

Отчет по лабораторной работе №1

Информационная безопасность

Софич Андрей Геннадьевич

Содержание

1	Цель работы	1
2	Задание.....	2
3	Выполнение лабораторной работы.....	2
4	Выполнение дополнительных заданий	6
5	Ответы на контрольные вопросы.....	8
6	Выводы	8
	Список литературы	8

Список иллюстраций

Рис. 1: Создание виртуальной машины	2
Рис. 2: Основные характеристики	2
Рис. 3: Выделение внутренней памяти	3
Рис. 4: Образ диска	3
Рис. 5: Запуск	4
Рис. 6: Выбор языка	4
Рис. 7: Выбор окружения	5
Рис. 8: Отключение kdump	5
Рис. 9: Настройка узла	6
Рис. 10: Установка	6
Рис. 11: Характеристики компьютера	7
Рис. 12: Тип гипервизора.....	7
Рис. 13: Тип файловой системы	7
Рис. 14: Последовательность монтирования файловых систем.....	7

Список таблиц

Элементы списка иллюстраций не найдены.

1 Цель работы

Целью работы является умение устанавливать операционную систему на виртуальную машину а также минимально настраивать работу сервисов

2 Задание

1. Установка и настройка ОП
2. Поиск информации с помощью команды dmesg ...

3 Выполнение лабораторной работы

Скачиваю приложение виртуальной машины а также версию Linux Rocky, на которой буду работать в дальнейшем, создаю виртуальную машину, сначала задаю имя и операционную систему (рис. 1).

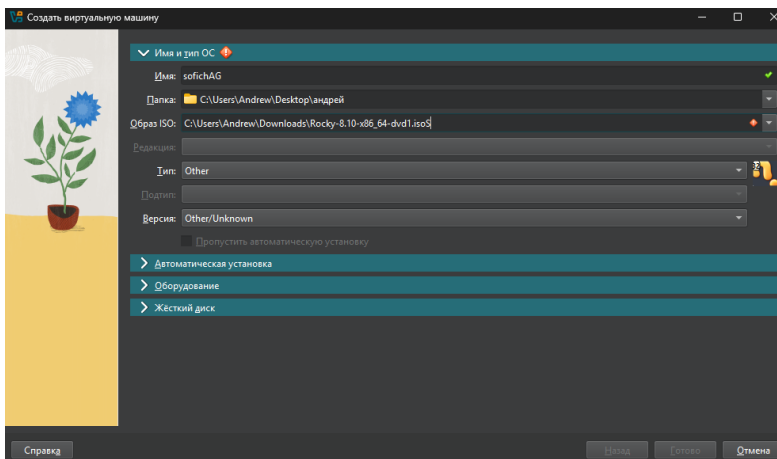


Рис. 1: Создание виртуальной машины

Выделяю на виртуальную машину оперативную память и процессоры (рис. 2).

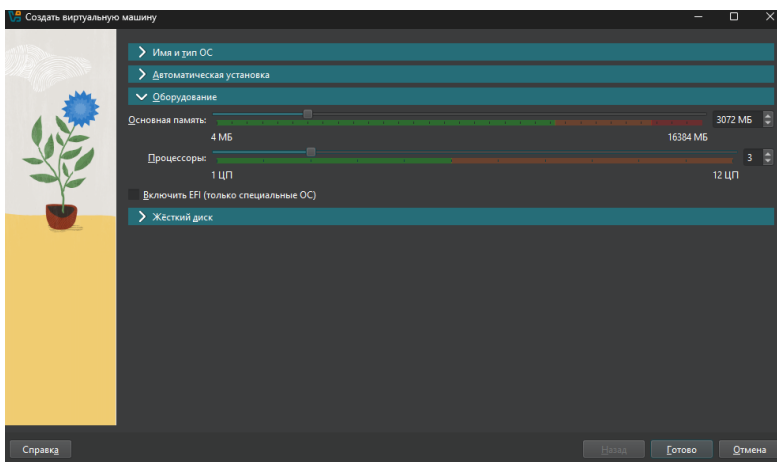


Рис. 2: Основные характеристики

Выделяю на виртуальную машину внутреннюю память- 40 гб (рис. 3).

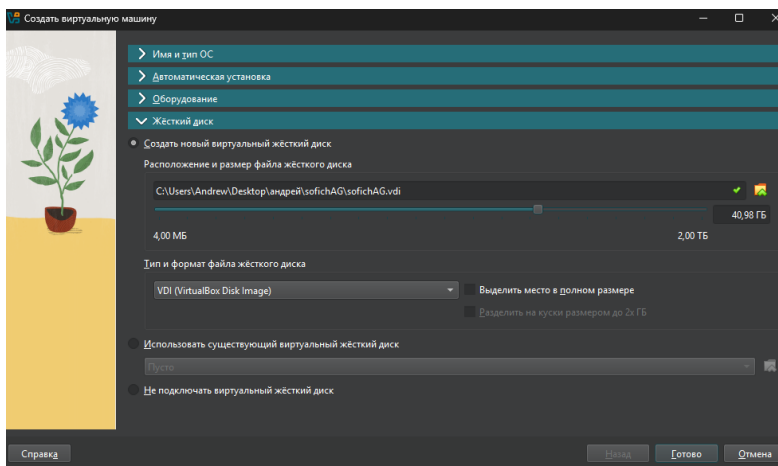


Рис. 3: Выделение внутренней памяти

В носителях подключаю образ диска, после установки он сам пропадет (рис. 4).

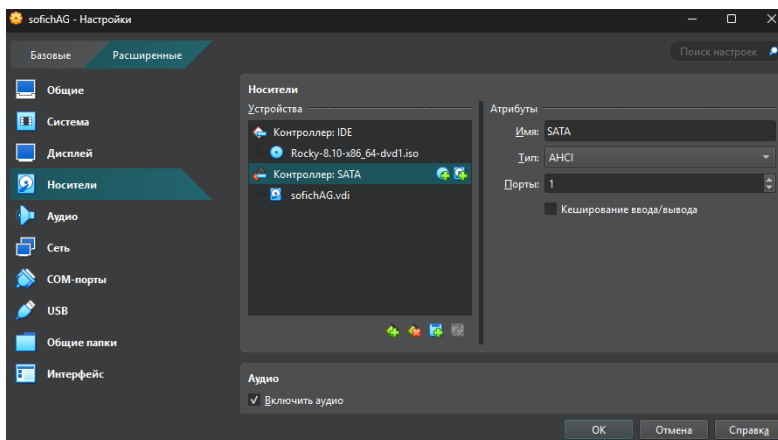


Рис. 4: Образ диска

Запускаю виртуальную машину (рис. 5).

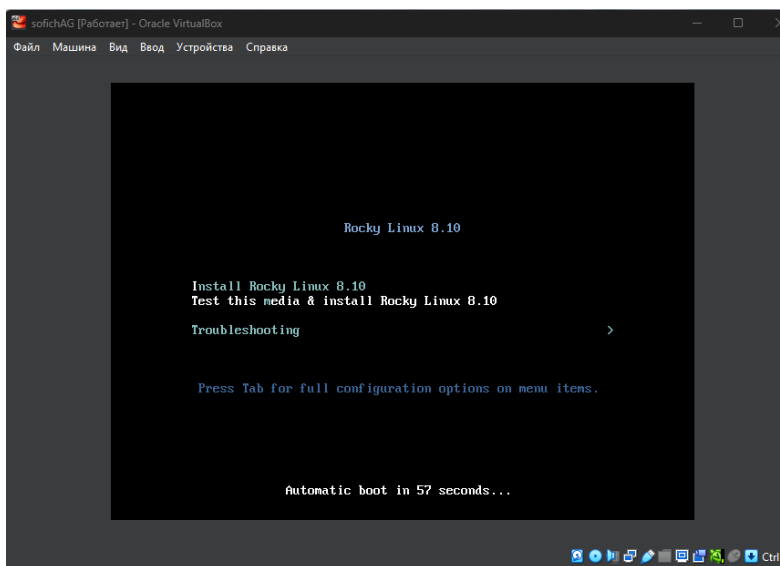


Рис. 5: Запуск

Начинаю настраивать систему, для начала выбираю язык (рис. 6).

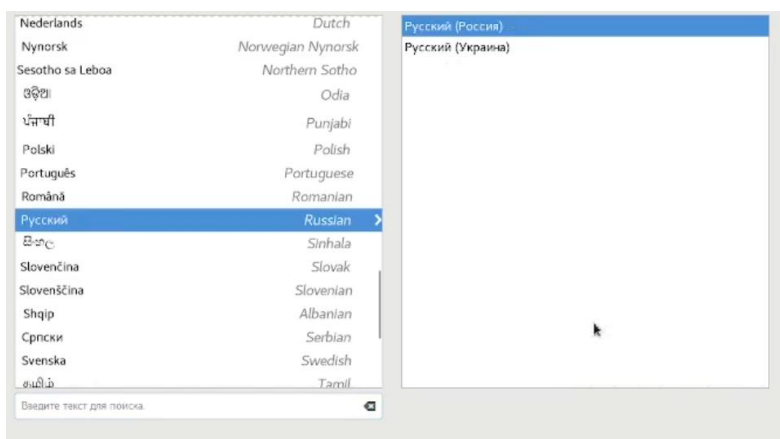


Рис. 6: Выбор языка

В соответствии с требованием лабораторной работы выбираю окружение сервер с GUB и средства разработки в дополнительном программном обеспечении (рис. 7).

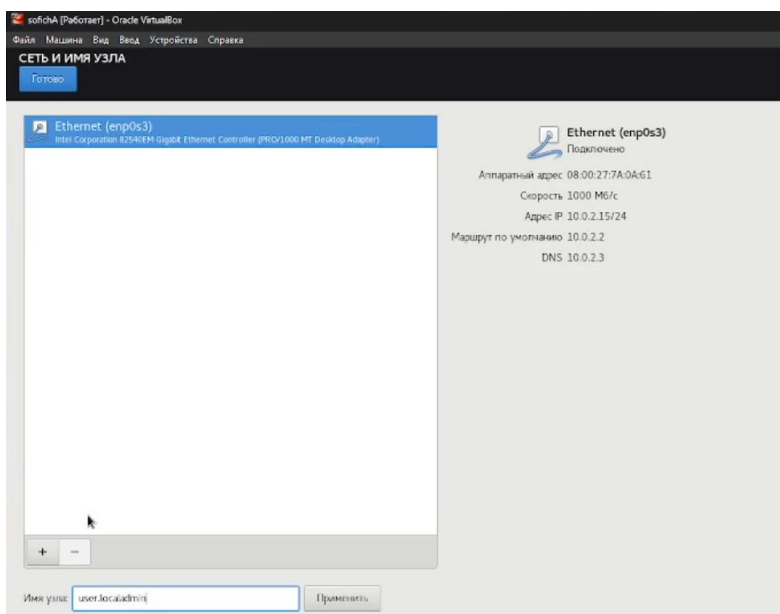


Рис. 9: Настройка узла

Создаю администратора и начинаю установку системы (рис. 10).

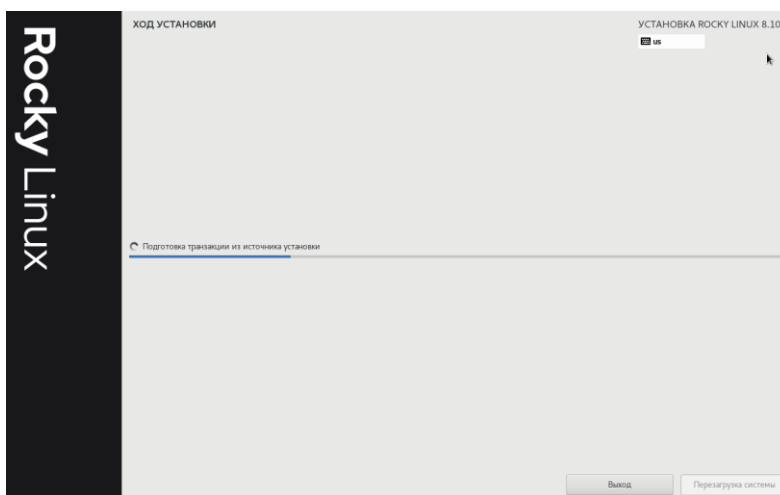


Рис. 10: Установка

4 Выполнение дополнительных заданий

Узнаю версию ядра- 4.18.0-553.el8_10.x86_64, частота процессора-3600.012 МГц, модель процессора-AMD Ryzen 5 3600, просматриваю доступную память (рис. 11).

```
[soficha@sofichaA ~]$ dmesg | grep -i "linux version"
[ 0.000000] Linux version 4.18.0-553.el8_10.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc version 8.5.0 20210514 (Red Hat 8.5.0-22) (GCC)) #1 SMP Fri May 24 13:05:10 UTC 2024
[soficha@sofichaA ~]$ dmesg | grep -i "Detected"
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] tsc: Detected 3600.012 MHz processor
[ 0.987381] hub 1-0:1.0: 12 ports detected
[ 0.998853] hub 2-0:1.0: 12 ports detected
[ 1.290087] system[3]: Detected virtualization oracle.
[ 1.290090] system[1]: Detected architecture x86-64.
[ 4.878923] system[1]: Detected virtualization oracle.
[ 4.878927] system[1]: Detected architecture x86-64.
[soficha@sofichaA ~]$ dmesg | grep -i "CPU0"
[ 0.122917] smpboot: CPU0: AMD Ryzen 5 3600 6-Core Processor (family: 0x17, model: 0x71, stepping: 0x0)
[soficha@sofichaA ~]$ dmesg | grep -i "Memory"
[ 0.000000] ACPI: Reserving FACP table memory at [mem 0xbfff00f0-0xbfff01e3]
[ 0.000000] ACPI: Reserving DSDT table memory at [mem 0xbfff0620-0xbfff2972]
[ 0.000000] ACPI: Reserving FACS table memory at [mem 0xbfff0200-0xbfff023f]
[ 0.000000] ACPI: Reserving FACS table memory at [mem 0xbfff0200-0xbfff023f]
[ 0.000000] ACPI: Reserving APIC table memory at [mem 0xbfff0240-0xbfff02ab]
[ 0.000000] ACPI: Reserving SSDT table memory at [mem 0xbfff02b0-0xbfff061b]
[ 0.000000] Early memory node ranges
[ 0.000000] PM: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.000000] PM: Registered nosave memory: [mem 0x0000f000-0x0009ffff]
[ 0.000000] PM: Registered nosave memory: [mem 0x000a0000-0x000effff]
[ 0.000000] PM: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[ 0.000000] Memory: 261120K/3145272K available (14339K kernel code, 5957K rwddata, 8568K rodata, 2820K init, 13792K bss, 156616K reserved, 0K cma-reserved)
[ 0.023536] Freeing SMP alternatives memory: 36K
[ 0.145134] x86/mm: Memory block size: 128MB
[ 0.757547] Freeing initrd memory: 52248K
[ 0.924511] Non-volatile memory driver v1.3
[ 1.259667] Freeing unused decrypted memory: 2028K
[ 1.260389] Freeing unused kernel image (initmem) memory: 2820K
[ 1.276377] Freeing unused kernel image (text/rodata gap) memory: 2016K
```

Рис. 11: Характеристики компьютера

Обнаружен гипервизор типа KVM (рис. 12).

```
[soficha@sofichaA ~]$ dmesg | grep -i "Hypervisor"
[ 0.000000] Hypervisor detected: KVM
[ 2.329976] vmwgfx 0000:00:02.0: [drm] *ERROR* vmwgfx seems to be running on an unsupported hypervisor.
```

Рис. 12: Тип гипервизора

Команда `sudo fdisk -l` показывает тип файловой системы (рис. 13).

```
[soficha@sofichaA ~]$ sudo fdisk -l
[sudo] пароль для soficha:
Диск /dev/sda: 40,1 GiB, 43037949952 байт, 84058496 секторов
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0x4a40e087

Устр-во    Загрузочный  начало      Конеч      Секторы  Размер  Идентификатор  Тип
/dev/sda1  *            2048        2099199    2097152    1G      83 Linux
/dev/sda2                2099200    84058111  81958912    39,1G    8e Linux LVM

Диск /dev/mapper/rl-root: 36 GiB, 38700843008 байт, 75587584 секторов
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт

Диск /dev/mapper/rl-swap: 3 GiB, 3258974208 байт, 6365184 секторов
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
```

Рис. 13: Тип файловой системы

Далее показана последовательность монтирования файловых систем (рис. 14).

```
[soficha@sofichaA ~]$ dmesg | grep -i "Mount"
[ 0.011136] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, vmalloc)
[ 0.011143] Mountpoint-cache hash table entries: 8192 (order: 4, 65536 bytes, vmalloc)
[ 4.009043] XFS (dm-0): Mounting V5 Filesystem
[ 4.049180] XFS (dm-0): Ending clean mount
[ 7.779729] XFS (sda1): Mounting V5 Filesystem
[ 8.403333] XFS (sda1): Ending clean mount
```

Рис. 14: 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 приобрел навыки установки операционной системы.

Список литературы