

Отчет по лабораторной работе №1

Операционные системы

Дворкина Ева Владимировна

Содержание

1	Цель работы	6
2	Задание	7
3	Выполнение лабораторной работы	8
3.1	Создание виртуальной машины	8
3.2	Установка операционной системы	13
3.3	Работа с операционной системой после установки	20
3.4	Установка программного обеспечения для создания документации	27
4	Выводы	28
5	Ответы на контрольные вопросы	29
6	Выполнение дополнительного задания	31
	Список литературы	34

Список иллюстраций

3.1	Окно Virtualbox	8
3.2	Создание виртуальной машины	9
3.3	Указание объема памяти	9
3.4	Жесткий диск	10
3.5	Тип жесткого диска	10
3.6	Размер жесткого диска	11
3.7	Формат хранения жесткого диска	12
3.8	Выбор образа оптического диска	12
3.9	Выбранный образ оптического диска	13
3.10	Окно загрузчика	13
3.11	Интерфейс начальной конфигурации	14
3.12	Запуск терминала	15
3.13	Выбор языка интерфейса	15
3.14	Выбор раскладки клавиатуры	16
3.15	Выбор часового пояса	16
3.16	Выбор места установки	17
3.17	Задание сетевого имени компьютера	17
3.18	Создание аккаунта администратора	18
3.19	Создание пользователя	19
3.20	Завершение установки операционной системы	19
3.21	Просмотр оптического диска	20
3.22	Отключение оптического диска	20
3.23	Вход в ОС	21
3.24	Запуск терминала	21
3.25	Обновления	22
3.26	Установка tmux и mc	22
3.27	Установка программного обеспечения для автоматического обновления	22
3.28	Запуск таймера	23
3.29	Поиск файла	23
3.30	Изменение файла	24
3.31	Перезагрузка виртуальной машины	24
3.32	Запуск терминального мультиплексора	24
3.33	Переключение на роль супер-пользователя	25
3.34	Установка пакета dkms	25
3.35	Примонтирование диска	25
3.36	Установка драйвера	25

3.37	Перезагрузка виртуальной машины	26
3.38	Поиск файла, вход в mc	26
3.39	Редактирование файла	26
3.40	Перезагрузка виртуальной машины	26
3.41	Переключение на роль супер-пользователя	27
3.42	Установка pandoc	27
3.43	Установка расширения pandoc	27
3.44	Установка texlive	27
6.1	Анализ последовательности загрузки системы	31
6.2	Поиск версии ядра	31
6.3	Поиск частоты процессора	32
6.4	Поиск модели процессора	32
6.5	Поиск объема доступной оперативной памяти	32
6.6	Поиск типа обнаруженного гипервизора	32
6.7	Поиск типа файловой системы корневого раздела	33
6.8	Последовательность монтирования файловых систем	33

Список таблиц

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

1. Создание виртуальной машины
2. Установка операционной системы
3. Работа с операционной системой после установки
4. Установка программного обеспечения для создания документации
5. Дополнительные задания

3 Выполнение лабораторной работы

3.1 Создание виртуальной машины

Virtualbox я устанавливала и настраивала при выполнении лабораторной работы в курсе “Архитектура компьютера и Операционные системы (раздел”Архитектура компьютера”)“, поэтому сразу открываю окно приложения (рис. 3.1).

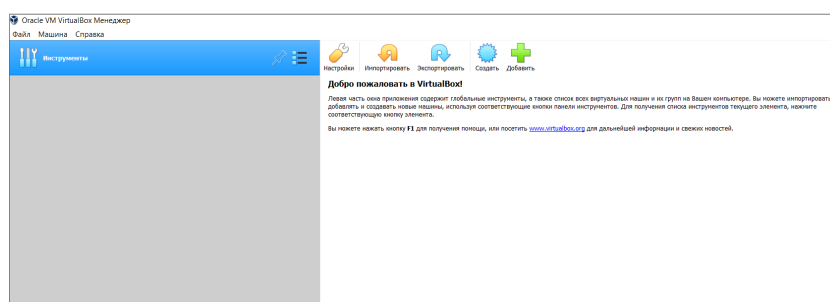


Рис. 3.1: Окно Virtualbox

Нажимая “создать”, создаю новую виртуальную машину, указываю ее имя, путь к папке машины по умолчанию меня устраивает, выбираю тип ОС и версию (рис. 3.2).

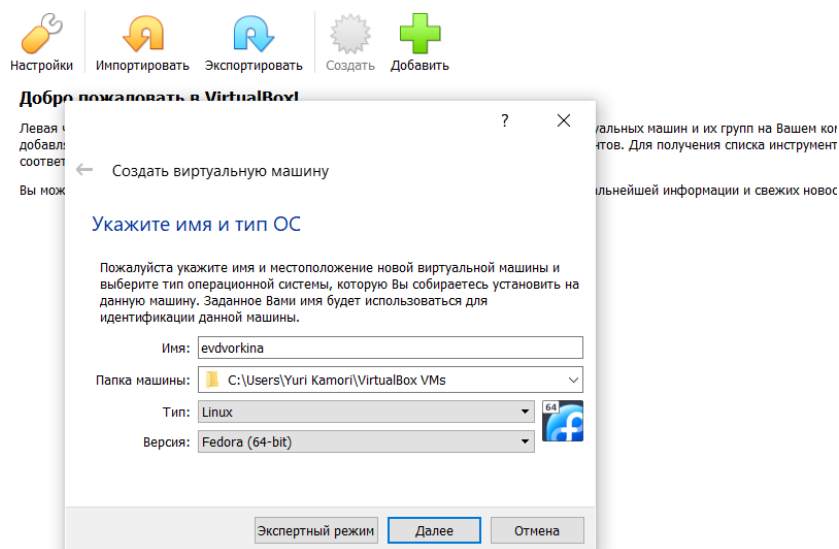


Рис. 3.2: Создание виртуальной машины

Указываю объем основной памяти виртуальной машины размером 4096МБ (рис. 3.3).

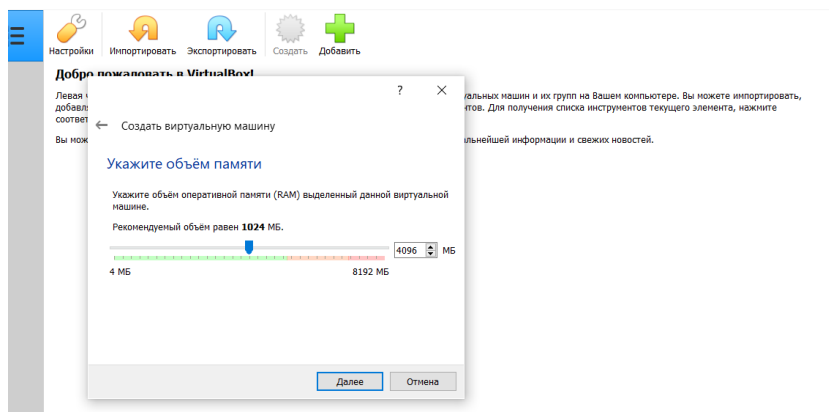


Рис. 3.3: Указание объема памяти

Выбираю создание нового виртуального жесткого диска (рис. 3.4).

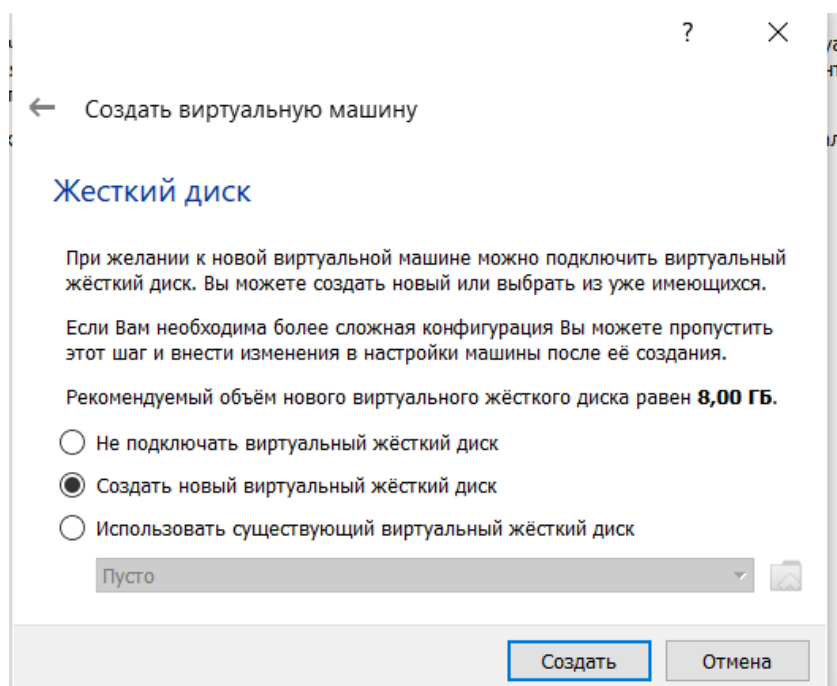


Рис. 3.4: Жесткий диск

Задаю конфигурацию жесткого диска: загрузочный VDI (рис. 3.5).

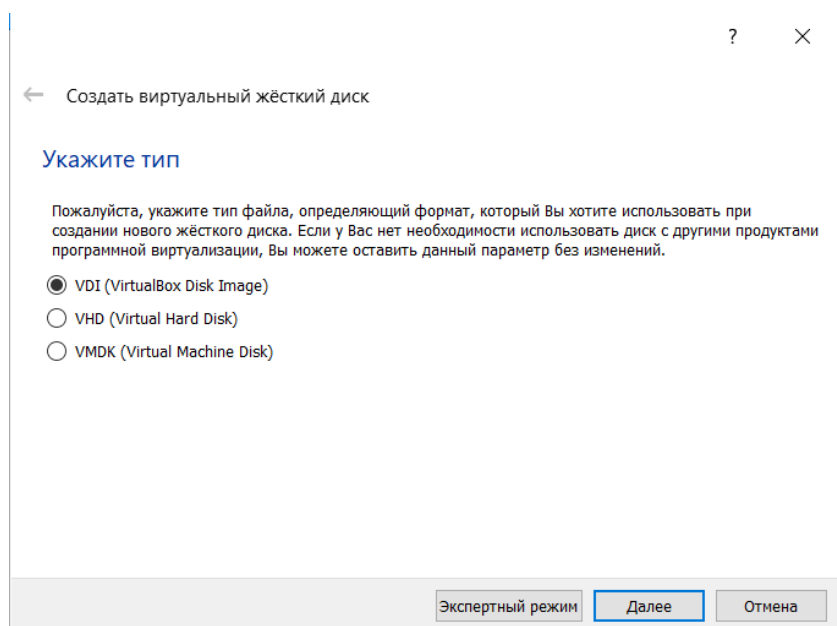


Рис. 3.5: Тип жесткого диска

Задаю размер диска - 80 ГБ, оставляю расположение жесткого диска по умолчанию, т. к. работаю на собственной технике и значение по умолчанию меня устраивает (рис. 3.6).

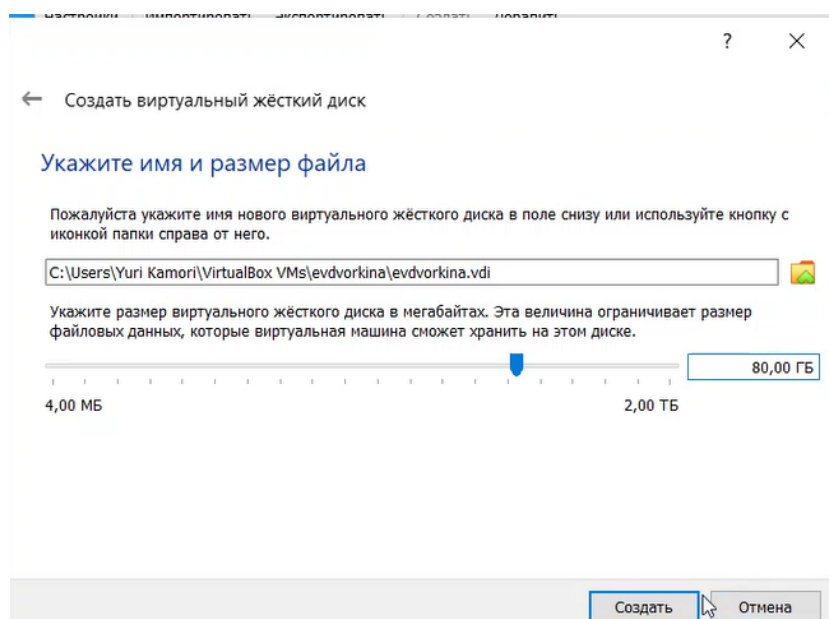


Рис. 3.6: Размер жесткого диска

Выбираю динамический виртуальный жесткого диска при указании формата хранения (рис. 3.7).

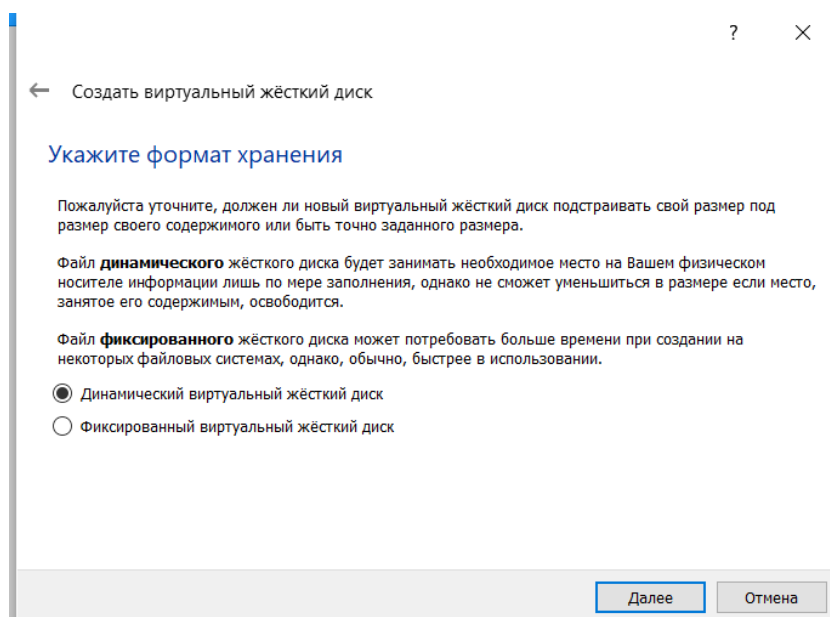


Рис. 3.7: Формат хранения жесткого диска

Выбираю в Virtualbox настройку своей виртуальной машины. Перехожу в “Носители”, добавляю новый привод оптических дисков и выбираю скачанный образ операционной системы Fedora (рис. 3.8).

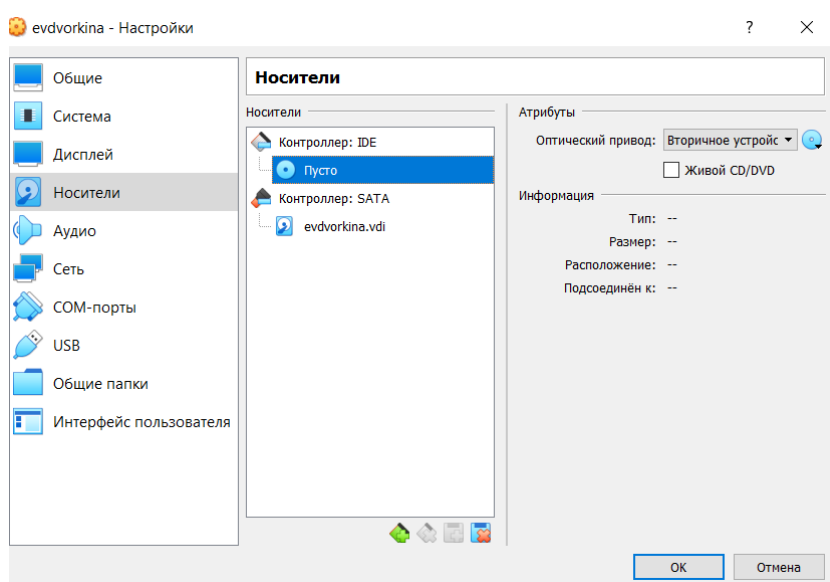


Рис. 3.8: Выбор образа оптического диска

Скачанный образ ОС был успешно выбран (рис. 3.9).

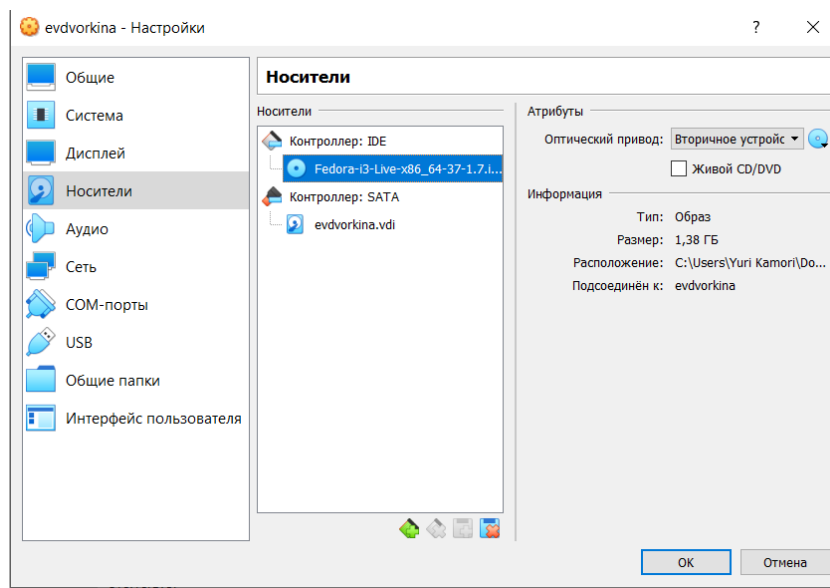


Рис. 3.9: Выбранный образ оптического диска

3.2 Установка операционной системы

Запускаю созданную виртуальную машину для установки (рис. 3.10).

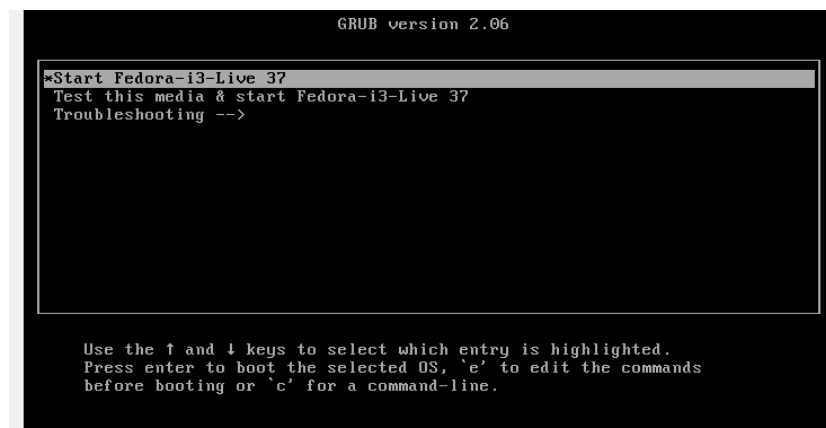


Рис. 3.10: Окно загрузчика

Вижу интерфейс начальной конфигурации. Нажимаю Enter для создания кон-

фигурации по умолчанию, далее нажимаю Enter, чтобы выбрать в качестве модификатора клавишу Win (рис. 3.11).

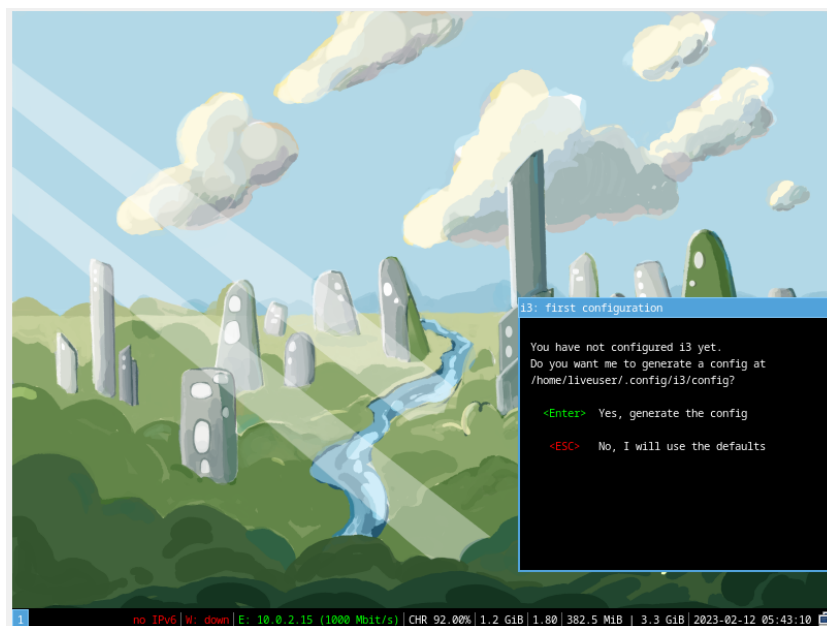


Рис. 3.11: Интерфейс начальной конфигурации

Нажимаю Win+Enter для запуска терминала. В терминале запускаю liveinst (рис. 3.12).

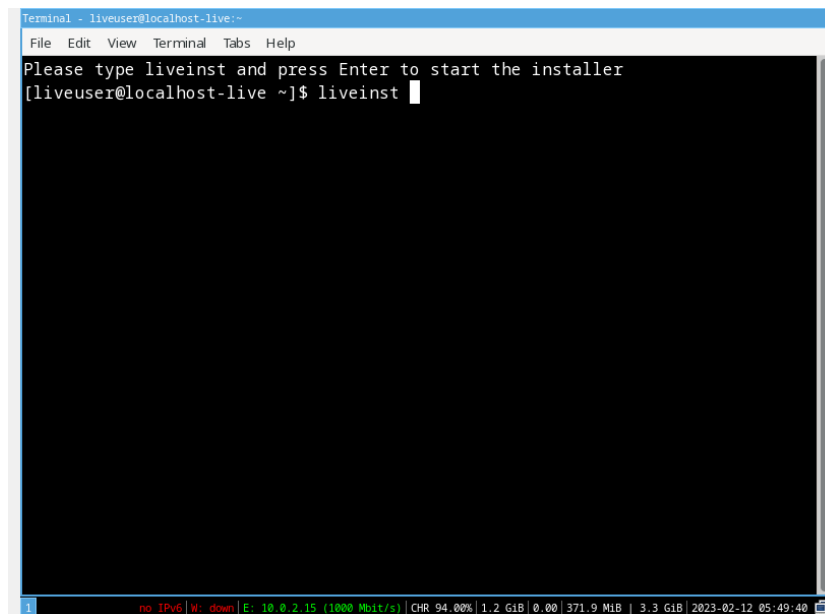


Рис. 3.12: Запуск терминала

Чтобы перейти к раскладке окон с табами, нажимаю Win+w. Выбираю язык для использования в процессе установки русски (рис. 3.13).

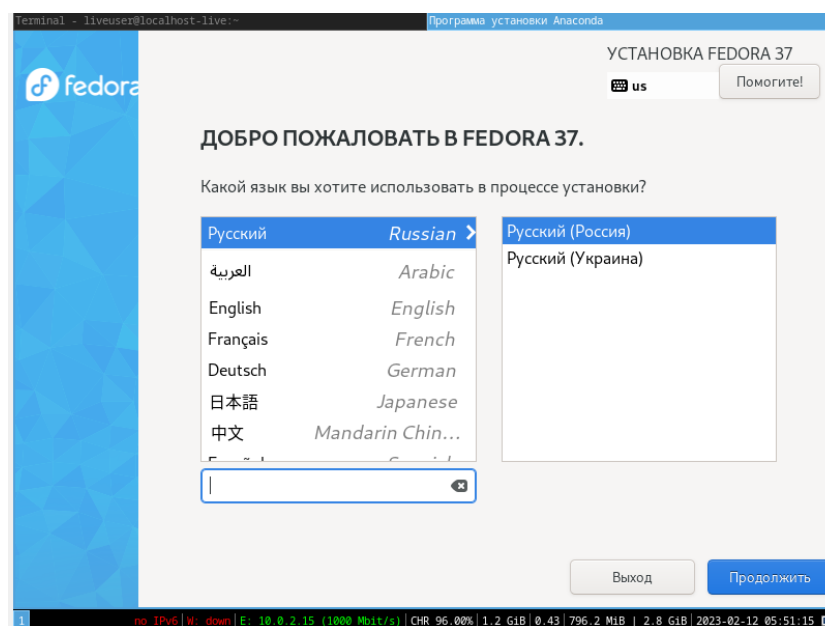


Рис. 3.13: Выбор языка интерфейса

Раскладку клавиатуры выбираю и русскую, и английскую (рис. 3.14).

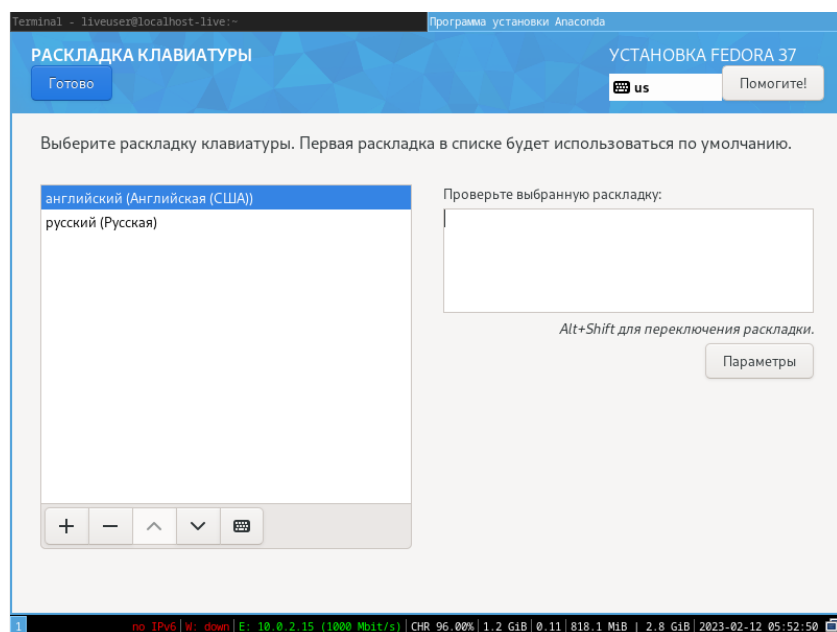


Рис. 3.14: Выбор раскладки клавиатуры

Корректирую часовой пояс, чтобы время на виртуальной машине совпадало с временем в моем регионе (рис. 3.15).

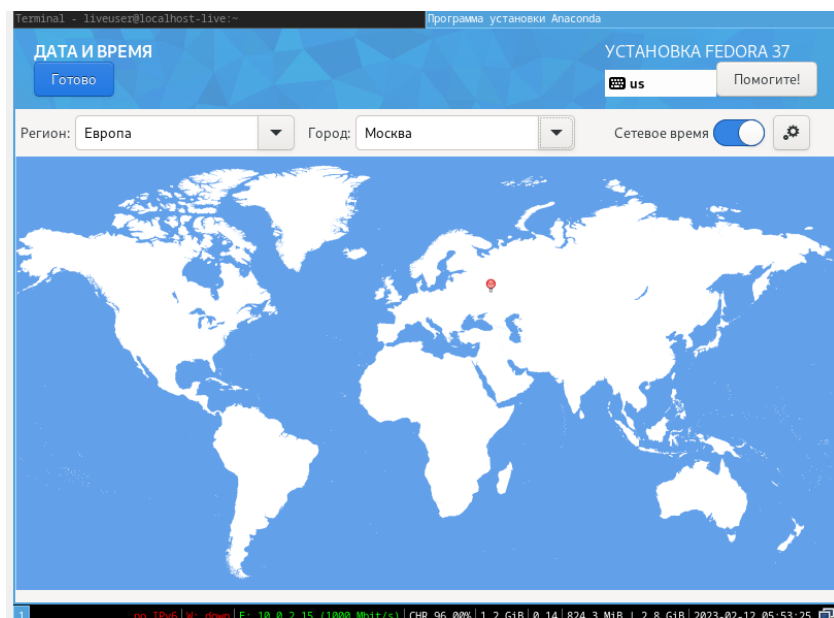


Рис. 3.15: Выбор часового пояса

Проверяю место установки и сохраняю значение по умолчанию (рис. 3.16).

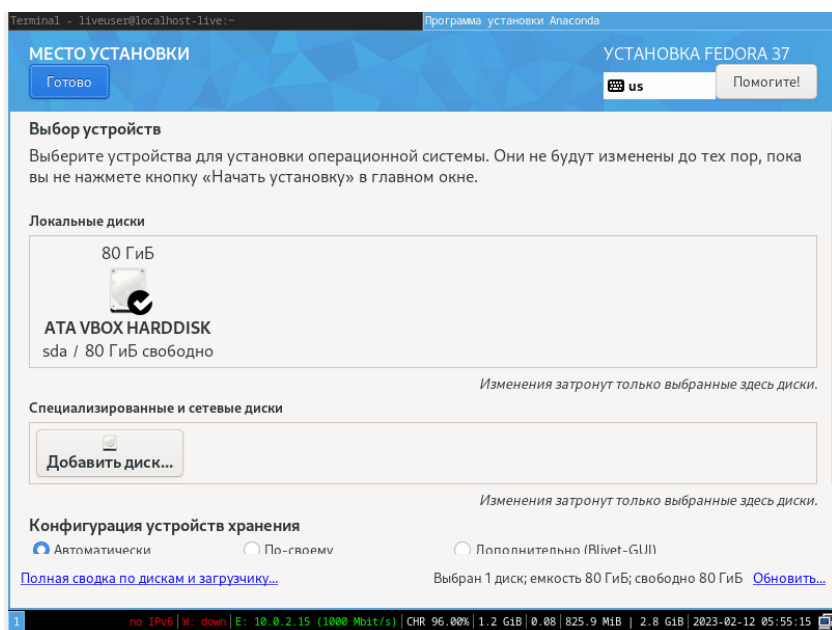


Рис. 3.16: Выбор места установки

Задаю сетевое имя компьютера в соответствии с соглашением об именовании (рис. 3.17).

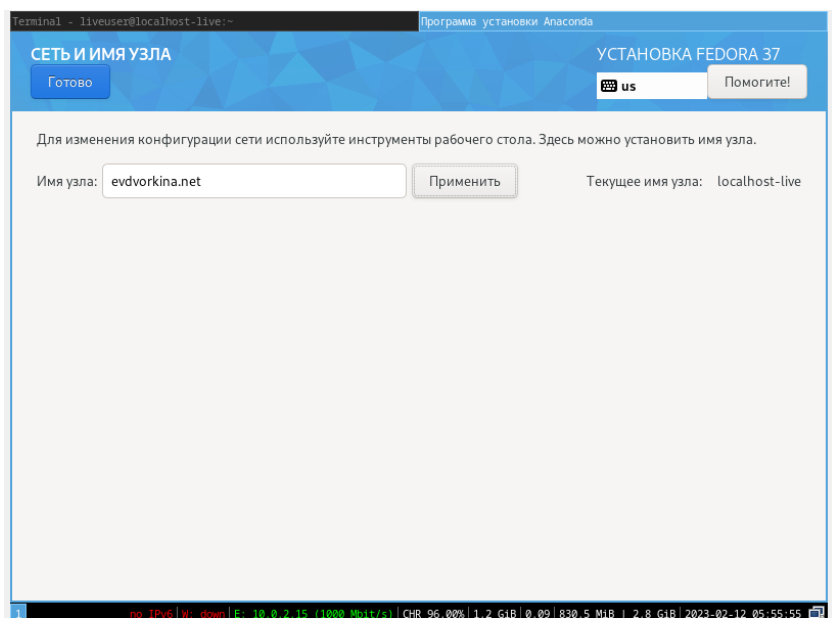


Рис. 3.17: Задание сетевого имени компьютера

Создаю аккаунт администратора и создаю пароль для супер-пользователя (рис. 3.18).

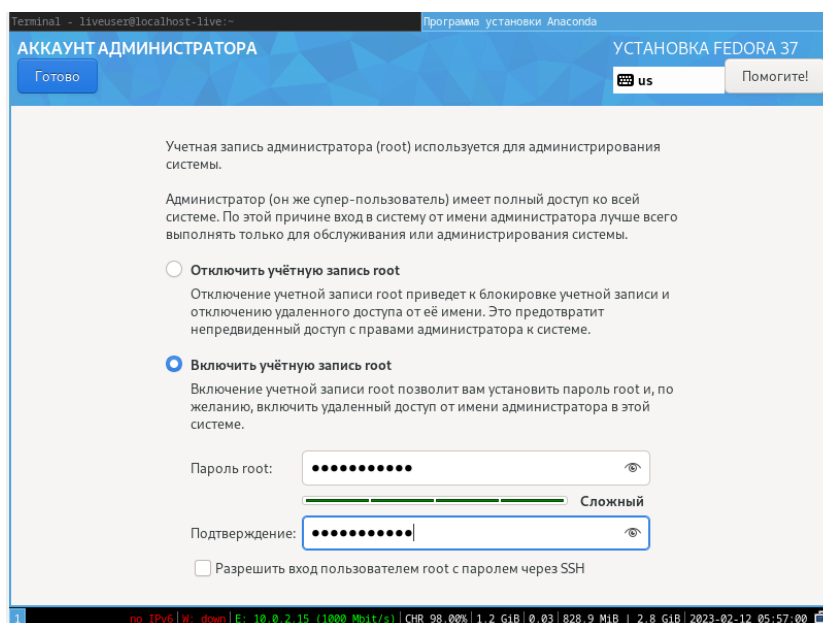


Рис. 3.18: Создание аккаунта администратора

Создаю пользователя, добавляю административные привилегии для этой учетной записи, чтобы я могла свободно выполнять команды как супер-пользователь (рис. 3.19).

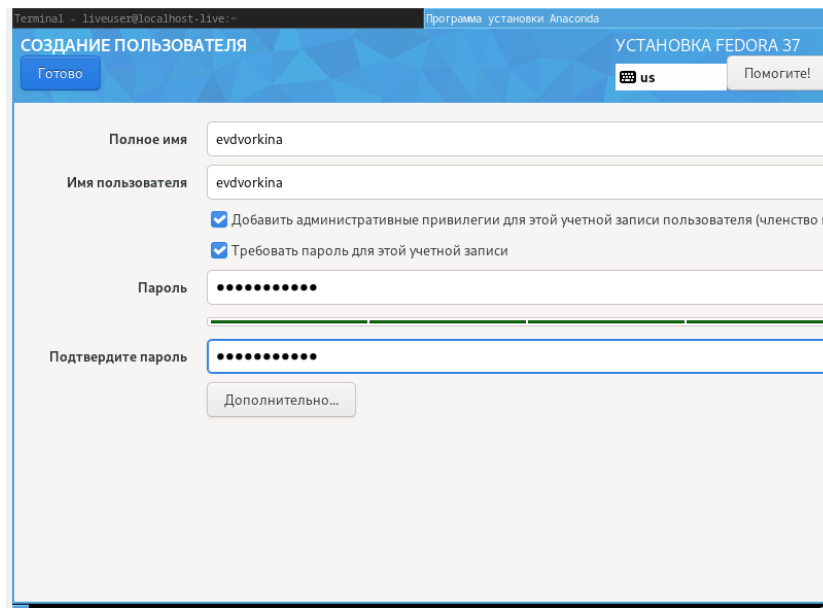


Рис. 3.19: Создание пользователя

Далее операционная система устанавливается. После установки нажимаю “завершить установку” (рис. 3.20).

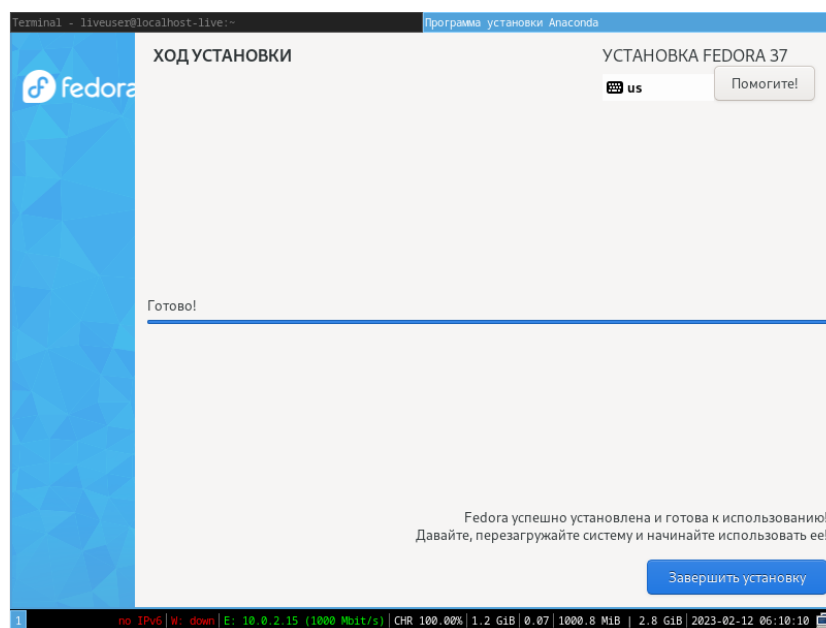


Рис. 3.20: Завершение установки операционной системы

Диск не отключался автоматически, поэтому отключаю носитель информации с образом (рис. 3.21).

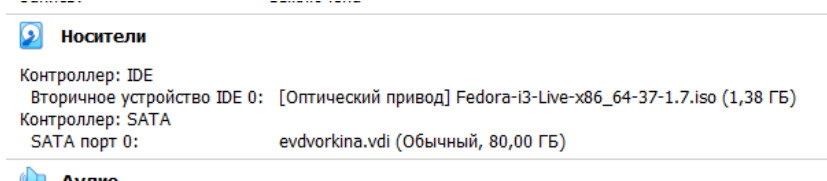


Рис. 3.21: Просмотр оптического диска

Носитель информации с образом отключен (рис. 3.22).

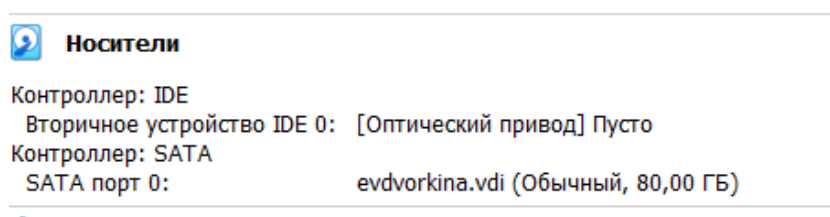


Рис. 3.22: Отключение оптического диска

3.3 Работа с операционной системой после установки

Запускаю виртуальную машину. Вхожу в ОС под заданной мной при установке учетной записью (рис. 3.23).

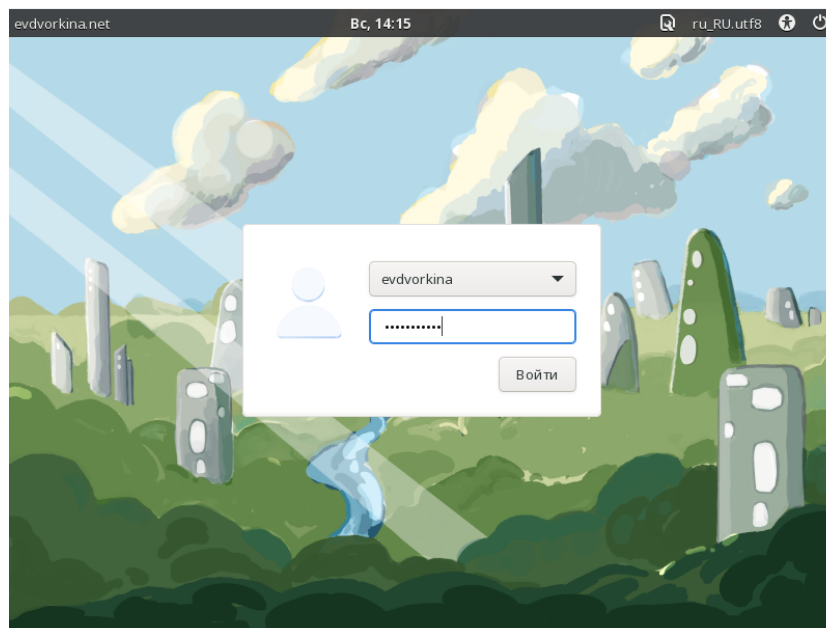


Рис. 3.23: Вход в ОС

Нажимаю Win+Enter для запуска терминала и переключаюсь на роль супер-пользователя(рис. 3.24).

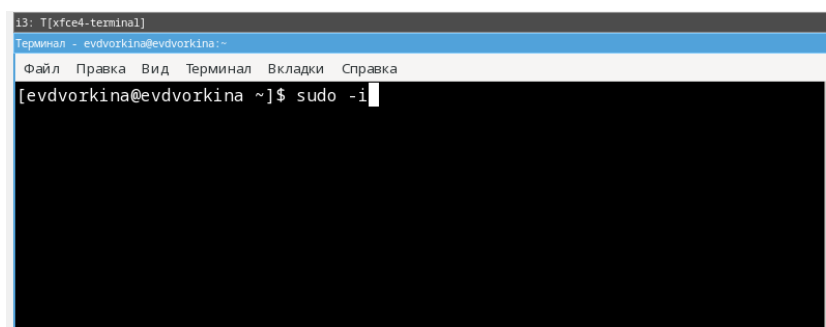


Рис. 3.24: Запуск терминала

Обновляю все пакеты (рис. 3.25).

```
[sudo] пароль для evdvorkina:
[root@evdvorkina ~]# dnf -y update
```

Рис. 3.25: Обновления

Устанавливаю программы для удобства работы в консоли: `tmux` для открытия нескольких “вкладок” в одном терминале, `mc` в качестве файлового менеджера в терминале (рис. 3.26).

```
[root@evdvorkina ~]# dnf install tmux mc
Последняя проверка окончания срока действия метаданных: 0:12:51 назад, Вс 12
фев 2023 14:19:33.
Пакет tmux-3.3a-1.fc37.x86_64 уже установлен.
Зависимости разрешены.
=====
Пакет      Архитектура  Версия      Репозиторий  Размер
=====
Установка:
mc          x86_64       1:4.8.28-3.fc37  fedora       1.9 M
Установка зависимостей:
gpm-libs   x86_64       1.20.7-41.fc37  fedora       20 k
```

Рис. 3.26: Установка `tmux` и `mc`

Устанавливаю программы для автоматического обновления (рис. 3.27).

```
[root@evdvorkina ~]# dnf install dnf-automatic
Последняя проверка окончания срока действия метаданных: 0:14:18 назад, Вс 12
фев 2023 14:19:33.
Зависимости разрешены.
=====
Пакет      Архитектура  Версия      Репозиторий  Размер
=====
Установка:
dnf-automatic noarch      4.14.0-1.fc37  fedora       47 k
```

Рис. 3.27: Установка программного обеспечения для автоматического обновления

Запускаю таймер (рис. 3.28).

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

Рис. 3.28: Запуск таймера

Перемещаюсь в директорию /etc/selinux, открываю md, ищу нужный файл (рис. 3.29).

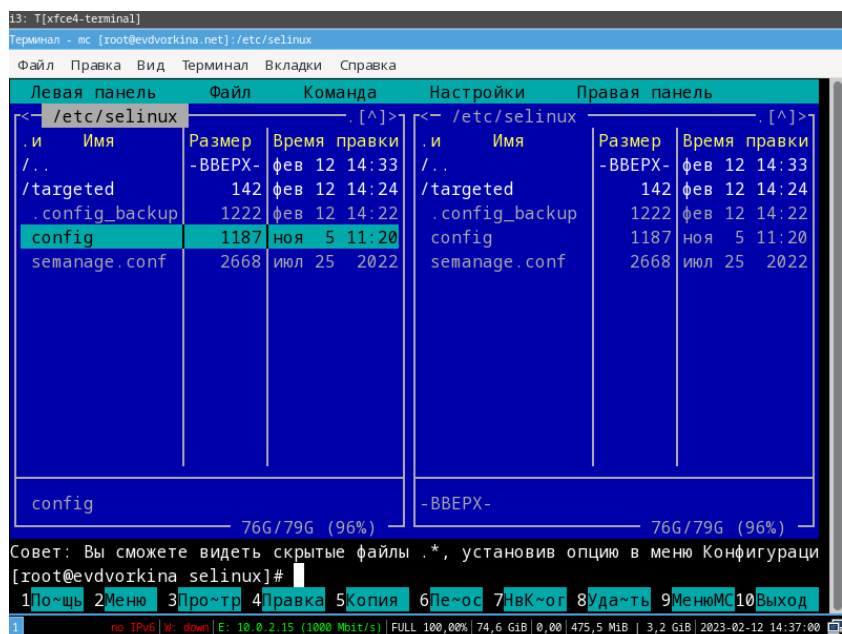
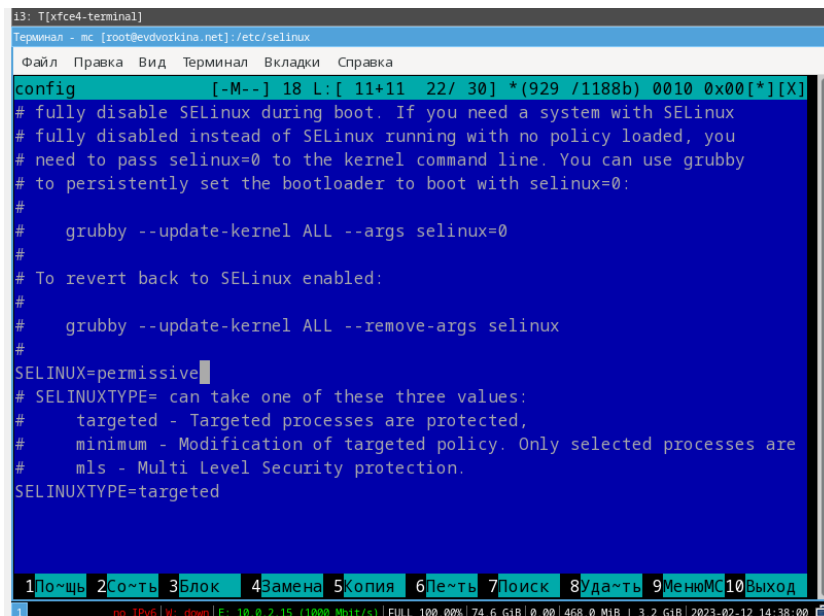


Рис. 3.29: Поиск файла

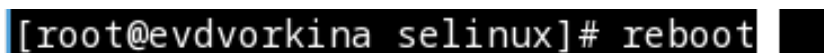
Изменяю открытый файл: SELINUX=enforcing меняю на значение SELINUX=permissive (рис. 3.30).



```
13: T[xfce4-terminal]
Терминал - mc [root@evdvorkina.net]:/etc/selinux
Файл Правка Вид Терминал Вкладки Справка
config [-M--] 18 L:[ 11+11 22/ 30] *(929 /1188b) 0010 0x00[*][X]
# 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 processes are
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
1 По~щъ 2 Со~ть 3 Блок 4 Замена 5 Копия 6 Ле~ть 7 Поиск 8 Уда~ть 9 МенюМС 10 Выход
1 no IPv6 | W: down | E: 10.0.2.15 (1000 Mbit/s) | FULL 100,00% | 74,6 GiB | 0,00 | 468,0 MiB | 3,2 GiB | 2023-02-12 14:38:00
```

Рис. 3.30: Изменение файла

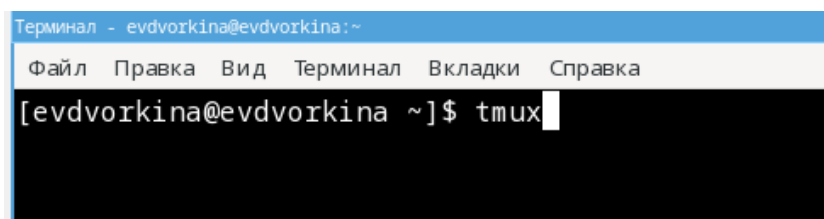
Перезагружаю виртуальную машину (рис. 3.31).



```
[root@evdvorkina selinux]# reboot
```

Рис. 3.31: Перезагрузка виртуальной машины

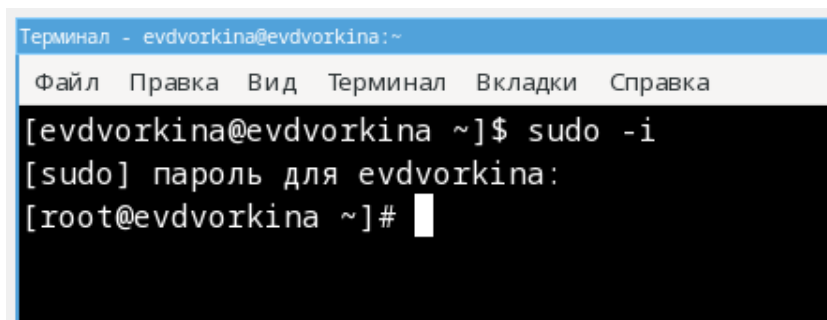
Снова вхожу в ОС, снова запускаю терминал, запускаю терминальный мультиплексор (рис. 3.32).



```
Терминал - evdvorkina@evdvorkina:~
Файл Правка Вид Терминал Вкладки Справка
[evdvorkina@evdvorkina ~]$ tmux
```

Рис. 3.32: Запуск терминального мультиплексора

Переключаюсь на роль супер-пользователя (рис. 3.33).



```
Терминал - evdvorkina@evdvorkina:~
Файл  Правка  Вид  Терминал  Вкладки  Справка
[evdvorkina@evdvorkina ~]$ sudo -i
[sudo] пароль для evdvorkina:
[root@evdvorkina ~]#
```

Рис. 3.33: Переключение на роль супер-пользователя

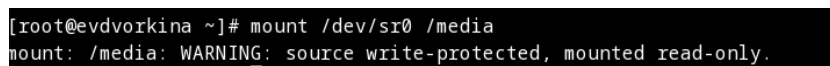
Устанавливаю пакет dkms (рис. 3.34).



```
Выполнено.
[root@evdvorkina ~]# dnf install dkms
```

Рис. 3.34: Установка пакета dkms

В меню виртуальной машины подключаю образ диска гостевой ОС и примонтирую диск с помощью утилиты mount (рис. 3.35).



```
[root@evdvorkina ~]# mount /dev/sr0 /media
mount: /media: WARNING: source write-protected, mounted read-only.
```

Рис. 3.35: Примонтирование диска

Устанавливаю драйвера (рис. 3.36).



```
[root@evdvorkina ~]# /media/VBoxLinuxAdditions.run
Verifying archive integrity... All good.
Uncompressing VirtualBox 6.1.38 Guest Additions for Linux....
VirtualBox Guest Additions installer
Copying additional installer modules ...
Installing additional modules ...
```

Рис. 3.36: Установка драйвера

Перезагружаю виртуальную машину (рис. 3.37).

```
the system is restarted
[root@evdvorkina ~]# reboot
```

Рис. 3.37: Перезагрузка виртуальной машины

Перехожу в директорию /etc/X11/xorg.conf.d, открываю tc для удобства, открываю файл 00-keyboard.conf (рис. 3.38).

```
[evdvorkina@evdvorkina ~]$ sudo -i
[sudo] пароль для evdvorkina:
[root@evdvorkina ~]# cd /etc/X11/xorg.conf.d/
[root@evdvorkina xorg.conf.d]# mc
```

Рис. 3.38: Поиск файла, вход в tc

Редактирую конфигурационный файл (рис. 3.39).

```
терминал - evdvorkina@evdvorkina ~
Файл  Правка  Вид  Терминал  Вкладки  Справка
00-keyboard.conf  [-M--] 32 L: [ 1+ 7  8/ 11] *(335 / 442b) 0110 0x06E
# Written by systemd-localed(8), read by systemd-localed and Xorg. It's
# probably wise not to edit this file manually. Use localectl(1) to
# instruct systemd-localed to update it.
Section "InputClass"
    Identifier "system-keyboard"
    MatchIsKeyboard "on"
    Option "XkbLayout" "us,ru"
    Option "XkbVariant" ",winkeys"
    Option "XkbOptions" "grp:alt_shift_toggle, compose:ralt,terminate:ctrl_alt_bksp"
EndSection
```

Рис. 3.39: Редактирование файла

Перезагружаю виртуальную машину (рис. 3.40).

```
[root@evdvorkina xorg.conf.d]# reboot
```

Рис. 3.40: Перезагрузка виртуальной машины

3.4 Установка программного обеспечения для создания документации

Запускаю терминал. Запускаю терминальный мультиплексор tmux, переключаюсь на роль супер-пользователя (рис. 3.41).

```
[evdvorkina@evdvorkina ~]$ sudo -i
[sudo] пароль для evdvorkina:
[root@evdvorkina ~]# dnf -i install pandoc
```

Рис. 3.41: Переключение на роль супер-пользователя

Устанавливаю pandoc с помощью утилиты dnf и флага -y, который автоматически на все вопросы системы отвечает “yes” (рис. 3.42).

```
[root@evdvorkina ~]# dnf -y install pandoc
Последняя проверка окончания срока действия метаданных: 0:38:54 назад, Вс 12 фев 2023 14:39:15.
Зависимости разрешены.
=====
Пакет                               Архитектура                     Версия
-----
Установка:
pandoc                               x86_64                           2.14.0.3-18.fc
Установка зависимостей:
pandoc-common                        noarch                           2.14.0.3-18.fc
=====
Результат транзакции
```

Рис. 3.42: Установка pandoc

Устанавливаю необходимые расширения для pandoc (рис. 3.43).

```
[root@evdvorkina ~]# pip install pandoc-fignos pandoc-eqnos pandoc-tablenos pandoc-secnos --user
0 | 0: bash "evdvorkina.net"
```

Рис. 3.43: Установка расширения pandoc

Устанавливаю дистрибутив texlive (рис. 3.44).

```
[root@evdvorkina ~]# dnf -y install texlive texlive-\\*
Последняя проверка окончания срока действия метаданных: 0:52:55 назад, Вс 12 фев 2023 14:39:15.
```

Рис. 3.44: Установка texlive

4 Выводы

При выполнении данной лабораторной работы я приобрела практические навыки установки операционной системы на виртуальную машину, а так же сделала настройки минимально необходимых для дальнейшей работы сервисов.

5 Ответы на контрольные вопросы

1. Учетная запись содержит необходимые для идентификации пользователя при подключении к системе данные, а так же информацию для авторизации и учета: системного имени (user name) (оно может содержать только латинские буквы и знак нижнее подчеркивание, еще оно должно быть уникальным), идентификатор пользователя (UID) (уникальный идентификатор пользователя в системе, целое положительное число), идентификатор группы (GID) (группа, к к-рой относится пользователь. Она, как минимум, одна, по умолчанию - одна), полное имя (full name) (Могут быть ФИО), домашний каталог (home directory) (каталог, в к-рый попадает пользователь после входа в систему и в к-ром хранятся его данные), начальная оболочка (login shell) (командная оболочка, к-рая запускается при входе в систему).
2. Для получения справки по команде: `–help`; для перемещения по файловой системе - `cd`; для просмотра содержимого каталога - `ls`; для определения объёма каталога - `du` ; для создания / удаления каталогов - `mkdir/rmdir`; для создания / удаления файлов - `touch/rm`; для задания определённых прав на файл / каталог - `chmod`; для просмотра истории команд - `history`
3. Файловая система - это порядок, определяющий способ организации и хранения и именования данных на различных носителях информации. Примеры: FAT32 представляет собой пространство, разделенное на три части: одна область для служебных структур, форма указателей в виде таблиц и зона для хранения самих файлов. ext3/ext4 - журналируемая файловая система, используемая в основном в ОС с ядром Linux.

4. С помощью команды `df`, введя ее в терминале. Это утилита, которая показывает список всех файловых систем по именам устройств, сообщает их размер и данные о памяти. Также посмотреть подмонтированные файловые системы можно с помощью утилиты `mount`.
5. Чтобы удалить зависший процесс, вначале мы должны узнать, какой у него `id`: используем команду `ps`. Далее в терминале вводим команду `kill < id процесса >`. Или можно использовать утилиту `killall`, что “убьет” все процессы, которые есть в данный момент, для этого не нужно знать `id` процесса.

6 Выполнение дополнительного задания

Ввожу в терминале команду `dmesg`, чтобы проанализировать последовательность загрузки системы (рис. 6.1).

```
tevdvorkina@tevdvorkina:~$ dmesg
Linux version 6.1.10-200.fc37.x86_64 (mockbuild@bkernel01.iad2.fedoraproject.org) (gcc (GCC) 12.2.1 20221121 (Red Hat 12.2.1-4), GNU ld version 2.38-25.fc37) #1 SMP PREEMPT_D
YNAMIC Mon Feb 6 23:56:48 UTC 2023
[ 0.000000] command line: BOOT_IMAGE=(hob gpt2)/vmlinuz-6.1.10-200.fc37.x86_64 root=UUID-929ee264-28d7-4fa2-863f-cd6ae852164 ro rootflags=subvol=root rhgb quiet
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
[ 0.000000] x86/fpu: xstate_offset[2]: 576, xstate_size[2]: 256
[ 0.000000] x86/fpu: Enabled xstate features 0x7, context size is 832 bytes, using 'standard' format.
[ 0.000000] signal: max sigframe size: 3776
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000009ffff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000000009fffc00-0x00000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000000000-0x0000000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000000000000-0x0000000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000000000000-0x0000000000000000ffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x000000000000000000000-0x0000000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000000000000-0x0000000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000000000000000-0x0000000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000000000000000000-0x0000000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000000000000000-0x0000000000000000ffff] usable
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMIOS 2.5 present
[ 0.000000] acpi: Imported GSH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using msrc 00000000 and 00000000
[ 0.000000] kvm-clock: using sched offset of 9974184431 cycles
[ 0.000000] clocksource: kvm-clock: mask 0xffffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881598591483 ns
[ 0.000000] acpi: Detected 1992.000 MHz processor
[ 0.000000] e820: update [mem 0x000000000-0x00000000ffff] usable --- reserved
[ 0.000000] e820: remove [mem 0x000000000-0x00000000ffff] usable
[ 0.000000] last_pfn = 0x120000 max_pfn=0x120000 max_arch_pfn = 0x00000000
[ 0.000000] Disabled
[ 0.000000] CPU MTRRs all blank - virtualized system
[ 0.000000] x86/PAT: Configuration [0-7]: WB WT UC- UC- WB WT UC- UC-
[ 0.000000] last_pfn = 0x120000 max_arch_pfn = 0x00000000
[ 0.000000] Found SMP MP-table at [mem 0x00000000-0x00000000ffff]
```

Рис. 6.1: Анализ последовательности загрузки системы

С помощью поиска, осуществляемого командой `dmesg | grep -i`, ищу версию ядра Linux: 6.1.10-200.fc37.x86_64 (рис. 6.2).

```
tevdvorkina@tevdvorkina:~$ dmesg | grep -i "linux version"
[ 0.000000] Linux version 6.1.10-200.fc37.x86_64 (mockbuild@bkernel01.iad2.fedoraproject.org) (gcc (GCC) 12.2.1
20221121 (Red Hat 12.2.1-4), GNU ld version 2.38-25.fc37) #1 SMP PREEMPT_DYNAMIC Mon Feb 6 23:56:48 UTC 2023
```

Рис. 6.2: Поиск версии ядра

К сожалению, если вводить “Detected Mhz processor” там, где нужно указывать, что я ищу, то мне ничего не выведется. Это происходит потому, что запрос не предусматривает дополнительные символы внутри него (я проверяла, будет ли работать он с маской - не будет). В таком случае я оставила одно из ключевых

слов (могла оставить два: "Mhz processor") и получила результат: 1992 Mhz (рис. 6.3).

```
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "Detected Mhz processor"
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "processor"
[ 0.000012] tsc: Detected 1992.000 MHz processor
[ 0.236538] smpboot: Total of 1 processors activated (3984.00 BogoMIPS)
[ 0.284426] ACPI: Added _OSI(Processor Device)
[ 0.284428] ACPI: Added _OSI(Processor Aggregator Device)
```

Рис. 6.3: Поиск частоты процессора

Аналогично ищу модель процессора (рис. 6.4).

```
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "CPU0"
[ 0.236431] smpboot: CPU0: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz (family: 0x6, model: 0x8e, stepping: 0xa)
```

Рис. 6.4: Поиск модели процессора

Объем доступной оперативной памяти ищу аналогично поиску частоты процессора, т. к. возникла та же проблема, что и там (рис. 6.5).

```
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "Memory: "
[ 0.027952] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.027955] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009ffff]
[ 0.027956] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000affff]
[ 0.027958] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[ 0.027960] PM: hibernation: Registered nosave memory: [mem 0xdfff0000-0xdfffffff]
[ 0.027961] PM: hibernation: Registered nosave memory: [mem 0xe0000000-0xfefbffff]
[ 0.027962] PM: hibernation: Registered nosave memory: [mem 0xfec00000-0xfec0ffff]
[ 0.027964] PM: hibernation: Registered nosave memory: [mem 0xfec01000-0xfedfffff]
[ 0.027965] PM: hibernation: Registered nosave memory: [mem 0xfec00000-0xfec0ffff]
[ 0.027966] PM: hibernation: Registered nosave memory: [mem 0xfec01000-0xfefbffff]
[ 0.027967] PM: hibernation: Registered nosave memory: [mem 0xfec00000-0xfec0ffff]
[ 0.062354] Memory: 3971024K/4193848K available (16393K kernel code, 3265K rdata, 12468K rodata, 3032K init, 4596K bss, 222564K reserved, 0K cma-reserved)
```

Рис. 6.5: Поиск объема доступной оперативной памяти

Нахожу тип обнаруженного гипервизора (рис. 6.6).

```
1.252143] Freeing unused kernel image (10data/data gap) memory
evdvorkina@evdvorkina ~]$ dmesg | grep -i "Hypervisor detected"
0.000000] Hypervisor detected: KVM
```

Рис. 6.6: Поиск типа обнаруженного гипервизора

Тип файловой системы корневого раздела можно посмотреть с помощью утилиты fdisk (рис. 6.7).


```
[evdvorkina@evdvorkina ~]$ sudo fdisk -l
[sudo] пароль для evdvorkina:
Диск /dev/sda: 80 GiB, 85899345920 байт, 167772160 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: gpt
Идентификатор диска: DAB00C34-381C-4EC1-9930-B40C01551F8D

Устр-во      начало      Конец      Секторы  Размер  Тип
/dev/sda1      2048        4095        2048      1M  BIOS boot
/dev/sda2      4096       2101247    2097152    1G  Файловая система Linux
/dev/sda3     2101248    167770111  165668864  79G  Файловая система Linux

Диск /dev/zram0: 3,83 GiB, 4108320768 байт, 1003008 секторов
Единицы: секторов по 1 * 4096 = 4096 байт
Размер сектора (логический/физический): 4096 байт / 4096 байт
Размер I/O (минимальный/оптимальный): 4096 байт / 4096 байт
```

Рис. 6.7: Поиск типа файловой системы корневого раздела

Последовательность монтирования файловых систем можно посмотреть, введя в поиск по результату dmesg слово mount (рис. 6.8).

```
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "mount"
[ 0.100238] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 0.100265] Mountpoint-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 0.130675] systemd[1]: Set up automount proc-sys-binfmt_misc automount - Arbitrary Executable File Formats File System Automount Point.
[ 0.150730] systemd[1]: Mounting dev-hugepages.mount - Huge Pages File System.
[ 0.152841] systemd[1]: Mounting dev-mqueue.mount - POSIX Message Queue File System.
[ 0.157245] systemd[1]: Mounting sys-kernel-debug.mount - Kernel Debug File System.
[ 0.157723] systemd[1]: Mounting sys-kernel-tracing.mount - Kernel Trace File System.
[ 0.282909] systemd[1]: Starting systemd-remount.service - Remount Root and Kernel File Systems...
[ 0.320096] systemd[1]: Mounted dev-hugepages.mount - Huge Pages File System.
[ 0.322886] systemd[1]: Mounted dev-mqueue.mount - POSIX Message Queue File System.
[ 0.326011] systemd[1]: Mounted sys-kernel-debug.mount - Kernel Debug File System.
[ 0.331819] systemd[1]: Mounted sys-kernel-tracing.mount - Kernel Trace File System.
[ 0.354751] systemd[1]: Finished systemd-remount.service - Remount Root and Kernel File Systems.
[ 0.359264] systemd[1]: Mounting sys-fs-fuse-connections.mount - FUSE Control File System.
[ 0.372610] systemd[1]: Mounting sys-kernel-config.mount - Kernel Configuration File System.
[ 0.373363] systemd[1]: ostree-remount.service - OSTree Remount OS/Bind Mounts was skipped because of a failed condition check (ConditionKernelCommandLine=ostree).
[ 7.901632] EXT4-fs (sda2): mounted filesystem with ordered data mode. Quota mode: none.
[evdvorkina@evdvorkina ~]$
```

Рис. 6.8: Последовательность монтирования файловых систем

Список литературы

1. Dash P. Getting started with oracle vm virtualbox. Packt Publishing Ltd, 2013. 86 p.
2. Colvin H. Virtualbox: An ultimate guide book on virtualization with virtualbox. CreateSpace Independent Publishing Platform, 2015. 70 p.
3. van Vugt S. Red hat rhcsa/rhce 7 cert guide : Red hat enterprise linux 7 (ex200 and ex300). Pearson IT Certification, 2016. 1008 p.
4. Робачевский А., Немнюгин С., Стесик О. Операционная система unix. 2-е изд. Санкт-Петербург: БХВ-Петербург, 2010. 656 p.
5. Немец Э. et al. Unix и Linux: руководство системного администратора. 4-е изд. Вильямс, 2014. 1312 p.
6. Колисниченко Д.Н. Самоучитель системного администратора Linux. СПб.: БХВ-Петербург, 2011. 544 p.
7. Robbins A. Bash pocket reference. O'Reilly Media, 2016. 156 p.