

Лабораторная работа №1

Операционные системы

Кирилюк Светлана Алексеевна

Содержание

1 Цель работы	5
2 Выполнение лабораторной работы	6
3 Выводы	21

Список иллюстраций

2.1	Папка машины	6
2.2	Хост-клавиши	6
2.3	Создание виртуальной машины (1)	7
2.4	Создание виртуальной машины (2)	7
2.5	Создание виртуальной машины (3)	8
2.6	Настройка виртуальной машины (1)	8
2.7	Настройка виртуальной машины (2)	9
2.8	Настройка виртуальной машины (3)	10
2.9	Настройка виртуальной машины (4)	10
2.10	Настройка виртуальной машины (5)	11
2.11	Установка виртуальной машины	12
2.12	Изъятие диска	13
2.13	Обновление пакетов	13
2.14	Программы для удобства работы в консоли	14
2.15	Програмное обеспечение	14
2.16	Запуск таймера	14
2.17	Отключение SELinux	15
2.18	Установка драйверов (1)	15
2.19	Установка драйверов (2)	16
2.20	Настройка раскладки клавиатуры	16
2.21	Установка pandoc	17
2.22	Установка texlive	17
2.23	Последовательность монтирования файловых систем (1)	17
2.24	Последовательность монтирования файловых систем (2)	18
2.25	Версия ядра	19
2.26	Частота процессора	19
2.27	Модель процессора	19
2.28	Объём доступной памяти	20
2.29	Тип обнаружения гипервизора	20
2.30	Тип файловой системы корневого каталога	20

Список таблиц

1 Цель работы

Приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых сервисов для дальнейшей работы.

2 Выполнение лабораторной работы

Перед началом работы я скачала образ операционной системы и установила VirtualBox на свой компьютер. Запустив VirtualBox я выбрала папку для хранения будущей машины (рис. 2.1) и настроила хост-клавиши (рис. 2.2).

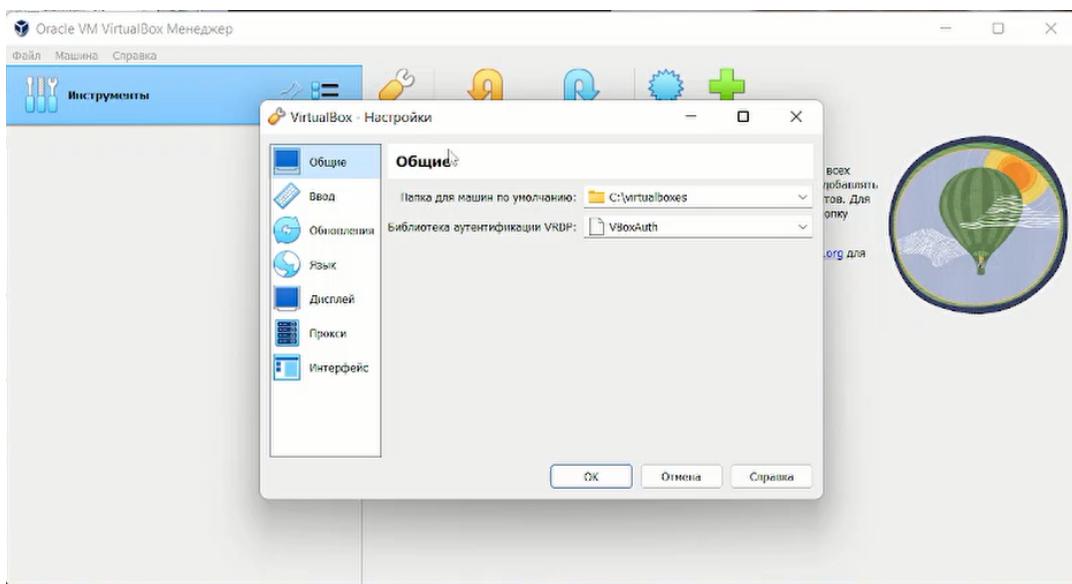


Рис. 2.1: Папка машины

Хост-клавиши

Рис. 2.2: Хост-клавиши

Затем я создала виртуальную машину, указала её название и образ (рис. 2.3), отрегулировала объём основной памяти (рис. 2.4) и создала новый виртуальный жёсткий диск (рис. 2.5).

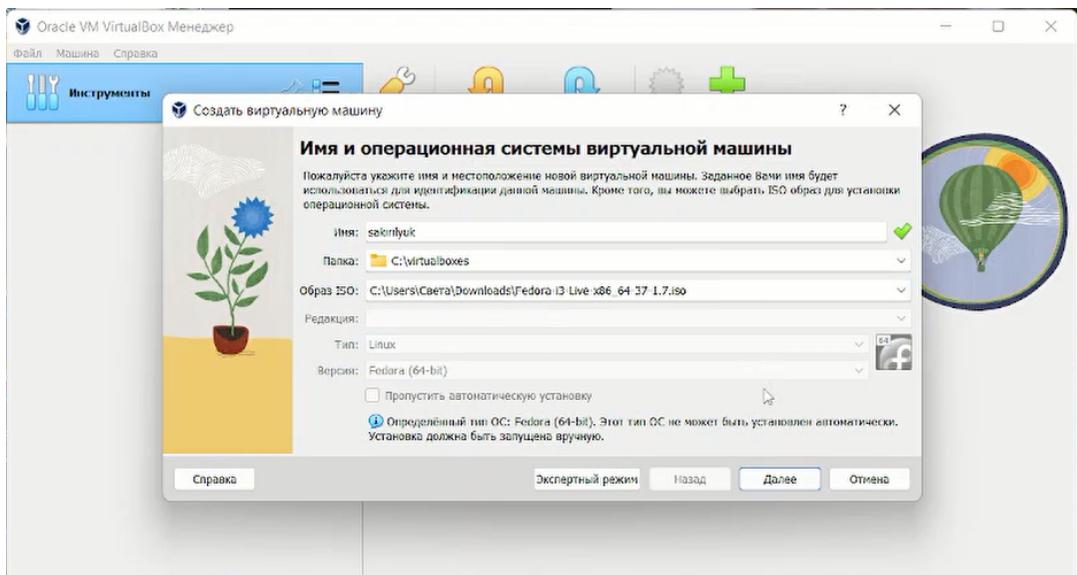


Рис. 2.3: Создание виртуальной машины (1)

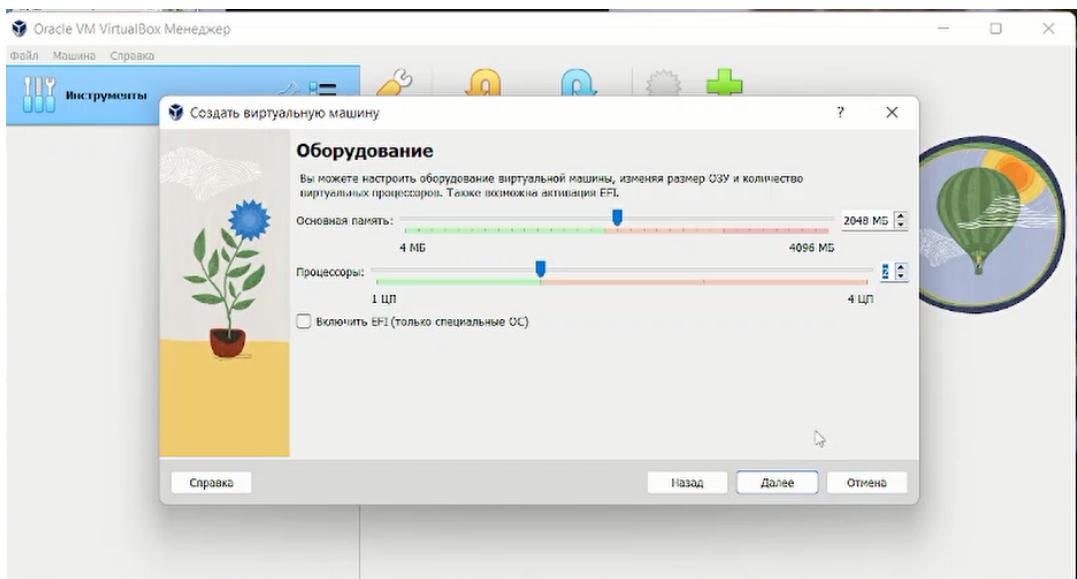


Рис. 2.4: Создание виртуальной машины (2)

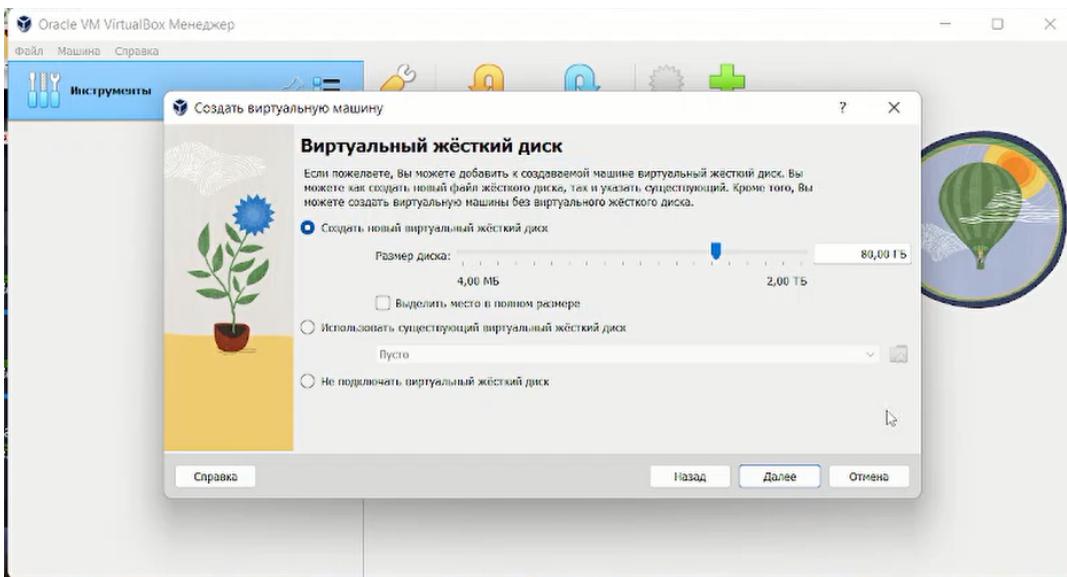


Рис. 2.5: Создание виртуальной машины (3)

Открыв созданную машину, я перешла в терминал и ввела команду для открытия окна с настройкой интерфейса (рис. 2.6). Здесь я настроила язык и часовой пояс, выбрала место установки (рис. 2.7), задала имя узла (рис. 2.8), создала учётную запись (рис. 2.9) и создала пользователя (рис. 2.10).

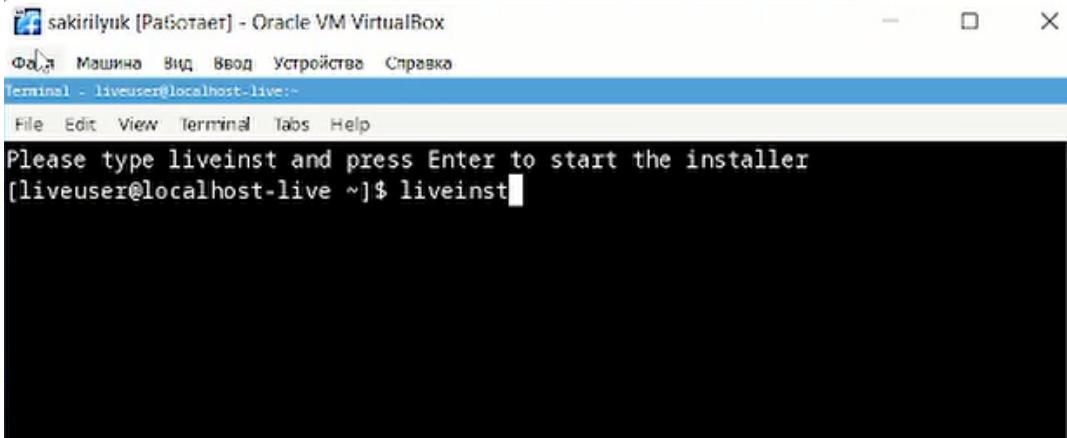


Рис. 2.6: Настройка виртуальной машины (1)

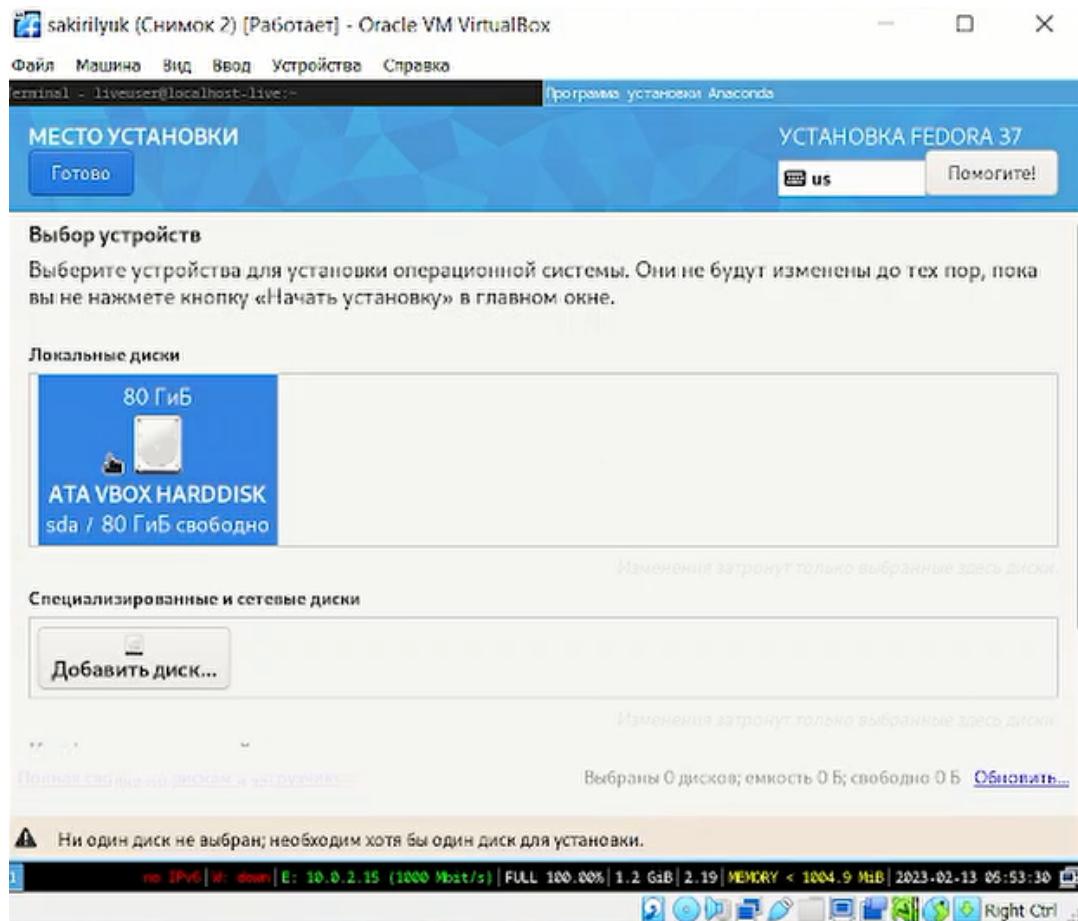


Рис. 2.7: Настройка виртуальной машины (2)

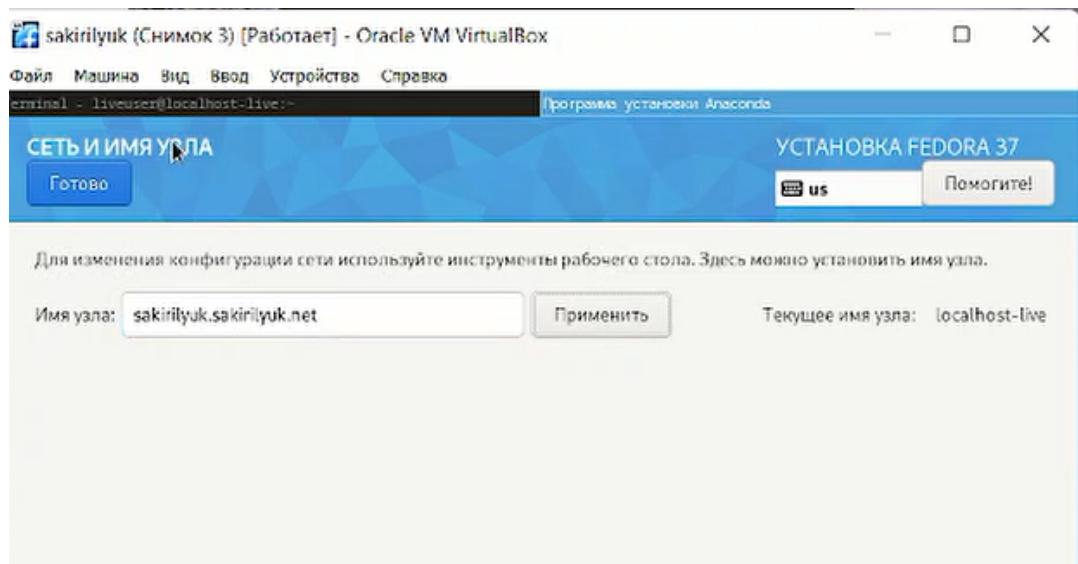


Рис. 2.8: Настройка виртуальной машины (3)

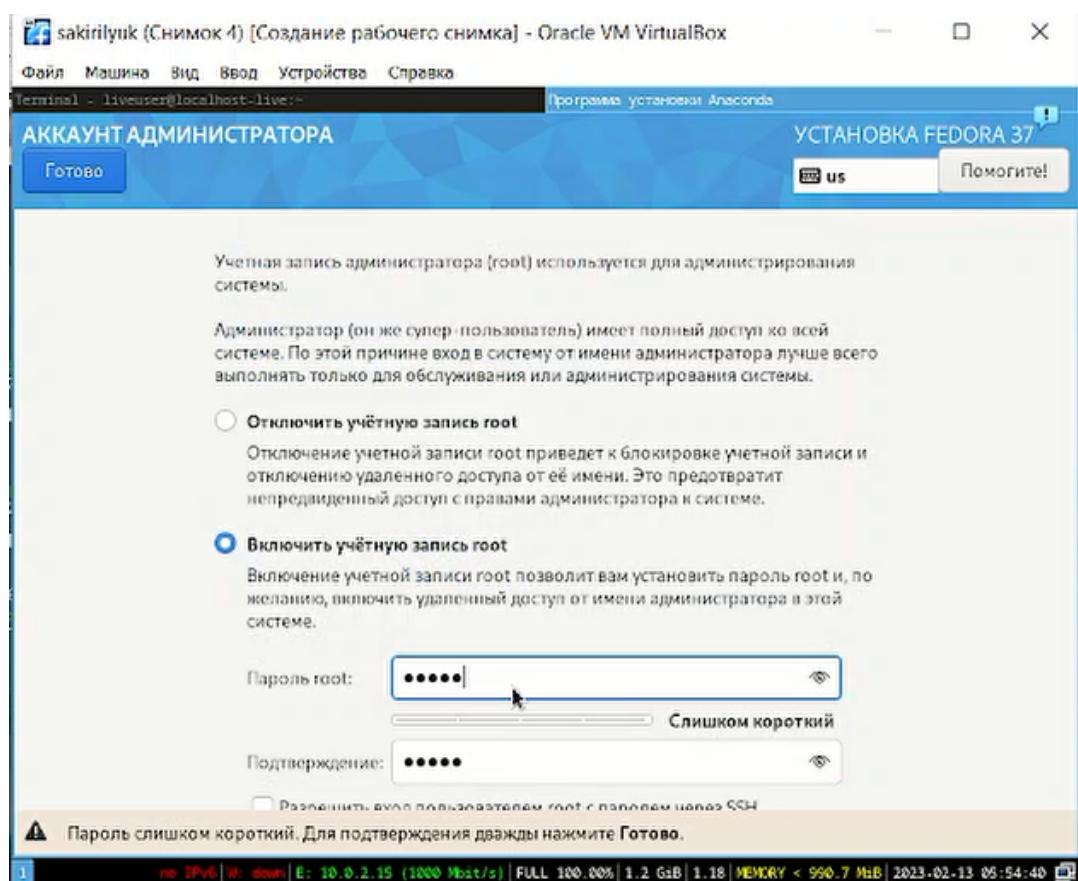


Рис. 2.9: Настройка виртуальной машины (4)

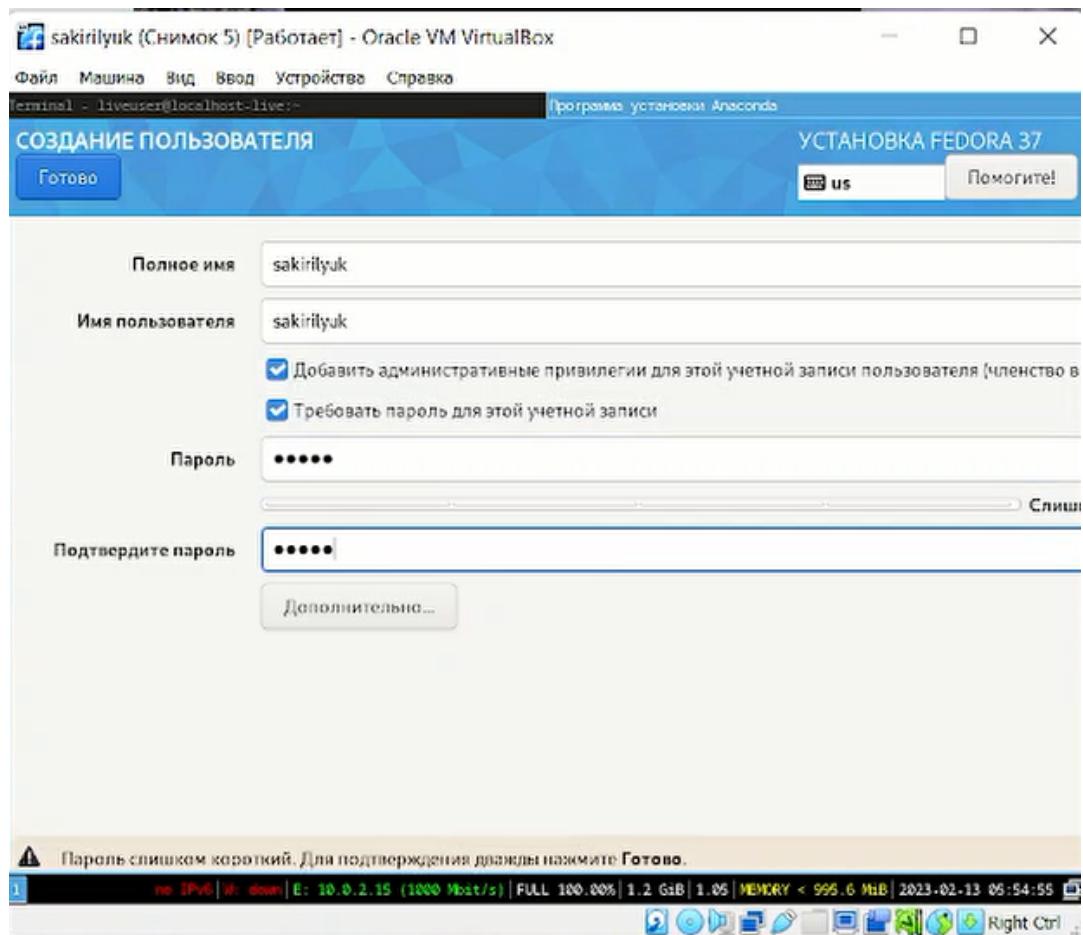


Рис. 2.10: Настройка виртуальной машины (5)

После настройки виртуальной машины я начала установку (рис. 2.11). После её окончания, я закрыла виртуальную машину и изъяла диск (рис. 2.12).

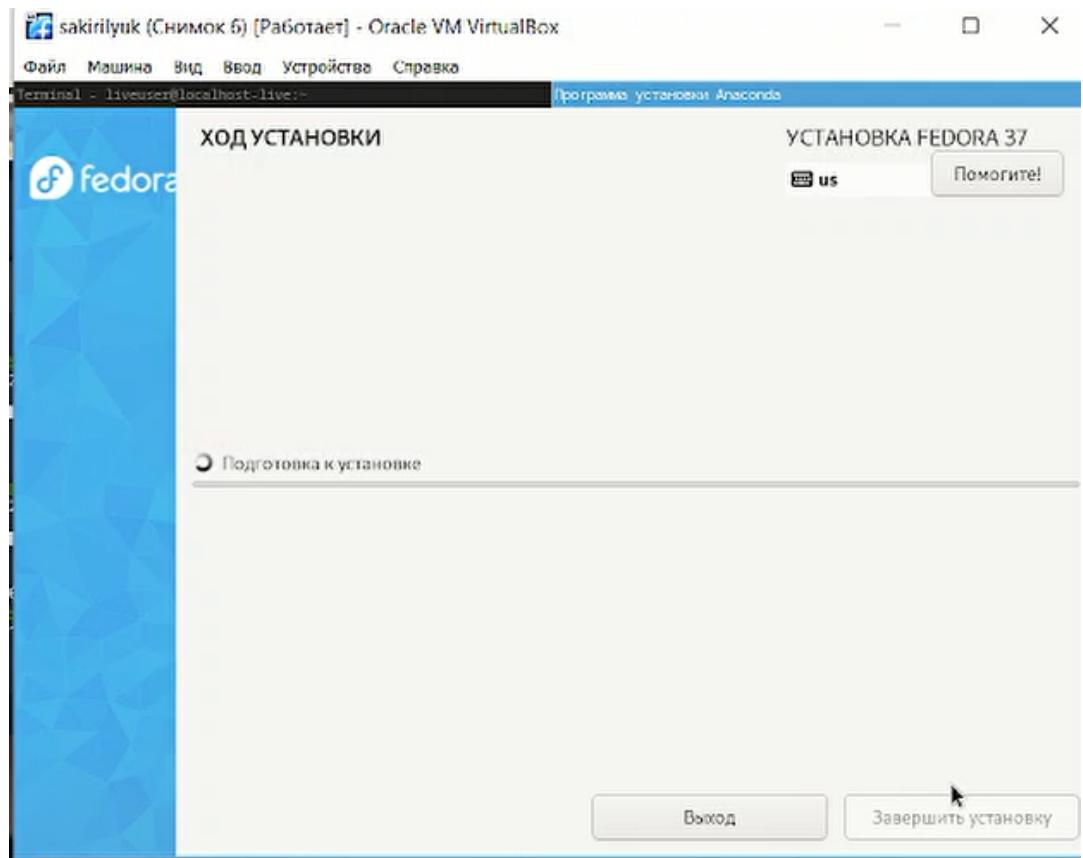


Рис. 2.11: Установка виртуальной машины

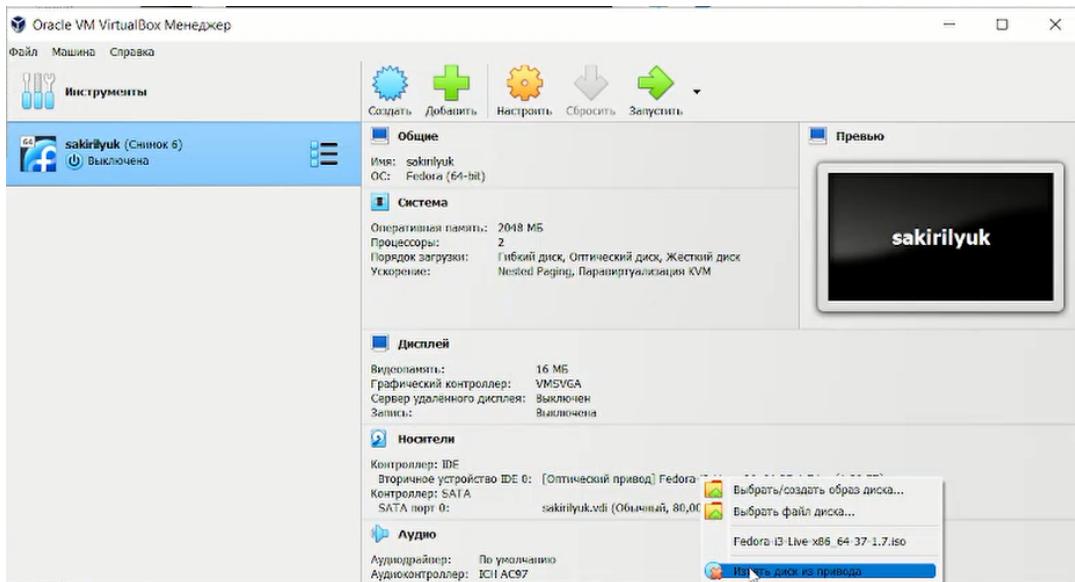


Рис. 2.12: Изъятие диска

Я снова запустила машину и приступила к обновлению всех пакетов (рис. 2.13). Затем я установила программы для удобства работы консоли (рис. 2.14) и программное обеспечение (рис. 2.15), после чего командой запустила таймер (рис. 2.16).

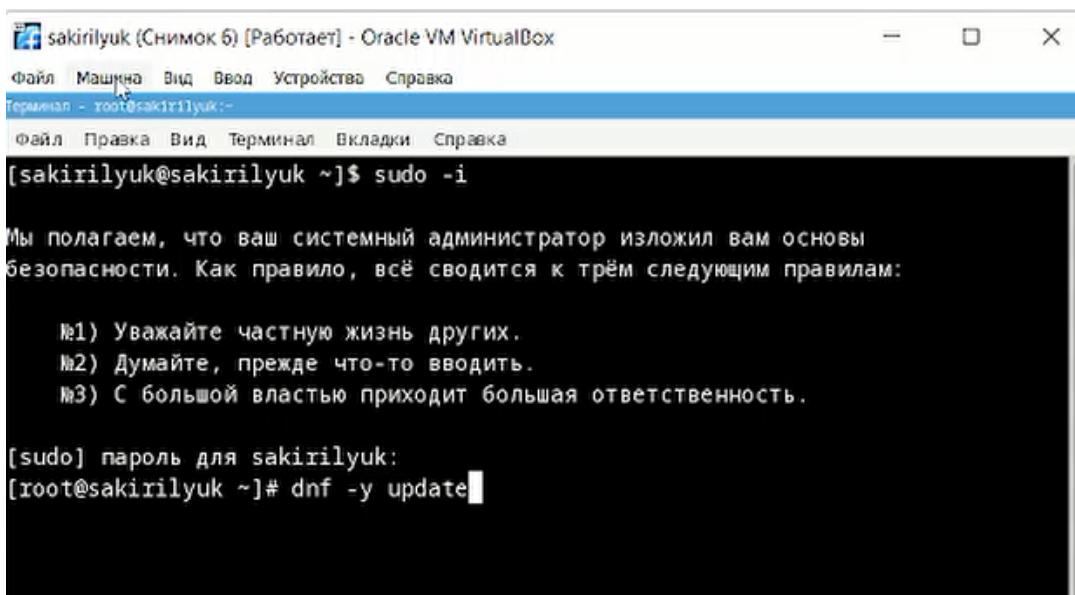
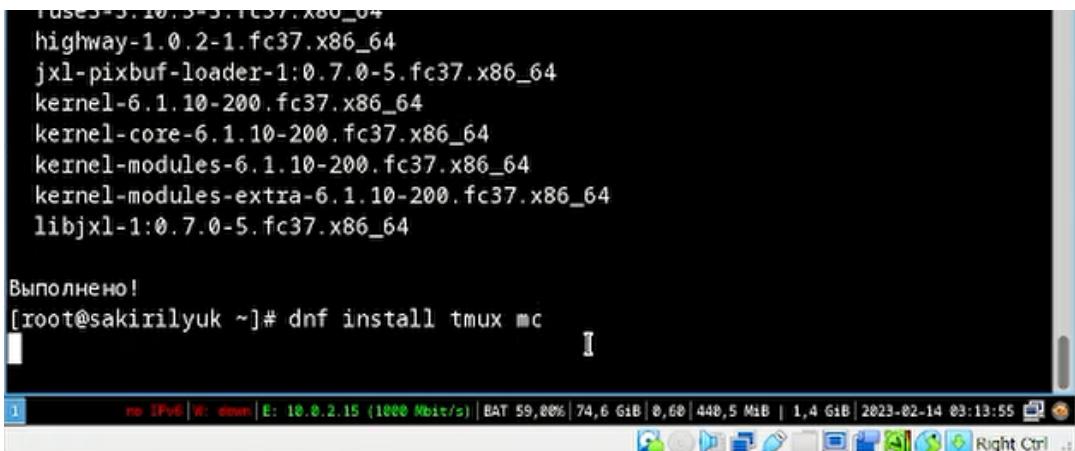


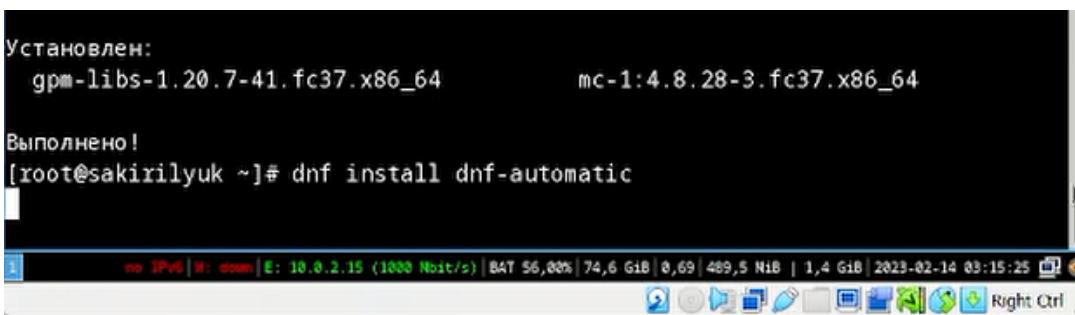
Рис. 2.13: Обновление пакетов



```
fuse3-3.10.3-3.fc37.x86_64
highway-1.0.2-1.fc37.x86_64
jxl-pixbuf-loader-1:0.7.0-5.fc37.x86_64
kernel-6.1.10-200.fc37.x86_64
kernel-core-6.1.10-200.fc37.x86_64
kernel-modules-6.1.10-200.fc37.x86_64
kernel-modules-extra-6.1.10-200.fc37.x86_64
libjxl-1:0.7.0-5.fc37.x86_64

Выполнено!
[root@sakirilyuk ~]# dnf install tmux mc
```

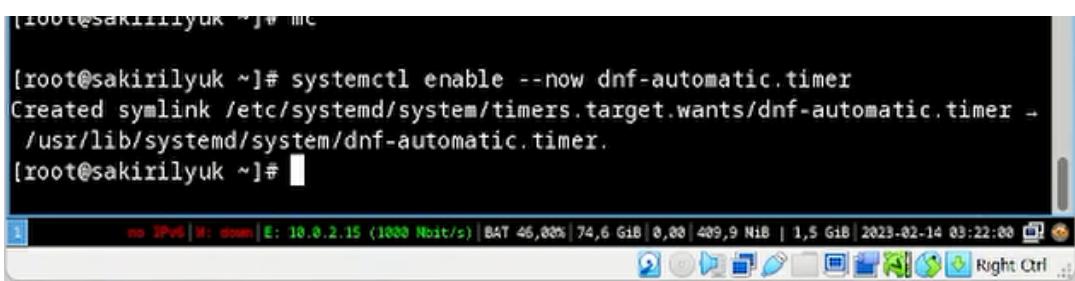
Рис. 2.14: Программы для удобства работы в консоли



```
Установлен:
grpm-libs-1.20.7-41.fc37.x86_64           mc-1:4.8.28-3.fc37.x86_64

Выполнено!
[root@sakirilyuk ~]# dnf install dnf-automatic
```

Рис. 2.15: Програмное обеспечение



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

Рис. 2.16: Запуск таймера

Далее я перешела к отключению SELinux, заменив значение SELINUX=enforcing на SELINUX=permissive в файле /etc/selinux/config (рис. 2.17).

```
config           [-M--] 18 L:[ 12*10 22/ 30] *(929 /1188b) 0010 0x00[*][X]
# 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
```

Рис. 2.17: Отключение SELinux

Переключившись на роль супер-пользователя, я установила драйвера (рис. 2.18), (рис. 2.19).

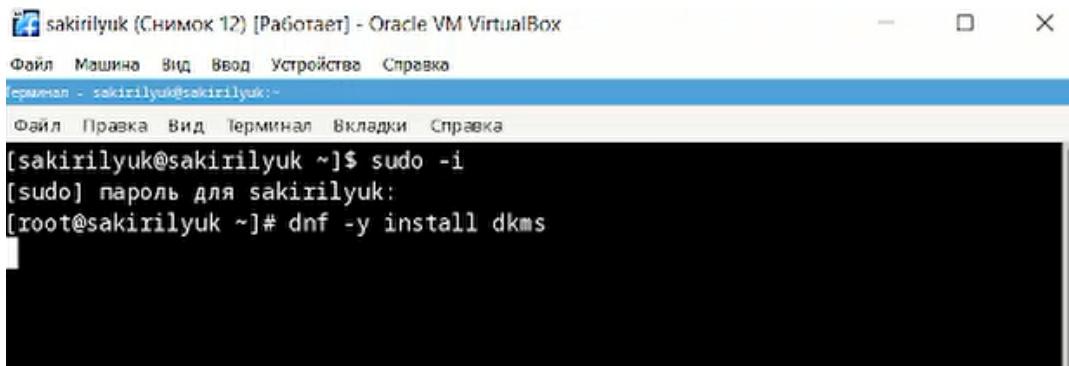


Рис. 2.18: Установка драйверов (1)

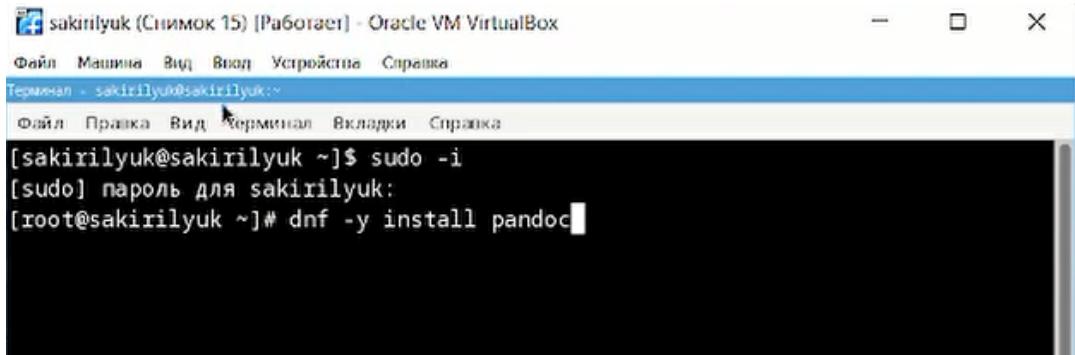
Рис. 2.19: Установка драйверов (2)

Затем я настроила раскладку клавиатуры, отредактировав конфигурационный файл (рис. 2.20).

```
00-keyboard.conf [-M--] 40 L:[ 1+ 2 3/ 10] *(180 / 436b) 0010 0x00[*][X]
# Written by systemd-located(8), read by systemd-located and Xorg. It's
# probably wise not to edit this file manually. Use localectl(1) to
# instruct systemd-located to update it. █
Section "InputClass"
    Identifier "system-keyboard"
    MatchIsKeyboard "on"
    Option "XkbLayout" "us,ru"
    Option "XkbVariant" ",winkeys"
    Option "XkbOptions" "grp:rctrl_toggle,compose:ralt,terminate:ctrl_alt
EndSection
```

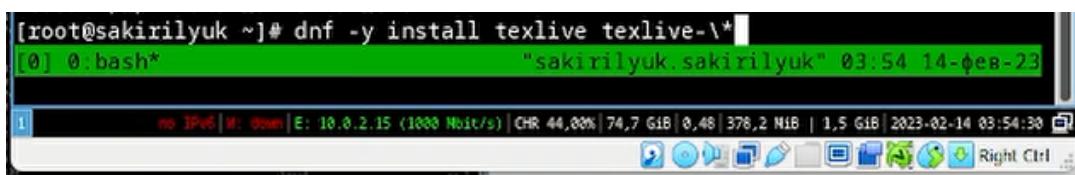
Рис. 2.20: Настройка раскладки клавиатуры

Также я установила pandoc (рис. 2.21) и texlive (рис. 2.22).



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

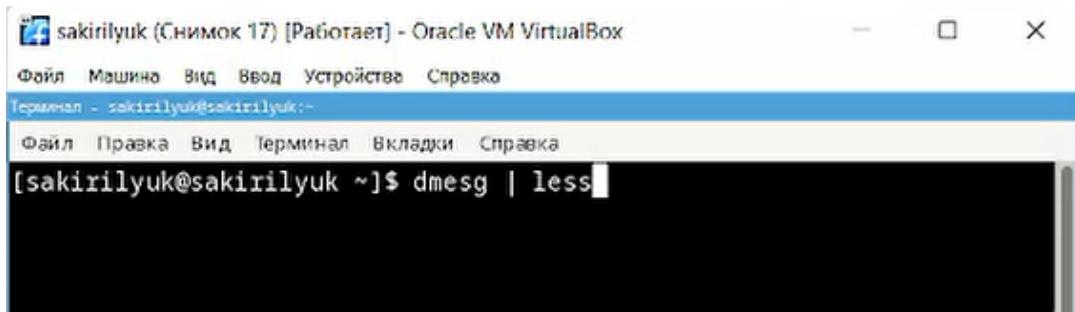
Рис. 2.21: Установка pandoc



```
[root@sakirilyuk ~]# dnf -y install texlive texlive-\*  
[0] 0:bash* "sakirilyuk.sakirilyuk" 03:54 14-фев-23
```

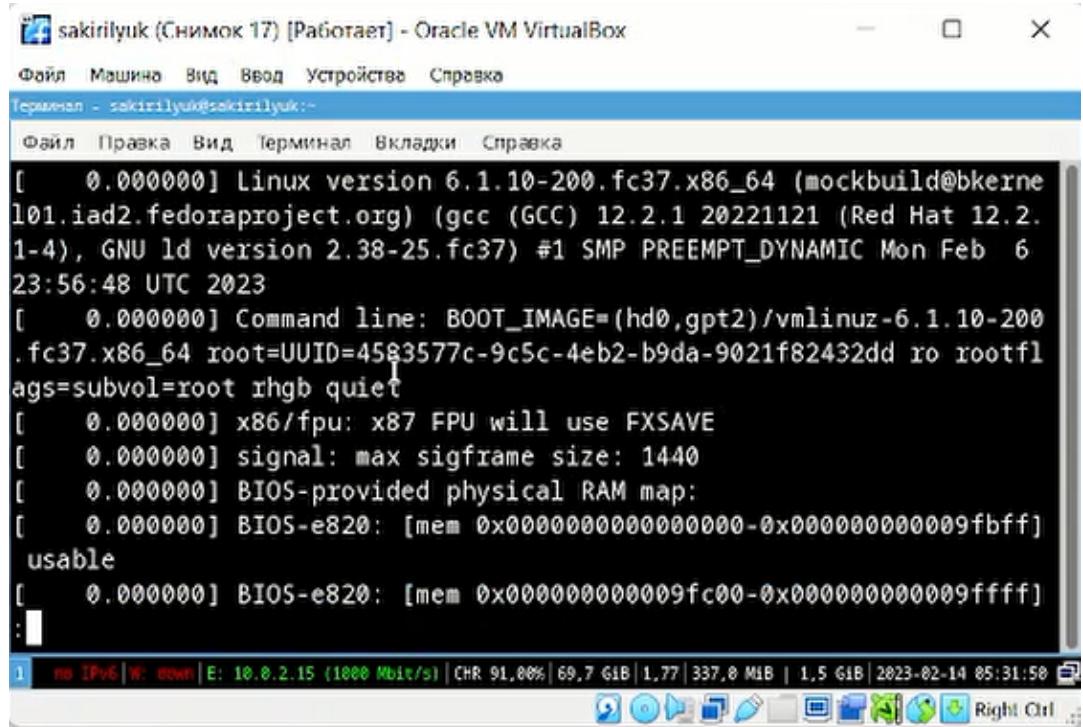
Рис. 2.22: Установка texlive

Затем я приступила к выполнению домашней работы. Введя необходимые команды я узнала информацию про последовательность монтирования файловых систем (рис. 2.23), (рис. 2.24), версию ядра (рис. 2.25), частоту процессора (рис. 2.26), его модель (рис. 2.27), объём доступной памяти (рис. 2.28), тип обнаружения гипервизора (рис. 2.29) и тип файловой системы корневого каталога (рис. 2.30).



```
[sakirilyuk@sakirilyuk ~]$ dmesg | less
```

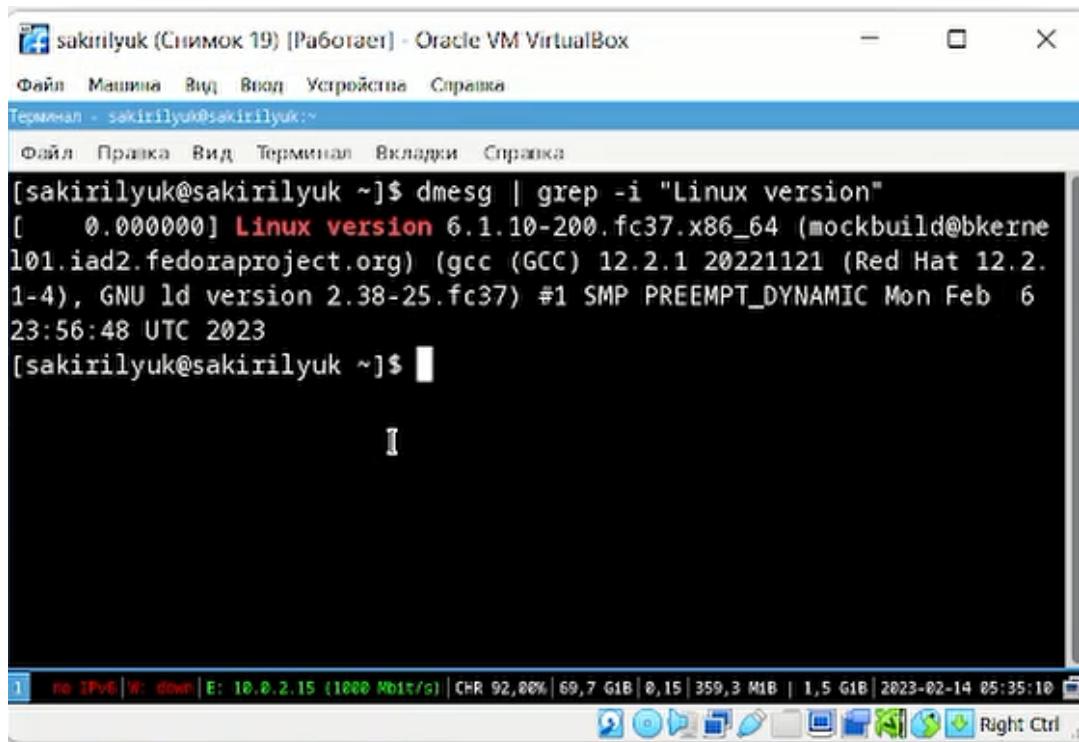
Рис. 2.23: Последовательность монтирования файловых систем (1)



The screenshot shows a terminal window titled "sakirilyuk (Снимок 17) [Работает] - Oracle VM VirtualBox". The window displays the boot process of a Red Hat 12.2.1 system. The logs include kernel version information, command line parameters (including root=UUID), processor configuration, memory maps, and BIOS-e820 entries. The terminal interface includes a menu bar with "Файл", "Машина", "Вид", "Ввод", "Устройства", "Справка", and a toolbar with icons for copy, paste, cut, and others.

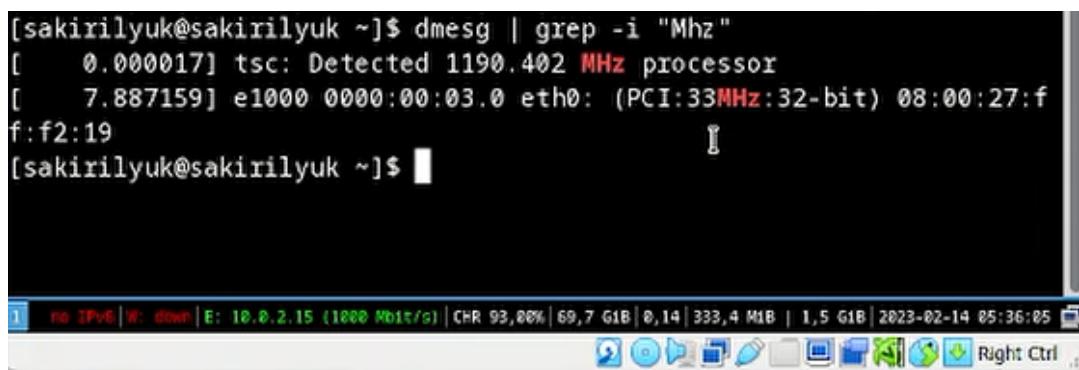
```
[ 0.000000] Linux version 6.1.10-200.fc37.x86_64 (mockbuild@bkerne
101.iad2.fedoraproject.org) (gcc (GCC) 12.2.1 20221121 (Red Hat 12.2.
1-4), GNU ld version 2.38-25.fc37) #1 SMP PREEMPT_DYNAMIC Mon Feb 6
23:56:48 UTC 2023
[ 0.000000] Command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.1.10-200
.fc37.x86_64 root=UUID=4583577c-9c5c-4eb2-b9da-9021f82432dd ro rootfl
ags=subvol=root rhgb quiet
[ 0.000000] x86/fpu: x87 FPU will use FXSAVE
[ 0.000000] signal: max sigframe size: 1440
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff]
usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff]
:
```

Рис. 2.24: Последовательность монтирования файловых систем (2)



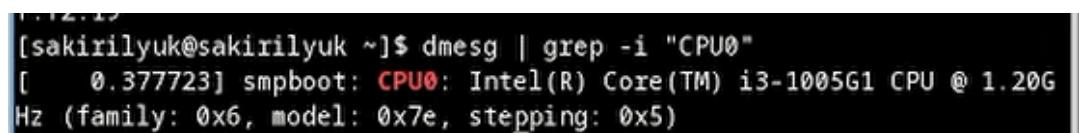
```
[sakirilyuk@sakirilyuk ~]$ dmesg | grep -i "Linux version"
[    0.000000] Linux version 6.1.10-200.fc37.x86_64 (mockbuild@bkerne
101.iad2.fedoraproject.org) (gcc (GCC) 12.2.1 20221121 (Red Hat 12.2.
1-4), GNU ld version 2.38-25.fc37) #1 SMP PREEMPT_DYNAMIC Mon Feb  6
23:56:48 UTC 2023
```

Рис. 2.25: Версия ядра



```
[sakirilyuk@sakirilyuk ~]$ dmesg | grep -i "Mhz"
[    0.000017] tsc: Detected 1190.402 MHz processor
[    7.887159] e1000 0000:00:03.0 eth0: (PCI:33MHz:32-bit) 08:00:27:f
f:f2:19
```

Рис. 2.26: Частота процессора



```
[sakirilyuk@sakirilyuk ~]$ dmesg | grep -i "CPU0"
[    0.377723] smpboot: CPU0: Intel(R) Core(TM) i3-1005G1 CPU @ 1.20G
```

Рис. 2.27: Модель процессора

```
[ 0.037438] /M: Initialization. Registered nosave memory. mem=0x00000000-0x0000ffff  
[ 0.096965] Memory: 1975752K/2096696K available (16393K kernel code, 3265K iwdata, 12468K rodata, 3032K init, 4596K bss, 120684K reserved, 0K cma-reserved)  
[ 0.282906] Freeing SMP alternatives memory: 44K
```

Рис. 2.28: Объём доступной памяти

```
[sakirilyuk@sakirilyuk ~]$ dmesg | grep -i "Hypervisor detected"  
[ 0.000000] Hypervisor detected: KVM  
[sakirilyuk@sakirilyuk ~]$
```

Рис. 2.29: Тип обнаружения гипервизора

```
[ 16.725525] EXT4-fs (sda2): mounted filesystem with ordered data mode. Quota mode: none.  
[sakirilyuk@sakirilyuk ~]$
```

Рис. 2.30: Тип файловой системы корневого каталога

Ответы на вопросы:

- 1)Учетная запись содержит сведения, необходимые для идентификации пользователя.
 - 2) 2.1 cat /etc/-release (~\$ cat /etc/-release PRETTY_NAME=“Debian GNU/Linux 10)
 - 2.2 cd (cd ~/work/study/2022-2023/“Операционные системы”)
 - 2.3 sudo (sudo du -sh /home/root-user/Downloads)
 - 2.4 mkdir - создание каталогов и файлов (mkdir -p ~/work/study/2022-2023/“Операционные системы”). rm - удаление каталогов и файлов (rm -r dir1)
 - 2.5 sudo -i
 - 2.6 history (history -c)
- 3)Файловая система - порядок, определяющий способ организации, хранения и именования данных на носителях информации в компьютерах и других электронных оборудовани- ях.
- 4)Выполнить команду dmesg/ 5)При помощи команд ps и grep определяем идентификатор процесса PID и удаляем его командами kill или killall

3 Выводы

В ходе работы я приобрела практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых сервисов для дальнейшей работы.