

# **лабораторной работе №1**

**Установка и Конфигурация ОС на Виртуальную Машину**

Дауд Амжад

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Выполнение дополнительной работы</b>	<b>15</b>
<b>5</b>	<b>Ответы на контрольные вопросы</b>	<b>19</b>
<b>6</b>	<b>Выводы</b>	<b>21</b>
	<b>Список литературы</b>	<b>22</b>

# Список иллюстраций

3.1	Оптический диск . . . . .	7
3.2	Объем оперативной памяти . . . . .	8
3.3	Жесткий диск . . . . .	8
3.4	Итог . . . . .	9
3.5	Носители . . . . .	9
3.6	Запуск машины . . . . .	10
3.7	Выбор языка . . . . .	10
3.8	Окно настроек . . . . .	11
3.9	Отключение kdump . . . . .	11
3.10	Создание пользователя . . . . .	11
3.11	Выбор окружения . . . . .	12
3.12	Выбор сети . . . . .	12
3.13	Установка . . . . .	13
3.14	Проверка носителей . . . . .	13
3.15	Окно выбора пользователя . . . . .	14
4.1	Версия ядра Linux . . . . .	15
4.2	Частота процессора . . . . .	15
4.3	Модель процессора . . . . .	16
4.4	Объем доступной оперативной памяти . . . . .	17
4.5	Тип гипервизора . . . . .	17
4.6	Тип файловой системы . . . . .	18
4.7	Последовательность монтирования файловых систем . . . . .	18

## **Список таблиц**

# 1 Цель работы

Приобретение практических навыков установки операционной системы на виртуальную машину.

Цель данного шаблона — максимально упростить подготовку отчётов по лабораторным работам. Модифицируя данный шаблон, студенты смогут без труда подготовить отчёт по лабораторным работам, а также познакомиться с основными возможностями разметки Markdown.

## 2 Задание

1. Установить и настроить Rocky Linux.
2. Найти следующую информацию:
  1. Версия Linux
  2. Частота процессора
  3. Модель процессора
  4. Объем доступной оперативной памяти
  5. Тип обнаруженного гипервизора
  6. Тип файловой системы корневого раздела
  7. Последовательность монтирования файловых систем

### 3 Выполнение лабораторной работы

В приложении VirtualBox создаю новую виртуальную машину. Указываю имя виртуальной машины и добавляю оптический диск.

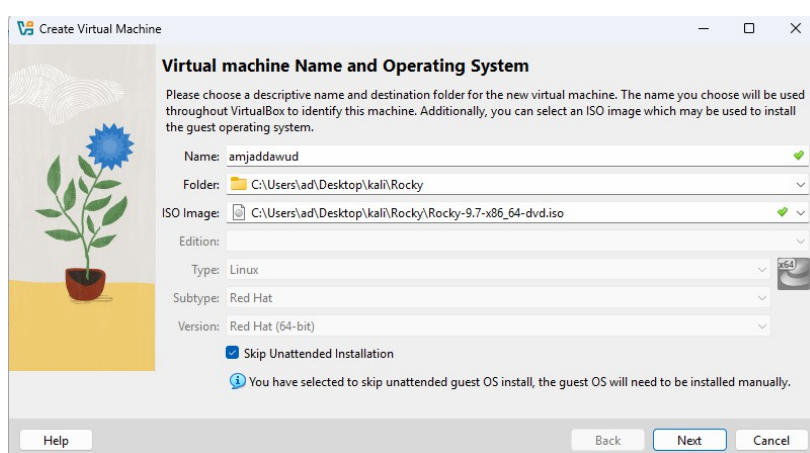


Рисунок 3.1: Оптический диск

Указываю объем памяти и создаю виртуальный жетский диск.

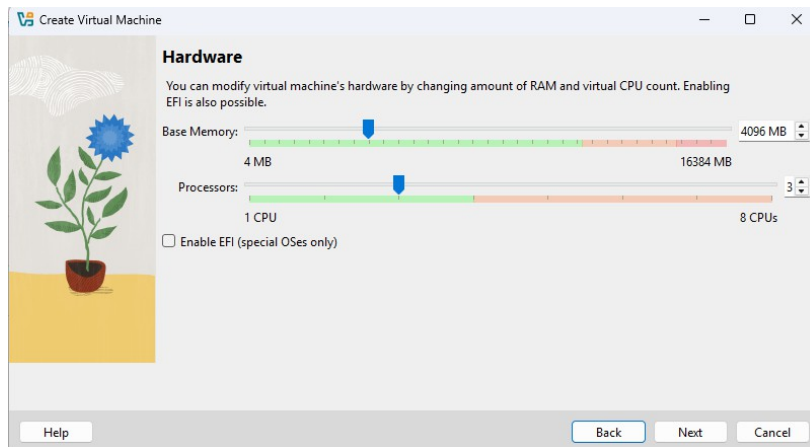


Рисунок 3.2: Объем оперативной памяти

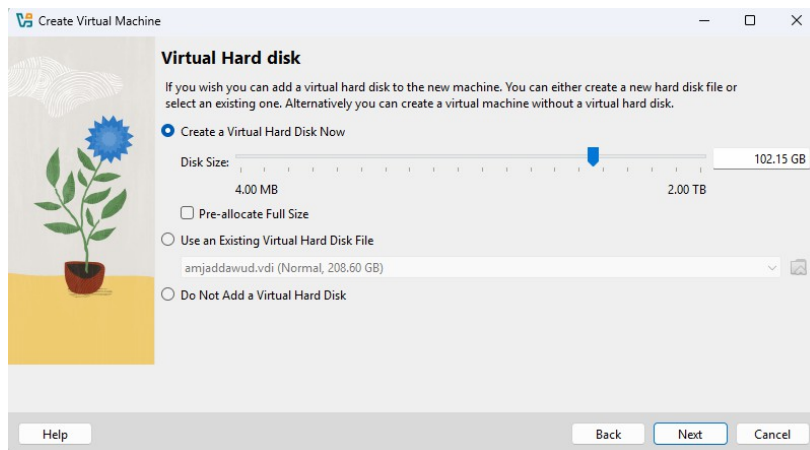


Рисунок 3.3: Жетский диск

Соглашаюсь с поставленными настройками.



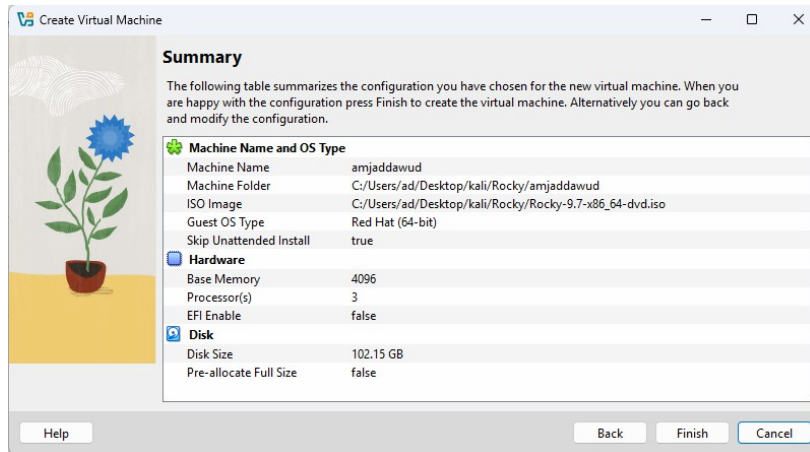


Рисунок 3.4: Итог

Проверяю подключения диска в носителях образ.

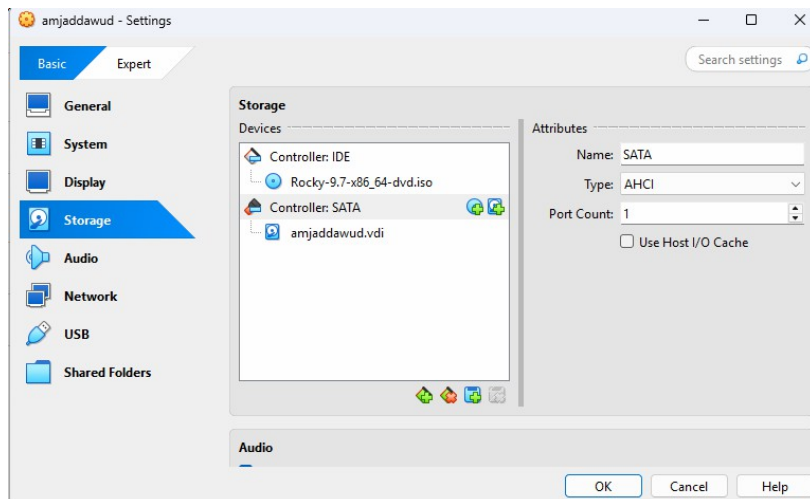


Рисунок 3.5: Носители

Запускаю машину и устанавливаю систему.

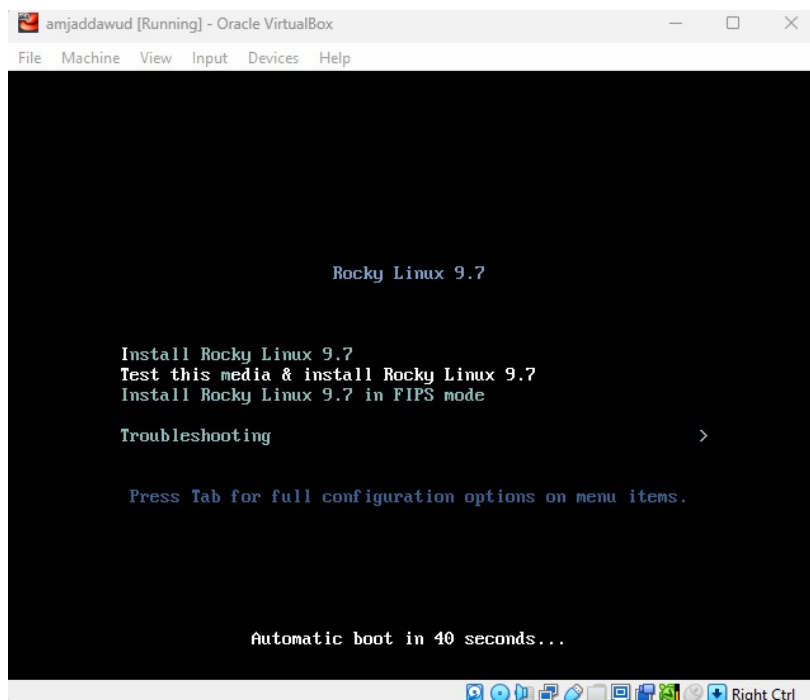


Рисунок 3.6: Запуск машины

Выбираю язык установки.

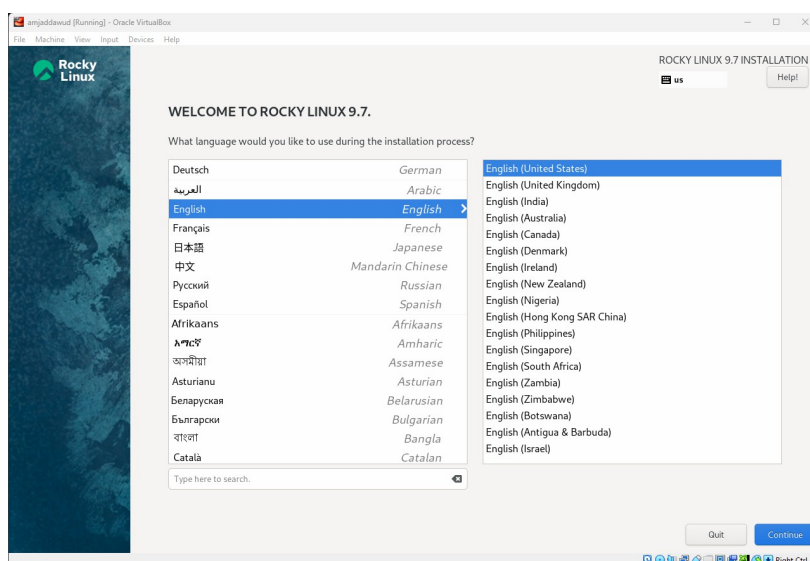


Рисунок 3.7: Выбор языка

Выбираю место установки, отключаю kdump, создаю пользователя (администратора) и устанавливаю пароль для администратора.

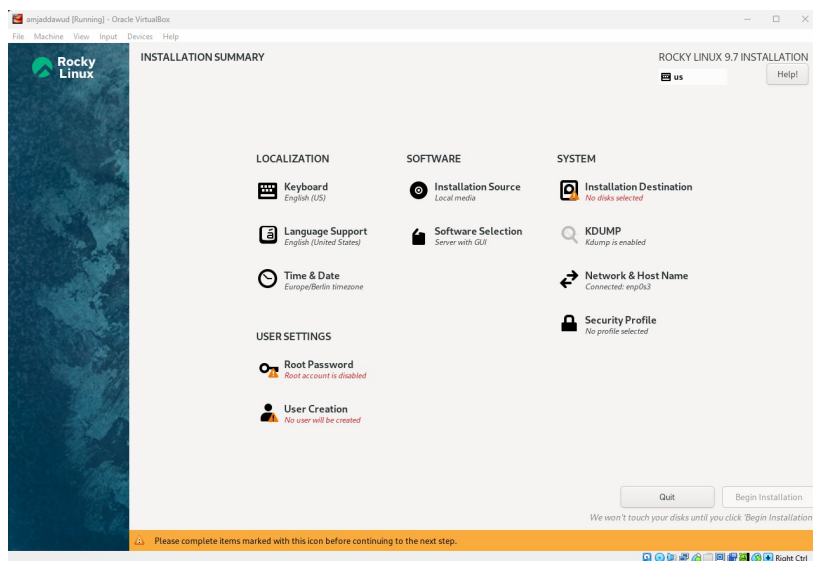


Рисунок 3.8: Окно настроек

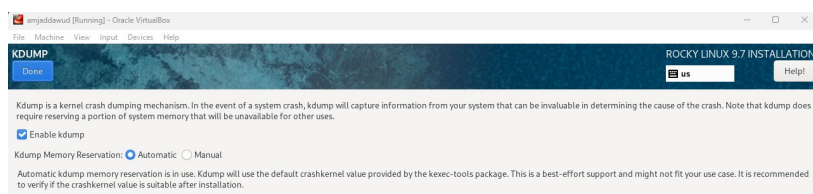


Рисунок 3.9: Отключение kdump

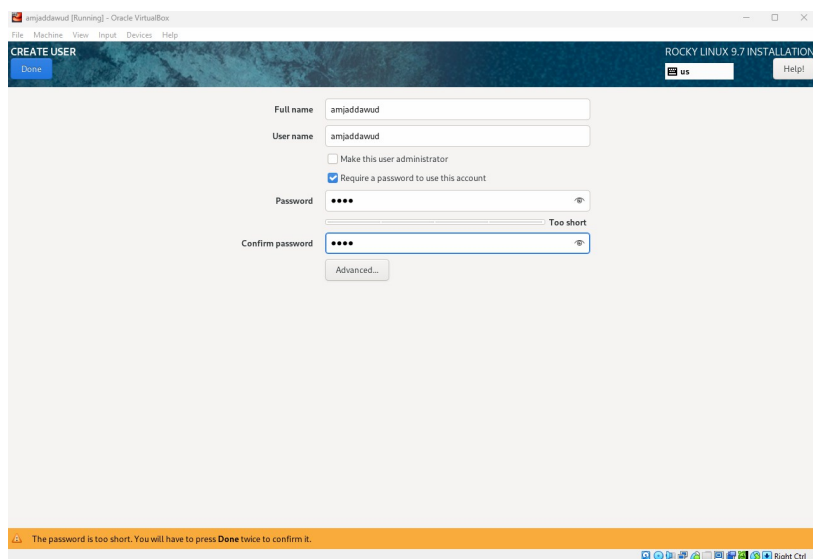


Рисунок 3.10: Создание пользователя

Выбираю окружение сервер с GUI и средства разработки в дополнительном программном обеспечении.

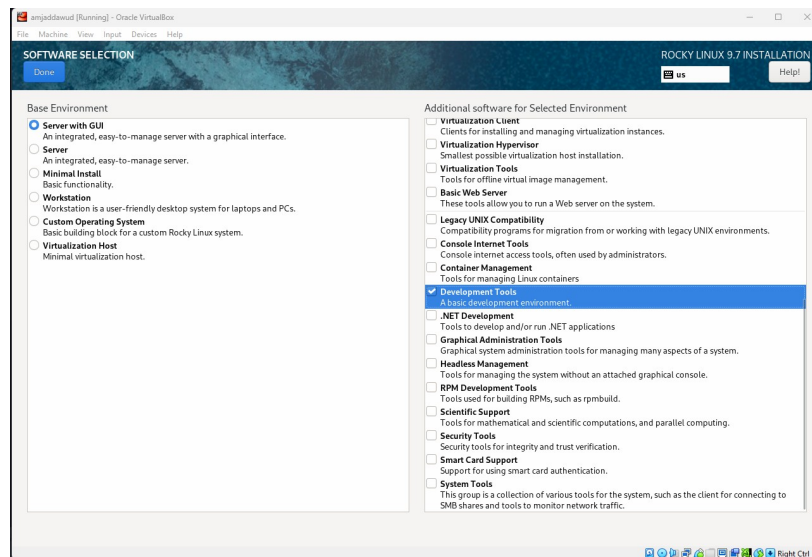


Рисунок 3.11: Выбор окружения

Указываю имя узла.

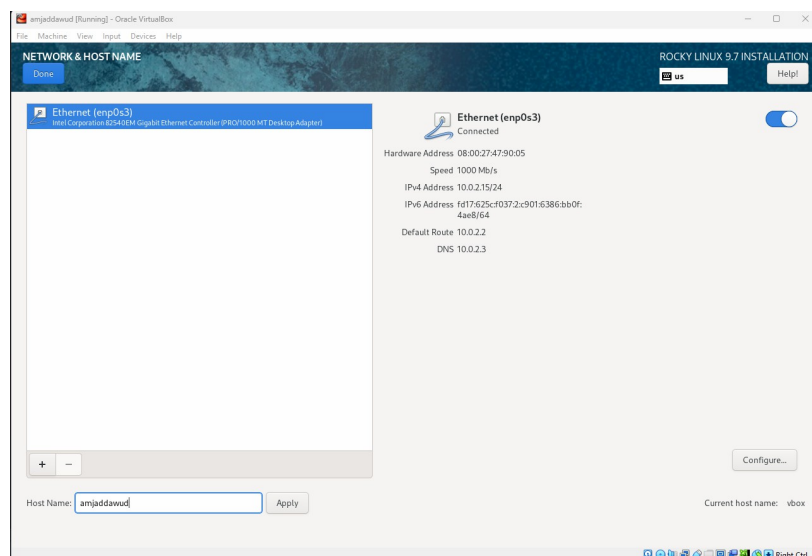


Рисунок 3.12: Выбор сети

Затем устанавливаю систему.

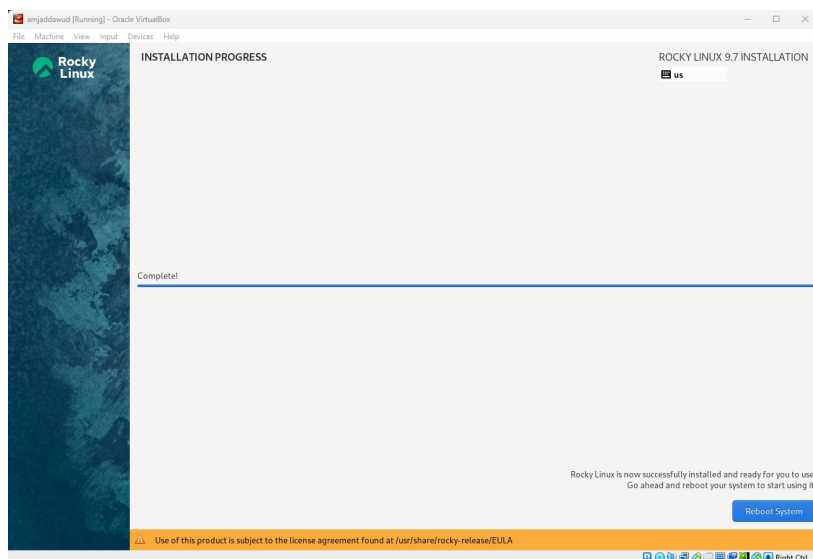


Рисунок 3.13: Установка

После завершения установки образ диска пропадет из носителей.

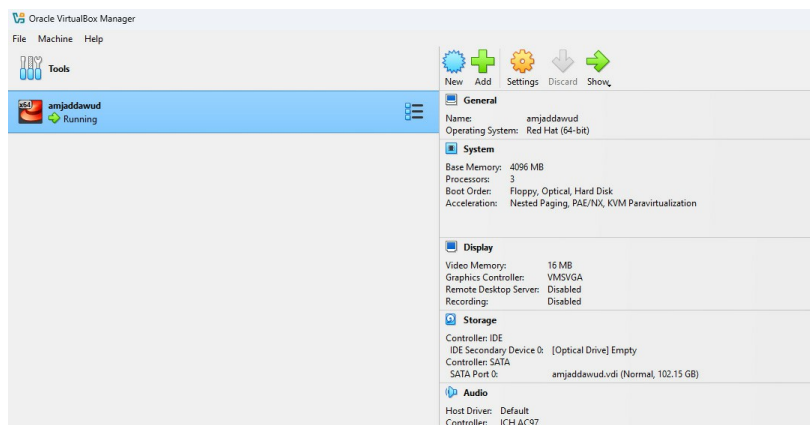


Рисунок 3.14: Проверка носителей

При запуске виртуальной машины появляется окно выбора пользователя.

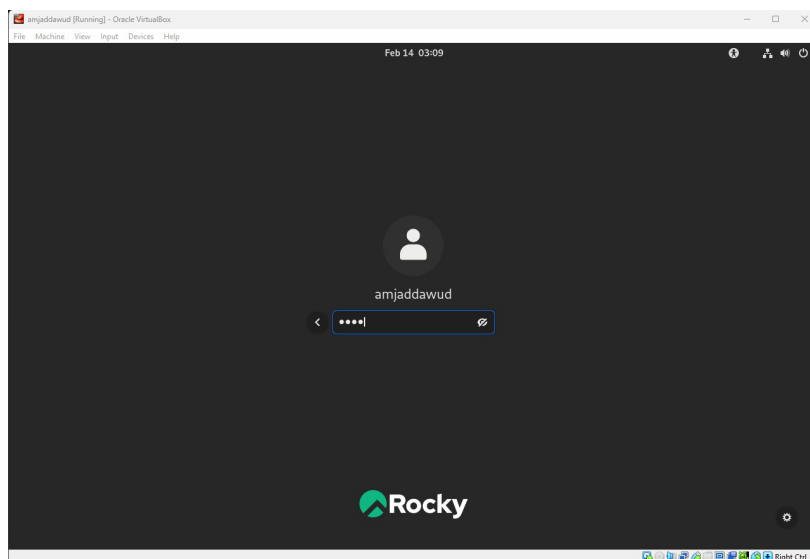


Рисунок 3.15: Окно выбора пользователя

## 4 Выполнение дополнительной работы

Запускаю в терминале: `dmesg | grep -i «Linux version»`, чтобы получить информацию о ядре.

```
[amjaddawud@amjaddawud ~]$ dmesg | grep -i "Linux version"
[ 0.000000] Linux version 5.14.0-611.5.1.el9_7.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 11.5.0 20240719 (Red Hat 11.5.0-11), GNU ld version 2.35.2-67.el9) #1 SMP PREEMPT_DYNAMIC Tue Nov 11 22:20:27 UTC 2025
[amjaddawud@amjaddawud ~]$
```

Рисунок 4.1: Версия ядра Linux

`dmesg | grep -i «detected»`, чтобы получить информацию о процессоре.

```
[amjaddawud@amjaddawud ~]$ dmesg | grep -i "Detected"
[ 0.000000] Hypervisor detected: KVM
[ 0.000007] tsc: Detected 3600.000 MHz processor
[ 0.024276] Warning: Deprecated Hardware is detected: x86_64-v2:GenuineIntel: Intel(R) Core(TM) i3-10100 CPU @ 3.60GHz will not be maintained in a future major release and may be disabled
[ 1.129968] hub 1-0:1.0: 12 ports detected
[ 1.201152] hub 2-0:1.0: 12 ports detected
[ 1.505833] systemd[1]: Detected virtualization oracle.
[ 1.506192] systemd[1]: Detected architecture x86-64.
[ 2.389564] Warning: Unmaintained driver is detected: e1000
[ 2.403392] Warning: Unmaintained driver is detected: e1000_init_module
[ 5.885811] systemd[1]: Detected virtualization oracle.
[ 5.886140] systemd[1]: Detected architecture x86-64.
[amjaddawud@amjaddawud ~]$
```

Рисунок 4.2: Частота процессора

`dmesg | grep -i «CPU»`, чтобы получить информацию о модели процессора.

```

[amjaddawud@amjaddawud ~]$ dmesg | grep -i "CPU"
[ 0.002056] CPU MTRRs all blank - virtualized system.
[ 0.013104] ACPI: SSDT 0x00000000DFFF02B0 00036C (v01 VBOX VBOXCPU 00000000
2 INTL 20100528)
[ 0.024109] CPU topo: Max. logical packages: 1
[ 0.024110] CPU topo: Max. logical dies: 1
[ 0.024110] CPU topo: Max. dies per package: 1
[ 0.024114] CPU topo: Max. threads per core: 1
[ 0.024114] CPU topo: Num. cores per package: 3
[ 0.024115] CPU topo: Num. threads per package: 3
[ 0.024115] CPU topo: Allowing 3 present CPUs plus 0 hotplug CPUs
[ 0.024276] Warning: Deprecated Hardware is detected: x86_64-v2:GenuineIntel:
Intel(R) Core(TM) i3-10100 CPU @ 3.60GHz will not be maintained in a future major
release and may be disabled
[ 0.028645] setup_percpu: NR_CPUS:8192 nr_cpumask_bits:3 nr_cpu_ids:3 nr_node
_ids:1
[ 0.032992] percpu: Embedded 64 pages/cpu s225280 r8192 d28672 u524288
[ 0.032998] pcpu-alloc: s225280 r8192 d28672 u524288 alloc=1*2097152
[ 0.033001] pcpu-alloc: [0] 0 1 2 -
[ 0.056927] SLUB: HWalign=64, Order=0-3, MinObjects=0, CPUs=3, Nodes=1
[ 0.067338] rcu: RCU restricting CPUs from NR_CPUS=8192 to nr_cpu_ids=3.
[ 0.067341] rcu: Adjusting geometry for rcu_fanout_leaf=16, nr_cpu_ids=3
[ 0.067346] RCU Tasks: Setting shift to 2 and lim to 1 rcu_task_cb_adjust=1 r
cu_task_cpu_ids=3.
[ 0.067348] RCU Tasks Rude: Setting shift to 2 and lim to 1 rcu_task_cb_adju
st=1 rcu_task_cpu_ids=3.
[ 0.067348] RCU Tasks Trace: Setting shift to 2 and lim to 1 rcu_task_cb_adju
st=1 rcu_task_cpu_ids=3.
[ 0.137684] MMIO Stale Data: Vulnerable: Clear CPU buffers attempted, no micr
ocode
[ 0.254047] smpboot: CPU0: Intel(R) Core(TM) i3-10100 CPU @ 3.60GHz (family:
0x6, model: 0xa5, stepping: 0x3)
[ 0.254684] Performance Events: unsupported p6 CPU model 165 no PMU driver, s
oftware events only.
[ 0.254684] smp: Bringing up secondary CPUs ...
[ 0.254684] .... node #0, CPUs: #1 #2
[ 0.265051] smp: Brought up 1 node, 3 CPUs
[ 0.275814] cpuidle: using governor menu
[ 0.288690] cryptd: max_cpu_qlen set to 1000
[ 0.296851] ACPI: _OSC evaluation for CPUs failed, trying _PDC
[ 1.218555] intel_pstate: CPU model not supported
[amjaddawud@amjaddawud ~]$

```

Рисунок 4.3: Модель процессора

`dmesg | grep -i «memory»`, чтобы получить информацию о памяти.



```

[amjaddawud@amjaddawud ~]$ dmesg | grep -i "Memory"
[ 0.013106] ACPI: Reserving FACP table memory at [mem 0xdfff00f0-0xdfff01e3]
[ 0.013107] ACPI: Reserving DSDT table memory at [mem 0xdfff0620-0xdfff2972]
[ 0.013108] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.013109] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.013109] ACPI: Reserving APIC table memory at [mem 0xdfff0240-0xdfff02a3]
[ 0.013110] ACPI: Reserving SSDT table memory at [mem 0xdfff02b0-0xdfff061b]
[ 0.015158] Early memory node ranges
[ 0.024263] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.024265] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009ffff]
[ 0.024265] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000effff]
[ 0.024266] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[ 0.024267] PM: hibernation: Registered nosave memory: [mem 0xdfff0000-0xdfffffff]
[ 0.024267] PM: hibernation: Registered nosave memory: [mem 0xe0000000-0xfebfffff]
[ 0.024268] PM: hibernation: Registered nosave memory: [mem 0xfec00000-0xfec0ffff]
[ 0.024268] PM: hibernation: Registered nosave memory: [mem 0xfec01000-0xfedfffff]
[ 0.024269] PM: hibernation: Registered nosave memory: [mem 0xfef00000-0xfef0ffff]
[ 0.024269] PM: hibernation: Registered nosave memory: [mem 0xfef01000-0xfffffbffff]
[ 0.024270] PM: hibernation: Registered nosave memory: [mem 0xffffc000-0xfffffbffff]
[ 0.143133] Freeing SMP alternatives memory: 40K
[ 0.269381] Memory: 3671452K/4193848K available (16384K kernel code, 5783K rw data, 13988K rodata, 4068K init, 7312K bss, 518504K reserved, 0K cma-reserved)
[ 0.271119] x86/mm: Memory block size: 128MB
[ 1.047548] Freeing initrd memory: 61080K
[ 1.116258] Non-volatile memory driver v1.3
[ 1.479204] Freeing unused decrypted memory: 2028K
[ 1.480044] Freeing unused kernel image (initmem) memory: 4068K
[ 1.481343] Freeing unused kernel image (rodata/data gap) memory: 348K
[ 3.119691] vmwgfx 0000:00:02.0: [drm] Legacy memory limits: VRAM = 16384 KiB, FIFO = 2048 KiB, surface = 507904 KiB
[ 3.119697] vmwgfx 0000:00:02.0: [drm] Maximum display memory size is 16384 K

```

Рисунок 4.4: Объем доступной оперативной памяти

`dmesg | grep -i «detected»`, чтобы получить информацию о гипервизоре.

```

[amjaddawud@amjaddawud ~]$ dmesg | grep -i "Detected"
[ 0.000000] Hypervisor detected: KVM
[ 0.000007] tsc: Detected 3600.000 MHz processor
[ 0.024276] Warning: Deprecated Hardware is detected: x86_64-v2:GenuineIntel:
Intel(R) Core(TM) i3-10100 CPU @ 3.60GHz will not be maintained in a future major
release and may be disabled
[ 1.129968] hub 1-0:1.0: 12 ports detected
[ 1.201152] hub 2-0:1.0: 12 ports detected
[ 1.505833] systemd[1]: Detected virtualization oracle.
[ 1.506192] systemd[1]: Detected architecture x86-64.
[ 2.389564] Warning: Unmaintained driver is detected: e1000
[ 2.403392] Warning: Unmaintained driver is detected: e1000_init_module
[ 5.885811] systemd[1]: Detected virtualization oracle.
[ 5.886140] systemd[1]: Detected architecture x86-64.

```

Рисунок 4.5: Тип гипервизора

`sudo fdisk -l`, чтобы получить информацию о файловой системе корневого раздела.

```
[root@amjaddawud ~]# sudo fdisk -l

Disk /dev/sda: 102.15 GiB, 109678034944 bytes, 214214912 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xd3330006

   Device   Boot    Start        End    Sectors    Size Id Type
   /dev/sda1 *      2048     2099199     2097152    16 83 Linux
   /dev/sda2          2099200 214214655 212115456 101.16 8e Linux LVM

Disk /dev/mapper/rl_vbox-root: 65.31 GiB, 70124568576 bytes, 136962048 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/rl_vbox-swap: 3.95 GiB, 4236247040 bytes, 8273920 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/rl_vbox-home: 31.89 GiB, 34238103552 bytes, 66871296 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[root@amjaddawud ~]#
```

Рисунок 4.6: Тип файловой системы

`dmesg | grep -i «mount»`, чтобы получить информацию о монтировании файловых систем.

```
[amjaddawud@amjaddawud ~]$ dmesg | grep -i "Mount"
[ 0.143506] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 0.143511] Mountpoint-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 5.259679] XFS (dm-0): Mounting V5 Filesystem 0599f80d-9005-4269-a757-52b562720327
[ 5.291964] XFS (dm-0): Ending clean mount
[ 6.430172] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 6.461807] systemd[1]: Mounting Huge Pages File System...
[ 6.464217] systemd[1]: Mounting POSIX Message Queue File System...
[ 6.466829] systemd[1]: Mounting Kernel Debug File System...
[ 6.468182] systemd[1]: Mounting Kernel Trace File System...
[ 6.538817] systemd[1]: Starting Remount Root and Kernel File Systems...
[ 8.424782] XFS (sda1): Mounting V5 Filesystem 6eebf755-86ef-435b-8f6a-105bd777e774
[ 8.431764] XFS (dm-2): Mounting V5 Filesystem ae009557-4288-44f5-ac33-7fe80ed7ac8f
[ 8.470074] XFS (sda1): Ending clean mount
[ 8.488743] XFS (dm-2): Ending clean mount
```

Рисунок 4.7: Последовательность монтирования файловых систем

## 5 Ответы на контрольные вопросы

1. Учетная запись содержит необходимые для идентификации пользователя при подключении к системе данные, а так же информацию для авторизации и учета: системного имени (user name) (оно может содержать только латинские буквы и знак нижнее подчеркивание, еще оно должно быть уникальным), идентификатор пользователя (UID) (уникальный идентификатор пользователя в системе, целое положительное число), идентификатор группы (GID) (группа, к к-рой относится пользователь. Она, как минимум, одна, по умолчанию - одна), полное имя (full name) (Могут быть ФИО), домашний каталог (home directory) (каталог, в к-рый попадает пользователь после входа в систему и в к-ром хранятся его данные), начальная оболочка (login shell) (командная оболочка, к-рая запускается при входе в систему).
2. Для получения справки по команде: `--help`; для перемещения по файловой системе - `cd`; для просмотра содержимого каталога - `ls`; для определения объёма каталога - `du` ; для создания / удаления каталогов - `mkdir/rmdir`; для создания / удаления файлов - `touch/rm`; для задания определённых прав на файл / каталог - `chmod`; для просмотра истории команд - `history`
3. Файловая система - это порядок, определяющий способ организации и хранения и именования данных на различных носителях информации. Примеры: FAT32 представляет собой пространство, разделенное на три части: одна область для служебных структур, форма указателей в виде таблиц и зона для хранения самих файлов. ext3/ext4 - журналируемая файловая система,

используемая в основном в ОС с ядром Linux.

4. С помощью команды `df`, введя ее в терминале. Это утилита, которая показывает список всех файловых систем по именам устройств, сообщает их размер и данные о памяти. Также посмотреть подмонтированные файловые системы можно с помощью утилиты `mount`.
5. Чтобы удалить зависший процесс, вначале мы должны узнать, какой у него `id`: используем команду `ps`. Далее в терминале вводим команду `kill < id процесса >`. Или можно использовать утилиту `killall`, что «убьет» все процессы, которые есть в данный момент, для этого не нужно знать `id` процесса.

## 6 Выводы

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

## **Список литературы**