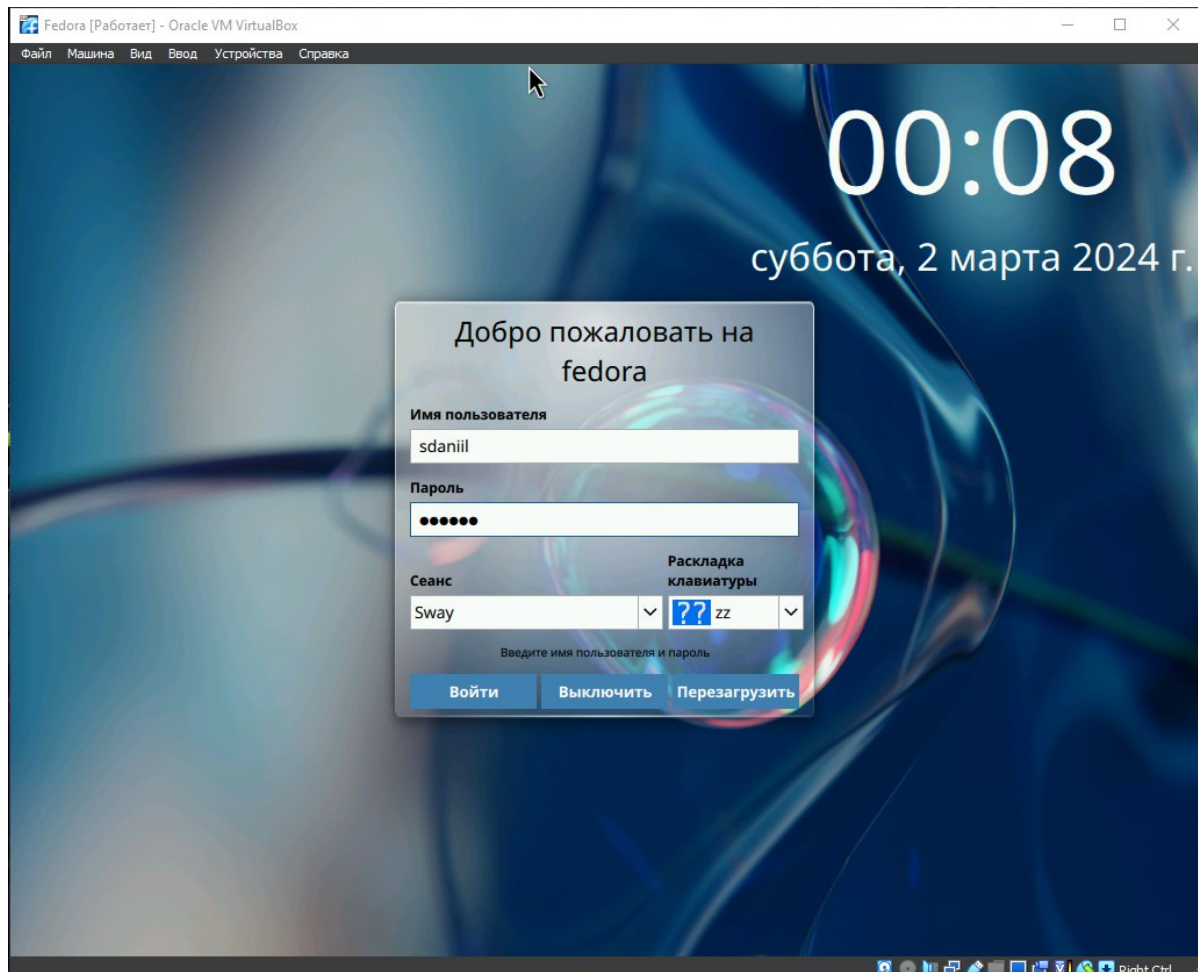


Рис. 4.12: Войдем в ОС

13) Нажмем комбинацию Win+Enter для запуска терминала. Переключимся на роль супер-пользователя с помощью команды `sudo -i` (рис. 4.13).

```
foot
[sdaniil@fedora ~]$ sudo -i

Мы полагаем, что ваш системный администратор изложил вам основы
безопасности. Как правило, всё сводится к трём следующим правилам:

    №1) Уважайте частную жизнь других.
    №2) Думайте, прежде чем что-то вводить.
    №3) С большой властью приходит большая ответственность.

По соображениям безопасности пароль, который вы введёте, не будет виден.

[sudo] пароль для sdaniil:
Попробуйте ещё раз.
[sudo] пароль для sdaniil:
[root@fedora ~]#
```

Рис. 4.13: Переключение на роль супер-пользователя

14) Обновим все пакеты с помощью команды: `dnf -y update` (рис. fig. 4.14).

```
foot
xorg-x11-server-common-1.20.14-30.fc39.x86_64
xorg-x11-xinit-1.4.2-1.fc39.x86_64
yt-dlp-2023.12.30-1.fc39.noarch
yt-dlp-bash-completion-2023.12.30-1.fc39.noarch
yum-4.19.0-1.fc39.noarch
zchunk-libs-1.4.0-1.fc39.x86_64
zenity-4.0.1-1.fc39.x86_64
zimg-3.0.5-1.fc39.x86_64
Установлен:
amd-ucode-firmware-20240220-1.fc39.noarch
cirrus-audio-firmware-20240220-1.fc39.noarch
gstreamer1-plugins-bad-free-libs-1.22.9-1.fc39.x86_64
intel-audio-firmware-20240220-1.fc39.noarch
kernel-6.7.6-200.fc39.x86_64
kernel-core-6.7.6-200.fc39.x86_64
kernel-modules-6.7.6-200.fc39.x86_64
kernel-modules-core-6.7.6-200.fc39.x86_64
kernel-modules-extra-6.7.6-200.fc39.x86_64
libdisplay-info-0.1.1-2.fc39.x86_64
libdovi-3.2.0-2.fc39.x86_64
libl3-1.0.4-2.fc39.x86_64
liblifter-0.4.1-1.fc39.x86_64
libvpl-1:2.10.2-1.fc39.x86_64
llvm-libs-17.0.6-3.fc39.x86_64
nxpwireless-firmware-20240220-1.fc39.noarch
python3-packaging-23.1-4.fc39.noarch
qt5-qttranslations-5.15.12-1.fc39.noarch
tiwilink-firmware-20240220-1.fc39.noarch
tpm2-tss-fapi-4.0.1-6.fc39.x86_64
wlroots0.16-0.16.2-1.fc39.x86_64
xcb-util-errors-1.0.1-1.fc39.x86_64

Выполнено!
[root@fedora ~]# util-linux                                x86_64 2.39.3-6.fc39
updates 1.2 M util-linux                                x86_64 2.39.3-6.fc39
updates 1.2 M util-linux                                x86_64 2.39.3-6.fc39
updates 1.2 M util-linux                                x86_64 2.39.3-6.fc39
updates 1.2 M util-linux                                x86_64 2.39.3-6.fc39 u
pdates 1.2 M util-linux                                x86_64 2.39.3-6.fc39 up
dates 1.2 M util-linux                                x86_64 2.39.3-6.fc39 upd
ates 1.2 M util-linux                                x86_64 2.39.3-6.fc39 upda
tes 1.
-bash: util-linux: команда не найдена
[root@fedora ~]#
```

Рис. 4.14: Обновление пакетов

15) Установим программы для удобства работы в консоли: `dnf -y install tmux` (рис. 4.15).

foot

```
perl-Getopt-Std-1.13-502.fc39.noarch
perl-HTTP-Tiny-0.088-3.fc39.noarch
perl-IO-1.52-502.fc39.x86_64
perl-IO-Socket-IP-0.42-1.fc39.noarch
perl-IO-Socket-SSL-2.083-3.fc39.noarch
perl-IPC-Open3-1.22-502.fc39.noarch
perl-MIME-Base64-3.16-500.fc39.x86_64
perl-Mozilla-CA-20230801-1.fc39.noarch
perl-NDBM_File-1.16-502.fc39.x86_64
perl-Net-SSLeay-1.92-10.fc39.x86_64
perl-POSIX-2.13-502.fc39.x86_64
perl-PathTools-3.89-500.fc39.x86_64
perl-Pod-Escapes-1:1.07-500.fc39.noarch
perl-Pod-Perldoc-3.28.01-501.fc39.noarch
perl-Pod-Simple-1:3.45-4.fc39.noarch
perl-Pod-Usage-4:2.03-500.fc39.noarch
perl-Scalar-List-Utils-5:1.63-500.fc39.x86_64
perl-SelectSaver-1.02-502.fc39.noarch
perl-Socket-4:2.037-3.fc39.x86_64
perl-Storable-1:3.32-500.fc39.x86_64
perl-Symbol-1.09-502.fc39.noarch
perl-Term-ANSIColor-5.01-501.fc39.noarch
perl-Term-Cap-1.18-500.fc39.noarch
perl-Text-ParseWords-3.31-500.fc39.noarch
perl-Text-Tabs+Wrap-2023.0511-3.fc39.noarch
perl-Time-Local-2:1.350-3.fc39.noarch
perl-URI-5.21-1.fc39.noarch
perl-base-2.27-502.fc39.noarch
perl-constant-1.33-501.fc39.noarch
perl-if-0.61.000-502.fc39.noarch
perl-interpreter-4:5.38.2-502.fc39.x86_64
perl-libnet-3.15-501.fc39.noarch
perl-libs-4:5.38.2-502.fc39.x86_64
perl-locale-1.10-502.fc39.noarch
perl-mro-1.28-502.fc39.x86_64
perl-overload-1.37-502.fc39.noarch
perl-overloading-0.02-502.fc39.noarch
perl-parent-1:0.241-500.fc39.noarch
perl-podlators-1:5.01-500.fc39.noarch
perl-vars-1.05-502.fc39.noarch
```

Выполнено!

[root@fedora ~]# util-linux

x86_64 2.39.3-6.fc39

updates 1.2 M

Рис. 4.15: Установка программ

16) Установка программного обеспечения: `dnf install dnf-automatic` (рис. 4.16).

```
dnf install: error: unrecognized arguments: -automatic
[root@fedora ~]# dnf -y install dnf-automatic
Последняя проверка окончания срока действия метаданных: 0:54:03 назад, С
6 02 мар 2024 01:03:59.
Зависимости разрешены.
=====
Пакет                Архитектура Версия                Репозиторий  Размер
=====
Установка:
dnf-automatic        noarch      4.19.0-1.fc39                updates      46 k
=====
Результат транзакции
=====
Установка 1 Пакет

Объем загрузки: 46 k
Объем изменений: 76 k
Загрузка пакетов:
dnf-automatic-4.19.0-1.fc39.noarch.rpm 573 kB/s | 46 kB    00:00
-----
Общий размер                87 kB/s | 46 kB    00:00
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
Тест транзакции проведен успешно.
Выполнение транзакции
Подготовка      : 1/1
Установка       : dnf-automatic-4.19.0-1.fc39.noarch 1/1
Запуск скрипта : dnf-automatic-4.19.0-1.fc39.noarch 1/1
Проверка        : dnf-automatic-4.19.0-1.fc39.noarch 1/1
```

Рис. 4.16: Установка ПО

17) Запустим таймер с помощью: `systemctl enable --now dnf-automatic.timer` (рис. 4.17).

```
[sdaniil@fedora ~]$ systemctl enable --now dnf-automatic.timer
Created symlink /etc/systemd/system/timers.target.wants/dnf-automatic.timer → /usr/lib/systemd/system/dnf-automatic.timer.
[sdaniil@fedora ~]$
```

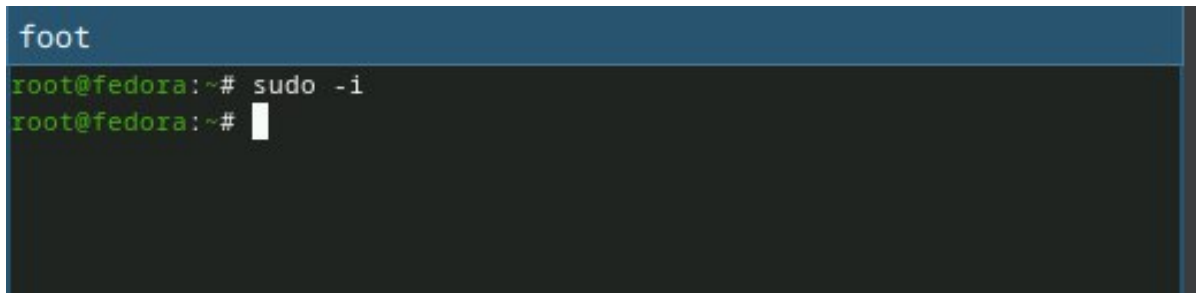
Рис. 4.17: Запуск таймера

- 18) В данном курсе мы не будем рассматривать работу с системой безопасности SELinux. Поэтому отключим его. В файле `/etc/selinux/config` заменим значение `SELINUX=enforcing` на значение `SELINUX=permissive`. После чего перезагрузим виртуальную машину: `reboot` (рис. 4.18).

```
foot
GNU nano 7.2 /etc/selinux/config Изменён
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
# See also:
# https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-
#
# NOTE: In earlier Fedora kernel builds, SELINUX=disabled would also
# fully disable SELinux during boot. If you need a system with SELinux
# fully disabled instead of SELinux running with no policy loaded, you
# need to pass selinux=0 to the kernel command line. You can use grubby
# to persistently set the bootloader to boot with selinux=0:
#
#   grubby --update-kernel ALL --args selinux=0
#
# To revert back to SELinux enabled:
#
#   grubby --update-kernel ALL --remove-args selinux
#
SELINUX=permissive
# SELINUXTYPE= can take one of these three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processe
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Рис. 4.18: Отключение системы безопасности SELinux

- 19) Далее нам нужно установить драйвера для VirtualBox. Войдите в ОС под заданной вами при установке учётной записью. Нажмем комбинацию Win+Enter для запуска терминала. Запустим терминальный мультиплексор tmux: tmux Переключимся на роль супер-пользователя: sudo -i (рис. 4.19).

A terminal window with a dark blue title bar containing the text 'foot'. The terminal background is black. The prompt 'root@fedora:~#' is shown in green. The command 'sudo -i' is entered in white. The prompt changes to 'root@fedora:~#' in white, indicating a successful switch to the root user. A white cursor is visible on the line following the second prompt.

```
foot
root@fedora:~# sudo -i
root@fedora:~#
```

Рис. 4.19: Запуск терминального мультиплексора и переключение на роль супер-пользователя

20) Установим средства разработки: `dnf -y group install "Development Tools"` (рис. 4.20).

foot

```
elfutils-devel-0.190-4.fc39.x86_64
elfutils-libelf-devel-0.190-4.fc39.x86_64
flex-2.6.4-13.fc39.x86_64
gc-8.2.2-4.fc39.x86_64
gcc-13.2.1-6.fc39.x86_64
gettext-0.22-2.fc39.x86_64
git-2.44.0-1.fc39.x86_64
git-core-2.44.0-1.fc39.x86_64
git-core-doc-2.44.0-1.fc39.noarch
glibc-devel-2.38-16.fc39.x86_64
glibc-headers-x86-2.38-16.fc39.noarch
guile22-2.2.7-9.fc39.x86_64
kernel-devel-6.7.6-200.fc39.x86_64
kernel-headers-6.7.3-200.fc39.x86_64
libserf-1.3.10-3.fc39.x86_64
libxcrypt-devel-4.4.36-2.fc39.x86_64
libzstd-devel-1.5.5-4.fc39.x86_64
m4-1.4.19-6.fc39.x86_64
make-1:4.4.1-2.fc39.x86_64
openssl-devel-1:3.1.1-4.fc39.x86_64
patch-2.7.6-22.fc39.x86_64
patchutils-0.4.2-11.fc39.x86_64
perl-Error-1:0.17029-13.fc39.noarch
perl-File-Find-1.43-502.fc39.noarch
perl-Git-2.44.0-1.fc39.noarch
perl-TermReadKey-2.38-18.fc39.x86_64
perl-lib-0.65-502.fc39.x86_64
subversion-1.14.3-1.fc39.x86_64
subversion-libs-1.14.3-1.fc39.x86_64
systemtap-5.1~pre17062192g5fd8daba-1.fc39.x86_64
systemtap-client-5.1~pre17062192g5fd8daba-1.fc39.x86_64
systemtap-devel-5.1~pre17062192g5fd8daba-1.fc39.x86_64
systemtap-runtime-5.1~pre17062192g5fd8daba-1.fc39.x86_64
tbb-2020.3-20.fc39.x86_64
xapian-core-libs-1.4.23-1.fc39.x86_64
xz-devel-5.4.4-1.fc39.x86_64
zlib-devel-1.2.13-4.fc39.x86_64
```

Выполнено!

root@fedora:~#

root@fedora:~#

Рис. 4.20: Установка средств разработки

21) Установим пакет DKMS: `dnf -y install dkms` (рис. 4.21).

```
foot
Установка:
  dkms                noarch      3.0.12-1.fc39      updates      80 k
Установка зависимостей:
  kernel-devel-matched x86_64     6.7.6-200.fc39     updates     161 k
Установка слабых зависимостей:
  openssl             x86_64     1:3.1.1-4.fc39     fedora      1.0 M

Результат транзакции
=====
Установка 3 Пакета

Объем загрузки: 1.2 М
Объем изменений: 1.8 М
Загрузка пакетов:
(1/3): kernel-devel-matched-6.7.6-200.f 622 kB/s | 161 kB    00:00
(2/3): dkms-3.0.12-1.fc39.noarch.rpm    305 kB/s | 80 kB     00:00
(3/3): openssl-3.1.1-4.fc39.x86_64.rpm  2.9 MB/s | 1.0 MB    00:00
-----
Общий размер                1.3 MB/s | 1.2 MB    00:00
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
Тест транзакции проведен успешно.
Выполнение транзакции
  Подготовка      :                               1/1
  Установка       : kernel-devel-matched-6.7.6-200.fc39.x86_64 1/3
  Установка       : openssl-1:3.1.1-4.fc39.x86_64             2/3
  Установка       : dkms-3.0.12-1.fc39.noarch                  3/3
  Запуск скрипта  : dkms-3.0.12-1.fc39.noarch                  3/3
Created symlink /etc/systemd/system/multi-user.target.wants/dkms.service
→ /usr/lib/systemd/system/dkms.service.

  Проверка        : openssl-1:3.1.1-4.fc39.x86_64             1/3
  Проверка        : dkms-3.0.12-1.fc39.noarch                  2/3
  Проверка        : kernel-devel-matched-6.7.6-200.fc39.x86_64 3/3

Установлен:
  dkms-3.0.12-1.fc39.noarch
  kernel-devel-matched-6.7.6-200.fc39.x86_64
  openssl-1:3.1.1-4.fc39.x86_64

Выполнено!
root@fedora:~#
```

[0] 0:sudo* "fedora" 02:10 02-мар-24

Рис. 4.21: Установка пакета DKMS

- 22) В меню виртуальной машины подключим образ диска дополнений гостевой ОС. (рис. 4.22).

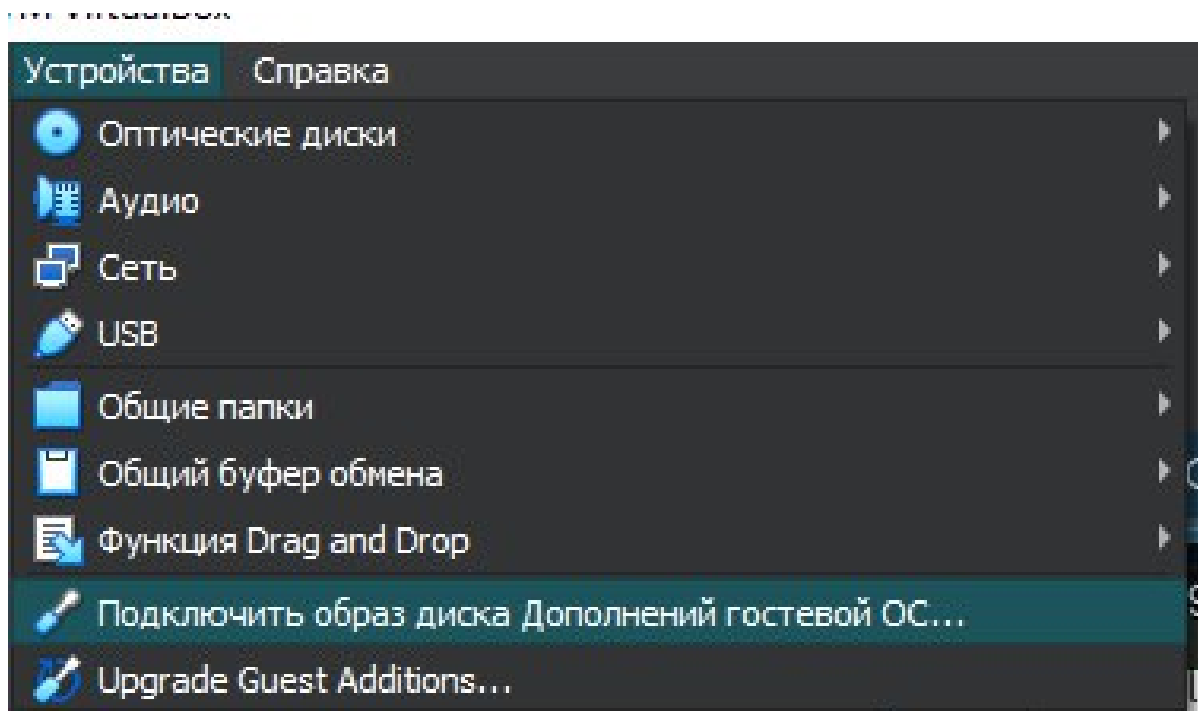


Рис. 4.22: Подключение образа диска дополнений гостевой ОС

- 23) Подмонтируйте диск: `mount /dev/sr0 /media` (рис. 4.23).

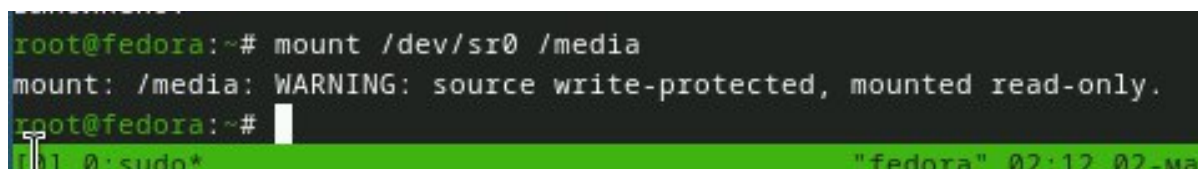


Рис. 4.23: Монтаж диска

- 24) Установим драйвера: `/media/VBoxLinuxAdditions.run` После чего перегрузим виртуальную машину: `reboot` (рис. 4.24).

```

root@fedora:~# /media/VBoxLinuxAdditions.run
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.0.10 Guest Additions for Linux 100%
VirtualBox Guest Additions installer
Removing installed version 7.0.10 of VirtualBox Guest Additions...
Copying additional installer modules ...
Installing additional modules ...
VirtualBox Guest Additions: Starting.
VirtualBox Guest Additions: Setting up modules
VirtualBox Guest Additions: Building the VirtualBox Guest Additions kernel
modules. This may take a while.
VirtualBox Guest Additions: To build modules for other installed kernels
, run
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup <version>
VirtualBox Guest Additions: or
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup all
VirtualBox Guest Additions: Building the modules for kernel
6.7.6-200.fc39.x86_64.

VirtualBox Guest Additions: Look at /var/log/vboxadd-setup.log to find o
ut what
went wrong
ValueError: File context for /opt/VBoxGuestAdditions-7.0.10/other/mount.
vboxsf already defined
VirtualBox Guest Additions: Running kernel modules will not be replaced
until
the system is restarted or 'rcvboxadd reload' triggered
VirtualBox Guest Additions: reloading kernel modules and services
VirtualBox Guest Additions: cannot reload kernel modules: one or more mo
dule(s)
is still in use
VirtualBox Guest Additions: kernel modules and services were not reload
ed
The log file /var/log/vboxadd-setup.log may contain further information.
root@fedora:~#
[0] 0:sudo* "fedora" 02:19 02-мар-24

```

Рис. 4.24: Установка драйверов

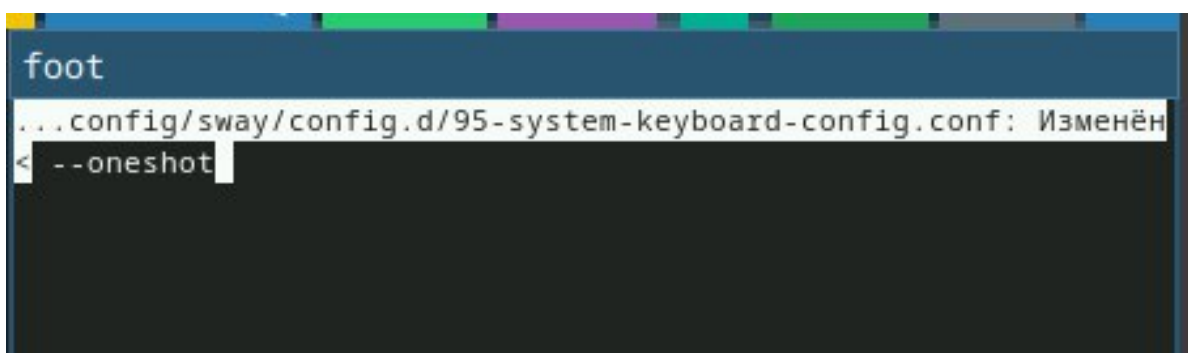
- 25) Войдем в ОС под заданной при установке учётной записью. Нажмем комбинацию Win+Enter для запуска терминала. После чего запустим терминальный мультиплексор tmux: tmux

Создадим конфигурационный файл `~/.config/sway/config.d/95-system-keyboard-config.conf`: (рис. 4.25).

```
sdaniil@fedora:~$ mkdir ~/.config/sway
sdaniil@fedora:~$ mkdir ~/.config/sway/config.d
sdaniil@fedora:~$ touch ~/.config/sway/config.d/95-system-keyboard-config.conf
sdaniil@fedora:~$
```

Рис. 4.25: Создание конфигурационного файла

26) Отредактируем конфигурационный файл `~/.config/sway/config.d/95-system-keyboard-config.conf`: (рис. 4.26).

A screenshot of a terminal window titled 'foot'. The terminal shows the path `...config/sway/config.d/95-system-keyboard-config.conf: Изменён` and the command `< --oneshot` being entered. The terminal has a dark background with a light blue header bar.

```
foot
...config/sway/config.d/95-system-keyboard-config.conf: Изменён
< --oneshot
```

Рис. 4.26: Редактирование конфигурационного файла

27) Переключимся на роль супер-пользователя: `sudo -i` (рис. 4.27).

```
sdaniil@fedora:~$ sudo -i
[sudo] пароль для sdaniil:
root@fedora:~#
```

Рис. 4.27: Супер-пользователь

- 28) Отредактируем конфигурационный файл `/etc/X11/xorg.conf.d/00-keyboard.conf`:
(рис. 4.28).

```
foot
/etc/X11/xorg.conf.d/00-keyboard.conf:
Section "InputClass"
    Identifier "system-keyboard"
    MatchIsKeyboard "on"
    Option "XkbLayout" "us,ru"
    Option "XkbVariant" ",winkeys"
    Option "XkbOptions" "grp:rctrl_tog>
EndSection
```

Рис. 4.28: Редактирование конфигурационного файла

После этого перезагрузим виртуальную машину.

- 29) Установим имя хоста: `hostnamectl set-hostname username` Проверим, что
имя хоста установлено верно: `hostnamectl` (рис. 4.29).


```
root@fedora:~# hostnamectl set-hostname sdaniil
root@fedora:~# hostnamectl
  Static hostname: sdaniil
            Icon name: computer-vm
            Chassis: vm
            Machine ID: 9ecf6202b1ca4ba49e1a0ca46d797242
            Boot ID: a9ac06f11fb943e3b67efbc547c2771c
    Virtualization: oracle
  Operating System: Fedora Linux 39 (Sway)
        CPE OS Name: cpe:/o:fedoraproject:fedora:39
    OS Support End: Tue 2024-11-12
OS Support Remaining: 8month 1w 4d
            Kernel: Linux 6.7.6-200.fc39.x86_64
    Architecture: x86-64
    Hardware Vendor: innotek GmbH
    Hardware Model: VirtualBox
    Firmware Version: VirtualBox
        Firmware Date: Fri 2006-12-01
        Firmware Age: 17y 3month
root@fedora:~#
```

Рис. 4.29: Установка имени хоста

30) Внутри виртуальной машины добавим своего пользователя в группу vboxsf: `gpasswd -a username vboxsf`

31) Создадим папку work в ОС Windows. Запустим командную строку на винд. после чего введем следующую команду, для добавления папки. (рис. 4.30).

```
C:\Users\Danil>"C:\Program Files\Oracle\VirtualBox\VBoxManage.exe" sharedfolder add "Fedora" --name=work --hostpath="C:/work" --automount
C:\Users\Danil>
```

Рис. 4.30: Подключение разделяемой папки

После этих действий перезагружаем виртуальную машину.

32)Нажмем комбинацию Win+Enter для запуска терминала. Запустим терминальный мультиплексор tmux: `tmux`

Переключимся на роль супер-пользователя: `sudo -i`

Средство `pandoc` для работы с языком разметки Markdown.

Установка с помощью менеджера пакетов: `dnf -y install pandoc` (рис. 4.31).

```
sdaniil@sdaniil:~$ sudo -i
[sudo] пароль для sdaniil:
root@sdaniil:~# dnf -y install pandoc
Последняя проверка окончания срока действия метаданных: 1:50:33 назад, С
6 02 мар 2024 01:03:59.
Зависимости разрешены.
=====
Пакет                Архитектура Версия                Репозиторий  Размер
=====
Установка:
pandoc                x86_64      3.1.3-25.fc39                updates      26 М
Установка зависимостей:
pandoc-common         noarch      3.1.3-25.fc39                updates      527 к
Результат транзакции
=====
Установка 2 Пакета
Объем загрузки: 26 М
```

Рис. 4.31: Установка `pandoc`

33) Для работы с перекрёстными ссылками мы используем пакет `pandoc-crossref`. Находим и скачиваем его с репозитория гит по ссылке указанной в лабораторной работе. (рис. 4.32).

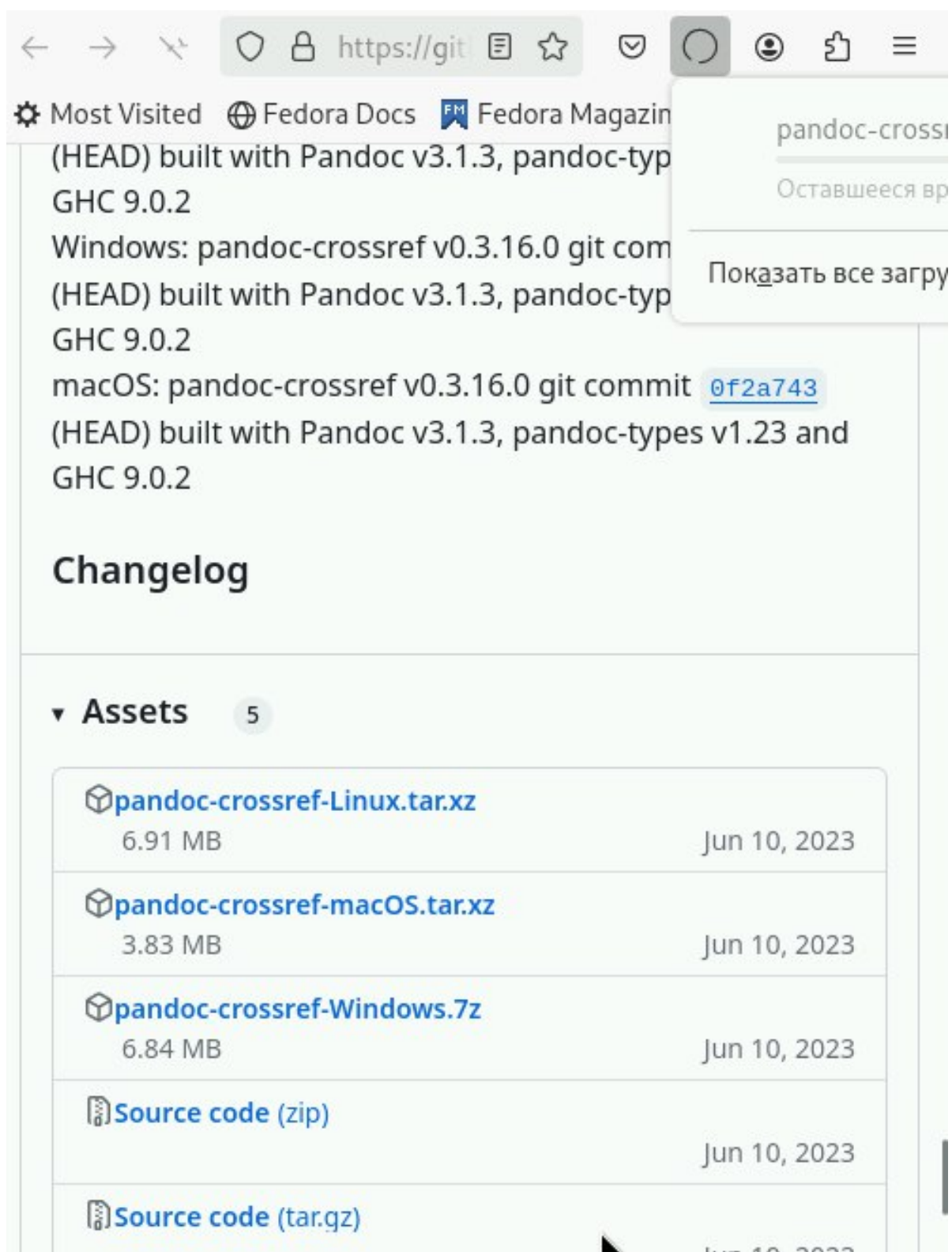


Рис. 4.32: Установка pandoc-crossref

34) После установки переходим в папку с установленным файлом и распаковываем архивы. (рис. 4.33 4.34).

```
[sdaniil@sdaniil ~]$ cd Загрузки
[sdaniil@sdaniil Загрузки]$ ls
'pandoc-crossref-Linux.tar(1).xz'  pandoc-crossref-Linux.tar.xz
[sdaniil@sdaniil Загрузки]$ tar -xvf pandoc-crossref-Linux.tar.xz
pandoc-crossref
pandoc-crossref.1
[sdaniil@sdaniil Загрузки]$
```

Рис. 4.33: Переход в папку загрузки

```
[sdaniil@sdaniil Загрузки]$ sudo mv pandoc-crossref /usr/local/bin
[sdaniil@sdaniil Загрузки]$
```

Рис. 4.34: Распаковка архивов

35) Установим дистрибутив TeXlive: `dnf -y install texlive-scheme-full` (рис. 4.35).

```
texlive-yhmath-11:svn54377-69.fc39.noarch
texlive-yinit-otf-11:svn40207-69.fc39.noarch
texlive-york-thesis-11:svn23348.3.6-69.fc39.noarch
texlive-youngtab-11:svn56500-69.fc39.noarch
texlive-yplan-11:svn34398-77.fc39.noarch
texlive-yquant-11:svn65933-69.fc39.noarch
texlive-ytableau-11:svn59580-69.fc39.noarch
texlive-zapfchan-11:svn61719-69.fc39.noarch
texlive-zapfding-11:svn61719-69.fc39.noarch
texlive-zbmath-review-template-11:svn59693-69.fc39.noarch
texlive-zebra-goodies-11:svn51554-69.fc39.noarch
texlive-zed-csp-11:svn17258.0-69.fc39.noarch
texlive-zennote-11:svn65549-69.fc39.noarch
texlive-zhlineskip-11:svn51142-69.fc39.noarch
texlive-zhlipsum-11:svn54994-69.fc39.noarch
texlive-zhmetrics-11:svn22207.r206-69.fc39.noarch
texlive-zhmetrics-uptex-11:svn40728-69.fc39.noarch
texlive-zhnumber-11:svn66115-69.fc39.noarch
texlive-zhspacing-11:svn41145-69.fc39.noarch
texlive-ziffer-11:svn32279.2.1-69.fc39.noarch
texlive-zitie-11:svn60676-69.fc39.noarch
texlive-zlmtt-11:svn64076-69.fc39.noarch
texlive-zootaxa-bst-11:svn50619-69.fc39.noarch
texlive-zref-11:svn62977-69.fc39.noarch
texlive-zref-check-11:svn63845-69.fc39.noarch
texlive-zref-clever-11:svn66021-69.fc39.noarch
texlive-zref-vario-11:svn65453-69.fc39.noarch
texlive-zwgetfdate-11:svn15878.0-69.fc39.noarch
texlive-zwpagelayout-11:svn63074-69.fc39.noarch
texlive-zx-calculus-11:svn60838-69.fc39.noarch
texlive-zxjafbfont-11:svn28539.0.2-69.fc39.noarch
texlive-zxjafont-11:svn62864-69.fc39.noarch
texlive-zxjatype-11:svn53500-69.fc39.noarch
texlive-zztex-11:svn55862-69.fc39.noarch
tk-1:8.6.12-5.fc39.x86_64
tre-0.8.0-41.20140228gitc2f5d13.fc39.x86_64
tre-common-0.8.0-41.20140228gitc2f5d13.fc39.noarch
tzdata-java-2024a-2.fc39.noarch
urw-base35-fonts-legacy-20200910-18.fc39.noarch
xpdf-libs-1:4.04-10.fc39.x86_64
zziplib-0.13.72-5.fc39.x86_64
```

Выполнено!

[root@sdaniil ~]# █

Рис. 4.35: Установка TeXLive

Домашнее задание

В окне терминала проанализируем последовательность загрузки системы, выполнив команду `dmesg`. Можно просто просмотреть вывод этой команды: `dmesg | less` (рис. 4.36).


```
foot
[ 0.000000] Linux version 6.7.6-200.fc39.x86_64 (mockbuild@1fbae28ea3
8d40908fb246e7adfe592f) (gcc (GCC) 13.2.1 20231205 (Red Hat 13.2.1-6), G
NU ld version 2.40-14.fc39) #1 SMP PREEMPT_DYNAMIC Fri Feb 23 18:27:29 U
TC 2024
[ 0.000000] Command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.7.6-200.fc3
9.x86_64 root=UUID=1148372d-daf9-4512-b238-679b871d2c7e ro rootflags=sub
vol=root nomodeset vga=791 rhgb quiet
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] us
able
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] re
served
[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] re
served
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000dfffffff] us
able
[ 0.000000] BIOS-e820: [mem 0x00000000dffff000-0x00000000dfffffff] AC
PI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] re
served
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee00fff] re
served
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] re
served
[ 0.000000] BIOS-e820: [mem 0x0000000100000000-0x000000021fffffffff] us
able
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] APIC: Static calls initialized
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox
12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using msrs 4b564d01 and 4b564d00
[ 0.000001] kvm-clock: using sched offset of 8641135075 cycles
[ 0.000003] clocksource: kvm-clock: mask: 0xffffffffffffffff max_cycl
es: 0x1cd42e4dfffb, max_idle_ns: 881590591483 ns
[ 0.000006] tsc: Detected 2611.202 MHz processor
[ 0.000784] e820: update [mem 0x00000000-0x00000fff] usable ==> reser
ved
[ 0.000787] e820: remove [mem 0x000a0000-0x000fffff] usable
[ 0.000791] last_pfn = 0x220000 max_arch_pfn = 0x40000000
[ 0.000797] MTRRs disabled by BIOS
[ 0.000799] x86/PAT: Configuration [0-7]: WB WC UC- UC WB WP UC-
:
```

Рис. 4.36: Просмотр вывода команды

С помощью grep:

```
dmesg | grep -i "то, что ищем"
```

Получим следующую информацию: (рис. 4.37).

Версия ядра Linux (Linux version).

Частота процессора (Detected Mhz processor).

Модель процессора (CPU0).

Объём доступной оперативной памяти (Memory available).

Тип обнаруженного гипервизора (Hypervisor detected).

Тип файловой системы корневого раздела.

Последовательность монтирования файловых систем.


```

Выполнено!
[root@sdaniil ~]# dmesg | less
[root@sdaniil ~]# dmesg | grep -i "Linux version"
[    0.000000] Linux version 6.7.6-200.fc39.x86_64 (mockbuild@1fbae28ea3
8d40908fb246e7adfe592f) (gcc (GCC) 13.2.1 20231205 (Red Hat 13.2.1-6), G
NU ld version 2.40-14.fc39) #1 SMP PREEMPT_DYNAMIC Fri Feb 23 18:27:29 U
TC 2024
[root@sdaniil ~]# dmesg | grep -i "Mhz processor"
[    0.000006] tsc: Detected 2611.202 MHz processor
[root@sdaniil ~]# dmesg | grep -i "CPU0"
[    0.188170] smpboot: CPU0: 11th Gen Intel(R) Core(TM) i5-11260H @ 2.6
0GHz (family: 0x6, model: 0x8d, stepping: 0x1)
[root@sdaniil ~]# dmesg | grep -i "Memory available"
[root@sdaniil ~]#
[root@sdaniil ~]# dmesg | grep -i "available"
[    0.001946] On node 0, zone DMA: 1 pages in unavailable ranges
[    0.001993] On node 0, zone DMA: 97 pages in unavailable ranges
[    0.022778] On node 0, zone Normal: 16 pages in unavailable ranges
[    0.023132] [mem 0xe0000000-0xfebfffff] available for PCI devices
[    0.028396] Booted with the nomodeset parameter. Only the system fram
ebuffer will be available
[    0.056080] Memory: 8084616K/8388152K available (20480K kernel code,
3276K rwdatas, 14748K rodata, 4588K init, 4892K bss, 303276K reserved, 0K
cma-reserved)
[root@sdaniil ~]# dmesg | grep -i "Hypervisor detected"
[    0.000000] Hypervisor detected: KVM
[root@sdaniil ~]# dmesg | grep -i "filesystem"
[root@sdaniil ~]# dmesg | grep -i "filesystem"
[    2.930176] BTRFS info (device sda3): first mount of filesystem 11483
72d-daf9-4512-b238-679b871d2c7e
[    7.306286] EXT4-fs (sda2): mounted filesystem e16d66a0-500b-41b7-adb
1-569cd5c5c803 r/w with ordered data mode. Quota mode: none.
[root@sdaniil ~]# █

```

Рис. 4.37: Использование команды grep

#Контрольные вопросы

- 1) Какую информацию содержит учётная запись пользователя?

Системное имя, идентификатор пользователя, идентификатор группы, полное имя, домашний каталог, начальная оболочка.

2) Укажите команды терминала и приведите примеры:

для получения справки по команде; `man (man ls)`

для перемещения по файловой системе; `cd (cd / -перемещение в корневой каталог)`

для просмотра содержимого каталога; `ls (ls / -содержимое корневого каталога)`

для определения объёма каталога; `du -s (du -s /etc)`

для создания / удаления каталогов / файлов;; `rm`

Пустые каталоги можно удалять командой `rmdir` (если добавить ключ `-s`, то можно удалять и не только пустые).

Также любые файлы можно удалять рекурсивно: `rm -r` для задания определённых прав на файл / каталог; `chmod (chmod 777 filename.txt)` для просмотра истории команд; `history`

3) Что такое файловая система? Приведите примеры с краткой характеристикой.

Файловая система - это способ организации и хранения данных на носителе информации, таком как жесткий диск или флэш-накопитель. Она определяет способ, которым файлы и каталоги структурируются, и как к ним обращаться.

Вот несколько примеров файловых систем в Linux:

ext4 (Fourth Extended Filesystem): Это одна из наиболее распространенных файловых систем в Linux. Она обеспечивает хорошую производительность и надежность, поддерживает большие размеры файлов и разделов. ext4 является стандартной файловой системой для многих дистрибутивов Linux.

Btrfs (B-tree File System): Это современная файловая система, которая поддерживает функции копирования на запись, снимков и сжатия данных. Btrfs предоставляет возможности по обнаружению и восстановлению поврежденных данных, а также управлению множеством дисков.

XFS (XFS File System): Эта файловая система изначально разработана для высокопроизводительных систем. Она обладает хорошей поддержкой больших файлов и разделов, а также высокой параллельной производительностью ввода-вывода.

ZFS (Zettabyte File System): Хотя ZFS не является частью ядра Linux из коробки из-за проблем лицензирования, он все равно доступен для установки и использования. ZFS предлагает мощные функции, такие как проверка целостности данных, снимки, моментальные копии и встроенное RAID.

F2FS (Flash-Friendly File System): Эта файловая система оптимизирована для использования на флэш-накопителях, таких как SSD. F2FS учитывает особенности флэш-памяти, такие как износ и способы записи, для повышения производительности и срока службы носителя.

4) Как посмотреть, какие файловые системы подмонтированы в ОС?

Командой `mount` 5) Как удалить зависший процесс? Узнайте идентификатор процесса (PID):

Вы можете использовать команду `ps aux | grep` для поиска запущенных процессов и их PID.

Например: `ps aux | grep firefox` Это покажет список процессов, связанных с Firefox, и их PID. Используйте команду `kill` для завершения процесса: Как только вы найдете PID зависшего процесса, используйте команду `kill` с этим PID для завершения процесса. Например `kill -9 -9` это сигнал, который немедленно завершает процесс. Обычно это работает, если процесс завис, и не реагирует на обычные сигналы завершения.

5 Выводы

В итоге выполнения данной лабораторной работы я приобрёл практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.