

# **Лабораторная работа №1**

**Основы информационной безопасности**

Тойчубекова Асель Нурлановна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Установка имени пользователя и названия хоста . . . . .	16
4.2	Выполнение домашнего задания . . . . .	16
<b>5</b>	<b>Выводы</b>	<b>20</b>
	<b>Список литературы</b>	<b>21</b>

# Список иллюстраций

4.1	Имя и операционная система виртуальной машины . . . . .	8
4.2	Оборудование . . . . .	9
4.3	Виртуальный жесткий диск . . . . .	9
4.4	Подключение образа оптического диска . . . . .	10
4.5	Запуск установки виртуальной машины . . . . .	10
4.6	Выбор языка интерфейса . . . . .	11
4.7	Раскладка клавиатуры . . . . .	11
4.8	Выбор программ . . . . .	12
4.9	Место установки ОС . . . . .	12
4.10	KDUMP . . . . .	13
4.11	Сетевое соединение . . . . .	13
4.12	Пароль root . . . . .	14
4.13	Пользователь и пароль . . . . .	14
4.14	Начало установки . . . . .	15
4.15	Подключение образа диска гостевой ОС . . . . .	15
4.16	Проверка имени пользователя . . . . .	16
4.17	Проверка имени хоста . . . . .	16
4.18	Загрузка системы . . . . .	17
4.19	Версия ядра Linux . . . . .	17
4.20	Частота процессора . . . . .	18
4.21	Модель процессора . . . . .	18
4.22	Объем доступной оперативной памяти . . . . .	18
4.23	Тип обнаруженного гипервизора . . . . .	19
4.24	Тип файловой системы корневого раздел . . . . .	19
4.25	Последовательность монтирования файловых систем . . . . .	19

## **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

- Установить на виртуальную машину VirtualBox операционную систему Linux (дистрибутив Rocky)
- Выполнить домашнее задание

## 3 Теоретическое введение

Linux — это семейство операционных систем на базе ядра Linux, известных своей стабильностью, безопасностью и открытым исходным кодом. Эти системы широко используются в серверных, облачных и корпоративных средах благодаря гибкости настройки и поддержке множества архитектур. В основе Linux лежит философия модульности и свободы выбора, позволяя пользователям адаптировать систему под свои нужды.

Rocky Linux — это один из дистрибутивов Linux, созданный как замена CentOS после его перехода на модель CentOS Stream. Разработанный сообществом, он ориентирован на стабильность и совместимость с Red Hat Enterprise Linux (RHEL), что делает его популярным в корпоративной среде. Rocky Linux предлагает длительную поддержку, надежность и предсказуемые обновления, что делает его отличным выбором для серверов и облачных решений.

## 4 Выполнение лабораторной работы

Для начала скачиваем DVD-образ операционной системы, соответствующий архитектуре нашего компьютера с сайта разработчика <https://rockylinux.org/download>. Затем создаем новую виртуальную машину, указываем имя, тип операционной системы — Linux, версию операционной системы — RedHat (64-bit) (рис. 4.1).

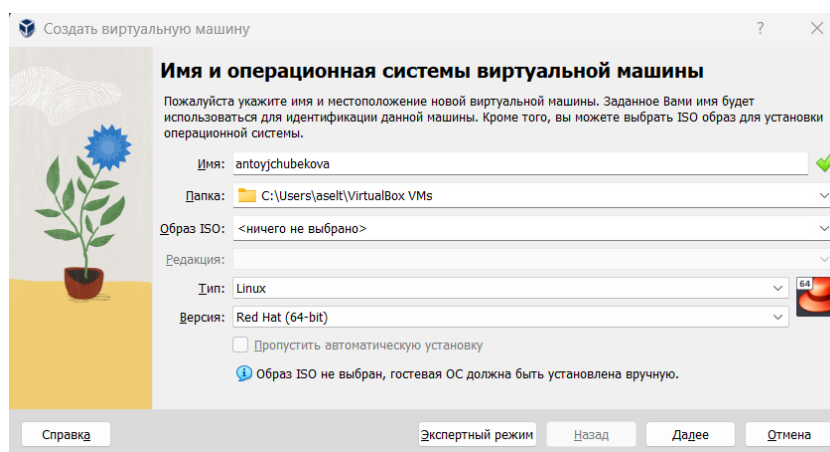


Рис. 4.1: Имя и операционная система виртуальной машины

Указываем размер основной памяти виртуальной машины -2048 МБ и число процессоров-4. (рис. 4.2).



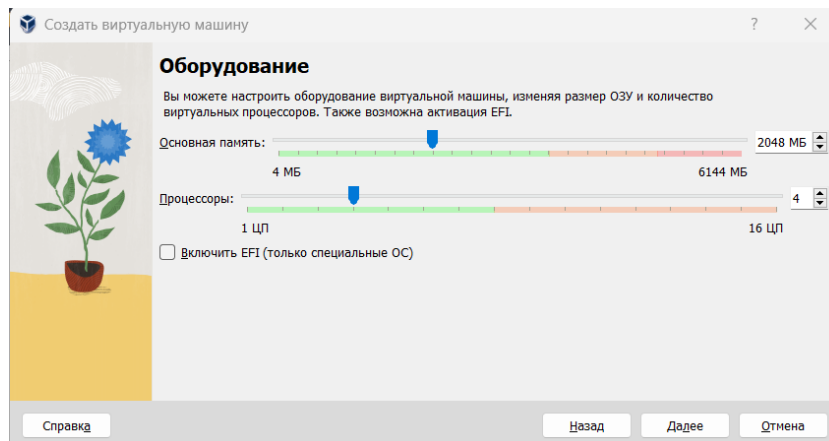


Рис. 4.2: Оборудование

Задаем размер виртуального жесткого диска-40ГБ. (рис. 4.3).

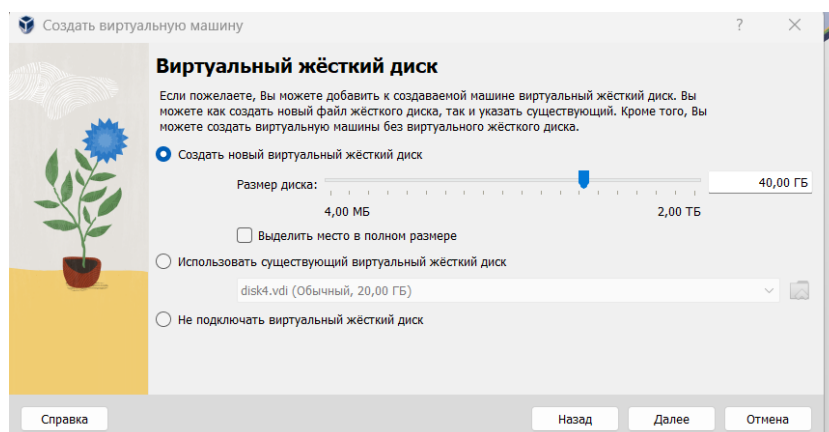


Рис. 4.3: Виртуальный жесткий диск

Заходим в настройки и добавляем новый привод оптических дисков и выбираем наш скаченный образ операционной системы. (рис. 4.4).

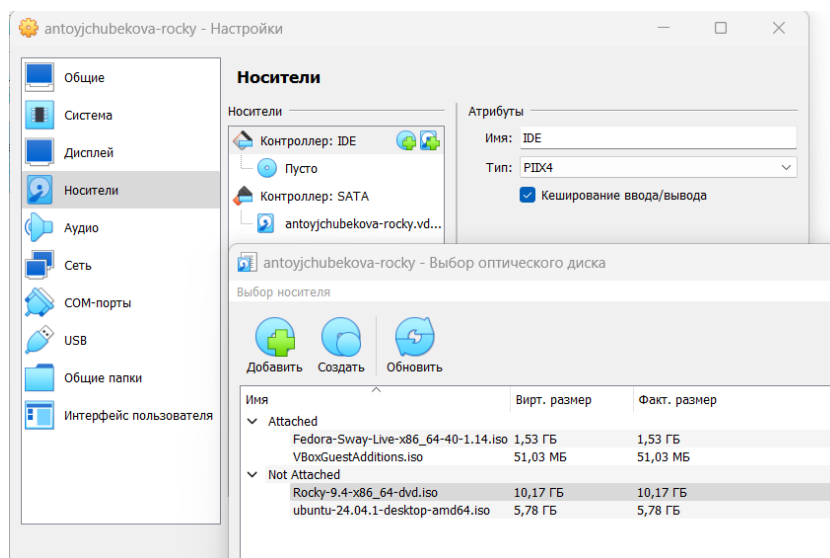


Рис. 4.4: Подключение образа оптического диска

Запустим виртуальную машину и в окне с меню переключимся на строку «Install Rocky Linux версия» и нажмем на Enter для запуска установки образа ОС. (рис. 4.5).

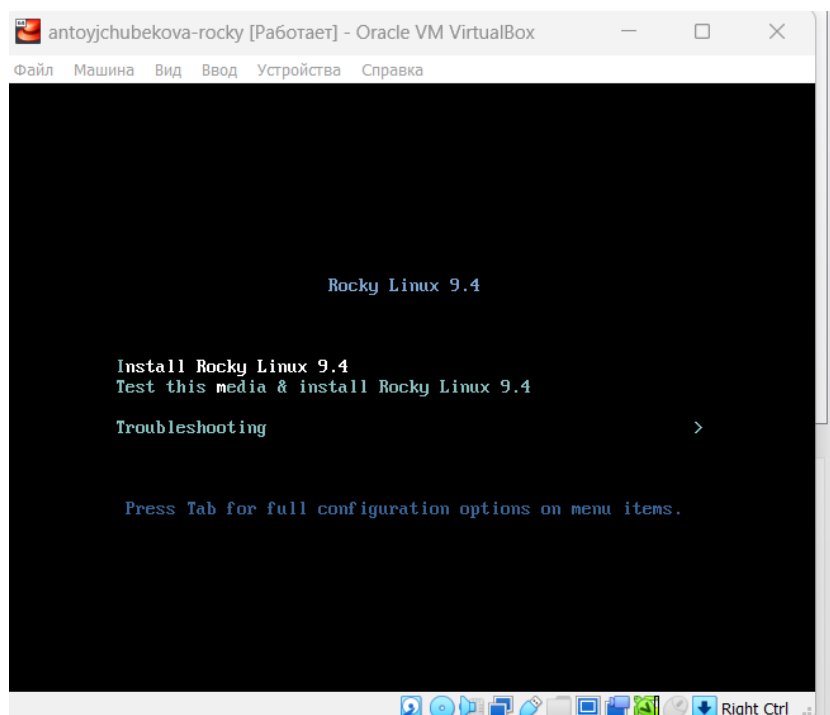


Рис. 4.5: Запуск установки виртуальной машины

Выберем English в качестве языка интерфейса и перейдем к настройкам установки операционной системы. (рис. 4.6).

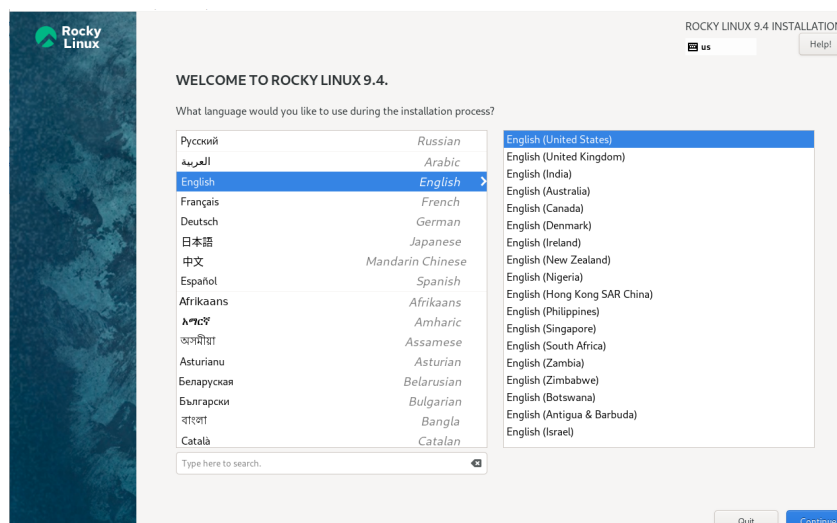


Рис. 4.6: Выбор языка интерфейса

Добавим в раскладку клавиатуры русский язык. (рис. 4.7).

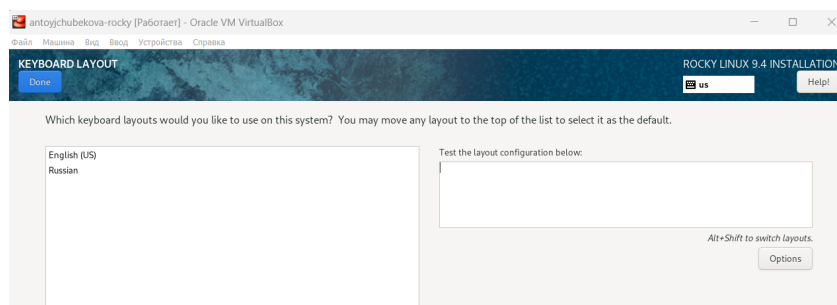


Рис. 4.7: Раскладка клавиатуры

В разделе выбора программ укажем в качестве базового окружения Server with GUI , а в качестве дополнения-Development Tools. (рис. 4.8).

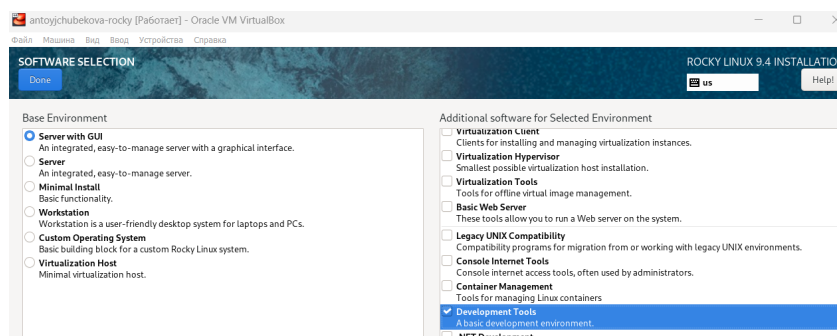


Рис. 4.8: Выбор программ

Место установки ОС оставляем без изменений. (рис. 4.9).

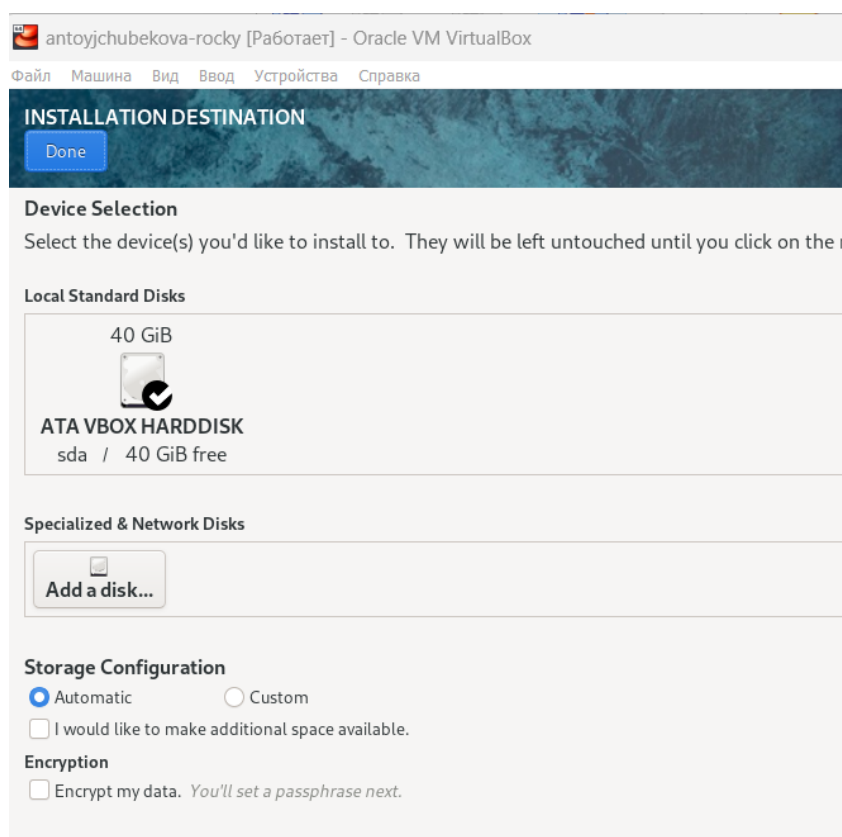


Рис. 4.9: Место установки ОС

Отключим KDUMP. (рис. 4.10).

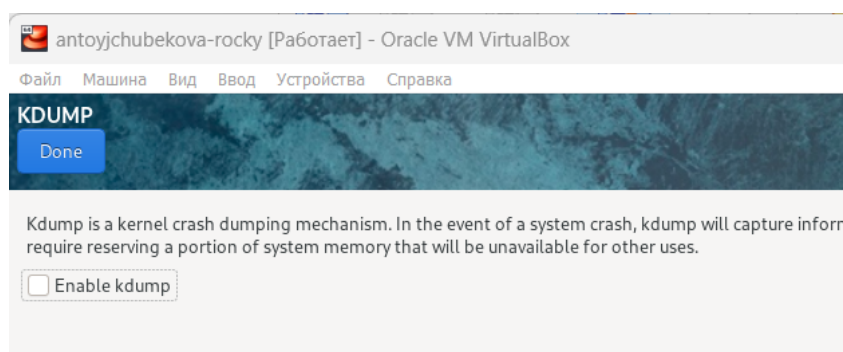


Рис. 4.10: KDUMP

Включим сетевое соединение и в качестве имени узла укажем antoyjchubekova.localdomain.  
(рис. 4.11).

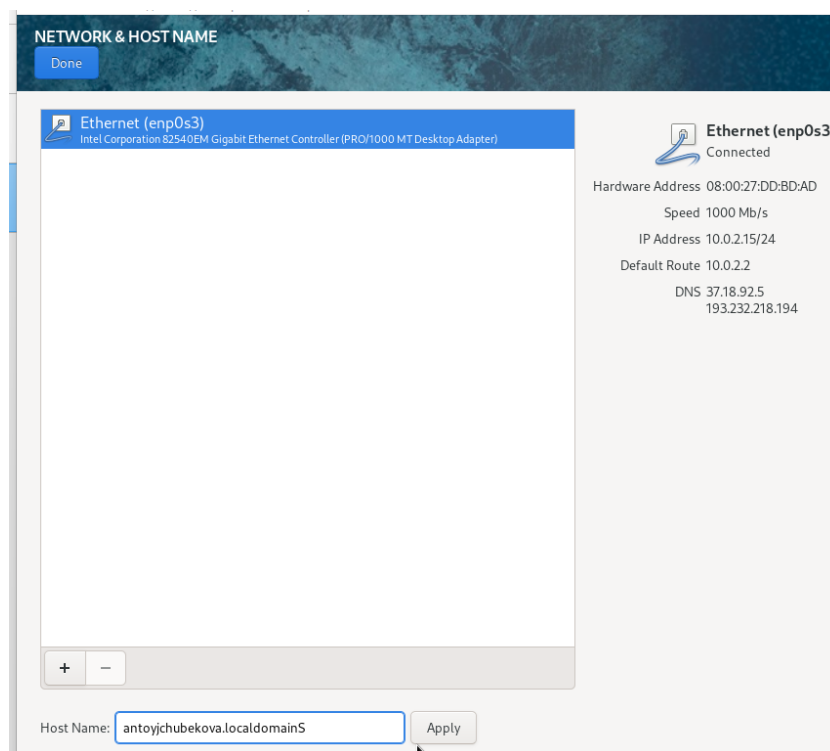


Рис. 4.11: Сетевое соединение

Указываем пароль root, разрешение на ввод пароля для root при использовании  
ssh. (рис. 4.12).

Рис. 4.12: Пароль root

Затем зададим локального пользователя с правами администратора и пароль для него. (рис. 4.13).

Рис. 4.13: Пользователь и пароль

После задания необходимых настроек нажмем на Begin Installation для начала установки образа системы. (рис. 4.14).

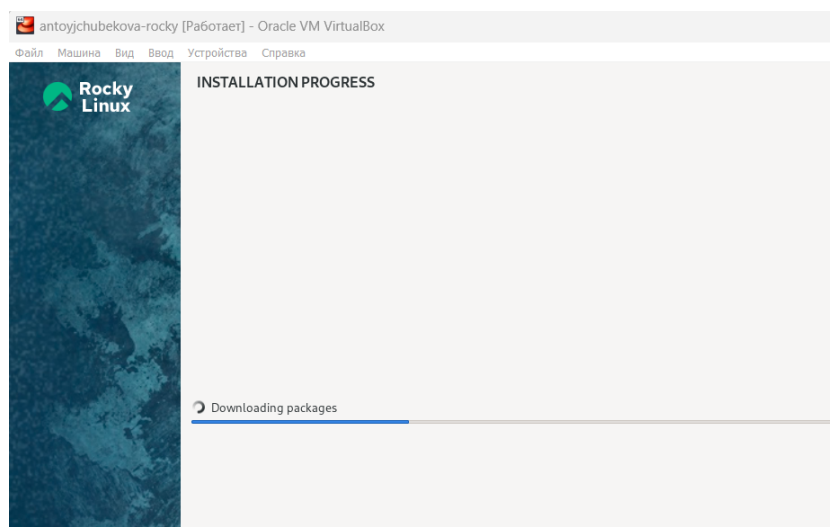


Рис. 4.14: Начало установки

После завершения установки войдем в ОС под заданной учетной записью. В меню Устройства виртуальной машины подключаем образ диска дополнительной гостевой ОС и запустим его. Затем корректно перезагрузим виртуальную машину. (рис. 4.15).

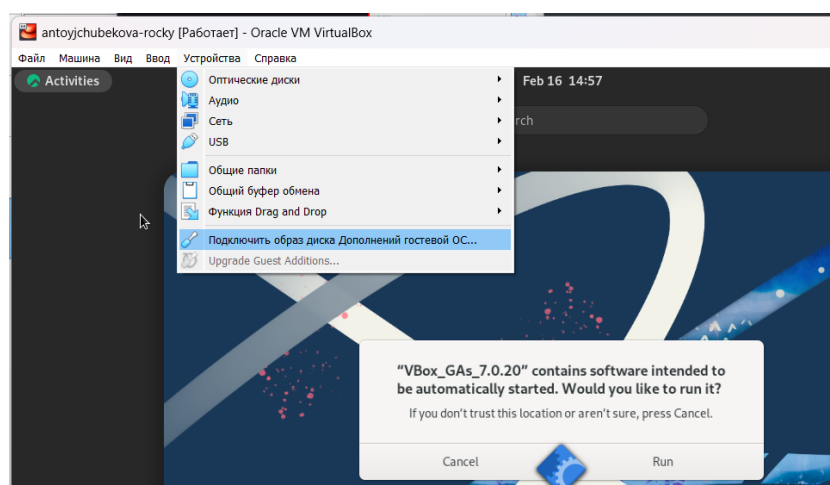


Рис. 4.15: Подключение образа диска гостевой ОС

## 4.1 Установка имени пользователя и названия хоста

Дальше мы можем установить имя пользователя и название хоста, чтобы она соответствовала нашему логину в дк. Для создание пользователя используется команда `adduser -G wheel username`, а для его пароля `passwd username`. С помощью команды `id -un` я проверила имя пользователя, она соответствовала логину (рис. 4.16), также с помощью команды `hostnamectl` проверила имя хоста, который также был корректным. (рис. 4.17).

```
[antoyjchubekova@antoyjchubekova ~]$ id -un
antoyjchubekova
[antoyjchubekova@antoyjchubekova ~]$
```

Рис. 4.16: Проверка имени пользователя

```
root@antoyjchubekova:~
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for antoyjchubekova:
[root@antoyjchubekova ~]# hostnamectl
Static hostname: antoyjchubekova.localdomains
Icon name: computer-vm
Chassis: vm
Machine ID: 9a85431ff0834b0b9f2b3ad05d545a5c
Boot ID: 12718d4d0d7647d6b4a43d66d2c1447c
Virtualization: oracle
Operating System: Rocky Linux 9.4 (Blue Onyx)
CPE OS Name: cpe:/o:rocky:rocky:9::baseos
Kernel: Linux 5.14.0-427.13.1.el9_4.x86_64
Architecture: x86-64
Hardware Vendor: innotek GmbH
Hardware Model: VirtualBox
Firmware Version: VirtualBox
```

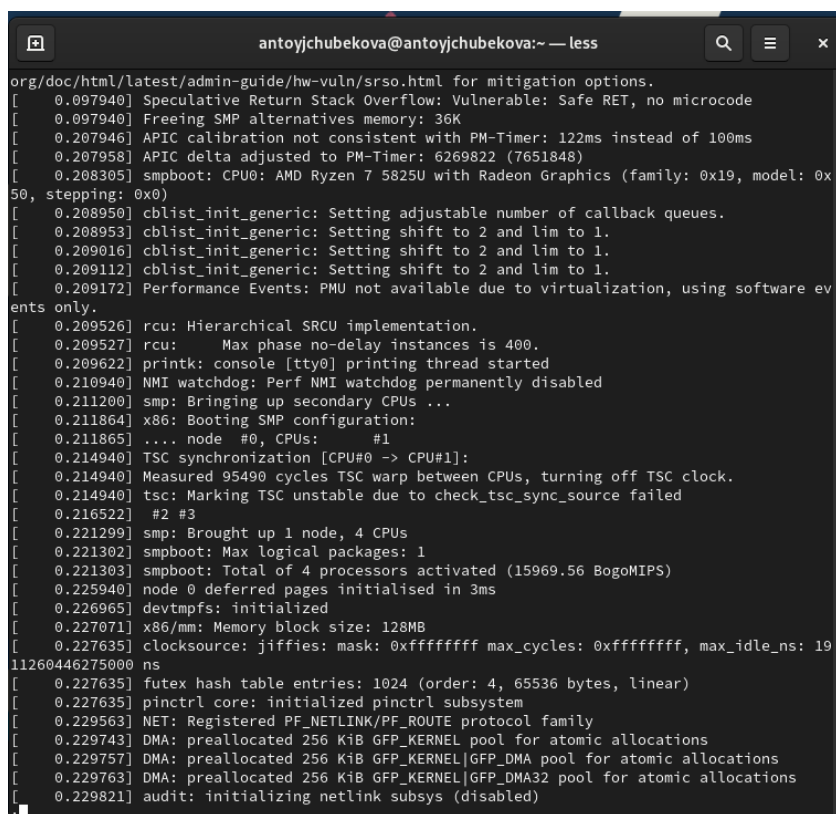
Рис. 4.17: Проверка имени хоста

## 4.2 Выполнение домашнего задания

С помощью команды `dmesg | less` выведем системные сообщения ядра, такие как информация о загрузке системы, подключении устройств, драйверах и ошибках.



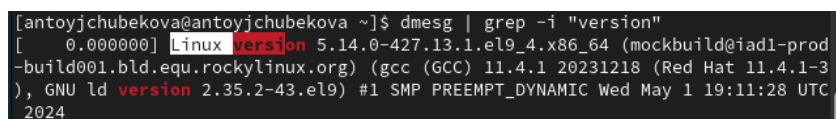
(рис. 4.18).



```
org/doc/html/latest/admin-guide/hw-vuln/srso.html for mitigation options.
[ 0.097940] Speculative Return Stack Overflow: Vulnerable: Safe RET, no microcode
[ 0.097940] Freeing SMP alternatives memory: 36K
[ 0.207946] APIC calibration not consistent with PM-Timer: 122ms instead of 100ms
[ 0.207958] APIC delta adjusted to PM-Timer: 6269822 (7651848)
[ 0.208305] smpboot: CPU0: AMD Ryzen 7 5825U with Radeon Graphics (family: 0x19, model: 0x50, stepping: 0x0)
[ 0.208950] cblist_init_generic: Setting adjustable number of callback queues.
[ 0.208953] cblist_init_generic: Setting shift to 2 and lim to 1.
[ 0.209016] cblist_init_generic: Setting shift to 2 and lim to 1.
[ 0.209112] cblist_init_generic: Setting shift to 2 and lim to 1.
[ 0.209172] Performance Events: PMU not available due to virtualization, using software events only.
[ 0.209526] rcu: Hierarchical SRCU implementation.
[ 0.209527] rcu: Max phase no-delay instances is 400.
[ 0.209622] printk: console [tty0] printing thread started
[ 0.210940] NMI watchdog: Perf NMI watchdog permanently disabled
[ 0.211200] smp: Bringing up secondary CPUs ...
[ 0.211864] x86: Booting SMP configuration:
[ 0.211865] .... node #0, CPUs: #1
[ 0.214940] TSC synchronization [CPU#0 -> CPU#1]:
[ 0.214940] Measured 95490 cycles TSC warp between CPUs, turning off TSC clock.
[ 0.214940] tsc: Marking TSC unstable due to check_tsc_sync_source failed
[ 0.216522] #2 #3
[ 0.221299] smp: Brought up 1 node, 4 CPUs
[ 0.221302] smpboot: Max logical packages: 1
[ 0.221303] smpboot: Total of 4 processors activated (15969.56 BogoMIPS)
[ 0.225940] node 0 deferred pages initialised in 3ms
[ 0.226965] devtmpfs: initialized
[ 0.227071] x86/mm: Memory block size: 128MB
[ 0.227635] clocksource: jiffies: mask: 0xffffffff max_cycles: 0xffffffff, max_idle_ns: 1911260446275000 ns
[ 0.227635] futex hash table entries: 1024 (order: 4, 65536 bytes, linear)
[ 0.227635] pinctrl core: initialized pinctrl subsystem
[ 0.229563] NET: Registered PF_NETLINK/PF_ROUTE protocol family
[ 0.229743] DMA: preallocated 256 KiB GFP_KERNEL pool for atomic allocations
[ 0.229757] DMA: preallocated 256 KiB GFP_KERNEL|GFP_DMA pool for atomic allocations
[ 0.229763] DMA: preallocated 256 KiB GFP_KERNEL|GFP_DMA32 pool for atomic allocations
[ 0.229821] audit: initializing netlink subsys (disabled)
```

Рис. 4.18: Загрузка системы

1. С помощью команды `dmesg | grep -i "version"` найдем версию ядра Linux. Мы видим, что версия - Linux version 5.14.0-427.13.1.el9\_4.x86\_64. (рис. 4.19).



```
[antoyjchubekova@antoyjchubekova ~]$ dmesg | grep -i "version"
[ 0.000000] Linux version 5.14.0-427.13.1.el9_4.x86_64 (mockbuild@iad1-prod-bu1d001.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
```

Рис. 4.19: Версия ядра Linux

2. С помощью той же команды `dmesg | grep -i "Mhz processor"` посмотрим частоту процессора. Мы видим, что частота процессора - 1996.195 Mhz. (рис. 4.20).

```
[antoyjchubekova@antoyjchubekova ~]$ dmesg | grep -i "Mhz processor"
[ 0.000011] tsc: Detected 1996.195 MHz processor
[antoyjchubekova@antoyjchubekova ~]$
```

Рис. 4.20: Частота процессора

3. С помощью команды `dmesg | grep -i "CPU0"` посмотрим модель процессора. Мы видим что, модель процессора-CPU: AMD Ryzen 7 5825U with Radeon Graphics. (рис. 4.21).

```
[antoyjchubekova@antoyjchubekova ~]$ dmesg | grep -i "CPU0"
[ 0.208305] smpboot: CPU0: AMD Ryzen 7 5825U with Radeon Graphics (family:
0x19, model: 0x50, stepping: 0x0)
[antoyjchubekova@antoyjchubekova ~]$
```

Рис. 4.21: Модель процессора

4. С помощью команды `dmesg | grep -i "available"` посмотрим объем доступной оперативной памяти. Мы видим что, объем доступной оперативной памяти-из 2096696K ( $\approx 2$  ГБ) оперативной памяти 260860K ( $\approx 255$  МБ) доступны. Остальное занято системой драйверами и процессорами (рис. 4.22).

```
[antoyjchubekova@antoyjchubekova ~]$ dmesg | grep -i "available"
[ 0.002327] On node 0, zone DMA: 1 pages in unavailable ranges
[ 0.002855] On node 0, zone DMA: 97 pages in unavailable ranges
[ 0.005398] On node 0, zone DMA32: 16 pages in unavailable ranges
[ 0.006847] [mem 0x80000000-0xfebfffff] available for PCI devices
[ 0.032651] Memory: 260860K/2096696K available (16384K kernel code, 5626K r
wdata, 11748K rodata, 3892K init, 5956K bss, 144136K reserved, 0K cma-reserved
)
```

Рис. 4.22: Объем доступной оперативной памяти

5. С помощью команды `dmesg | grep -i "Hypervisor detected"` тип обнаруженного гипервизора. Гипервизор (или виртуализатор) — это программное обеспечение, которое позволяет создавать и управлять виртуальными машинами (VM). Он работает как слой между аппаратным обеспечением и операционными системами, предоставляя виртуальные ресурсы для каждой из них. Мы видим, что у нас гипервизор 1 типа KVM (рис. 4.23).

```
[antoychubekova@antoychubekova ~]$ dmesg | grep -i "Hypervisor"
[ 0.000000] Hypervisor detected: KVM
[ 2.682671] vmwgfx 0000:00:02.0: [drm] *ERROR* vmwgfx seems to be running on an unsupported hypervisor.
[antoychubekova@antoychubekova ~]$
```

Рис. 4.23: Тип обнаруженного гипервизора

6. Далее посмотрим тип файловой системы корневого раздела с командой `dmesg | grep -i "filesystem"`. Мы видим, что тип файловой системы-Mounting V5 filesystem. (рис. 4.24).

```
[antoychubekova@antoychubekova ~]$ dmesg | grep -i "filesystem"
[ 4.887291] XFS (dm-0): Mounting V5 Filesystem a56f0ca0-69a6-47ad-8adb-ce243810d58e
[ 8.616411] XFS (sda1): Mounting V5 Filesystem 2bd1339c-6fe6-4e86-a110-9d0f521542c9
```

Рис. 4.24: Тип файловой системы корневого раздел

7. С помощью команды `mesg | grep -i "mount"` посмотрим последовательность монтирования файловых систем. (рис. 4.25).

```
[antoychubekova@antoychubekova ~]$ dmesg | grep -i "mount"
[ 0.097230] Mount-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[ 0.097307] Mountpoint-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[ 4.887291] XFS (dm-0): Mounting V5 Filesystem a56f0ca0-69a6-47ad-8adb-ce243810d58e
[ 7.074563] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 7.116730] systemd[1]: Mounting Huge Pages File System...
[ 7.119668] systemd[1]: Mounting POSIX Message Queue File System...
[ 7.122965] systemd[1]: Mounting Kernel Debug File System...
[ 7.127305] systemd[1]: Mounting Kernel Trace File System...
[ 7.244178] systemd[1]: Starting Remount Root and Kernel File Systems...
[ 7.262157] systemd[1]: Mounted Huge Pages File System.
[ 7.262999] systemd[1]: Mounted POSIX Message Queue File System.
[ 7.264223] systemd[1]: Mounted Kernel Debug File System.
[ 7.264586] systemd[1]: Mounted Kernel Trace File System.
[ 8.616411] XFS (sda1): Mounting V5 Filesystem 2bd1339c-6fe6-4e86-a110-9d0f521542c9
[antoychubekova@antoychubekova ~]$
```

Рис. 4.25: Последовательность монтирования файловых систем

## **5 Выводы**

В ходе выполнения лабораторной работы № 1 я научилась навыкам установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

## Список литературы

1. Купер М. Искусство программирования на языке сценариев командной оболочки. — 2004. — URL: [https://www.opennet.ru/docs/RUS/bash\\_scripting\\_guide/](https://www.opennet.ru/docs/RUS/bash_scripting_guide/).
2. Newham C. Learning the bash Shell: Unix Shell Programming. — O'Reilly Media, 2005. — (In a Nutshell).
3. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ-Петербург, 2010.
4. Колисниченко Д. Н. Самоучитель системного администратора Linux. — СПб. : БХВПетербург, 2011. — (Системный администратор).
5. Dash P. Getting Started with Oracle VM VirtualBox. — Packt Publishing Ltd, 2013.
6. Colvin H. VirtualBox: An Ultimate Guide Book on Virtualization with VirtualBox. — CreateSpace Independent Publishing Platform, 2015.
7. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — (Классика Computer Science).
8. GNU Bash Manual. — 2016. — URL: <https://www.gnu.org/software/bash/manual/>.
9. Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016.
10. Vugt S. van. Red Hat RHCSA/RHCE 7 cert guide : Red Hat Enterprise Linux 7 (EX200 and EX300). — Pearson IT Certification, 2016. — (Certification Guide).
11. Zarrelli G. Mastering Bash. — Packt Publishing, 2017.