

Презентация по лабораторной работе No.1

Операционные системы

Джаллох Ишмаил

02 март 2025

Российский университет дружбы народов, Москва, Россия

Объединённый институт ядерных исследований, Дубна, Россия

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

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

Выполнение лабораторной работы

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:001?)).

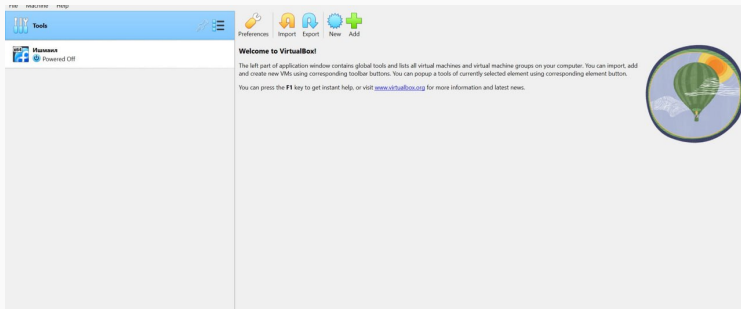
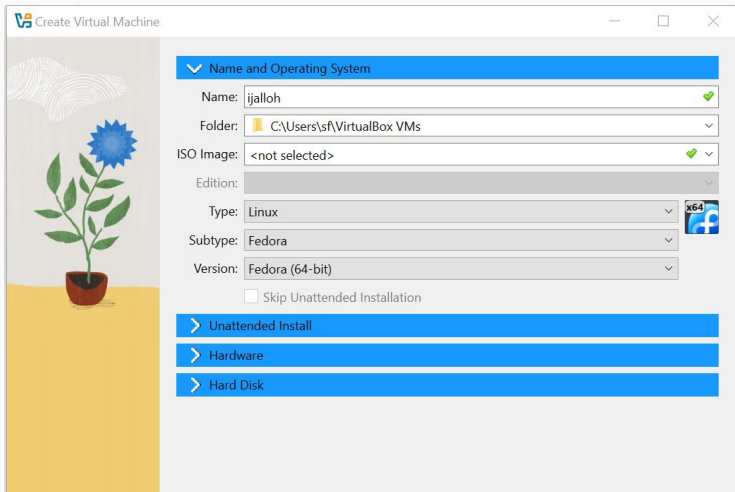


Рис. 1: Окно Virtualbox

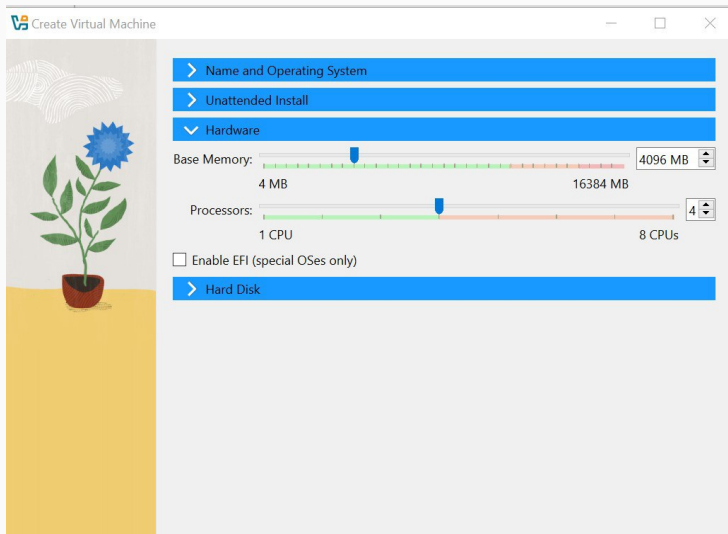
Создание виртуальной машины

Нажимая “создать”, создаю новую виртуальную машину, указываю ее имя, путь к папке машины по умолчанию меня устраивает, выбираю тип ОС и версию (рис. (fig:002?)).



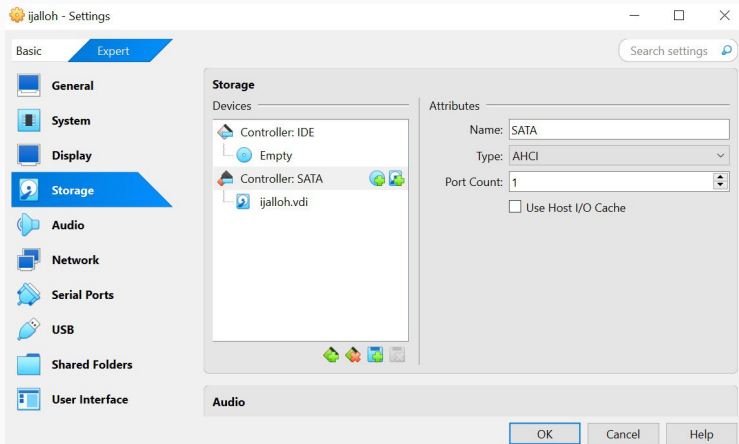
Создание виртуальной машины

Указываю объем основной памяти виртуальной машины размером 4096МБ (рис. (fig:003?)).



Создание виртуальной машины

Выбираю в Virtualbox настройку своей виртуальной машины. Перехожу в “Носители”, добавляю новый привод оптических дисков и выбираю скачанный образ операционной системы Fedora (рис. (fig:004?)).



Скаченный образ ОС был успешно выбран (рис. (fig:005?)).

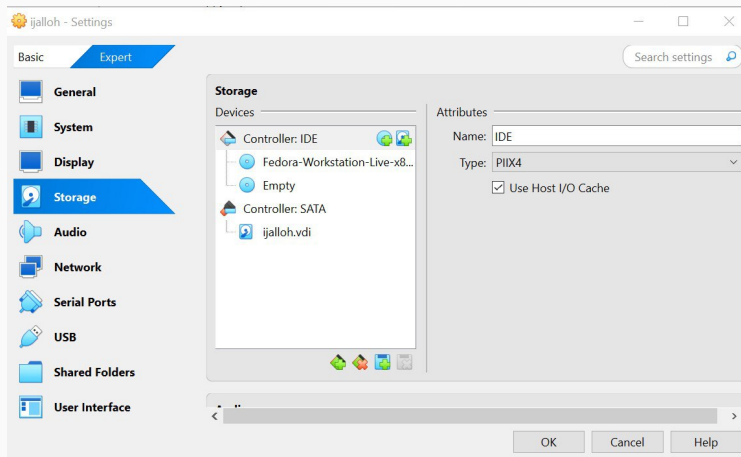


Рис. 5: Выбранный образ оптического диска

Установка операционной системы

Чтобы перейти к раскладке окон с табами, нажимаю Win+w. Выбираю язык для использования в процессе установки русски (рис. (fig:006?)).

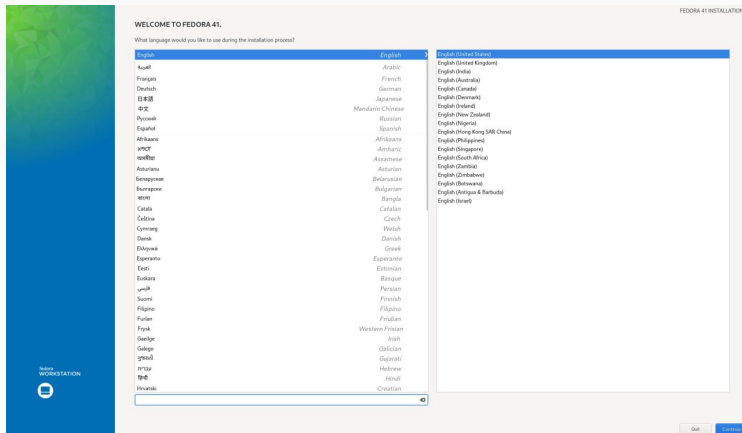


Рис. 6: Выбор языка интерфейса

Раскладку клавиатуры выбираю и русскую, и английскую (рис. (fig:007?)).

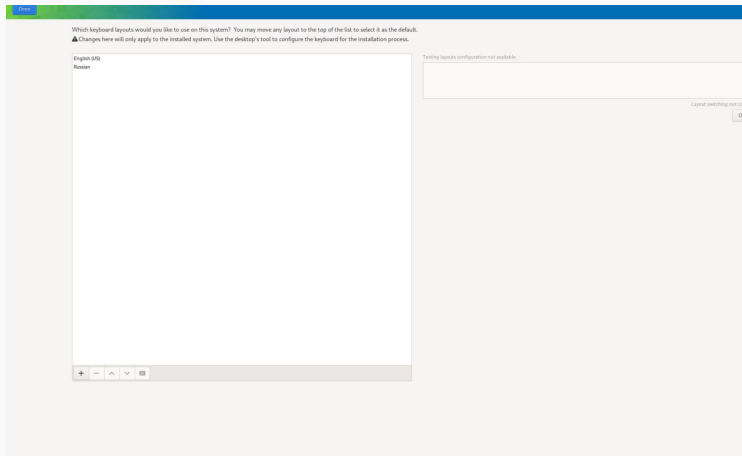
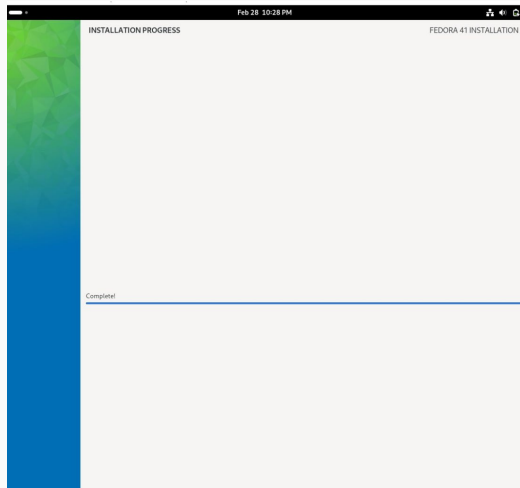



Рис. 7: Выбор раскладки клавиатуры

Далее операционная система устанавливается. После установки нажимаю “завершить установку” (рис. (fig:008?)).




Нажимаю Win+Enter для запуска терминала и переключаюсь на роль супер-пользователя(рис. (fig:028?)).

A screenshot of a terminal window with a dark background. The prompt is 'ijalloh@vbox:~\$' in green text, followed by the command 'sudo -i' in white text.

```
ijalloh@vbox:~$ sudo -i
```

Рис. 9: Запуск терминала

Обновляю все пакеты (рис. (fig:029?)).



```
root@vbox:~# dnf -y update
```

Рис. 10: 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:030?)).


```
root@vbox:~# dnf install tmux mc
Updating and loading repositories:
Repositories loaded.
Package "tmux-3.5a-2.fc41.x86_64" is already installed.
```

Package	Arch	Version	Repository
Installing:			
<code>mc</code>	x86_64	1:4.8.32-1.fc41	updates
Installing dependencies:			
<code>gpm-libs</code>	x86_64	1.20.7-48.fc41	fedora

```
Transaction Summary:
```

Рис. 11: Установка `tmux` и `mc`

Устанавливаю программы для автоматического обновления (рис. (fig:031?)).



```
root@vbox:~# dnf install dnf-automatic
Updating and loading repositories:
```

Рис. 12: Установка программного обеспечения для автоматического обновления

Запускаю таймер (рис. (fig:032?)).



```
root@vbox:~# 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. 13: Запуск таймера

Работа с операционной системой после установки

Перемещаюсь в директорию /etc/selinux, открываю md, ищу нужный файл (рис. (fig:033?)).

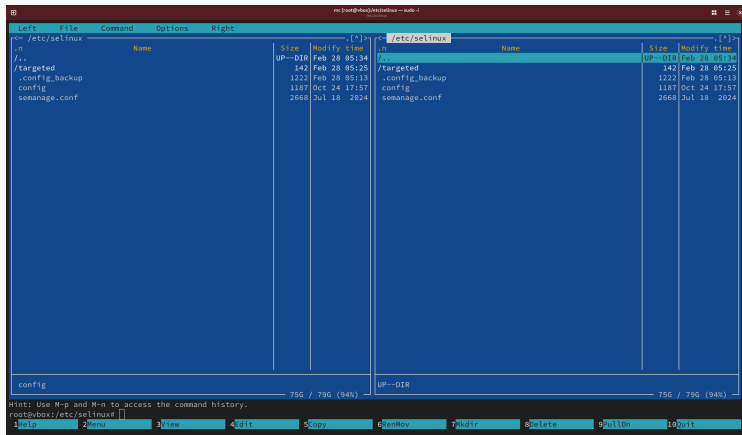
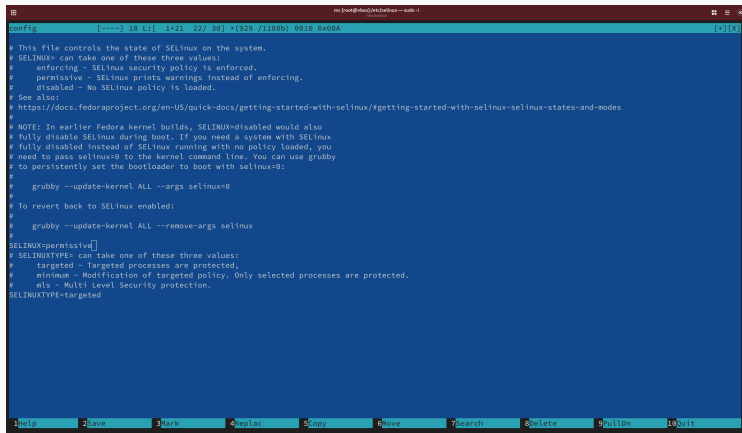


Рис. 14: Поиск файла

Работа с операционной системой после установки

Изменяю открытый файл: SELINUX=enforcing меняю на значение SELINUX=permissive (рис. (fig:034?)).



```
nc[root@rhel6:/etc/selinux] -- sudo -i
config [---] 18 L: [ 1+21 22/ 30] *(929 /1188b) 0010 0x00A [X] [X]

# 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. 15: Изменение файла

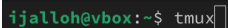
Перезагружаю виртуальную машину (рис. (fig:037?)).



```
ijalloh@vbox:/etc/selinux$ reboot
```

Рис. 16: Перезагрузка виртуальной машины

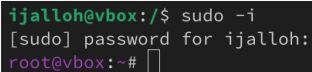
Снова вхожу в ОС, снова запускаю терминал, запускаю терминальный мультиплексор (рис. (fig:038?)).

A screenshot of a terminal window with a dark background. The prompt is 'ijalloh@vbox:~\$' in green text, followed by the command 'tmux' in white text and a white cursor. The rest of the terminal area is black.

```
ijalloh@vbox:~$ tmux
```

Рис. 17: Запуск терминального мультиплексора


Переключаюсь на роль супер-пользователя (рис. (fig:039?)).

A terminal window with a dark background. The first line shows the user 'ijalloh' at host 'vbox' with a green prompt character '\$' followed by the command 'sudo -i'. The second line shows the prompt '[sudo]' followed by 'password for ijalloh:'. The third line shows the user 'root' at host 'vbox' with a purple prompt character '#' followed by a tilde '~' and a cursor icon.

```
ijalloh@vbox:/$ sudo -i
[sudo] password for ijalloh:
root@vbox:~#
```

Рис. 18: Переключение на роль супер-пользователя

Устанавливаю пакет dkms (рис. (fig:0340?)).

A terminal window with a dark background. The prompt is 'root@vbox:~#'. The command 'dnf install dkms' is entered, followed by a cursor. The terminal has a blue bar on the left and a green bar at the bottom.

```
root@vbox:~# dnf install dkms
```

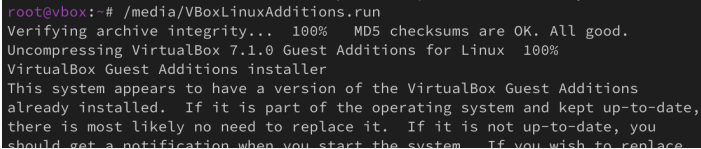
Рис. 19: Установка пакета dkms

В меню виртуальной машины подключаю образ диска гостевой ОС и примонтирую диск с помощью утилиты `mount` (рис. (fig:041?)).

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

Рис. 20: Примонтирование диска


Устанавливаю драйвера (рис. (fig:042?)).

A terminal window with a dark background and light-colored text. The text shows the execution of the command to install VirtualBox Guest Additions, including progress bars for verification and uncompressing, and a message from the installer about the current state of the system.

```
root@vbox:~# /media/VBoxLinuxAdditions.run
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.1.0 Guest Additions for Linux 100%
VirtualBox Guest Additions installer
This system appears to have a version of the VirtualBox Guest Additions
already installed. If it is part of the operating system and kept up-to-date,
there is most likely no need to replace it. If it is not up-to-date, you
should get a notification when you start the system. If you wish to replace
```

Рис. 21: Установка драйвера

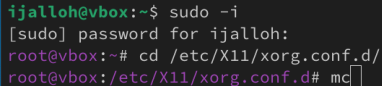
Перезагружаю виртуальную машину (рис. (fig:043?)).



```
root@vbox:~# reboot
```

Рис. 22: Перезагрузка виртуальной машины

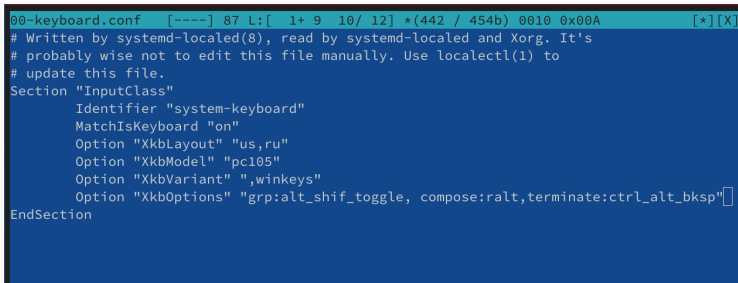
Перехожу в директорию /etc/X11/xorg.conf.d, открываю tc для удобства, открываю файл 00-keyboard.conf (рис. (fig:044?)).



```
ijalloh@vbox:~$ sudo -i
[sudo] password for ijalloh:
root@vbox:~# cd /etc/X11/xorg.conf.d/
root@vbox:/etc/X11/xorg.conf.d# mc
```

Рис. 23: Поиск файла, вход в tc

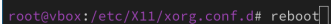
Редактирую конфигурационный файл (рис. (fig:045?)).



```
00-keyboard.conf  [----] 87 L:[ 1+ 9 10/ 12] *(442 / 454b) 0010 0x00A [X] [X]
# Written by systemd-locale(8), read by systemd-locale and Xorg. It's
# probably wise not to edit this file manually. Use localectl(1) to
# update this file.
Section "InputClass"
    Identifier "system-keyboard"
    MatchIsKeyboard "on"
    Option "XkbLayout" "us,ru"
    Option "XkbModel" "pc105"
    Option "XkbVariant" ",winkeys"
    Option "XkbOptions" "grp:alt_shif_toggle, compose:ralt,terminate:ctrl_alt_bksp"
EndSection
```

Рис. 24: Редактирование файла

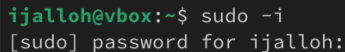
Перезагружаю виртуальную машину (рис. (fig:046?)).

A terminal window with a dark background. The prompt is 'root@vbox:/etc/X11/xorg.conf.d#' and the command 'reboot' is entered, followed by a cursor.

```
root@vbox:/etc/X11/xorg.conf.d# reboot
```

Рис. 25: Перезагрузка виртуальной машины

Запускаю терминал. Запускаю терминальный мультиплексор tmux, переключаюсь на роль супер-пользователя (рис. (fig:047?)).



```
ijalloh@vbox:~$ sudo -i  
[sudo] password for ijalloh:
```

Рис. 26: Переключение на роль супер-пользователя

Устанавливаю pandoc с помощью утилиты dnf и флага -y, который автоматически на все вопросы системы отвечает “yes” (рис. (fig:048?)).

```
root@vbox:~# dnf -y install pandoc
Updating and loading repositories:
Repositories loaded.
Package Arch Version Repository Size
Installing:
pandoc x86_64 3.1.11.1-32.fc41 fedora 185.0 MiB
Installing dependencies:
pandoc-common noarch 3.1.11.1-31.fc41 fedora 1.9 MiB

Transaction Summary:
Installing: 2 packages

Total size of inbound packages is 27 MiB. Need to download 27 MiB.
After this operation, 187 MiB extra will be used (install 187 MiB, remove 0 B)
```

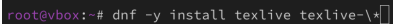
Рис. 27: Установка pandoc

Устанавливаю необходимые расширения для pandoc (рис. (fig:049?)).

```
root@vbox:~# pip install pandoc-fignos pandoc-eqnos pandoc-tablenos pandoc-secnos --user
```

Рис. 28: Установка расширения pandoc

Устанавливаю дистрибутив texlive (рис. (fig:051?)).

A terminal window with a dark background. The prompt is 'root@vbox:~#'. The command entered is 'dnf -y install texlive texlive-*'.

```
root@vbox:~# dnf -y install texlive texlive-*
```

Рис. 29: Установка texlive

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.

Выполнение дополнительного задания

Ввожу в терминале команду `dmesg`, чтобы проанализировать последовательность загрузки системы (рис. (fig:053?)).

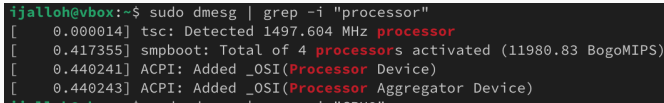
```
ijalloh@vbox:~$ sudo dmesg
[    0.000000] Linux version 6.13.4-200.fc41.x86_64 (mockbuild@leec6c3659654d339658e
9322f9b7a5a) (gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7), GNU ld version 2.43.1-5.
fc41) #1 SMP PREEMPT_DYNAMIC Sat Feb 22 16:09:10 UTC 2025
[    0.000000] Command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.13.4-200.fc41.x86_64 ro
ot=UUID=bdba74b3-304a-432f-b6ba-7ffed7265137 ro rootflags=subvol=root rhgb quiet
[    0.000000] BIOS-provided physical RAM map:
[    0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[    0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[    0.000000] BIOS-e820: [mem 0x0000000000100000-0x0000000000dfffff] usable
[    0.000000] BIOS-e820: [mem 0x000000000dfff0000-0x000000000dfffffff] ACPI data
[    0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee00fff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000100000000-0x0000000011ffffffff] usable
[    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] DMI: Memory slots populated: 0/0
[    0.000000] Hypervisor detected: KVM
[    0.000000] kvm-clock: Using msrs 4b564d01 and 4b564d00
[    0.000004] kvm-clock: using sched offset of 13468702786523 cycles
[    0.000008] clocksource: kvm-clock: mask: 0xffffffffffffffff max_cycles: 0x1cd42e
4dffb, max_idle_ns: 881590591483 ns
```

С помощью поиска, осуществляемого командой 'dmesg | grep -i', ищу версию ядра Linux: 6.1.10-200.fc37.x86_64 (рис. (fig:054?)).

```
ijalloh@vbox:~$ sudo dmesg | grep -i "Linux Version"
[    0.000000] Linux version 6.13.4-200.fc41.x86_64 (mockbuild@leec6c3659654d339658e9322f9b7a5a) (gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7), GNU ld version 2.43.1-5.fc41) #1 SMP PREEMPT_DYNAMIC Sat Feb 22 16:09:10 UTC 2025
```

Рис. 31: Поиск версии ядра

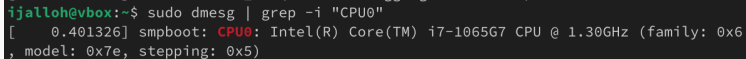
К сожалению, если вводить “Detected Mhz processor” там, где нужно указывать, что я ищу, то мне ничего не выведется. Это происходит потому, что запрос не предусматривает дополнительные символы внутри него (я проверяла, будет ли работать он с маской - не будет). В таком случае я оставила одно из ключевых слов (могла оставить два: “Mhz processor”) и получила результат: 1992 Mhz (рис. (fig:055?)).

A terminal window with a dark background and light green text. The prompt is 'ijalloh@vbox:~\$'. The command 'sudo dmesg | grep -i "processor"' has been executed. The output shows four lines of kernel messages, each starting with a timestamp in brackets. The word 'processor' is highlighted in red in the original image.

```
ijalloh@vbox:~$ sudo dmesg | grep -i "processor"
[ 0.000014] tsc: Detected 1497.604 MHz processor
[ 0.417355] smpboot: Total of 4 processors activated (11980.83 BogoMIPS)
[ 0.440241] ACPI: Added _OSI(Processor Device)
[ 0.440243] ACPI: Added _OSI(Processor Aggregator Device)
```

Рис. 32: Поиск частоты процессора

Аналогично ищу модель процессора (рис. (fig:056?)).

A terminal window with a dark background. The prompt is 'ijalloh@vbox:~\$'. The command entered is 'sudo dmesg | grep -i "CPU0"'. The output shows a kernel message: '[0.401326] smpboot: CPU0: Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz (family: 0x6, model: 0x7e, stepping: 0x5)'.

```
ijalloh@vbox:~$ sudo dmesg | grep -i "CPU0"
[ 0.401326] smpboot: CPU0: Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz (family: 0x6
, model: 0x7e, stepping: 0x5)
```

Рис. 33: Поиск модели процессора

Выполнение дополнительного задания

Объем доступной оперативной памяти ищут аналогично поиску частоты процессора, т. к. возникла та же проблема, что и там (рис. (fig:057?)).

```
ijalloh@vbox:~$ sudo dmesg | grep -i "memory"
[ 0.000000] DMI: Memory slots populated: 0/0
[ 0.003229] ACPI: Reserving FACP table memory at [mem 0xdfff00f0-0xdfff01e3]
[ 0.003230] ACPI: Reserving DSDT table memory at [mem 0xdfff0620-0xdfff2972]
[ 0.003231] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.003232] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.003233] ACPI: Reserving APIC table memory at [mem 0xdfff0240-0xdfff02ab]
[ 0.003233] ACPI: Reserving SSDT table memory at [mem 0xdfff02b0-0xdfff061b]
[ 0.005795] Early memory node ranges
[ 0.015115] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[
[ 0.015118] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009ffff]
[
[ 0.015119] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000effff]
[
[ 0.015119] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[
[ 0.015120] PM: hibernation: Registered nosave memory: [mem 0xdfff0000-0xdfffffff]
[
[ 0.015121] PM: hibernation: Registered nosave memory: [mem 0xe0000000-0xfebfffff]
[
[ 0.015122] PM: hibernation: Registered nosave memory: [mem 0xf0000000-0xf000ffff]
```

Рис. 34: Поиск объема доступной оперативной памяти

Нахожу тип обнаруженного гипервизора (рис. (fig:058?)).



```
ijalloh@vbox:~$ sudo dmesg | grep -i "Hypervisor"
[ 0.000000] Hypervisor detected: KVM
[ 0.262577] SRBDS: Unknown: Dependent on hypervisor status
```

Рис. 35: Поиск типа обнаруженного гипервизора

Выполнение дополнительного задания

Последовательность монтирования файловых систем можно посмотреть, введя в поиск по результату dmesg слово mount (рис. (fig:060?)).

```
ijalloh@vbox:~$ sudo dmesg | grep -i "mount"
[ 0.301056] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 0.301066] Mountpoint-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 3.821677] BTRFS: device label fedora devid 1 transid 494 /dev/sda3 (8:3) scanned by mount (453)
[ 3.829386] BTRFS info (device sda3): first mount of filesystem bdba74b3-304a-432f-b6ba-7ffed7265137
[ 6.692759] systemd[1]: run-credentials-systemd\x2djournald.service.mount: Deactivated successfully.
[ 6.706902] systemd[1]: Set up automount proc-sys-fs-binfmt_misc.automount - Arbitrary Executable File Formats File System Automount Point.
[ 6.725839] systemd[1]: Listening on systemd-mountfsd.socket - DDI File System Mounter Socket.
[ 6.748344] systemd[1]: Mounting dev-hugepages.mount - Huge Pages File System...
[ 6.756638] systemd[1]: Mounting dev-mqueue.mount - POSIX Message Queue File System...
[ 6.760263] systemd[1]: Mounting sys-kernel-debug.mount - Kernel Debug File System...
[ 6.764210] systemd[1]: Mounting sys-kernel-tracing.mount - Kernel Trace File System...
[ 6.874804] systemd[1]: Starting systemd-remount-fs.service - Remount Root and Kernel File Systems...
[ 6.906630] systemd[1]: Mounted dev-hugepages.mount - Huge Pages File System.
[ 6.908418] systemd[1]: Mounted dev-mqueue.mount - POSIX Message Queue File System.
```