

# Отчет по лабораторной работе № 1

## Установка и конфигурация операционной системы на виртуальную машину

Шатохина Виктория Сергеевна

### Содержание

Цель работы .....	1
Теоретическое введение .....	1
Выполнение лабораторной работы .....	1
Домашнее задание .....	8
Заключение .....	12
Ответы на вопросы .....	12
Библиографическая справка .....	15

### Цель работы

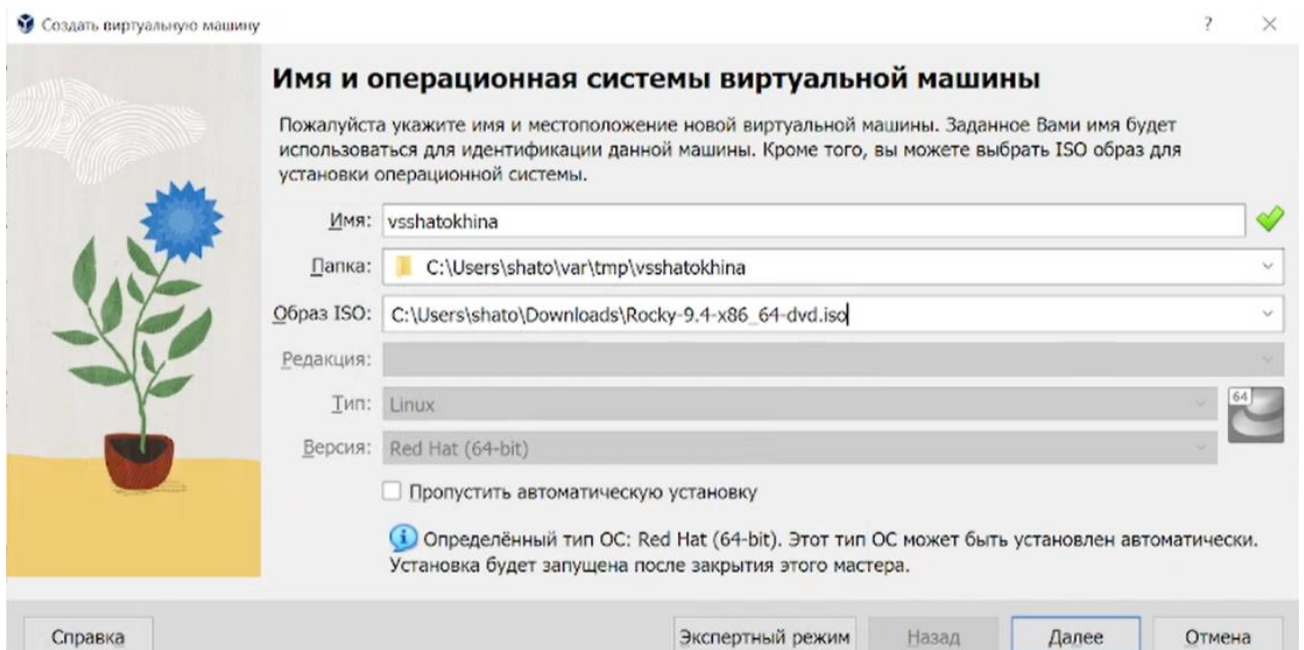
Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

### Теоретическое введение

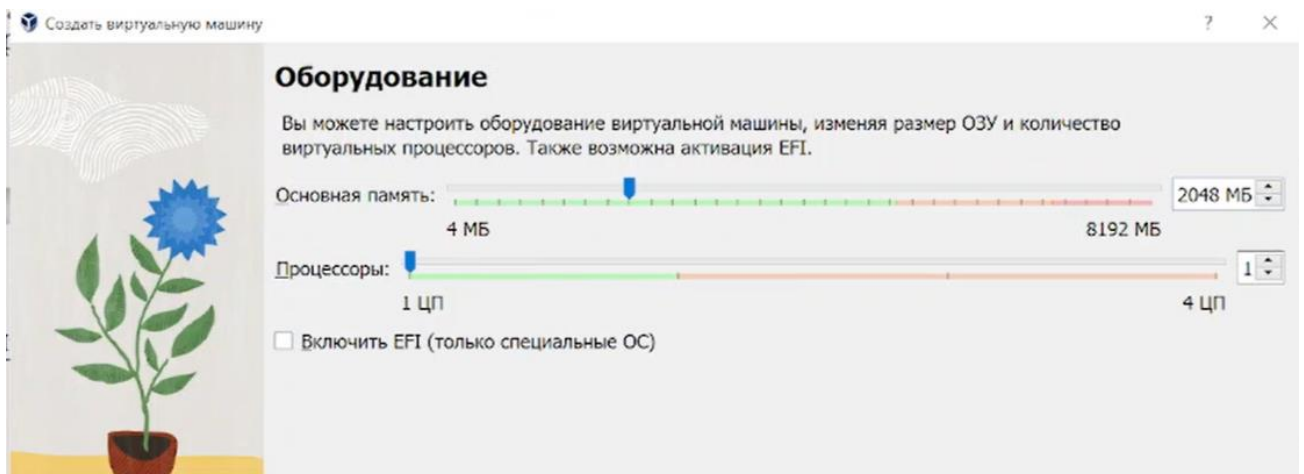
Программа VirtualBox предоставляет широкий спектр возможностей для работы с виртуальными машинами. Это решение подходит для тестирования новых операционных систем, запуска старых приложений или изоляции потенциально опасного программного обеспечения. Благодаря интуитивно понятному интерфейсу и богатому функционалу, VirtualBox стал выбором многих пользователей по всему миру[1].

### Выполнение лабораторной работы

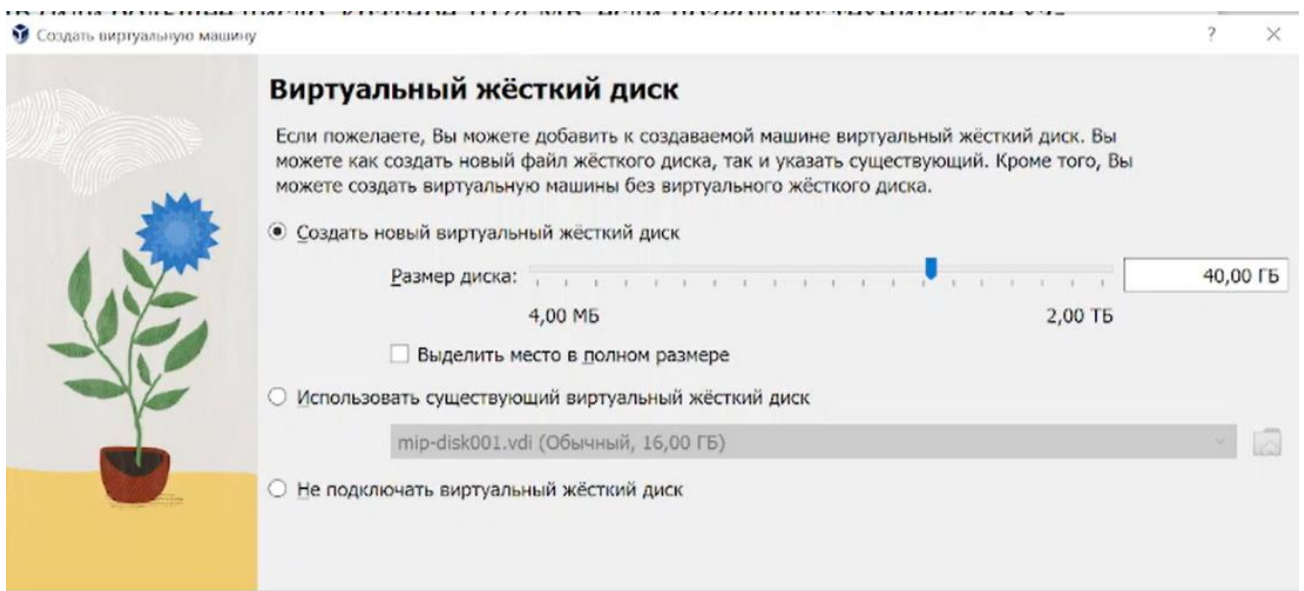
Запускаем виртуальную машину, нажимаем кнопку “создать” и выбираем скаченный образ ISO: См. [рис. 1](#)



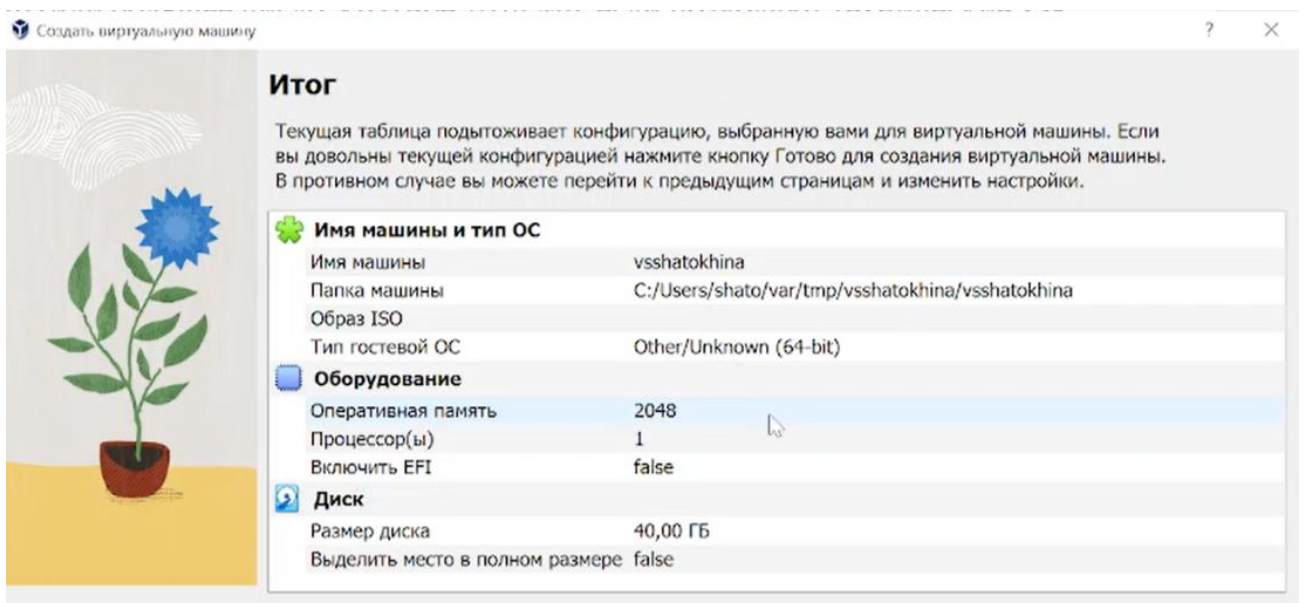
Настраиваем оборудование ВМ, изменяя размер ОЗУ: См. [рис. 3](#)



Создаём новый виртуальный жёсткий диск размером 40 Гб: См. [рис. 4](#)

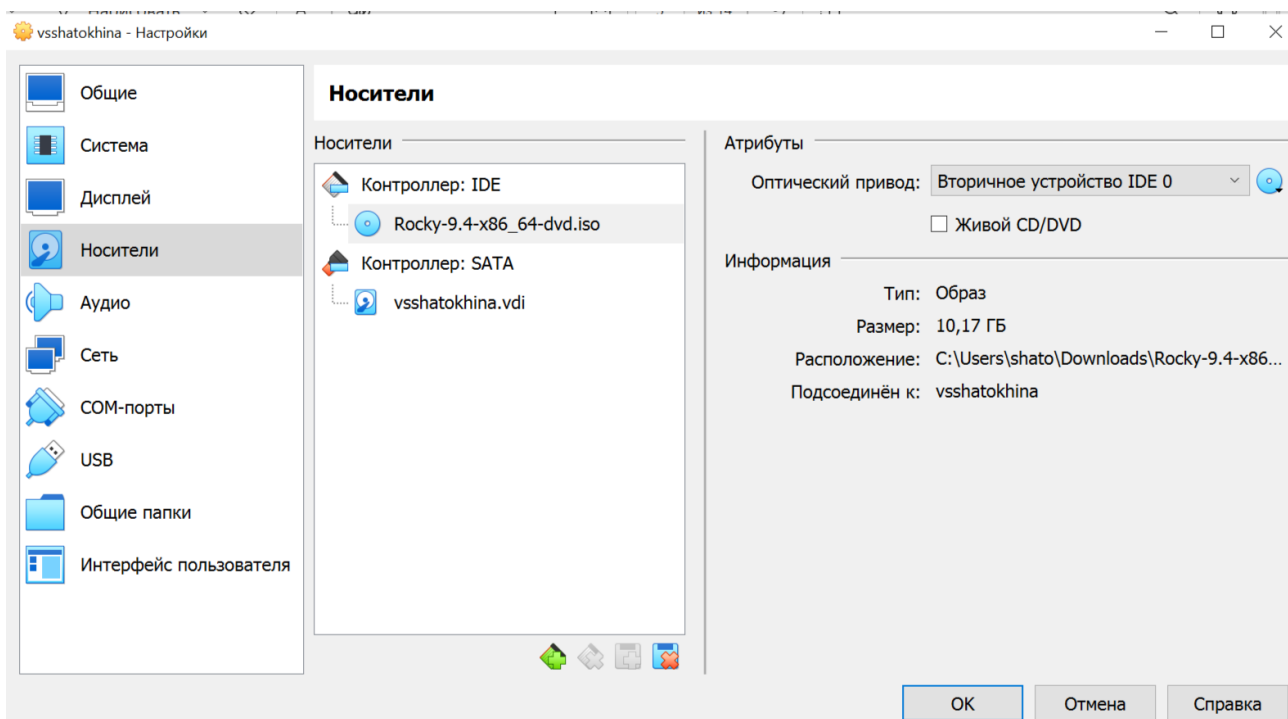


Проверяем итоговую конфигурацию для виртуальной машины: См. [рис. 5](#)

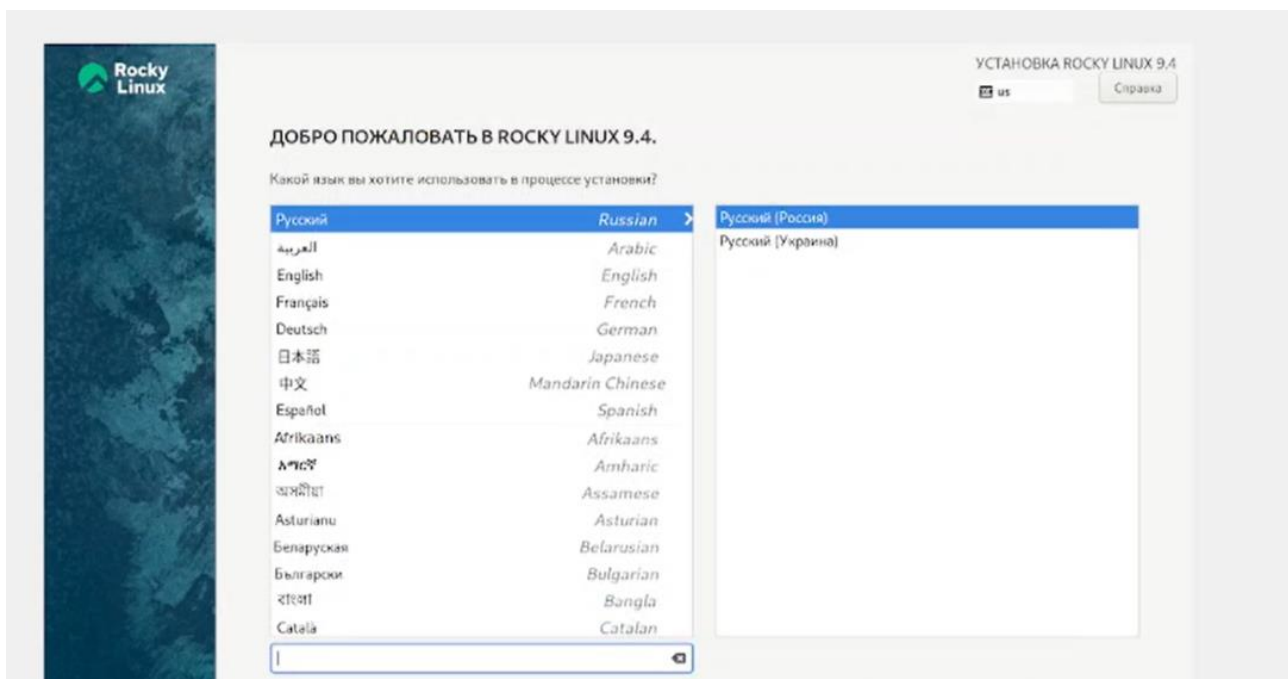


Меняем контроллер на скаченный образ Rocky: См. [рис. 6](#)

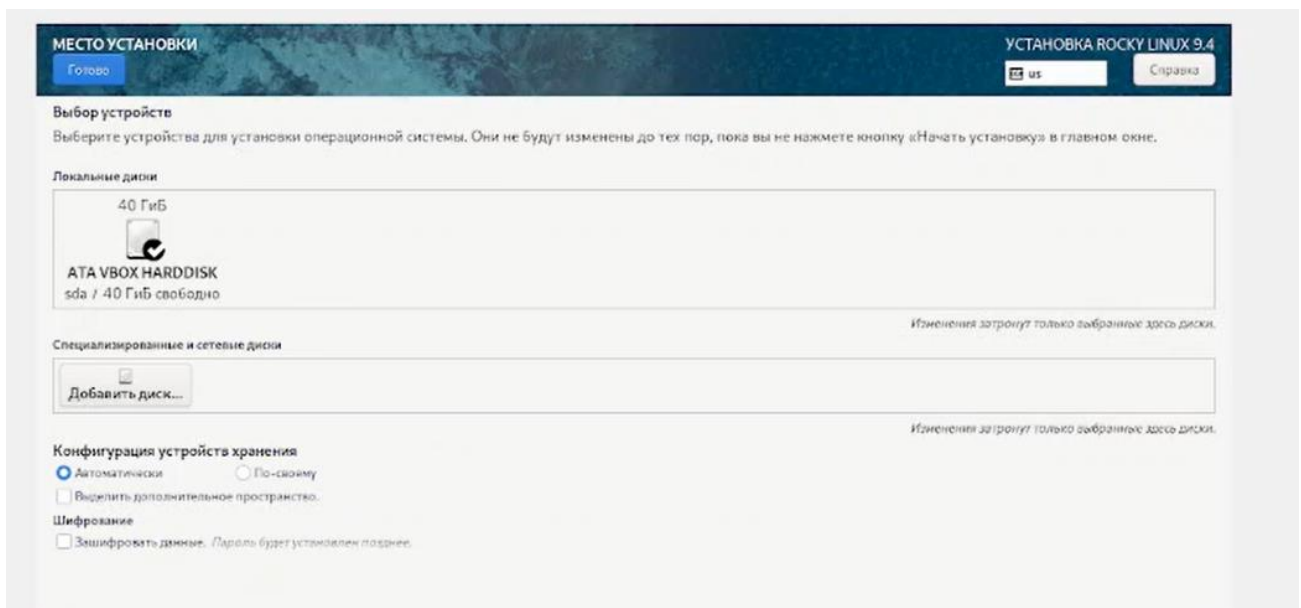
## Носители



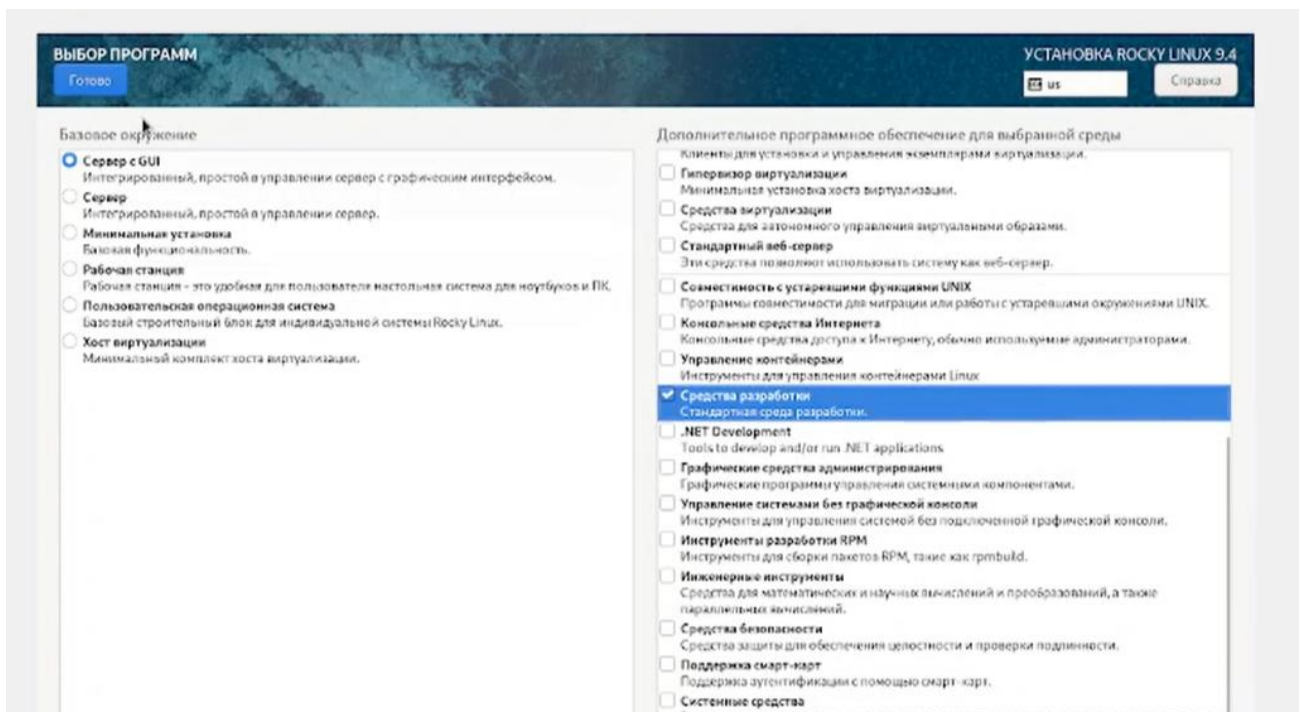
Попадаем в стартовое меню установки, выбираем английский язык: См. [рис. 7](#)



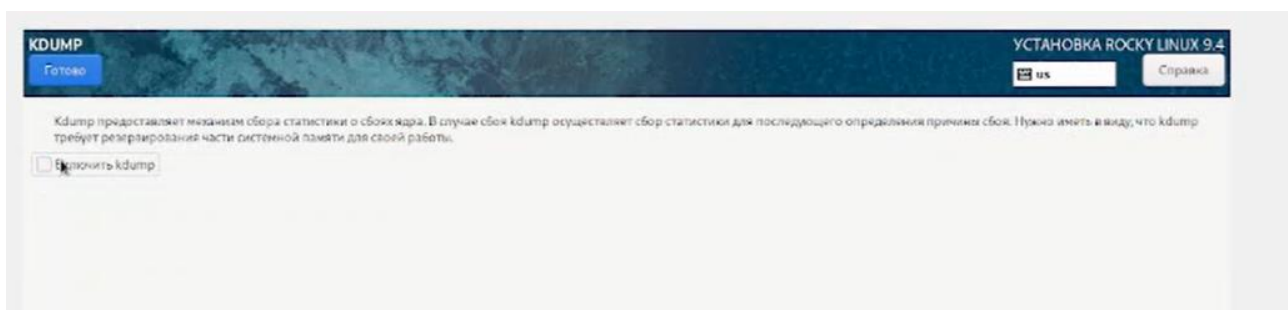
В Installation Destination выбираем диск: См. [рис. 8](#)



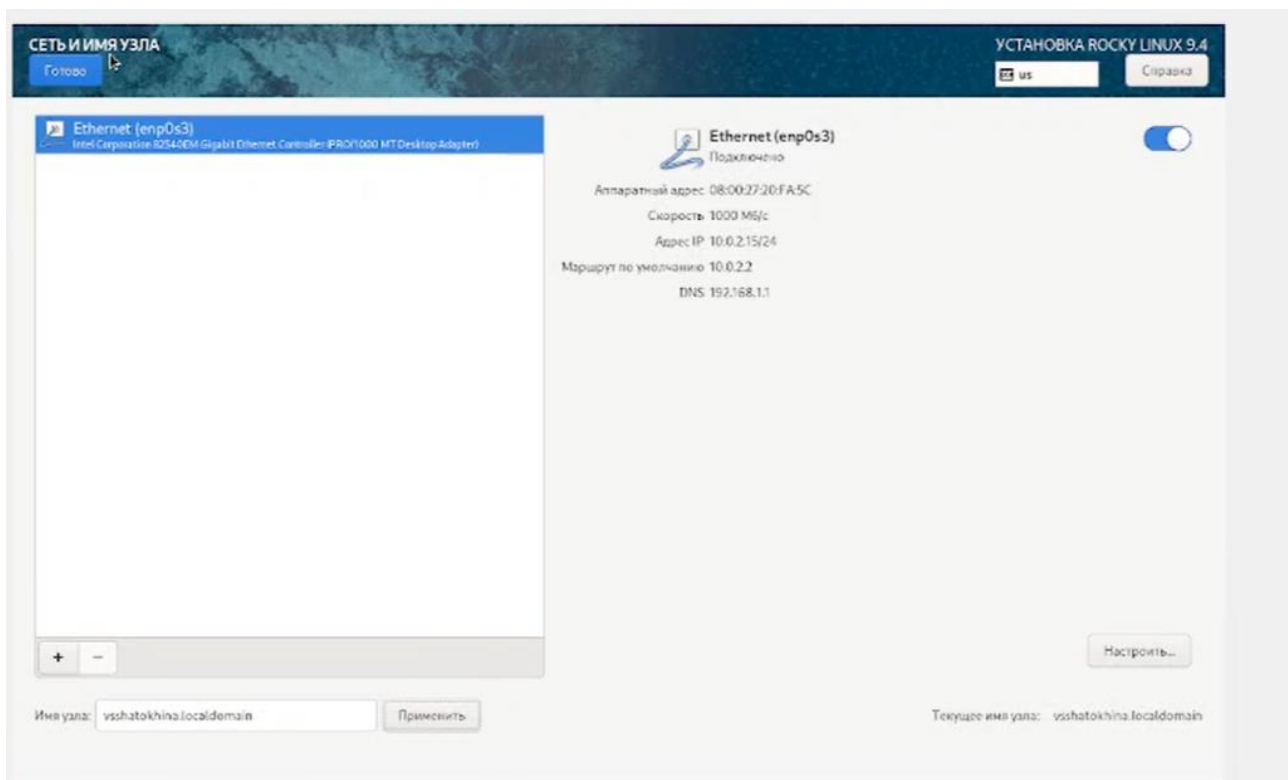
В Software Selection выбираем Server with GUI. В дополнительном ПО отмечаем Development Tools: См. [рис. 9](#)



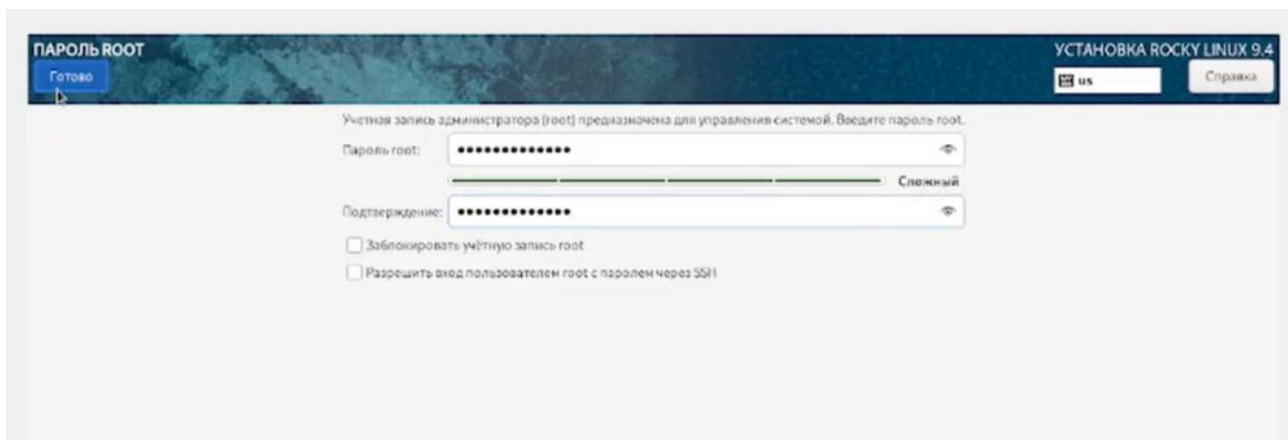
Заходим в KDUMP и отключаем его: См. [рис. 10](#)



Заходим в Network&Host Name и прописываем host name: См. [рис. 11](#)

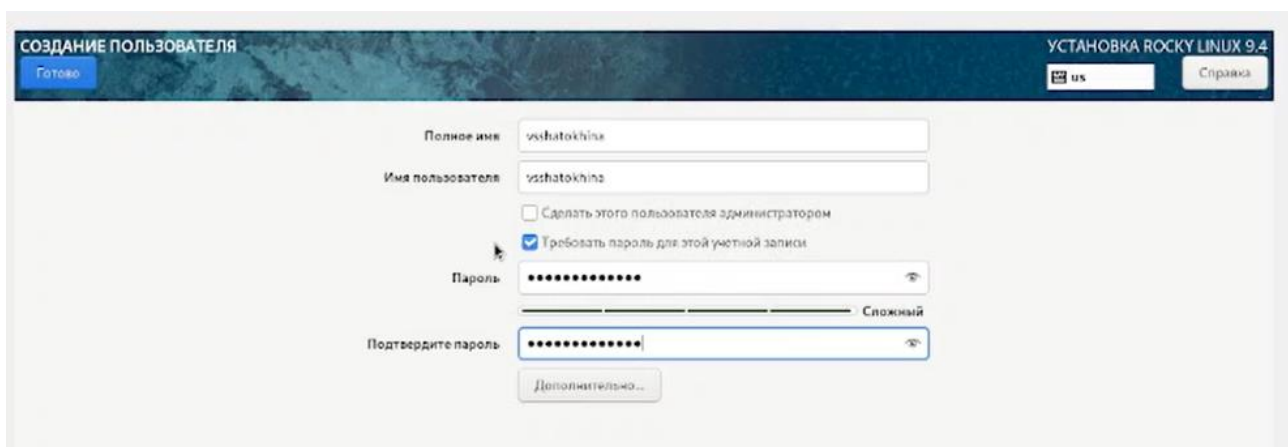


В разделе Root Password задаём пароль: См. [рис. 12](#)



Завершаем настройки во вкладке Create User: См. [рис. 13](#)





Создание пользователя

Готово

УСТАНОВКА ROCKY LINUX 9.4

us Справка

Полное имя: vshatokhina

Имя пользователя: vshatokhina

☐ Сделать этого пользователя администратором

☒ Требовать пароль для этой учетной записи

Пароль: [маскированный] Сложный

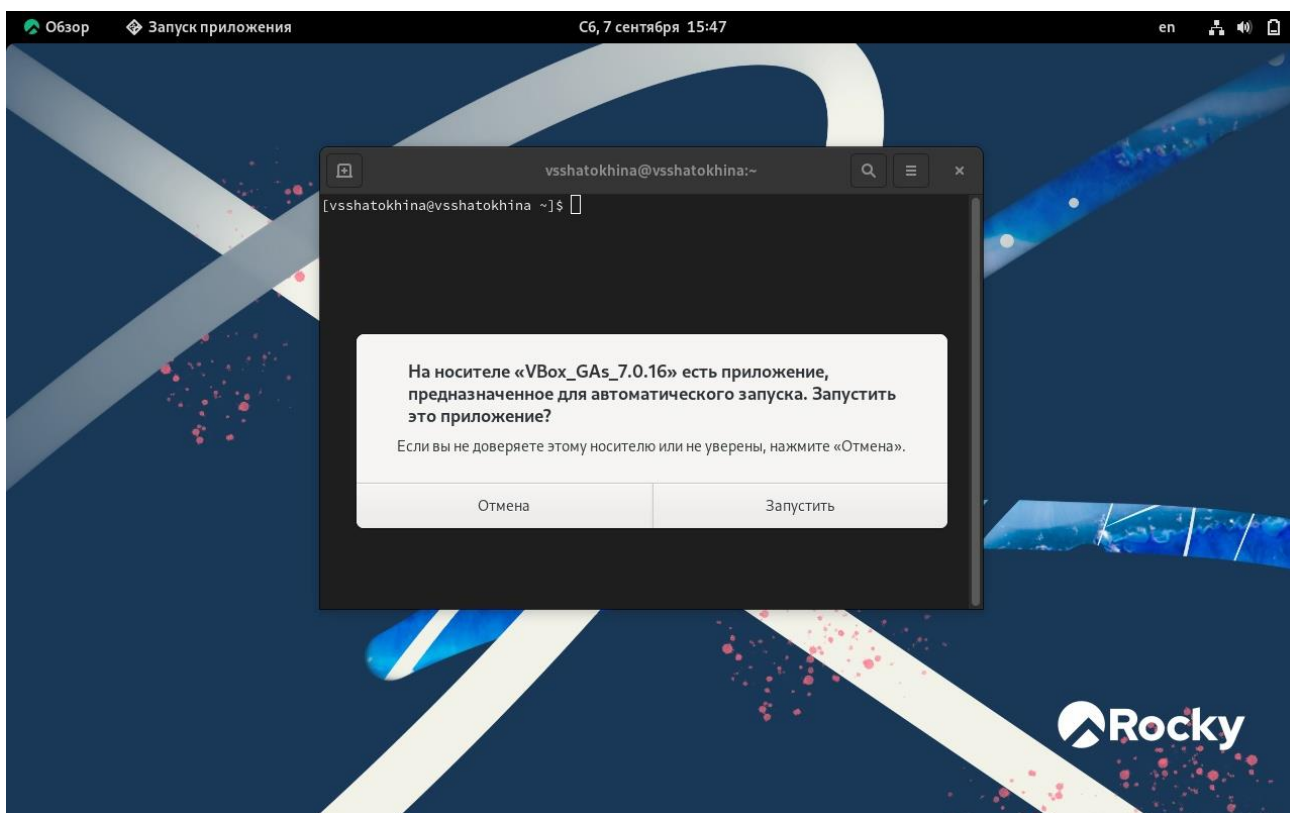
Подтвердите пароль: [маскированный]

Дополнительно...

Запускаем установку и ждем перезагрузки системы: См. [рис. 14](#)



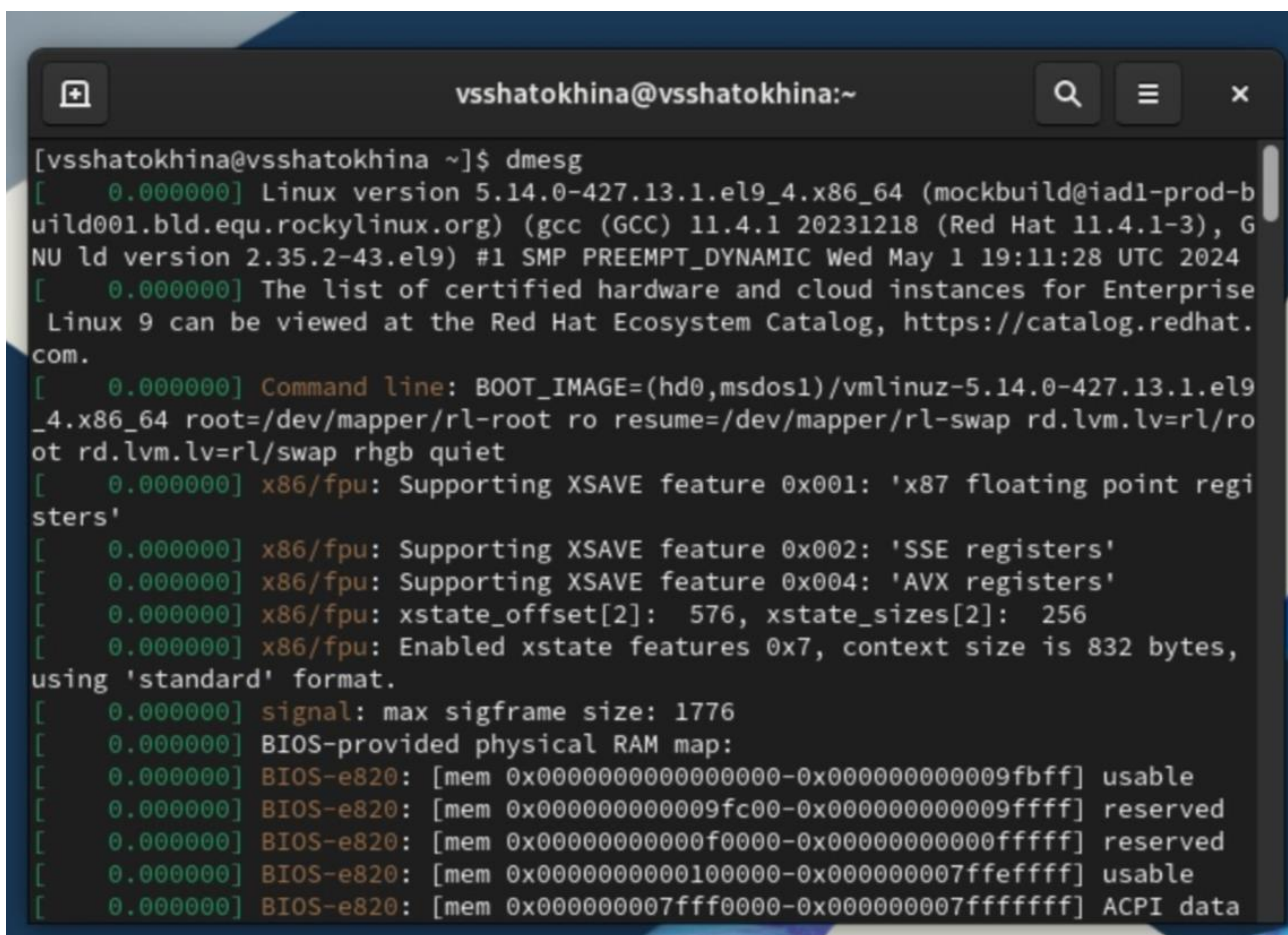
Запускаем образ диска дополнений гостевой ОС: См. [рис. 16](#)



## Домашнее задание

Посмотрим последовательность загрузки системы, выполнив команду `dmesg`: См. [рис. 17](#)

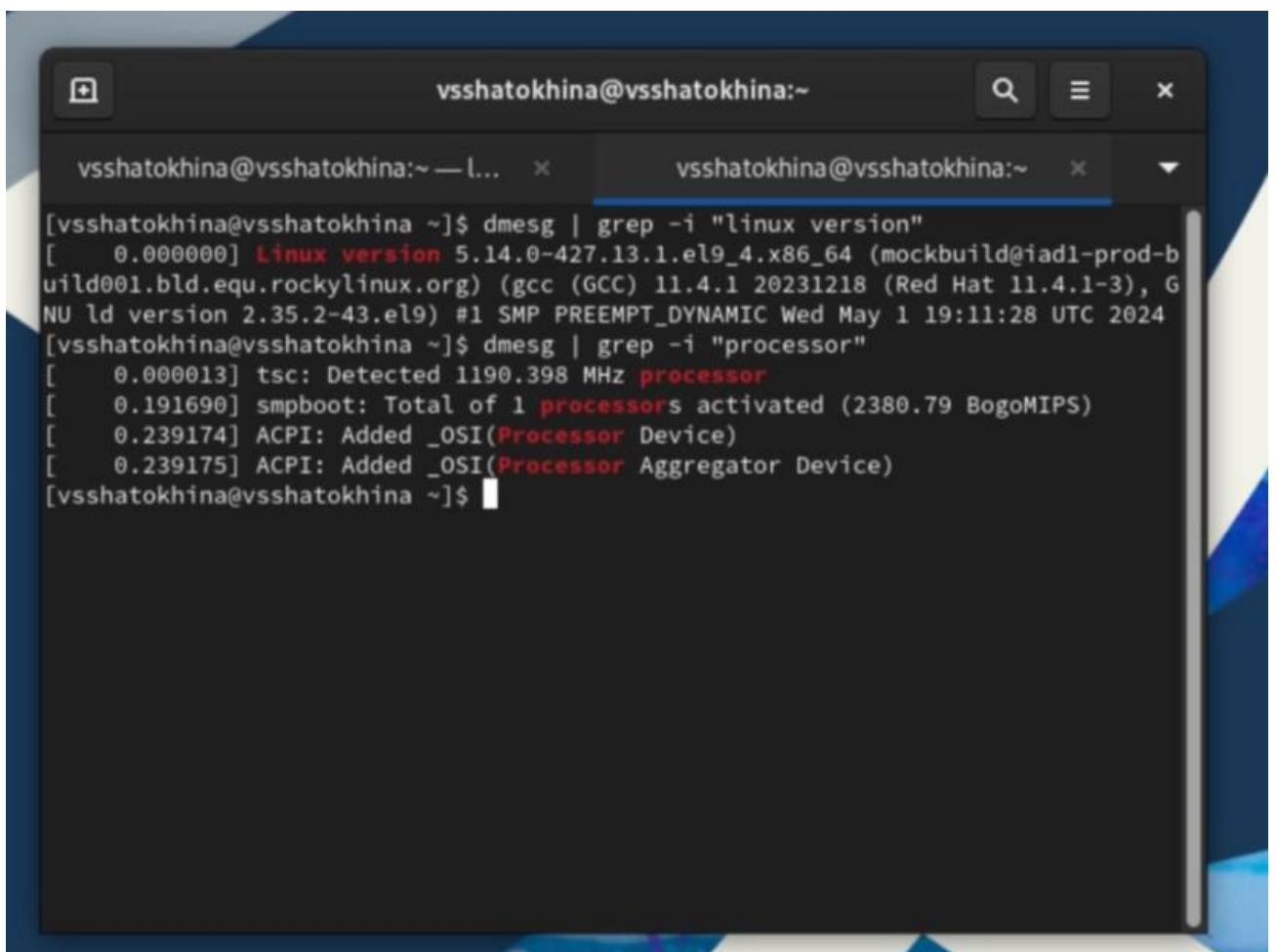


A terminal window titled 'vsshatokhina@vsshatokhina:~' with search, menu, and close icons. It displays the output of the 'dmesg' command. The output includes Linux version information (5.14.0-427.13.1.el9\_4.x86\_64), hardware details (gcc 11.4.1, GNU ld 2.35.2), a message about Red Hat Enterprise Linux 9 certification, the boot command line, x86/fpu XSAVE features (X87, SSE, AVX), signal frame size, and BIOS-provided physical RAM map details.

```
[vsshatokhina@vsshatokhina ~]$ dmesg
[    0.000000] Linux version 5.14.0-427.13.1.el9_4.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3), GNU ld version 2.35.2-43.el9) #1 SMP PREEMPT_DYNAMIC Wed May 1 19:11:28 UTC 2024
[    0.000000] The list of certified hardware and cloud instances for Enterprise Linux 9 can be viewed at the Red Hat Ecosystem Catalog, https://catalog.redhat.com.
[    0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.14.0-427.13.1.el9_4.x86_64 root=/dev/mapper/rl-root ro resume=/dev/mapper/rl-swap rd.lvm.lv=rl/root rd.lvm.lv=rl/swap 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_sizes[2]: 256
[    0.000000] x86/fpu: Enabled xstate features 0x7, context size is 832 bytes, using 'standard' format.
[    0.000000] signal: max sigframe size: 1776
[    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 0x0000000000100000-0x0000000007ffefffff] usable
[    0.000000] BIOS-e820: [mem 0x0000000007fff0000-0x0000000007ffffffffff] ACPI data
```

Получим следующую информацию: См. [рис. 18](#), См. [рис. 19](#), См. [рис. 20](#), См. [рис. 21](#)

1. Версия ядра Linux (Linux version).
2. Частота процессора (Detected Mhz processor).
3. Модель процессора (CPU0).
4. Объем доступной оперативной памяти (Memory available).
5. Тип обнаруженного гипервизора (Hypervisor detected).
6. Тип файловой системы корневого раздела.
7. Последовательность монтирования файловых систем



```
vsshatokhina@vsshatokhina:~  
vsshatokhina@vsshatokhina:~ — l... x vsshatokhina@vsshatokhina:~ x  
[vsshatokhina@vsshatokhina ~]$ dmesg | grep -i "linux version"  
[ 0.000000] Linux version 5.14.0-427.13.1.el9_4.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3), GNU ld version 2.35.2-43.el9) #1 SMP PREEMPT_DYNAMIC Wed May 1 19:11:28 UTC 2024  
[vsshatokhina@vsshatokhina ~]$ dmesg | grep -i "processor"  
[ 0.000013] tsc: Detected 1190.398 MHz processor  
[ 0.191690] smpboot: Total of 1 processors activated (2380.79 BogomIPS)  
[ 0.239174] ACPI: Added _OSI(Processor Device)  
[ 0.239175] ACPI: Added _OSI(Processor Aggregator Device)  
[vsshatokhina@vsshatokhina ~]$
```

```

vsshatokhina@vsshatokhina:~
vsshatokhina@vsshatokhina:~ — l...
vsshatokhina@vsshatokhina ~]$ dmesg | grep -i "memory"
[ 0.001846] ACPI: Reserving FACP table memory at [mem 0x7fff00f0-0x7fff01e3]
[ 0.001847] ACPI: Reserving DSDT table memory at [mem 0x7fff0610-0x7fff2962]
[ 0.001848] ACPI: Reserving FACS table memory at [mem 0x7fff0200-0x7fff023f]
[ 0.001849] ACPI: Reserving FACS table memory at [mem 0x7fff0200-0x7fff023f]
[ 0.001849] ACPI: Reserving APIC table memory at [mem 0x7fff0240-0x7fff0293]
[ 0.001850] ACPI: Reserving SSDT table memory at [mem 0x7fff02a0-0x7fff060b]
[ 0.002635] Early memory node ranges
[ 0.004042] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x0000
0fff]
[ 0.004044] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009
ffff]
[ 0.004045] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000e
ffff]
[ 0.004046] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000f
ffff]
[ 0.014634] Memory: 260860K/2096696K available (16384K kernel code, 5626K rwd
ata, 11748K rodata, 3892K init, 5956K bss, 145400K reserved, 0K cma-reserved)
[ 0.088612] Freeing SMP alternatives memory: 36K
[ 0.203499] x86/mm: Memory block size: 128MB
[ 0.564882] Non-volatile memory driver v1.3
[ 1.390570] Freeing initrd memory: 57168K
[ 1.722766] Freeing unused decrypted memory: 2028K
[ 1.728285] Freeing unused kernel image (initmem) memory: 3892K

```

```
vsshatokhina@vsshatokhina:~  
[vsshatokhina@vsshatokhina ~]$ dmesg | grep -i "memory available"  
[vsshatokhina@vsshatokhina ~]$ dmesg | grep -i "hypervisor "  
[    0.000000] Hypervisor detected: KVM  
[    0.068170] SRBDS: Unknown: Dependent on hypervisor status  
[    0.068171] GDS: Unknown: Dependent on hypervisor status  
[vsshatokhina@vsshatokhina ~]$ df -T  
Файловая система    Тип      1K-блоков  Использовано  Доступно  Использовано%  Смонт  
ровано в  
devtmpfs             devtmpfs    4096         0      4096         0% /dev/  
tmpfs                tmpfs      1007480         0  1007480         0% /dev/  
shm  
tmpfs                tmpfs      402992      8404   394588         3% /run  
/dev/mapper/rl-root  xfs      38678528  5962560 32715968        16% /  
/dev/sda1            xfs      983040    275748   707292        29% /boot  
tmpfs                tmpfs      201496      128   201368         1% /run/  
user/1000  
/dev/sr0             iso9660     52274     52274         0      100% /run/  
media/vsshatokhina/VBox_GAs_7.0.16  
[vsshatokhina@vsshatokhina ~]$ dmesg | grep -i "Mounted"  
[    7.874931] systemd[1]: Mounted Huge Pages File System.  
[    7.879376] systemd[1]: Mounted POSIX Message Queue File System.  
[    7.882764] systemd[1]: Mounted Kernel Debug File System.  
[    7.887254] systemd[1]: Mounted Kernel Trace File System.  
[vsshatokhina@vsshatokhina ~]$
```

## Заключение

Приобрели практические навыки установки операционной системы на виртуальную машину, настроили минимально необходимые для дальнейшей работы сервисы.

## Ответы на вопросы

1. Какую информацию содержит учётная запись пользователя?

Учётная запись пользователя в Linux содержит следующую информацию:

- Имя пользователя (username) — уникальное имя пользователя в системе.
- Идентификатор пользователя (UID) — уникальный числовой идентификатор для каждого пользователя.
- Идентификатор группы (GID) — идентификатор основной группы, к которой принадлежит пользователь.
- Домашний каталог (home directory) — директория, в которой пользователь хранит свои файлы и настройки.

- Интерпретатор команд (shell) — программа, запускаемая по умолчанию при входе пользователя в систему.
- Пароль пользователя — обычно хранится в хешированном виде в файле /etc/shadow.

Эти данные обычно содержатся в файле /etc/passwd, а зашифрованные пароли — в файле /etc/shadow.

2. Укажите команды терминала и приведите примеры: – для получения справки по команде;

– для перемещения по файловой системе;

– для просмотра содержимого каталога;

– для определения объёма каталога;

– для создания / удаления каталогов / файлов;

– для задания определённых прав на файл / каталог;

– для просмотра истории команд.

Для получения справки по команде: `man` Пример: `man ls` — получить справку по команде `ls`.

Для перемещения по файловой системе: `cd` Пример: `cd /home/user` — перейти в каталог /home/user.

Для просмотра содержимого каталога: `ls` [опции] Пример: `ls -la /home/user` — показать все файлы и каталоги, включая скрытые, с подробной информацией.

Для определения объёма каталога: `du -sh` Пример: `du -sh /home/user` — показать общий размер каталога /home/user.

Для создания / удаления каталогов / файлов:

`mkdir` - Создание каталога

`rmdir` - Удаление пустого каталога

`touch` - Создание пустого файла

`rm` - Удаление файла

`rm -r` - Рекурсивное удаление каталога и его содержимого

Примеры:

```
mkdir new_folder
```

```
touch new_file.txt
```

```
rm new_file.txt
```

```
rm -r new_folder
```

Для задания определённых прав на файл / каталог: `chmod`

Пример: `chmod 755 script.sh` — установить права `rwxr-xr-x` на файл `script.sh`.

Для просмотра истории команд: `history`

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

Файловая система — это метод и структура, по которым данные хранятся, организуются и управляются на носителе информации (жесткий диск, SSD, USB-накопитель и т.д.).

Примеры файловых систем:

**EXT4 (Fourth Extended Filesystem):** Одна из самых популярных файловых систем в Linux. Поддерживает журналирование, большие объемы данных, улучшенную производительность. Хорошо подходит для большинства стандартных Linux-установок.

**NTFS (New Technology File System):** Файловая система, используемая в операционных системах Windows. Поддерживает большие файлы, разрешения, шифрование и сжатие.

**FAT32 (File Allocation Table 32):** Универсальная файловая система, поддерживаемая практически всеми операционными системами. Ограничение на размер файла — до 4 ГБ.

**XFS:** Журналируемая файловая система с высокой производительностью, разработанная для систем с большими объемами данных. Хорошо подходит для серверных систем и больших файловых хранилищ.

**ZFS:** Передовая файловая система, поддерживающая большой объем данных, снапшоты, клонирование и защиту данных. Разработана для высоконадежных систем.

### 4. Как посмотреть, какие файловые системы подмонтированы в ОС?

Чтобы посмотреть, какие файловые системы подмонтированы в операционной системе Linux, можно воспользоваться несколькими способами. Во-первых, можно использовать команду `mount`, которая выводит список всех текущих монтированных файловых систем, их устройства, точки монтирования, типы файловых систем и параметры монтирования. Например, при вводе команды `mount` в терминале вы получите информацию о том, какие файловые системы были смонтированы и в каком порядке.

Во-вторых, можно просмотреть содержимое файла `/proc/mounts`, который также содержит сведения о всех монтированных файловых системах, включая псевдо-файловые системы вроде `proc` и `sysfs`. Это можно сделать, выполнив команду `cat /proc/mounts` в терминале. Наконец, команду `df -h` можно использовать для отображения информации о свободном и занятом пространстве на смонтированных файловых системах, что может быть полезным для мониторинга состояния системы.

### 5. Как удалить зависший процесс?



Для удаления зависшего процесса в Linux сначала нужно определить его идентификатор (PID). Это можно сделать с помощью команды `ps aux`, которая выводит список всех запущенных процессов в системе вместе с их PID, или с помощью утилит `top` или `htop`, которые предоставляют интерактивный список процессов.

После того как PID зависшего процесса известен, можно использовать команду `kill` для отправки сигнала завершения процессу. Если процесс не реагирует на обычный сигнал завершения, его можно принудительно завершить, используя команду `kill -9`. Этот сигнал (-9) немедленно завершает процесс, игнорируя любые попытки его сохранения или корректного завершения.

## Библиографическая справка

[1] Документация по VirtualBox: <https://www.virtualbox.org/>