

Отчет по лабораторной работе №1

Операционные системы

Бельчуг Александр Константинович

Содержание

1	Цель работы	6
2	Задание	7
3	Выполнение лабораторной работы	8
3.1	Создание виртуальной машины	8
3.2	Установка операционной системы	12
3.3	Работа с операционной системой после установки	16
3.4	Установка программного обеспечения для создания документации	21
4	Выводы	22
5	Ответы на контрольные вопросы	23
6	Выполнение дополнительного задания	25

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	11
3.8	Выбор образа оптического диска	12
3.9	Выбранный образ оптического диска	12
3.10	Окно загрузчика	13
3.11	Интерфейс начальной конфигурации	13
3.12	Запуск терминала	14
3.13	Выбор языка интерфейса	14
3.14	Выбор раскладки клавиатуры	15
3.15	Выбор места установки	15
3.16	Создание аккаунта администратора	16
3.17	Запуск терминала	16
3.18	Обновления	17
3.19	Установка tmux и mc	17
3.20	Установка программного обеспечения для автоматического обновления	17
3.21	Запуск таймера	18
3.22	Поиск файла	18
3.23	Изменение файла	18
3.24	Перезагрузка виртуальной машины	19
3.25	Запуск терминального мультиплексора	19
3.26	Переключение на роль супер-пользователя	19
3.27	Создание кнфигурационный файл	19
3.28	Отредактируйте конфигурационный файл	20
3.29	Поиск файла, вход в mc	20
3.30	Редактирование файла	20
3.31	Перезагрузка виртуальной машины	20
3.32	Переключение на роль супер-пользователя	21
3.33	Установка pandoc	21
3.34	Установка расширения pandoc-crossref	21
3.35	Установка texlive	21

6.1	Анализ последовательности загрузки системы	25
6.2	Поиск версии ядра	25
6.3	Поиск частоты процессора	25
6.4	Поиск модели процессора	26
6.5	Поиск объема доступной оперативной памяти	26
6.6	Поиск типа обнаруженного гипервизора	26
6.7	Поиск типа файловой системы корневого раздела	27
6.8	Последовательность монтирования файловых систем	27

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. fig. 3.1).

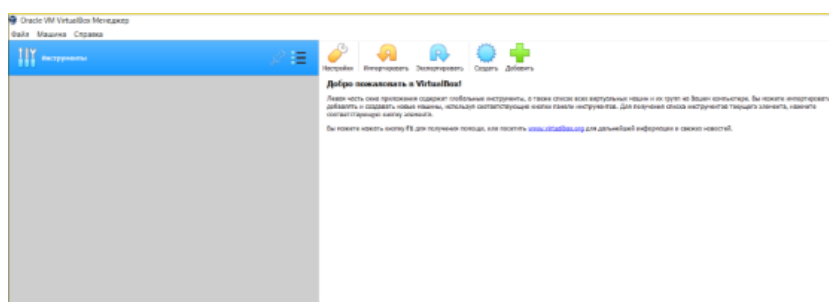


Рис. 3.1: Окно Virtualbox

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

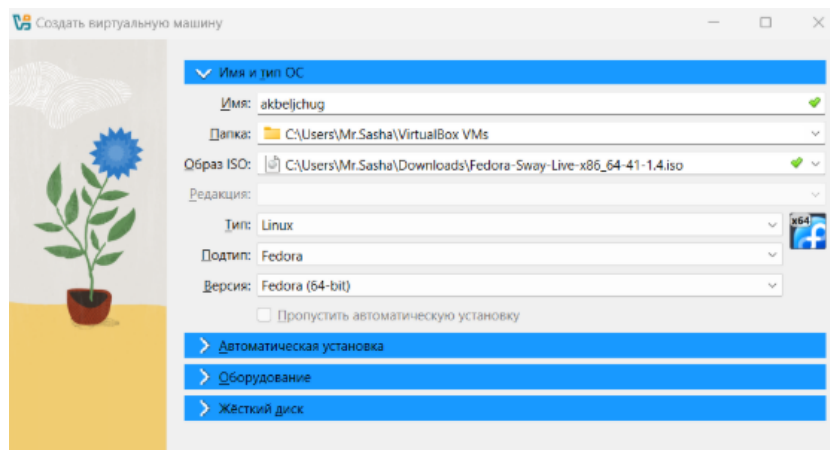


Рис. 3.2: Создание виртуальной машины

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

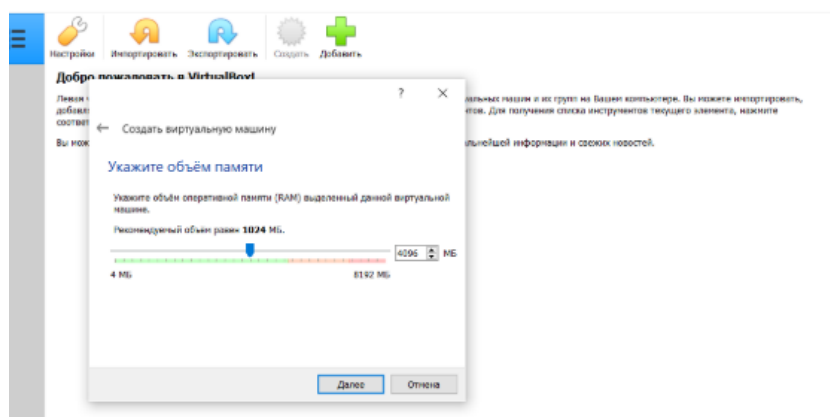


Рис. 3.3: Указание объема памяти

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

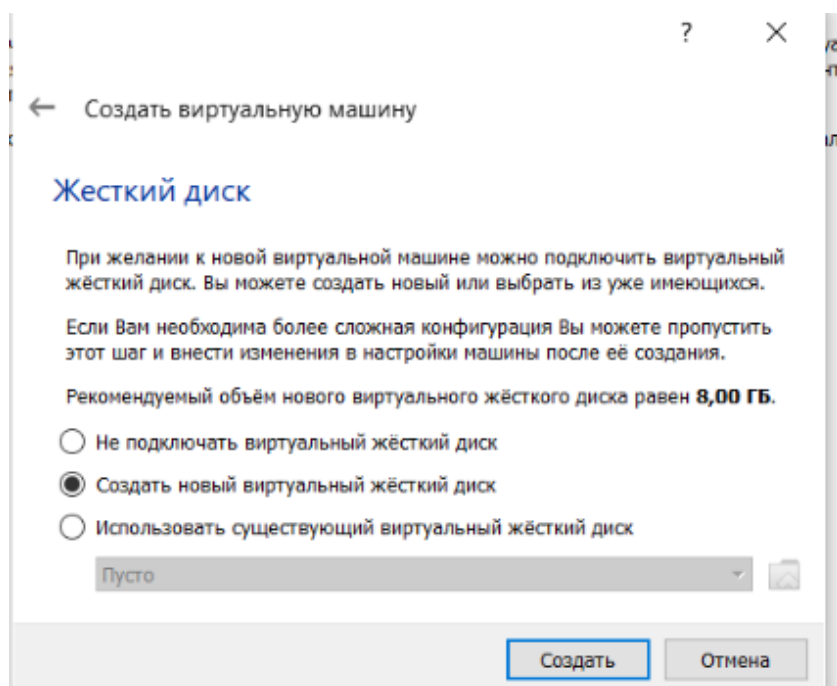


Рис. 3.4: Жесткий диск

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

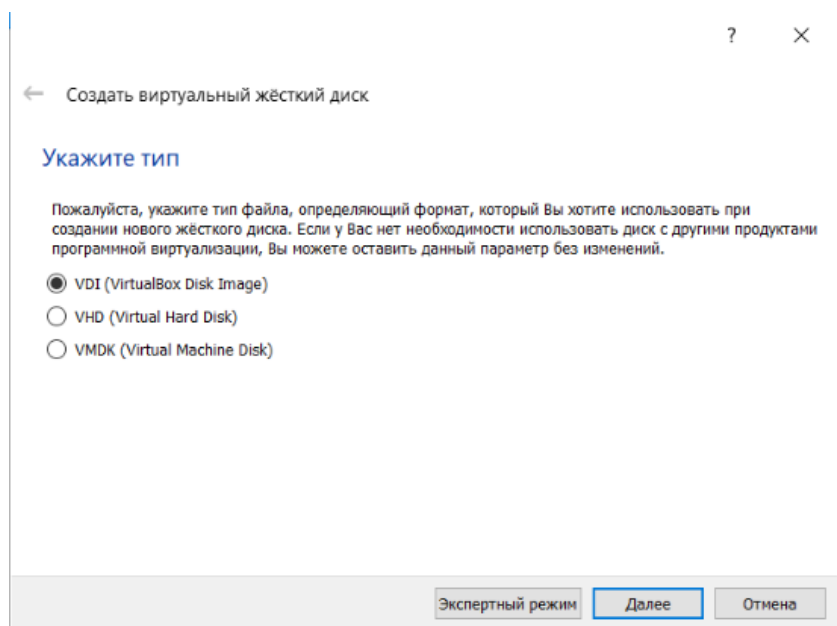


Рис. 3.5: Тип жесткого диска

Задаю размер диска - 80 ГБ, оставляю расположение жесткого диска по умол-

чанию, т. к. работаю на собственной технике и значение по умолчанию меня устраивает (рис. fig. 3.6).

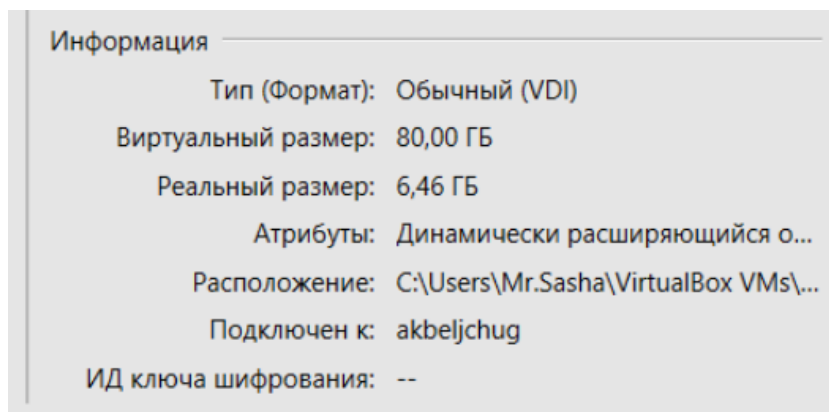


Рис. 3.6: Размер жесткого диска

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

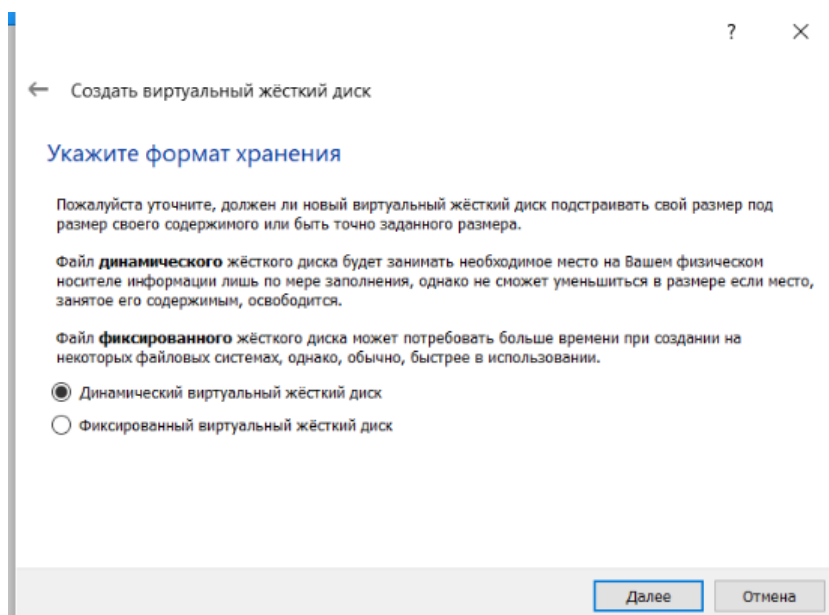


Рис. 3.7: Формат хранения жесткого диска

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

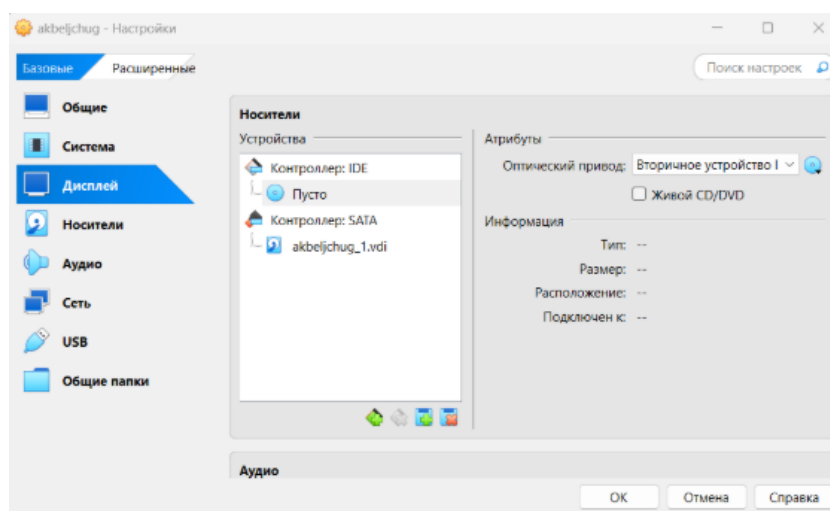


Рис. 3.8: Выбор образа оптического диска

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

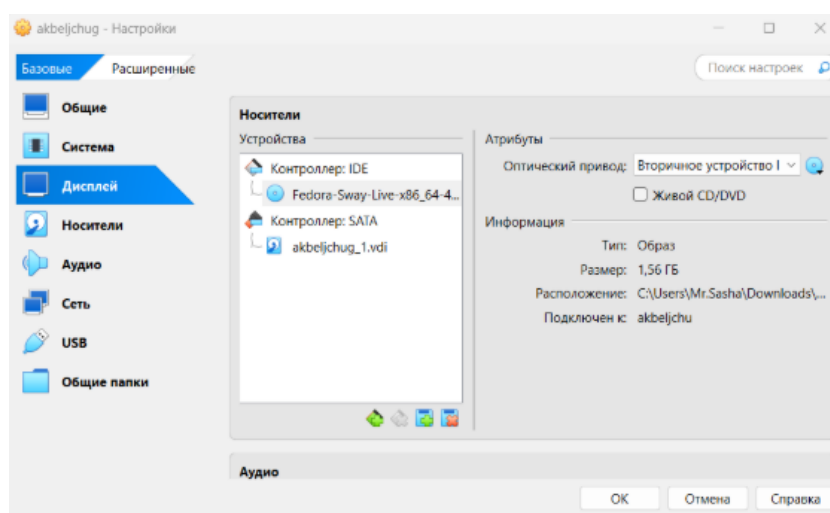


Рис. 3.9: Выбранный образ оптического диска

3.2 Установка операционной системы

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

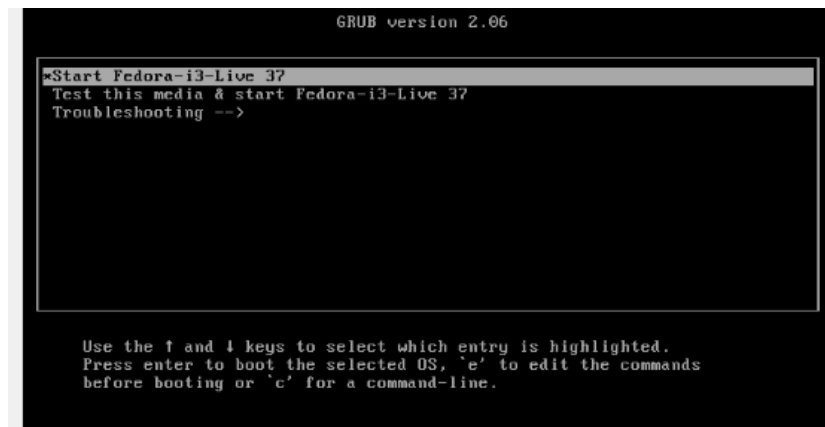


Рис. 3.10: Окно загрузчика

Вижу интерфейс начальной конфигурации. Нажимаю Enter для создания конфигурации 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. fig. 3.12).

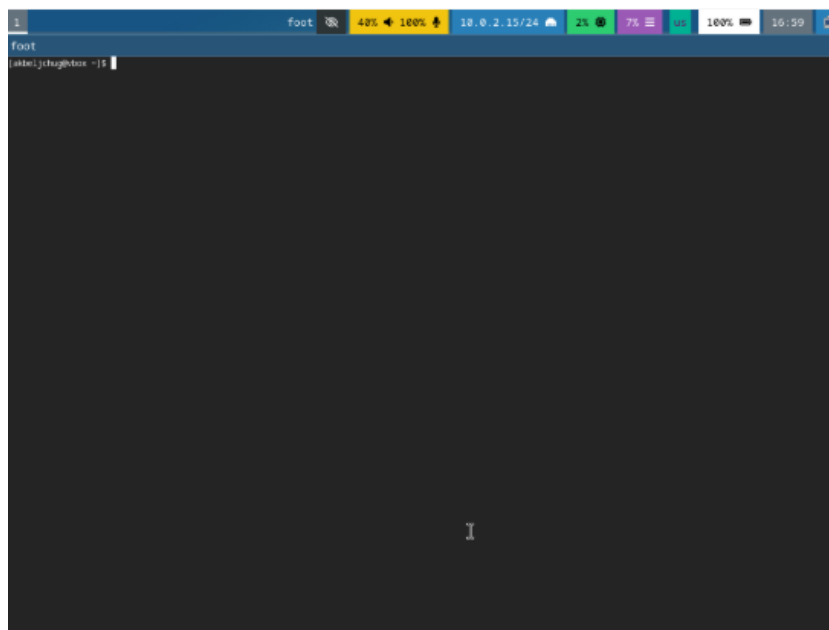


Рис. 3.12: Запуск терминала

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

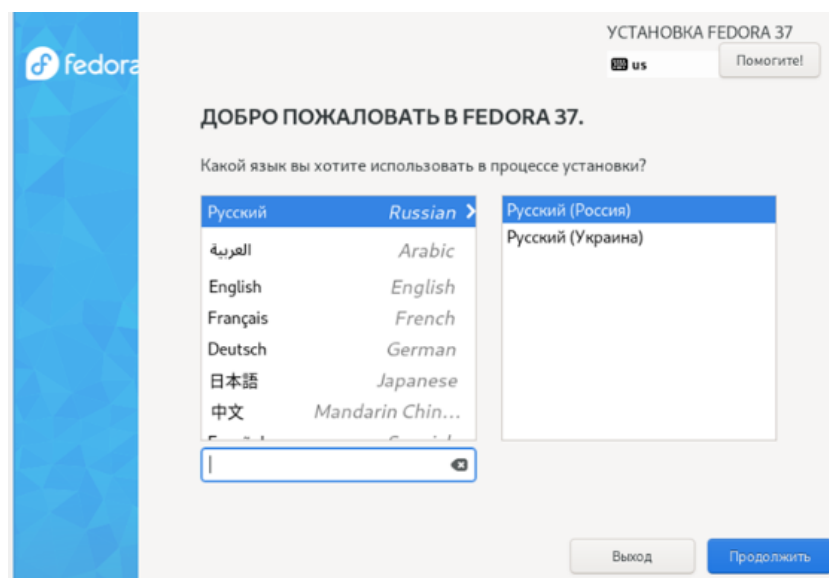


Рис. 3.13: Выбор языка интерфейса

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

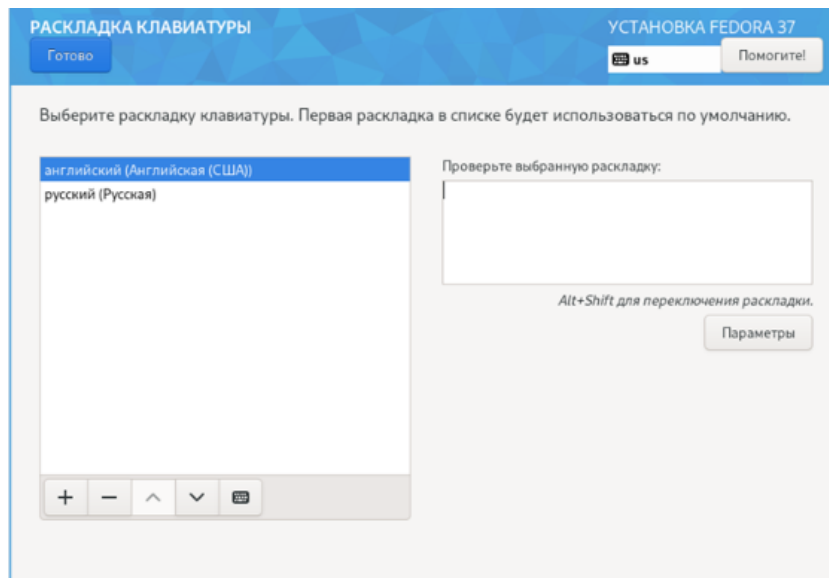


Рис. 3.14: Выбор раскладки клавиатуры

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

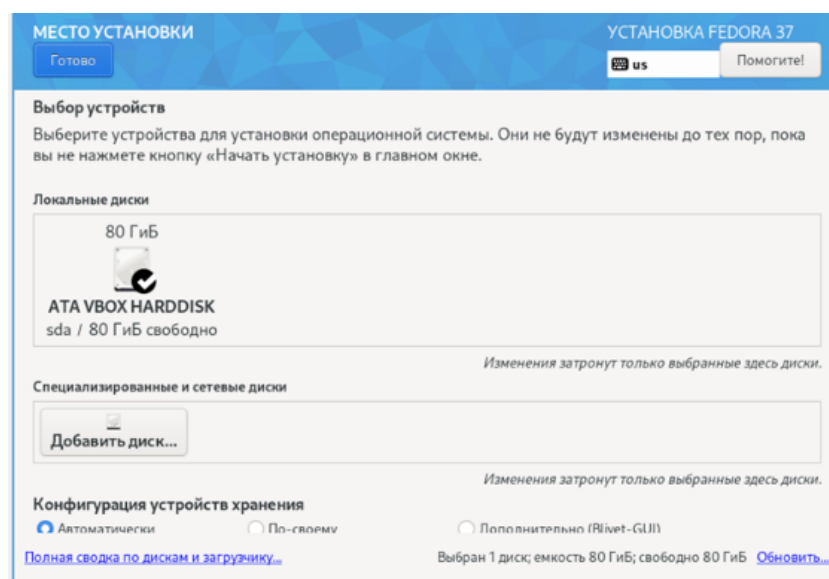


Рис. 3.15: Выбор места установки

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

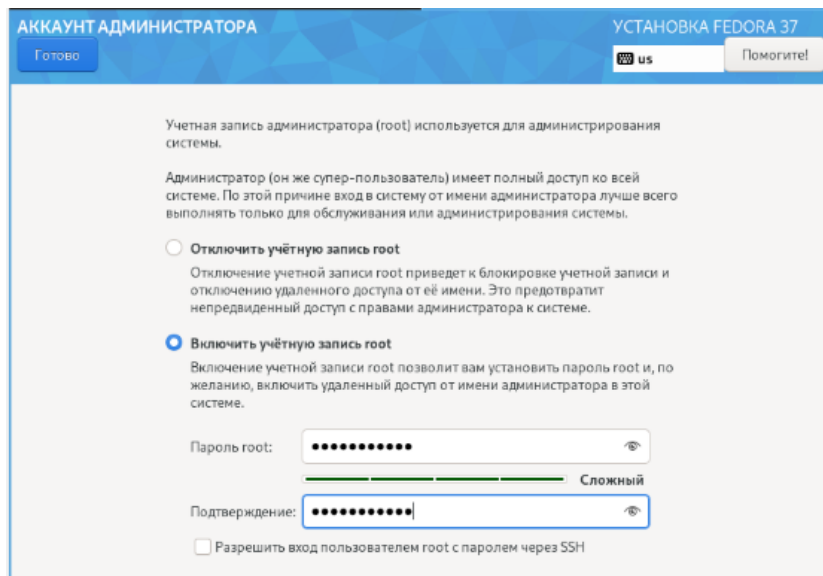


Рис. 3.16: Создание аккаунта администратора

3.3 Работа с операционной системой после установки

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

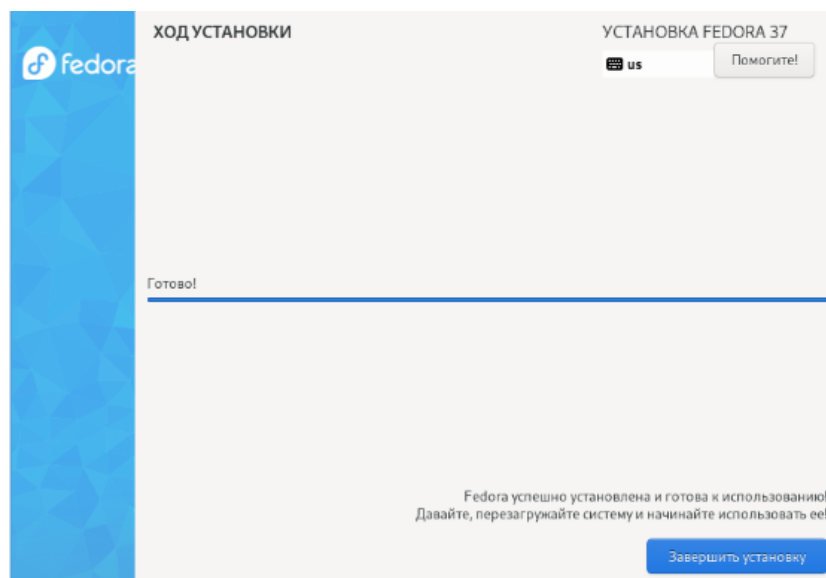


Рис. 3.17: Запуск терминала

Обновляю все пакеты (рис. fig. 3.25).

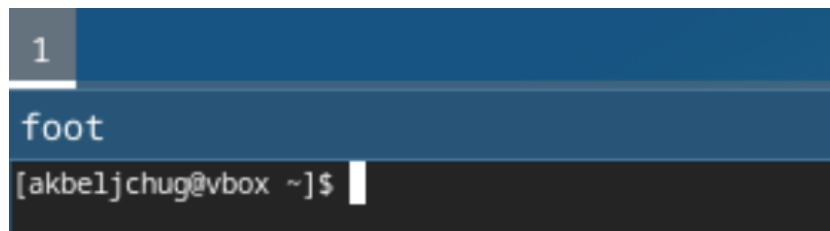


Рис. 3.18: Обновления

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

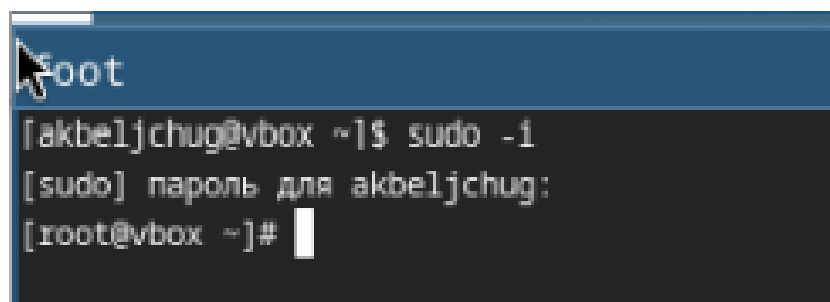


Рис. 3.19: Установка tmux и mc

Устанавливаю программы для автоматического обновления (рис. fig. 3.27).

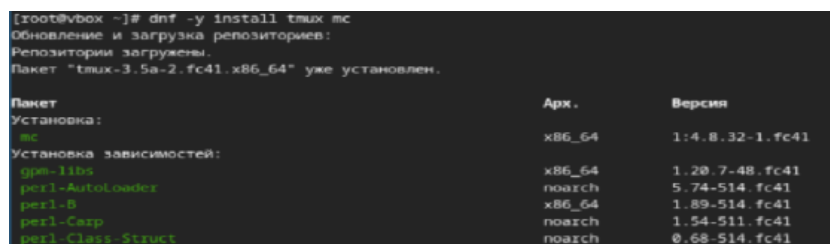


Рис. 3.20: Установка программного обеспечения для автоматического обновления

Запускаю таймер (рис. fig. 3.28).

```

[root@vbox ~]# dnf install dnf-automatic
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет:
Установка:
dnf5-plugin-automatic
Аrch. Версия
x86_64 5.2.10.0-2.fc41
Сводка транзакции:
Установка: 1 пакета
Общий размер входящих пакетов составляет 141 KiB. Необходимо загрузить 141 KiB.
После этой операции будут использоваться дополнительные 179 KiB (установка 179 KiB, удаление 0 B).
Is this ok [y/N]: y
[1/1] dnf5-plugin-automatic-0:5.2.10.0-2.fc41.x86_64
[1/1] Total
Выполнение транзакции
[1/3] Проверить файлы пакета
[2/3] Подготовить транзакции
[3/3] Установка dnf5-plugin-automatic-0:5.2.10.0-2.fc41.x86_64
Успешно!
[root@vbox ~]# systemctl enable --now dnf-automatic.timer

```

Рис. 3.21: Запуск таймера

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

```

[root@vbox ~]# systemctl enable --now dnf-automatic.timer
Created symlink '/etc/systemd/system/timers.target.wants/dnf5-automatic.timer' - '/usr/lib/systemd/system/dnf5-automatic.timer'.

```

Рис. 3.22: Поиск файла

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

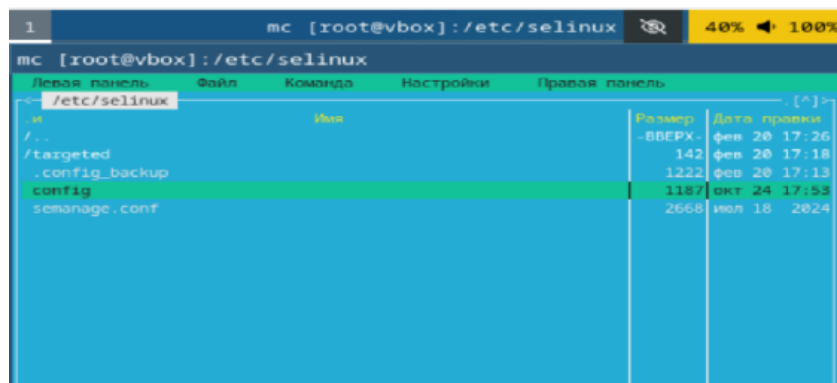


Рис. 3.23: Изменение файла

Перезагружаю виртуальную машину (рис. fig. 3.31).

```
mc [root@vbox] : /etc/selinux
config [----] 18 L[ 1421 22/ 30] *(929 /1500) 0010 0x00A

# 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-selinux/#getting-started-with-selinux-selinux-states-and-modes
#
# 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 processes are protected.
# mls - Multi level Security protection.
SELINUXTYPE=targeted
```

Рис. 3.24: Перезагрузка виртуальной машины

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

```
[root@vbox ~]# reboot
```

Рис. 3.25: Запуск терминального мультиплексора

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

```
[akbeljchug@vbox ~]$ tmux
```

Рис. 3.26: Переключение на роль супер-пользователя

Создание крнфигурационнй файл (рис. fig. 3.34).

```
root
[akbeljchug@vbox ~]$ sudo -i
[sudo] пароль для akbeljchug:
[root@vbox ~]#
```

Рис. 3.27: Создание крнфигурационнй файл

Отредактируйте конфигурационный файл (рис. fig. ??).

```
akbeljchug@vbox: ~$ mkdir -p ~/.config/sway
akbeljchug@vbox: ~$ mkdir -p ~/.config/sway/config.d
akbeljchug@vbox: ~$ mkdir -p ~/.config/sway/config.d/95-system-keyboard-config.conf
akbeljchug@vbox: ~$ touch ~/.config/sway/config.d/95-system-keyboard-config.conf
akbeljchug@vbox: ~$ exec_always /usr/libexec/sway-systemd/locale1-xkb-config --oneshot
```

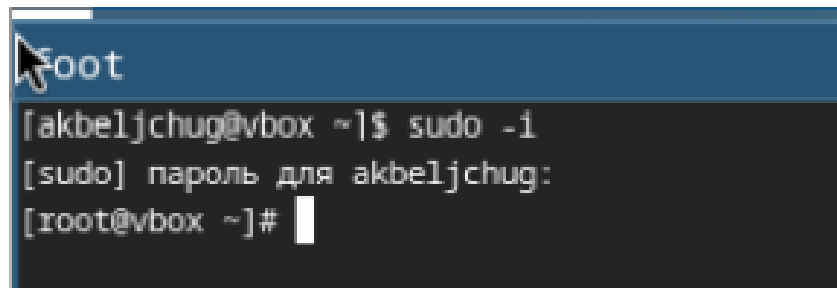
Рис. 3.28: Отредактируйте конфигурационный файл

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

```
[root@vbox ~]# exec_always /usr/libexec/sway-systemd/locale1-xkb-config --oneshot
```

Рис. 3.29: Поиск файла, вход в mc

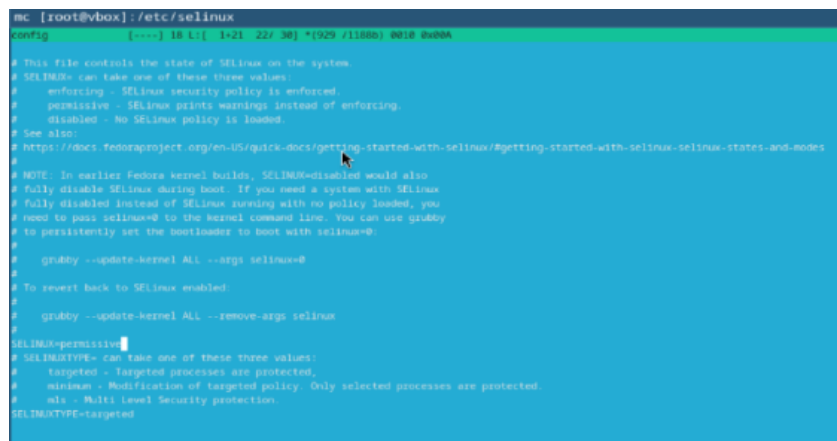
Редактирую конфигурационный файл (рис. fig. 6.2).



```
akbeljchug@vbox ~]$ sudo -i
[sudo] пароль для akbeljchug:
[root@vbox ~]#
```

Рис. 3.30: Редактирование файла

Перезагружаю виртуальную машину (рис. fig. 6.3).



```
mc [root@vbox] /etc/selinux
config [-----] 18 L 1 1+21 22/ 30 *(909 /11580) 9010 900A

# This file controls the state of SELinux on the system.
# SELinux can take one of 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-selinux/#getting-started-with-selinux-states-and-modes
# 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 boot loader 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 protected.
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Рис. 3.31: Перезагрузка виртуальной машины

3.4 Установка программного обеспечения для создания документации

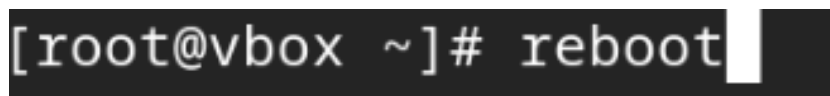
Запускаю терминал. Запускаю терминальный мультиплексор tmux, переключаюсь на роль супер-пользователя (рис. fig. 6.4).



```
section "InputClass"
    Identifier "system-keyboard"
    MatchIsKeyboard "on"
    Option "XkbLayout" "us,ru"
    Option "XkbVariant" ",winkeys"
    Option "XkbOptions" "grp:ctrl_toggle,compose:ralt,terminate:ctrl_alt_bksp"
EndSection
```

Рис. 3.32: Переключение на роль супер-пользователя

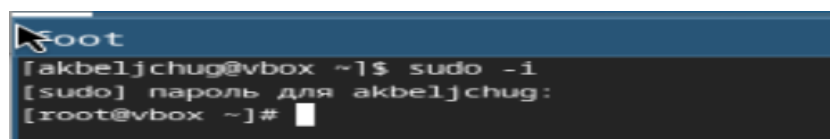
Устанавливаю pandoc с помощью утилиты dnf и флага -y, который автоматически на все вопросы системы отвечает “yes” (рис. fig. 6.5).



```
[root@vbox ~]# reboot
```

Рис. 3.33: Установка pandoc

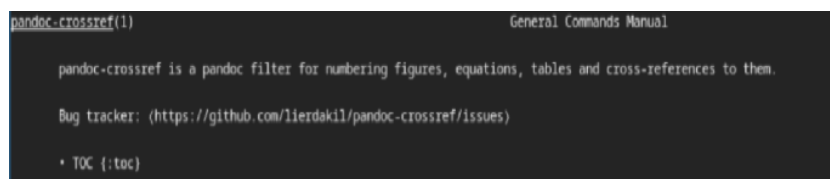
Устанавливаю pandoc-crossref (рис. fig. 6.6).



```
root
[akbeljchug@vbox ~]$ sudo -i
[sudo] пароль для akbeljchug:
[root@vbox ~]#
```

Рис. 3.34: Установка расширения pandoc-crossref

Устанавливаю дистрибутив texlive (рис. fig. 6.7).



```
pandoc-crossref(1)                                General Commands Manual

pandoc-crossref is a pandoc filter for numbering figures, equations, tables and cross-references to them.

Bug tracker: (https://github.com/lierdakil/pandoc-crossref/issues)

• TOC (:toc)
```

Рис. 3.35: 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. fig. 6.8).

```
root@vbox ~]# dnf -y install texlive-scheme-full
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет "texlive-scheme-full-11:svn540/4-/3.fc41.noarch" уже установлен.
```

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. fig. ??).

```
root@vbox ~]# dmesg | grep -i "Linux version"
[ 0.000000] Linux version 6.12.15-200.fc41.x86_64 (mockbuild@fc4102bca6b4b5181a31926b883
base) (gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7), GNU ld version 2.43.1-5.fc41) #1 SMP PR
EEMPT_DYNAMIC Tue Feb 18 15:24:05 UTC 2025
```

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. fig. ??).

```
root@vbox ~]# dmesg | grep -i "processor"
[ 0.000009] tsc: Detected 1689.602 MHz processor
[ 0.652299] smpboot: Total of 4 processors activated (13516.81 BogoMIPS)
[ 0.683360] ACPI: Added _OSI(Processor Device)
[ 0.683362] ACPI: Added _OSI(Processor Aggregator Device)
root@vbox ~]#
```

Рис. 6.3: Поиск частоты процессора

Аналогично ищу модель процессора (рис. fig. ??).

```
[root@vbox ~]# dmesg | grep -i "CPU0"
[ 0.644253] smpboot: CPU0: Intel(R) N95 (family: 0x6, model: 0xbe, stepping: 0x0)
```

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. fig. 6.4).

```
[root@vbox ~]# dmesg | grep -i "Memory:"
[ 0.273336] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.273338] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000fffff]
[ 0.273340] PM: hibernation: Registered nosave memory: [mem 0xdd618000-0xdd618fff]
[ 0.273341] PM: hibernation: Registered nosave memory: [mem 0xde59c000-0xde5e4fff]
[ 0.273342] PM: hibernation: Registered nosave memory: [mem 0xdeaed000-0xdead6cfff]
[ 0.273343] PM: hibernation: Registered nosave memory: [mem 0xdead6d000-0xdead7efff]
[ 0.273344] PM: hibernation: Registered nosave memory: [mem 0xdead7f000-0xdeadfefff]
[ 0.273345] PM: hibernation: Registered nosave memory: [mem 0xdf16e000-0xdf16efff]
[ 0.273345] PM: hibernation: Registered nosave memory: [mem 0xffff0000-0xffffffff]
[ 0.537836] Freeing SMP alternatives memory: 48K
[ 0.654294] Memory: 9061016K/9400000K available (22528K kernel code, 4428K rdata, 16752K
rodata, 4884K init, 4724K bss, 330968K reserved, 0K cma-reserved)
[ 1.646138] Freeing initrd memory: 26356K
[ 2.612821] Freeing unused decrypted memory: 2028K
[ 2.614072] Freeing unused kernel image (initram) memory: 4884K
[ 2.614975] Freeing unused kernel image (rodata/data gap) memory: 1680K
```

Рис. 6.5: Поиск объема доступной оперативной памяти

Нахожу тип обнаруженного гипервизора (рис. fig. 6.4).

```
[root@vbox ~]# 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. fig. 6.5).

```
[root@vbox ~]# sudo fdisk -l
Disk /dev/sda: 80 GiB, 85899345920 bytes, 167772160 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: BBF6BA2E-1191-429A-A133-9222FEE0C945

Device      Start      End      Sectors  Size Type
/dev/sda1    2048     1230847   1228800   600M EFI System
/dev/sda2   1230848   3327999   2097152    1G Linux extended boot
/dev/sda3   3328000  167770111 164442112 78,4G Linux filesystem

Disk /dev/zram0: 8 GiB, 8589934592 bytes, 2097152 sectors
Units: sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

Рис. 6.7: Поиск типа файловой системы корневого раздела

Последовательность монтирования файловых систем можно посмотреть, введя в поиск по результату dmesg слово mount (рис. fig. 6.6).

```
[root@vbox ~]# dmesg | grep -i "mount"
[ 0.530164] Mount cache hash table entries: 32768 (order: 6, 262144 bytes, linear)
[ 0.539198] Mountpoint-cache hash table entries: 32768 (order: 6, 262144 bytes, linear)
[ 3.851889] BTRFS: device label fedora devid 1 transid 449 /dev/sda3 (8:3) scanned by mount (488)
[ 3.854568] BTRFS info (device sda3): first mount of filesystem eb2d38c9-b2a5-4873-ae61-5b8a9fa21c3a
[ 7.002176] systemd[1]: run-credentials-systemd\x2djournal.service.mount: Deactivated successfully.
[ 7.016217] systemd[1]: Set up automount proc-sys-fs-binfmt_misc.automount - Arbitrary Executable File Formats File System Automount Point.
[ 7.029858] systemd[1]: Listening on systemd-remountfsd.socket - DDI File System Mounter Socket.
[ 7.044428] systemd[1]: Mounting dev-hugepages.mount - Huge Pages File System...
[ 7.047842] systemd[1]: Mounting dev-mqueue.mount - POSIX Message Queue File System...
[ 7.050539] systemd[1]: Mounting sys-kernel-debug.mount - Kernel Debug File System...
[ 7.054481] systemd[1]: Mounting sys-kernel-tracing.mount - Kernel Trace File System...
[ 7.144872] systemd[1]: Starting systemd-remount-fs.service - Remount Root and Kernel File Systems...
[ 7.228628] systemd[1]: Mounted dev-hugepages.mount - Huge Pages File System.
[ 7.228619] systemd[1]: Mounted dev-mqueue.mount - POSIX Message Queue File System.
[ 7.229358] systemd[1]: Mounted sys-kernel-debug.mount - Kernel Debug File System.
[ 7.229122] systemd[1]: Mounted sys-kernel-tracing.mount - Kernel Trace File System.
[ 7.237684] systemd[1]: Finished systemd-remount-fs.service - Remount Root and Kernel File Systems.
[ 7.241361] systemd[1]: Mounting sys-fs-fuse-connections.mount - FUSE Control File System
```

Рис. 6.8: Последовательность монтирования файловых систем

...