

# **Отчёт по лабораторной работе №1**

**Установка и конфигурация операционной системы на  
виртуальную машину**

Лабси Мохаммед

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Ход выполнения</b>	<b>6</b>
2.1	Установка операционной системы Rocky Linux . . . . .	6
2.2	Установка дополнений гостевой ОС и анализ загрузки системы . .	13
2.3	Контрольные вопросы . . . . .	16
<b>3</b>	<b>Заключение</b>	<b>20</b>

# Список иллюстраций

2.1	Параметры созданной виртуальной машины в Oracle VirtualBox . .	7
2.2	Выбор языка установки Rocky Linux . . . . .	8
2.3	Настройка раскладки клавиатуры . . . . .	8
2.4	Выбор базовой среды и дополнительного программного обеспечения	9
2.5	Выбор диска и параметров разметки . . . . .	10
2.6	Настройка Kdump . . . . .	10
2.7	Настройка сети и имени узла . . . . .	11
2.8	Настройка учётной записи root . . . . .	11
2.9	Создание пользовательской учётной записи . . . . .	12
2.10	Завершение установки Rocky Linux . . . . .	12
2.11	Установка VirtualBox Guest Additions из командной строки . . . . .	14
2.12	Анализ вывода dmesg: версия ядра, частота CPU, память и гипервизор	15
2.13	Вывод команды mount и последовательность монтирования файло- вых систем . . . . .	16

## **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 Ход выполнения

### 2.1 Установка операционной системы Rocky Linux

1. В среде виртуализации **Oracle VirtualBox** была создана виртуальная машина с именем **Mohammed**.

В качестве типа операционной системы выбран **Red Hat (64-bit)**.

Для виртуальной машины выделено **4096 МБ оперативной памяти, 4 процессора**, а также создан виртуальный жёсткий диск объёмом **40 ГБ**.

В качестве загрузочного носителя подключён ISO-образ **Rocky-10.0-x86\_64-dvd1.iso**.

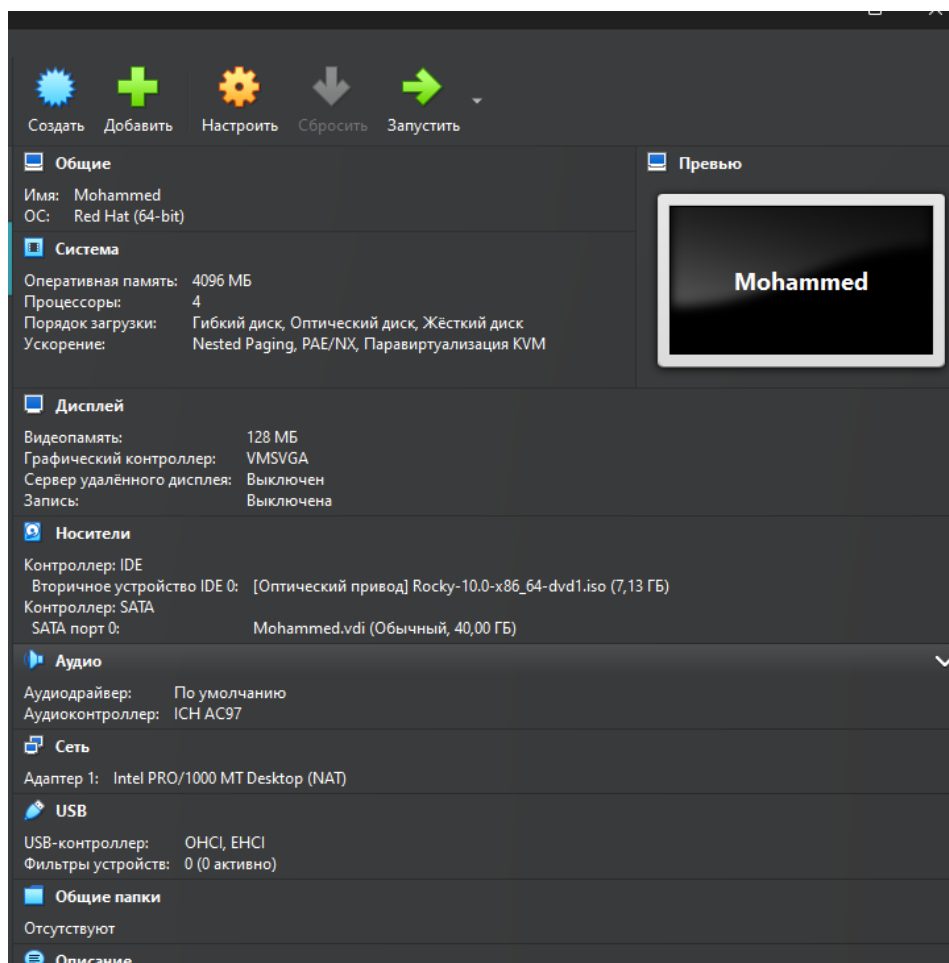


Рис. 2.1: Параметры созданной виртуальной машины в Oracle VirtualBox

2. После запуска виртуальной машины был загружен установщик **Rocky Linux 10.0**.

На начальном этапе установки выбран язык интерфейса **English (United States)** для дальнейшего процесса установки операционной системы.

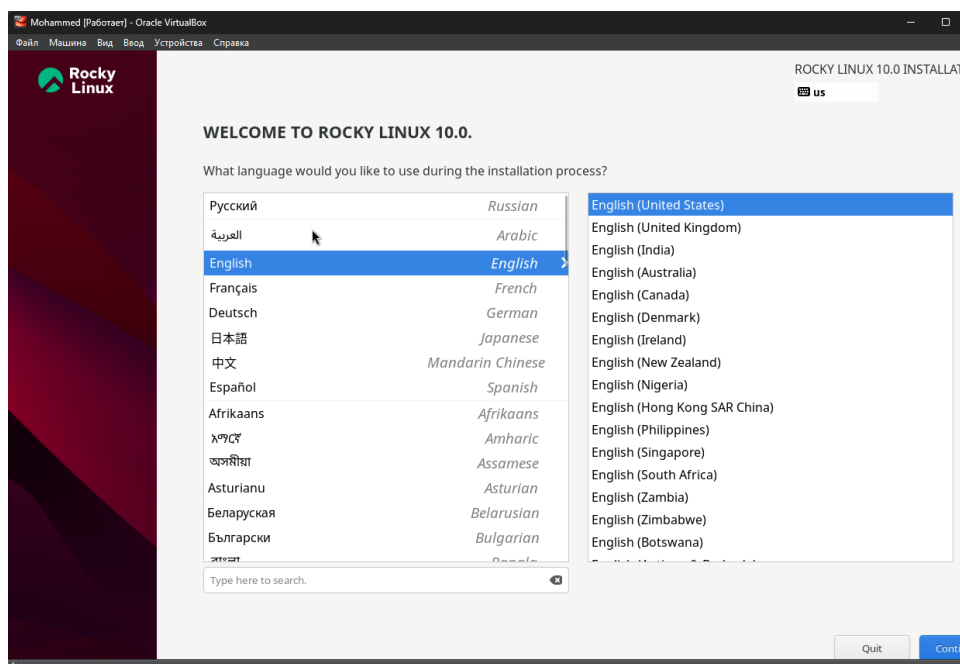


Рис. 2.2: Выбор языка установки Rocky Linux

3. На этапе настройки раскладки клавиатуры были добавлены две раскладки: **English (US)** и **Russian**, что позволяет использовать английскую и русскую раскладки при работе в системе.

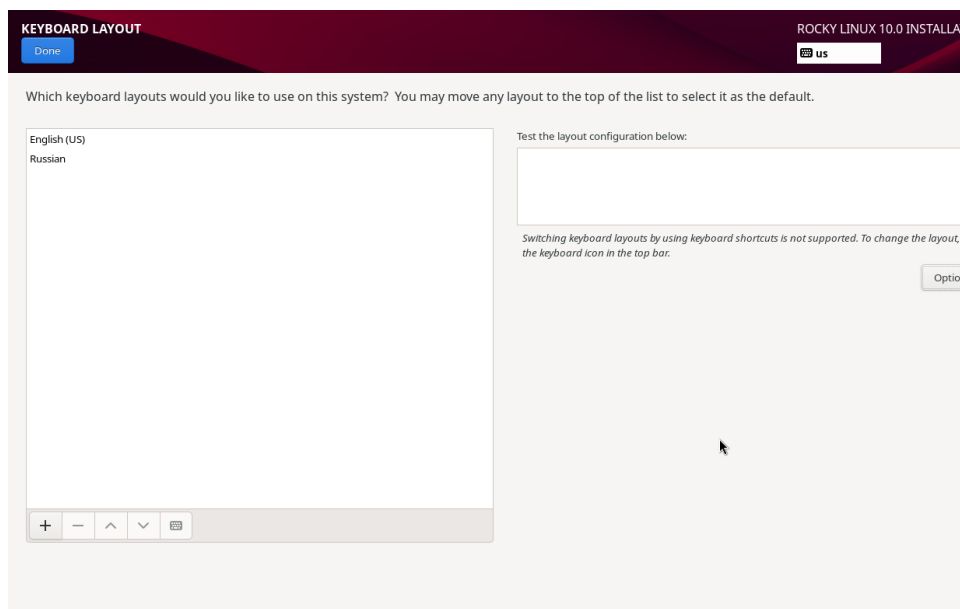


Рис. 2.3: Настройка раскладки клавиатуры



4. В разделе выбора программного обеспечения в качестве базовой среды была выбрана конфигурация **Server with GUI**.

Дополнительно был отмечен набор **Development Tools**, обеспечивающий наличие базового окружения для разработки программного обеспечения.

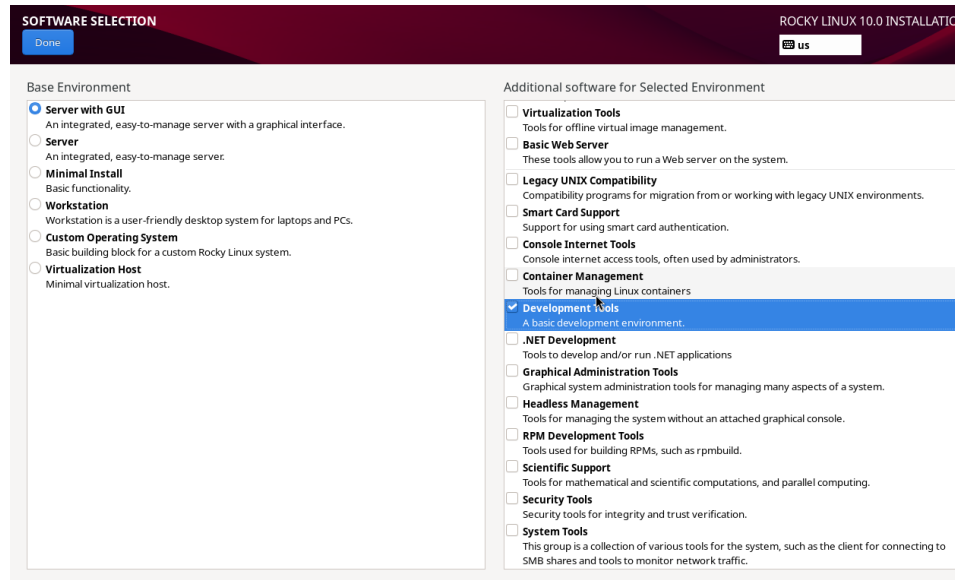


Рис. 2.4: Выбор базовой среды и дополнительного программного обеспечения

5. В разделе **Installation Destination** в качестве целевого устройства выбран виртуальный диск **ATA VBOX HARDDISK** объёмом **40 ГБ**.

Схема разметки диска оставлена в режиме **Automatic**, шифрование данных не включалось.

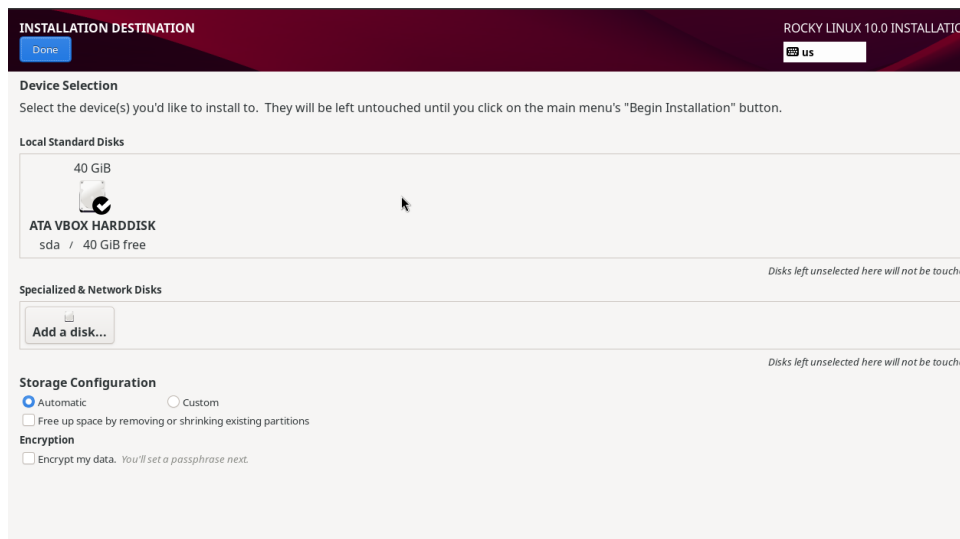


Рис. 2.5: Выбор диска и параметров разметки

6. На этапе настройки **KDUMP** параметр аварийного дампа ядра был оставлен отключённым, так как его использование не является обязательным для выполнения поставленных задач.

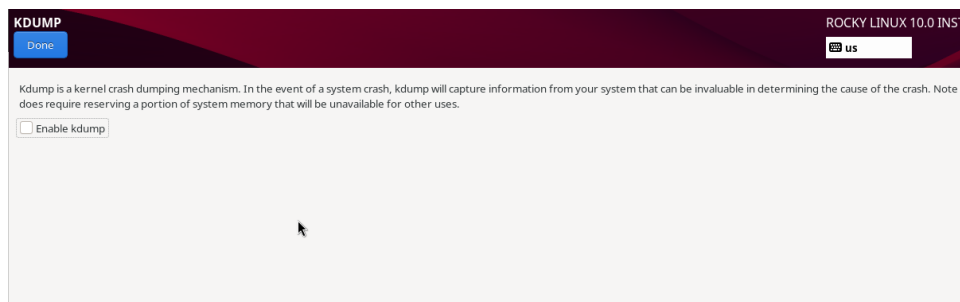


Рис. 2.6: Настройка Kdump

7. В разделе **Network & Host Name** был активирован сетевой интерфейс **Ethernet (enp0s3)**, обеспечивающий доступ виртуальной машины к сети. Имя узла установлено как **mohammed.localdomain**, параметры сети получены автоматически по DHCP.

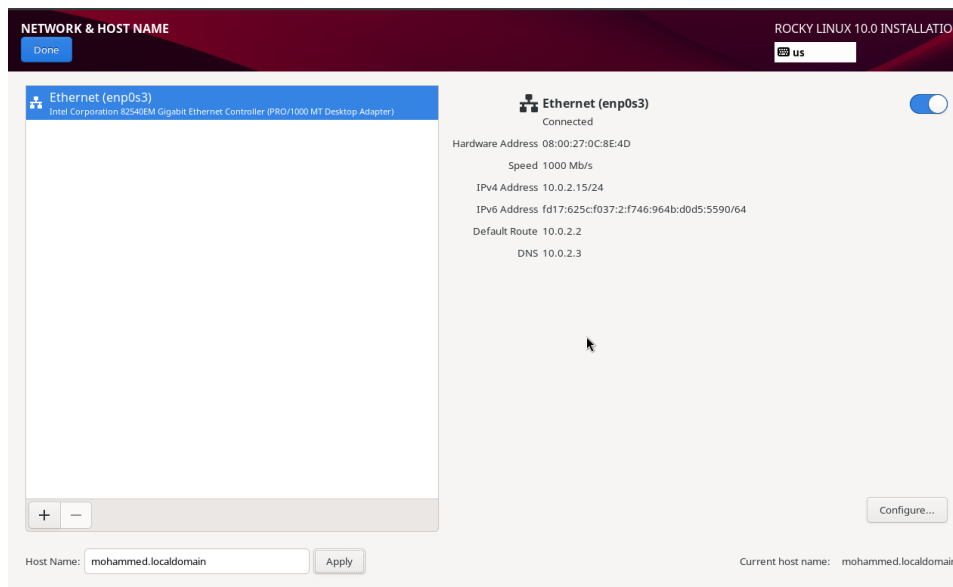


Рис. 2.7: Настройка сети и имени узла

8. В разделе настройки учётной записи суперпользователя был включён **root account**, задан пароль администратора, а также разрешён вход по **SSH** с использованием пароля.

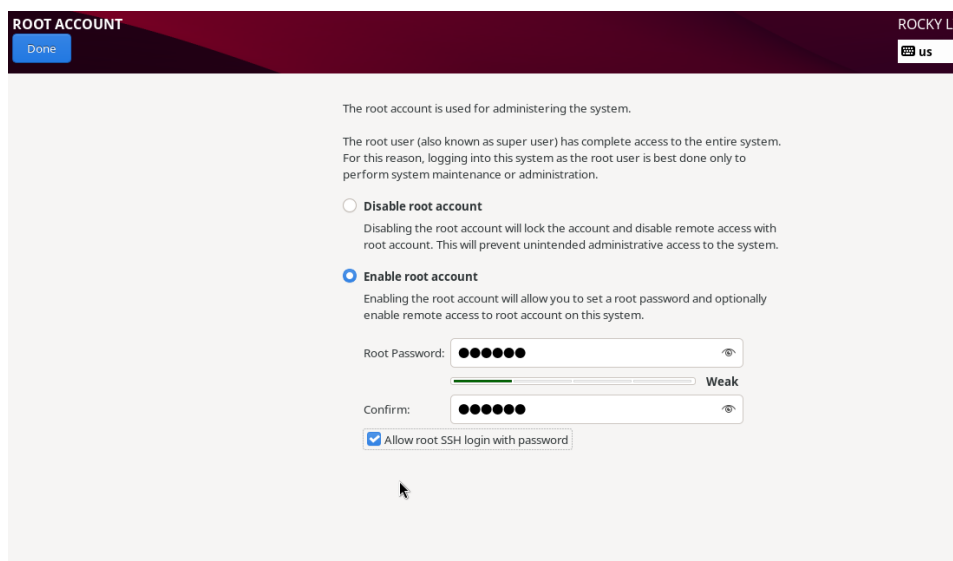


Рис. 2.8: Настройка учётной записи root

9. Далее был создан пользователь **mohammed**.  
Для пользователя заданы пароль и административные привилегии (вклю-

чение в группу **wheel**), что позволяет выполнять команды с повышенными правами.

CREATE USER

Done

Full name mohammed

User name mohammed

☒ Add administrative privileges to this user account (wheel group membership)

☒ Require a password to use this account

Password ●●●●●● Weak

Confirm password ●●●●●●

Advanced...

Рис. 2.9: Создание пользовательской учётной записи

10. После завершения всех этапов настройки был запущен процесс установки операционной системы.

Установка завершилась успешно, о чём свидетельствует сообщение **Complete!**, после чего система была готова к перезагрузке и дальнейшей работе.

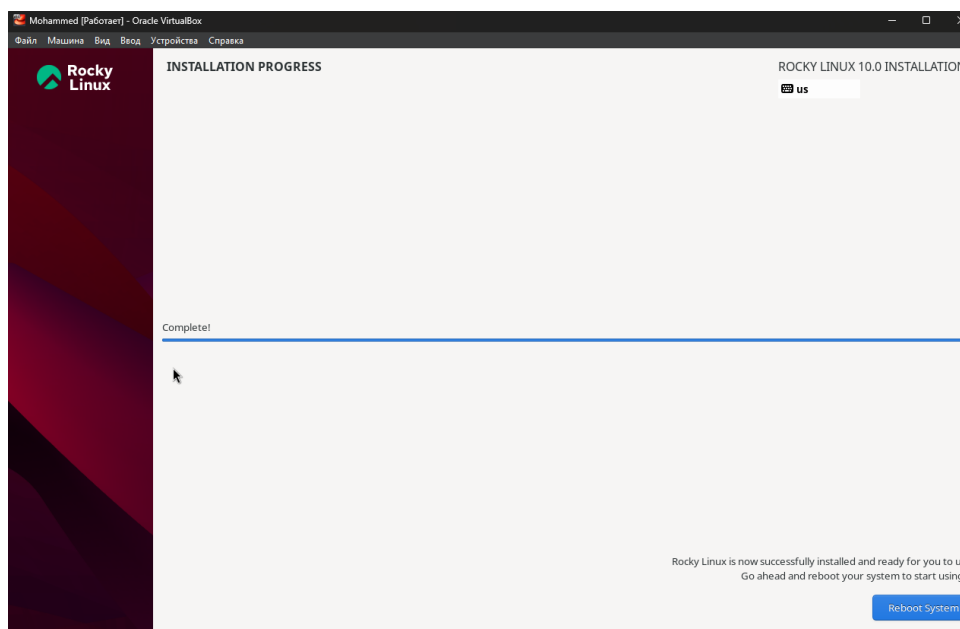


Рис. 2.10: Завершение установки Rocky Linux

## 2.2 Установка дополнений гостевой ОС и анализ загрузки системы

1. После завершения установки операционной системы и загрузки графического окружения был открыт терминал.

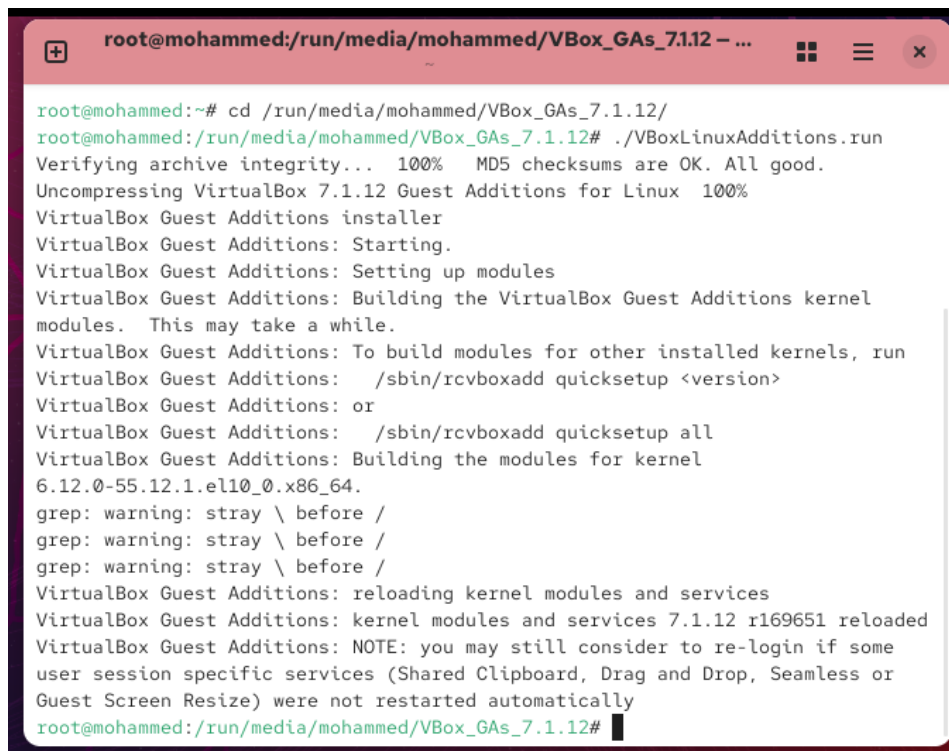
Пользователь получил права администратора с помощью команды **sudo -i**, после чего выполнен переход в каталог с подключённым образом дополнений гостевой ОС:

**/run/media/mohammed/VBox\_GAs\_7.1.12/.**

В данном каталоге был запущен установочный скрипт **./VBoxLinuxAdditions.run**, в результате чего началась установка **VirtualBox Guest Additions**.

В процессе установки были выполнены проверка целостности архива, распаковка файлов и сборка модулей ядра для текущей версии Linux.

По завершении установки модули ядра и сервисы VirtualBox Guest Additions были успешно перезагружены.



```
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12 - ...
root@mohammed:~# cd /run/media/mohammed/VBox_GAs_7.1.12/
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12# ./VBoxLinuxAdditions.run
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.1.12 Guest Additions for Linux 100%
VirtualBox Guest Additions installer
VirtualBox Guest Additions: Starting.
VirtualBox Guest Additions: Setting up modules
VirtualBox Guest Additions: Building the VirtualBox Guest Additions kernel
modules. This may take a while.
VirtualBox Guest Additions: To build modules for other installed kernels, run
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup <version>
VirtualBox Guest Additions: or
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup all
VirtualBox Guest Additions: Building the modules for kernel
6.12.0-55.12.1.el10_0.x86_64.
grep: warning: stray \ before /
grep: warning: stray \ before /
grep: warning: stray \ before /
VirtualBox Guest Additions: reloading kernel modules and services
VirtualBox Guest Additions: kernel modules and services 7.1.12 r169651 reloaded
VirtualBox Guest Additions: NOTE: you may still consider to re-login if some
user session specific services (Shared Clipboard, Drag and Drop, Seamless or
Guest Screen Resize) were not restarted automatically
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12#
```

Рис. 2.11: Установка VirtualBox Guest Additions из командной строки

2. Для анализа последовательности загрузки системы была использована команда **dmesg** с фильтрацией вывода при помощи **grep**.

В результате выполнения команды

**dmesg | grep -i "Linux version"**

была получена информация о версии ядра операционной системы:

**Linux version 6.12.0-55.12.1.el10\_0.x86\_64.**

3. Частота процессора была определена с помощью команды

**dmesg | grep -i "MHz".**

В выводе зафиксировано сообщение **Detected 3187.196 MHz processor**, что указывает на частоту процессора около **3187 МГц**.

4. Для анализа доступной оперативной памяти выполнена команда

**dmesg | grep -i "avail".**

Согласно выводу, объем доступной оперативной памяти составляет

**3961196K / 4193848K available**, что соответствует приблизительно **4 ГБ ОЗУ**.

5. Тип используемого гипервизора был определён с помощью команды **dmesg | grep -i "Hyper"**.

В журнале загрузки зафиксировано сообщение **Hypervisor detected: KVM**, что указывает на использование гипервизора **KVM**.

```
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12# dmesg | grep -i "Linux version"
[ 0.000000] Linux version 6.12.0-55.12.1.el10_0.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7), GNU ld version 2.41-53.el10) #1 SMP PREEMPT_DYNAMIC Fri May 23 17:41:02 UTC 2025
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12# dmesg | grep -i "MHz"
[ 0.000003] tsc: Detected 3187.196 MHz processor
[ 9.516160] e1000 0000:00:03:0 eth0: (PCI:33MHz:32-bit) 08:00:27:0c:8e:4d
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12# dmesg | grep -i "avail"
[ 0.004020] On node 0, zone DMA: 1 pages in unavailable ranges
[ 0.004034] On node 0, zone DMA: 97 pages in unavailable ranges
[ 0.007437] On node 0, zone Normal: 16 pages in unavailable ranges
[ 0.007682] [mem 0xe0000000-0xfefbffff] available for PCI devices
[ 0.154265] Memory: 3961196K/4193848K available (18432K kernel code, 5782K rdata, 14104K rodata, 4320K init, 6792K bss, 228116K reserved, 0K cma-reserved)
[ 0.863589] vmwgfx 0000:00:02:0: [drm] Available shader model: Legacy.
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12# dmesg | grep -i "Hyper"
[ 0.000000] Hypervisor detected: KVM
[ 0.862423] vmwgfx 0000:00:02:0: [drm] *ERROR* vmwgfx seems to be running on an unsupported hypervisor.
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12#
```

Рис. 2.12: Анализ вывода dmesg: версия ядра, частота CPU, память и гипервизор

6. Для определения типа файловой системы корневого раздела и анализа последовательности монтирования файловых систем была выполнена команда **mount**.

В результате установлено, что корневой раздел / смонтирован с файловой системой **xfs**.

7. Анализ вывода команды **mount** показал последовательность монтирования основных файловых систем, включая **xfs**, **devtmpfs**, **tmpfs**, **sysfs**, **proc**, **cgroup2** и другие системные точки монтирования, что подтверждает корректную инициализацию файловой подсистемы при загрузке операционной системы.

```

root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12#
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12# mount
/dev/mapper/rl_vbox-root on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=495716,mode=755,inode64)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=800560k,nr_inodes=819200,mode=755,inode64)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=36,pgpr=1,timeout=0,minproto=5,maxproto=5,direct,pipe
_ino=1909)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,nosuid,nodev,relatime,seclabel,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
tmpfs on /run/credentials/systemd-journald.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymfollow,seclabel,size
e=1024k,nr_inodes=1024,mode=700,inode64,noswap)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
/dev/sda2 on /boot type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=400280k,nr_inodes=100070,mode=700,uid=1000,g
id=1000,inode64)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
portal on /run/user/1000/doc type fuse.portal (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
/dev/sr0 on /run/media/mohammed/VBox_GAs_7.1.12 type iso9660 (ro,nosuid,nodev,relatime,nojoliet,check=s,map=n,blocksize
=2048,uid=1000,gid=1000,dmode=500,fmode=400,iocharset=utf8,uhelper=udisks2)
tmpfs on /run/user/0 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=400280k,nr_inodes=100070,mode=700,inode64)
root@mohammed:/run/media/mohammed/VBox_GAs_7.1.12#

```

Рис. 2.13: Вывод команды mount и последовательность монтирования файловых систем

## 2.3 Контрольные вопросы

### 1. Укажите команды терминала и приведите примеры.

#### – Для получения справки по команде

Для просмотра справочной информации используется команда **man** или параметр **–help**.

Пример:

**man ls** — отображает полное руководство по команде *ls*.

**ls –help** — выводит краткую справку с описанием доступных параметров.

#### – Для перемещения по файловой системе

Используется команда **cd**.

Пример:

**cd /home/mohammed** — переход в каталог */home/mohammed*.

**cd ..** — переход в родительский каталог.

#### – Для просмотра содержимого каталога



Применяется команда **ls**.

Пример:

**ls** — выводит список файлов и каталогов текущей директории.

**ls -l** — отображает подробную информацию о файлах (права, владелец, размер).

– **Для определения объёма каталога**

Используется команда **du**.

Пример:

**du -sh /var/log** — показывает суммарный размер каталога */var/log* в удобочитаемом формате.

– **Для создания и удаления каталогов и файлов**

Для каталогов применяются команды **mkdir** и **rmdir**, для файлов — **touch** и **rm**.

Пример:

**mkdir testdir** — создание каталога *testdir*.

**touch file.txt** — создание пустого файла *file.txt*.

**rm file.txt** — удаление файла.

**rm -r testdir** — рекурсивное удаление каталога с содержимым.

– **Для задания определённых прав на файл или каталог**

Используется команда **chmod**.

Пример:

**chmod 755 script.sh** — установка прав на выполнение для владельца и чтение/выполнение для остальных.

– **Для просмотра истории команд**

Применяется команда **history**.

Пример:

**history** — выводит список ранее введённых команд в текущей оболочке.

2. **Какую информацию содержит учётная запись пользователя? Какие ко-**

### **манды позволяют посмотреть информацию о пользователе?**

Учётная запись пользователя содержит имя пользователя, идентификатор пользователя (UID), идентификатор основной группы (GID), домашний каталог, используемую оболочку и дополнительные группы.

Посмотреть информацию о пользователе позволяют команды **id**, **whoami**, **getent passwd имя\_пользователя**.

Например, команда **id mohammed** отображает UID, GID и список групп пользователя *mohammed*.

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

Файловая система — это способ организации и хранения данных на носителе информации, определяющий структуру каталогов, правила доступа и методы управления файлами.

Примеры файловых систем:

- **ext4** — широко используемая файловая система Linux, отличается надёжностью и поддержкой журналирования.
- **xfs** — высокопроизводительная файловая система, оптимизированная для работы с большими объёмами данных.
- **vfat** — файловая система, совместимая с различными операционными системами, часто используется на съёмных носителях.

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

Для просмотра подмонтированных файловых систем используются команды **mount**, **df -T** или **lsblk -f**.

Например, команда **mount** отображает список всех смонтированных файловых систем, их типы и точки монтирования.

### **5. Как удалить зависший процесс?**

Для завершения зависшего процесса используется команда **kill** с указанием идентификатора процесса (PID).

Пример:

**kill 1234** — попытка корректного завершения процесса с PID 1234.

**kill -9 1234** — принудительное завершение процесса, если он не реагирует на стандартные сигналы.

## 3 Заключение

В ходе выполнения работы была успешно установлена операционная система **Rocky Linux** на виртуальную машину в среде **Oracle VirtualBox**. Были выполнены базовые настройки системы, установлены дополнения гостевой ОС и настроены учётные записи пользователей. Также проведён анализ процесса загрузки системы и получены основные сведения о конфигурации аппаратных ресурсов и файловой системе. Полученные навыки позволяют использовать виртуальную машину для дальнейшего изучения и практической работы с операционными системами семейства Linux.