

Отчёт по лабораторной работе 1

Установка и Конфигурация ОС на Виртуальную Машину

Гомес Лопес Теофания

Содержание

1 Цель работы	5
2 Задание	6
3 Выполнение лабораторной работы	7
4 Выполнение дополнительной работы	13
5 Ответы на контрольные вопросы	16
6 Выводы	18

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 Выбор языка	10
3.7 Окно настроек	10
3.8 Отключение kdump	10
3.9 Создание пользователя	11
3.10 Выбор окружения	11
3.11 Выбор сети	11
3.12 Установка	12
4.1 dmesg	13
4.2 Частота процессора	13
4.3 Модель процессора	14
4.4 Объем доступной оперативной памяти	14
4.5 Тип гипервизора	14
4.6 Тип файловой системы	14
4.7 Последовательность монтирования файловых	15

Список таблиц

1 Цель работы

Получение практических навыков установки операционной системы на виртуальную машину

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

Для начала я создаю новую виртуальную машину в VirtualBox. Потом мне нужно указать её имя и добавить оптический диск.

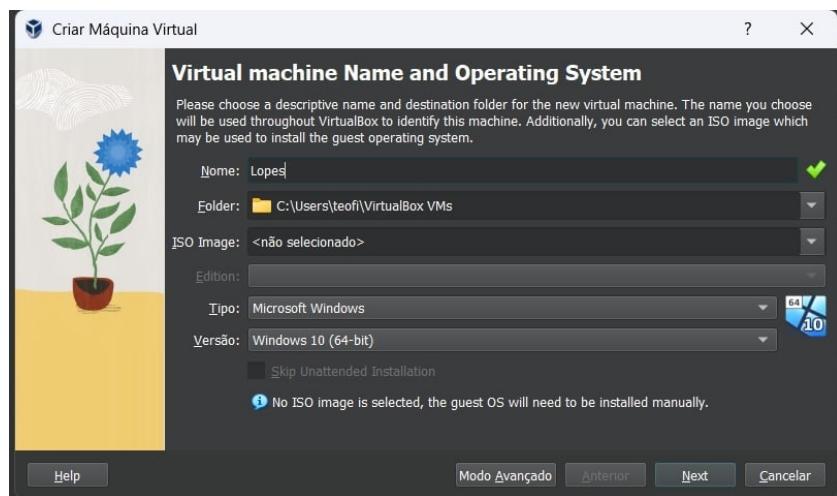


Рис. 3.1: Оптический диск

Указываю объем памяти и создаю виртуальный жесткий диск.

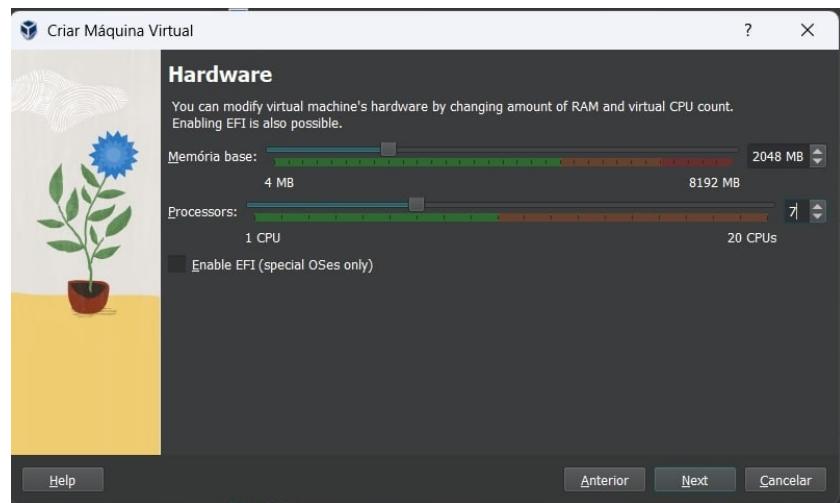


Рис. 3.2: Объем оперативной памяти

Жесткий диск.

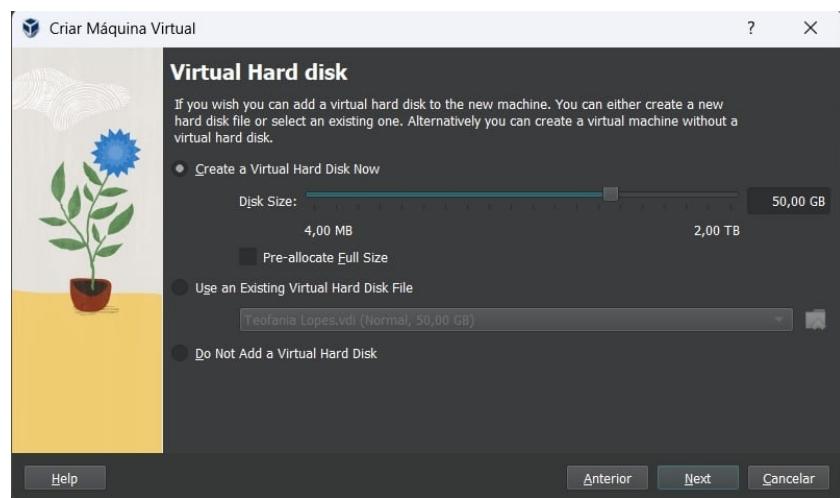


Рис. 3.3: Жесткий диск

Я соглашаюсь с этими настройками.

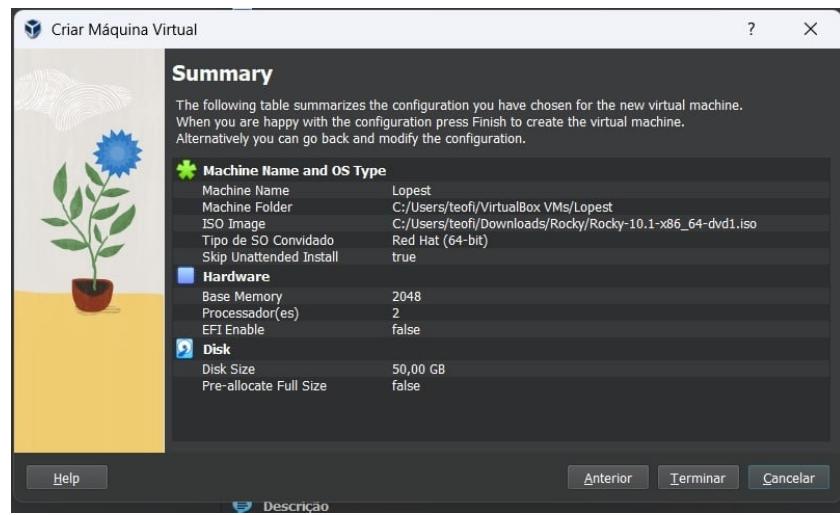


Рис. 3.4: Итог

Проверяю подключения диска в носителях образ.

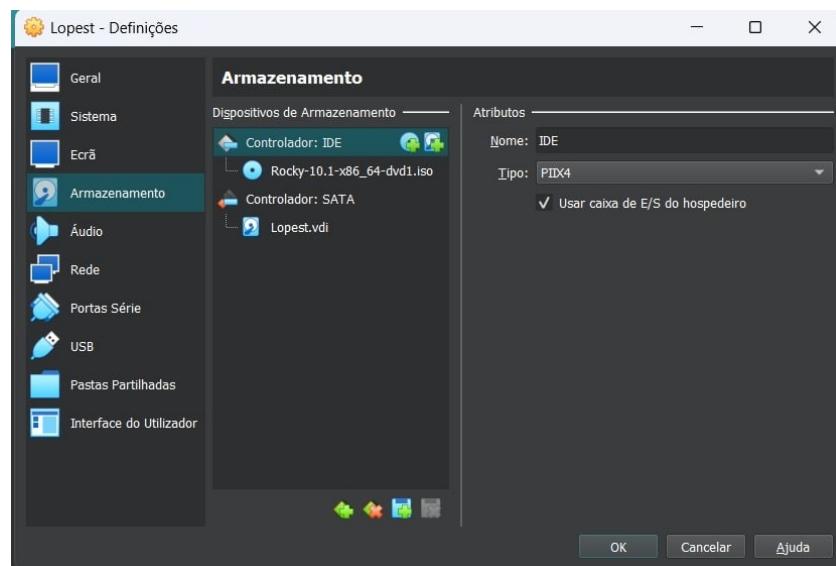


Рис. 3.5: Носители

Выбираю язык установки.

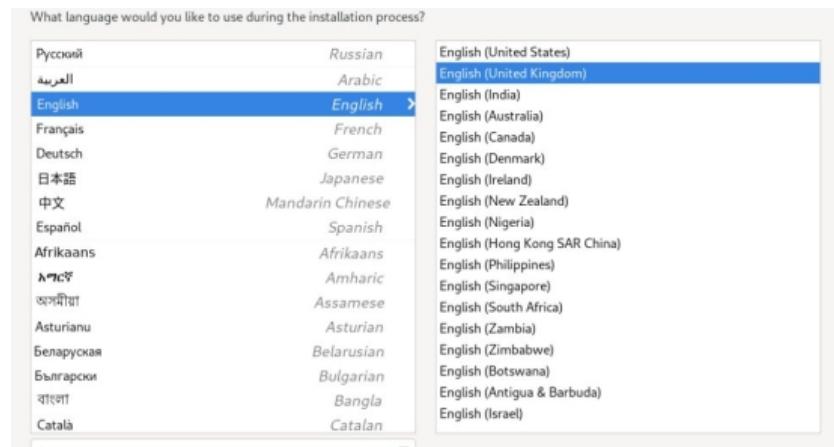


Рис. 3.6: Выбор языка

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

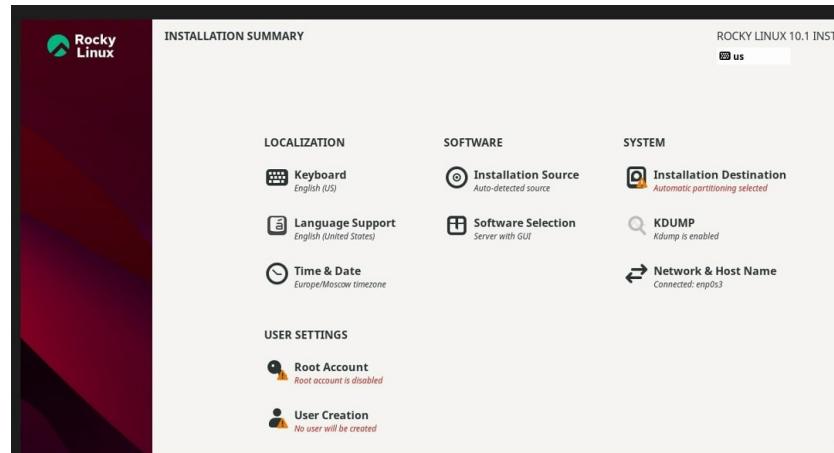


Рис. 3.7: Окно настроек

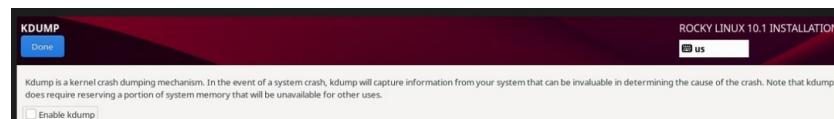


Рис. 3.8: Отключение kdump

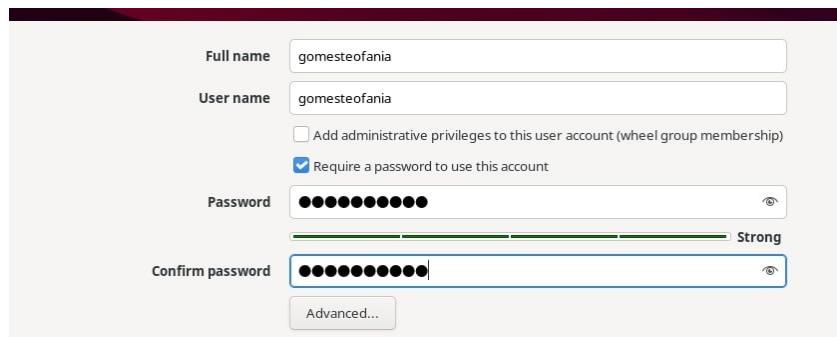


Рис. 3.9: Создание пользователя

В качестве окружения я выбираю сервер с графическим интерфейсом (GUI) и дополнительно устанавливаю средства разработки.

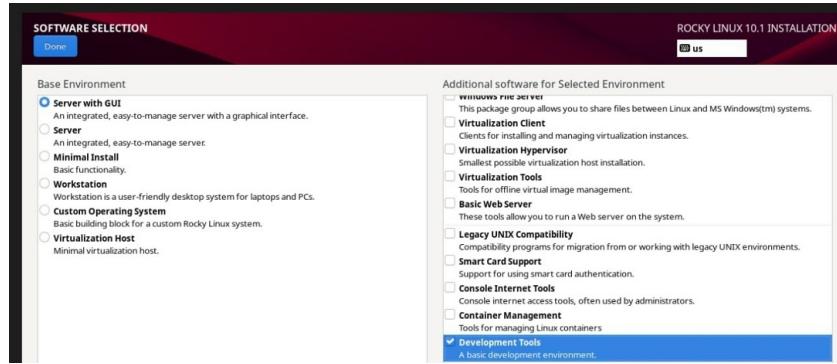


Рис. 3.10: Выбор окружения

Указываю имя узла.

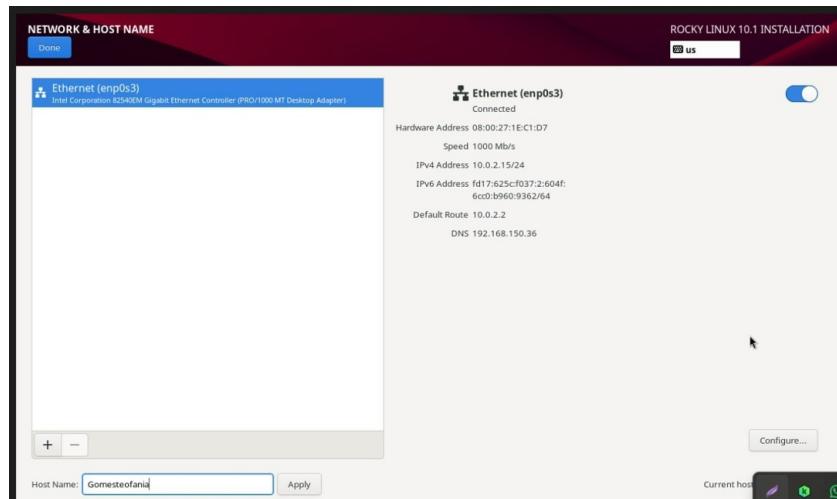


Рис. 3.11: Выбор сети

Затем устанавливаю систему.

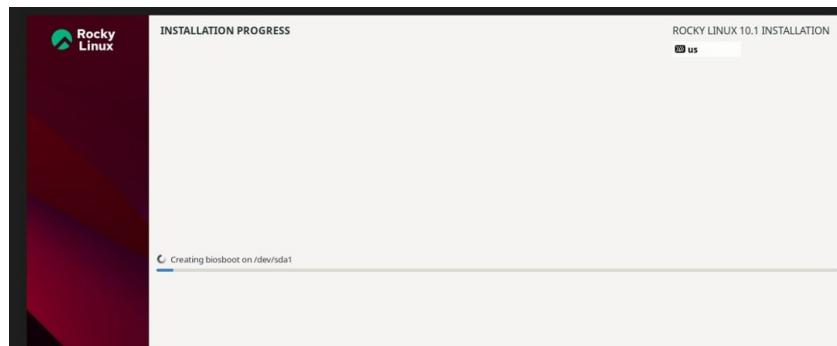
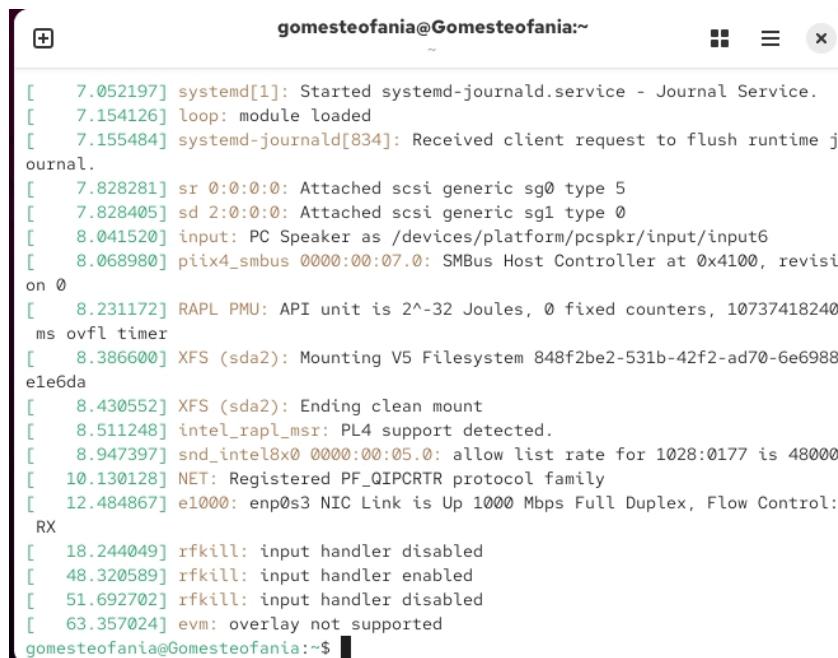


Рис. 3.12: Установка

4 Выполнение дополнительной работы

Запускаю в терминале: dmesg



```
7.052197] systemd[1]: Started systemd-journald.service - Journal Service.
[ 7.154126] loop: module loaded
[ 7.155484] systemd-journald[834]: Received client request to flush runtime journal.
[ 7.828281] sr 0:0:0:0: Attached scsi generic sg0 type 5
[ 7.828405] sd 2:0:0:0: Attached scsi generic sg1 type 0
[ 8.041520] input: PC Speaker as /devices/platform/pcspkr/input/input6
[ 8.068980] piix4_smbus 0000:00:07.0: SMBus Host Controller at 0x4100, revision 0
[ 8.231172] RAPL PMU: API unit is 2^-32 Joules, 0 fixed counters, 10737418240 ms ovfl timer
[ 8.386600] XFS (sda2): Mounting V5 Filesystem 848f2be2-531b-42f2-ad70-6e6988e1e6da
[ 8.430552] XFS (sda2): Ending clean mount
[ 8.511248] intel_rapl_msr: PL4 support detected.
[ 8.947397] snd_intel8x0 0000:00:05.0: allow list rate for 1028:0177 is 48000
[ 10.130128] NET: Registered PF_QIPCRTR protocol family
[ 12.484867] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[ 18.244049] rfkill: input handler disabled
[ 48.320589] rfkill: input handler enabled
[ 51.692702] rfkill: input handler disabled
[ 63.357024] evm: overlay not supported
gomesteofania@Gomesteofania:~$
```

Рис. 4.1: dmesg

dmesg | grep -i "detected", чтобы получить информацию о процессоре.

```
[ 05.357024] evm: overlay not supported
gomesteofania@Gomesteofania:~$ sudo dmesg | grep -i "Detected"
[ 0.000000] Hypervisor Detected: KVM
[ 0.000016] tsc: Detected 2495.998 MHz processor
```

Рис. 4.2: Частота процессора

dmesg | grep -i "CPU", чтобы получить информацию о модели процессора.

```
gomesteofania@Gomesteofania:~$ sudo dmesg | grep -i "CPU"
[    0.001516] CPU MTTRs all blank - virtualized system.
[    0.013577] ACPI: SSDT 0x00000007FFF02A0 000038 (v01 VBOX   VBOXCPU) 00000000
2 VBOX 0002A122)
```

Рис. 4.3: Модель процессора

dmesg | grep -i “memory”, чтобы получить информацию о памяти.

```
gomesteofania@Gomesteofania:~$ sudo dmesg | grep -i "memory"
[    0.000000] DMI: Memory slots populated: 0/0
[    0.013579] ACPI: Reserving FACP table memory at [mem 0x7fff00f0-0x7fff01e3]
[    0.013580] ACPI: Reserving DSDT table memory at [mem 0x7fff02e0-0x7fff26321]
```

Рис. 4.4: Объем доступной оперативной памяти

dmesg | grep -i “detected”, чтобы получить информацию о гипервизоре.

```
gomesteofania@Gomesteofania:~$ sudo dmesg | grep -i "Detected"
[    0.000000] Hypervisor detected: KVM
```

Рис. 4.5: Тип гипервизора

sudo fdisk -l, чтобы получить информацию о файловой системе корневого разделяемого устройства.

```
gomesteofania@Gomesteofania:~$ sudo fdisk -l
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 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: gpt
Disk identifier: C4FE44AB-FAAD-4BBE-9D6E-1C7B5FD93FDD

Device      Start      End  Sectors Size Type
/dev/sda1    2048     4095     2048  1M BIOS boot
/dev/sda2    4096  2101247  2097152   1G Linux extended boot
/dev/sda3  2101248 41940991 39839744  19G Linux LVM
```

Рис. 4.6: Тип файловой системы

dmesg | grep -i “mount”, чтобы получить информацию о монтировании файловых систем.

```
gomesteofania@Gomesteofania:~$ sudo dmesg | grep -i "Mount"
[    0.099150] Mount-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[    0.099150] Mountpoint-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[    4.140555] XFS (dm-0): Mounting V5 Filesystem cd69a111-2f35-4935-ad9c-a766a642c04d
[    5.777429] systemd[1]: Set up automount proc-sys-fs-binfmt_misc.automount - Arbitrary Executable File Formats File System Automount Point.
[    5.791430] systemd[1]: Mounting dev-hugepages.mount - Huge Pages File System...
[    5.793692] systemd[1]: Mounting dev-mqueue.mount - POSIX Message Queue File System...
[    5.799438] systemd[1]: Mounting sys-kernel-debug.mount - Kernel Debug File System...
```

Рис. 4.7: Последовательность монтирования файловых

5 Ответы на контрольные вопросы

1. Учетная запись содержит необходимые для идентификации пользователя при подключении к системе данные, а также 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 (CID) (группа, к которой относится пользователь. Она, как минимум, одна, по умолчанию - одна), полное имя (full name) (Могут быть ФИО), домашний каталог (home directory) (каталог, в который попадает пользователь после входа в систему и в котором хранятся его данные), начальная оболочка (login shell) (командная оболочка, которая 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 объема каталога - du ; для создания / удаления каталогов - mkdir/rmdir; для создания / удаления файлов - touch/rm; для задания определенных 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

На практике я научилась устанавливать операционную систему на виртуальную машину и настраивать минимально необходимые сервисы для работы.