

Отчёт по лабораторной работе 1

Простейший вариант

Арфонос Дмитрий

Содержание

| | | |
|---|--------------------------------|----|
| 1 | Цель работы | 5 |
| 2 | Выполнение лабораторной работы | 6 |
| 3 | Самостоятельная работа | 22 |
| 4 | Вывод | 25 |

Список иллюстраций

| | | |
|------|--|----|
| 2.1 | Настройка каталогов | 6 |
| 2.2 | Настройка хост клавиш | 7 |
| 2.3 | Создание виртуальной машины | 7 |
| 2.4 | Установка оперативки | 8 |
| 2.5 | жесткий диск | 8 |
| 2.6 | видеопамять | 9 |
| 2.7 | уч запись | 10 |
| 2.8 | дата и время | 11 |
| 2.9 | Конфигурация | 11 |
| 2.10 | установка | 12 |
| 2.11 | образ опт диска | 13 |
| 2.12 | Запуск | 14 |
| 2.13 | Запуск root | 15 |
| 2.14 | обновление | 15 |
| 2.15 | обновление | 16 |
| 2.16 | установка программ | 16 |
| 2.17 | установка программ | 17 |
| 2.18 | установка таймера | 17 |
| 2.19 | Отключение SELinux | 17 |
| 2.20 | Установите средства разработки: | 18 |
| 2.21 | Установите средства разработки: | 18 |
| 2.22 | установка программ | 18 |
| 2.23 | установка пдрайвера | 19 |
| 2.24 | перезагрузка | 19 |
| 2.25 | установка имени хоста | 20 |
| 2.26 | создание общей папки | 20 |
| 2.27 | установка программ | 20 |
| 2.28 | установка программ | 21 |
| 3.1 | последовательность загрузки | 22 |
| 3.2 | Версия ядра Linux | 22 |
| 3.3 | Частота процессора | 23 |
| 3.4 | Модель процессора | 23 |
| 3.5 | Объём доступной оперативной памяти | 23 |
| 3.6 | гипервизор | 23 |
| 3.7 | тип файловой системы | 24 |

| | | |
|-----|--|----|
| 3.8 | Последовательность монтирования файловых систем. | 24 |
|-----|--|----|

1 Цель работы

Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

2 Выполнение лабораторной работы

1 Настроиваю каталог для виртуальной машины (рис. [2.1])

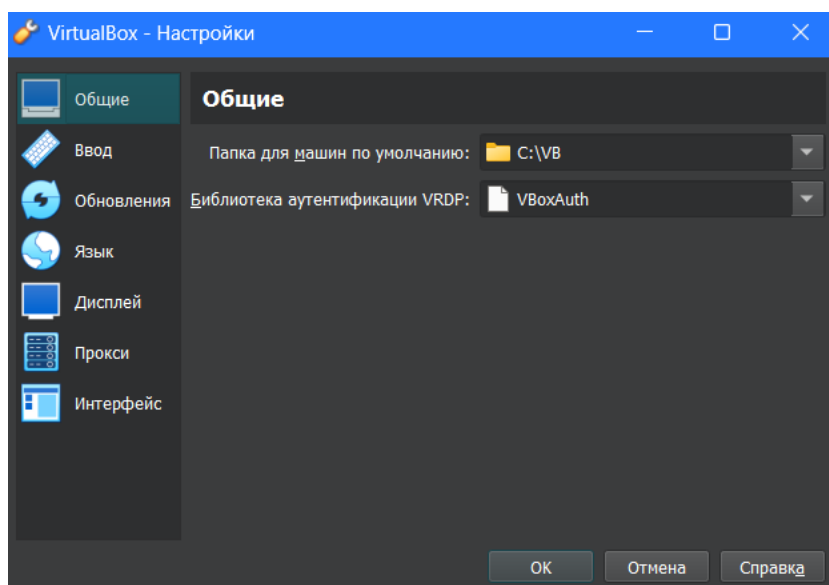


Рис. 2.1: Настройка каталогов

2 Настроиваю хост-клавиши с ctrl на num-lock (рис. [2.2])

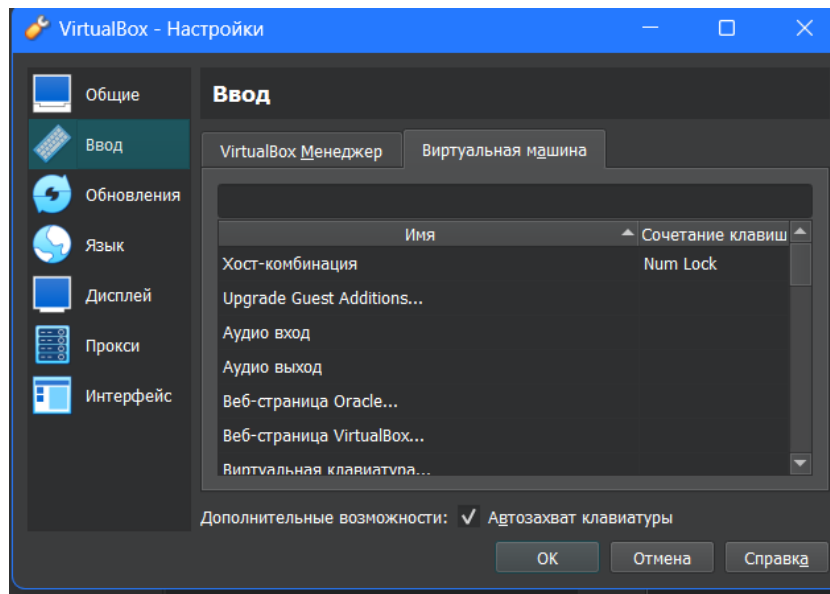


Рис. 2.2: Настройка хост клавиш

3 Создаю виртуальную машину: выбираю установленный привод оптических дисков и даю ей имя своего логина(рис. [2.3])

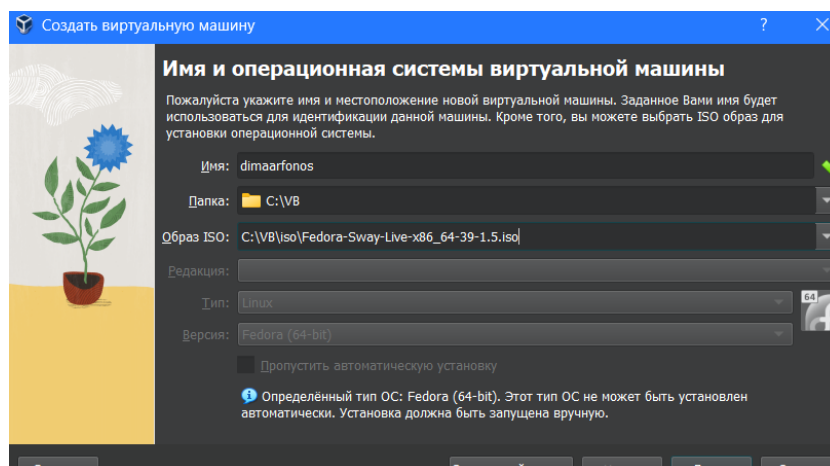


Рис. 2.3: Создание виртуальной машины

4 Устанавливаю объем оперативной памяти 2 гб и 3 цп (рис. [2.4])

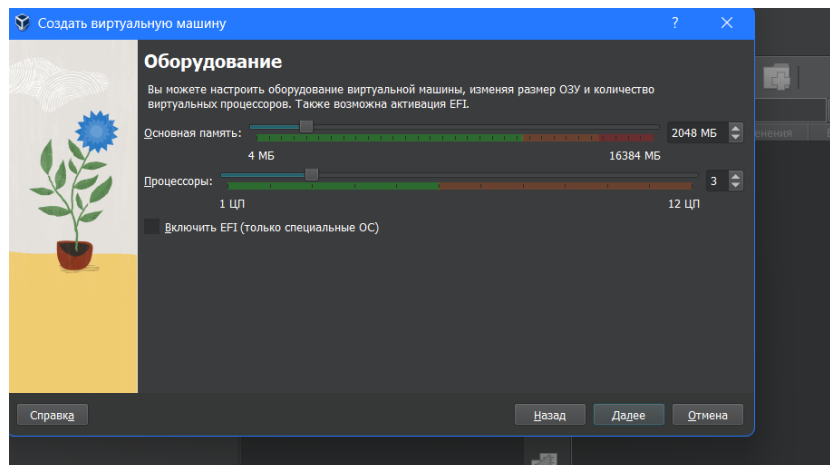


Рис. 2.4: Установка оперативки

5 Устанавливаю объем жесткого диска 80 гб (рис. [2.5])

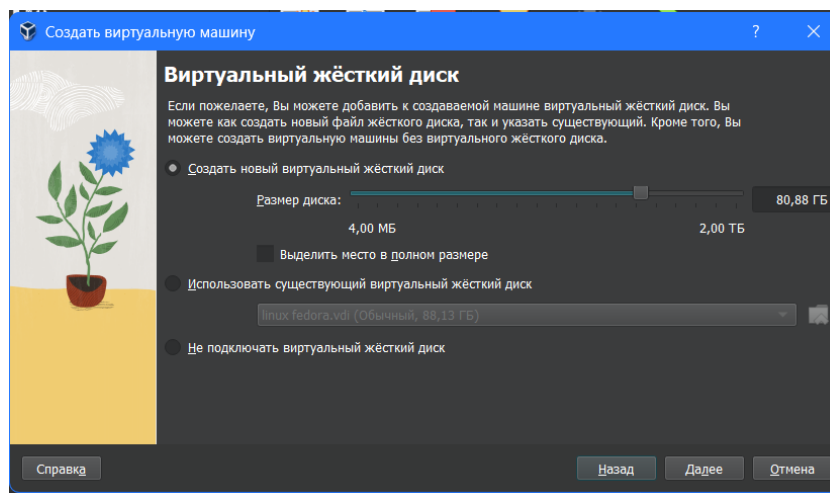


Рис. 2.5: жесткий диск

6 Устанавливаю объем видеопамяти(рис. [2.6])

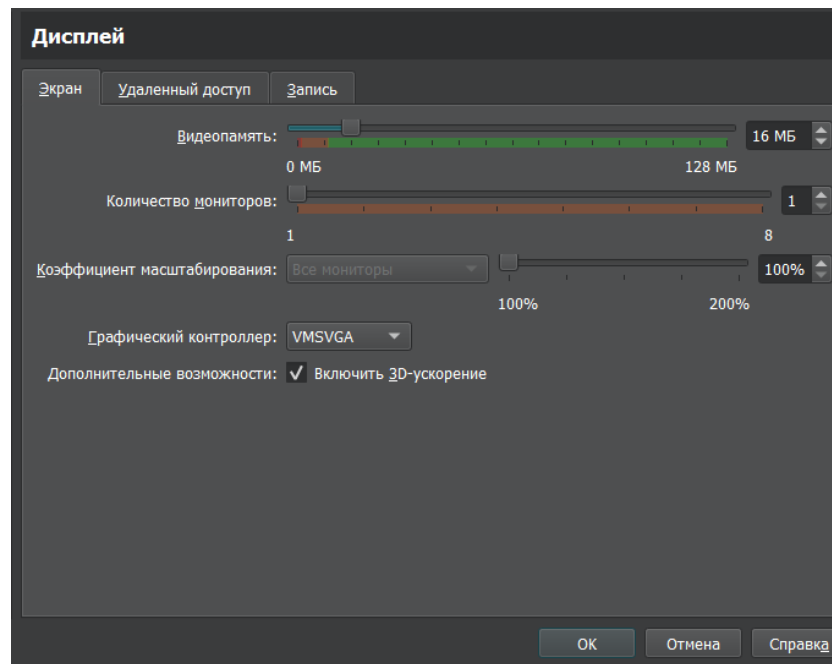


Рис. 2.6: видеопамять

7 Создаю учетную запись для виртуальной машины(рис. [2.7])

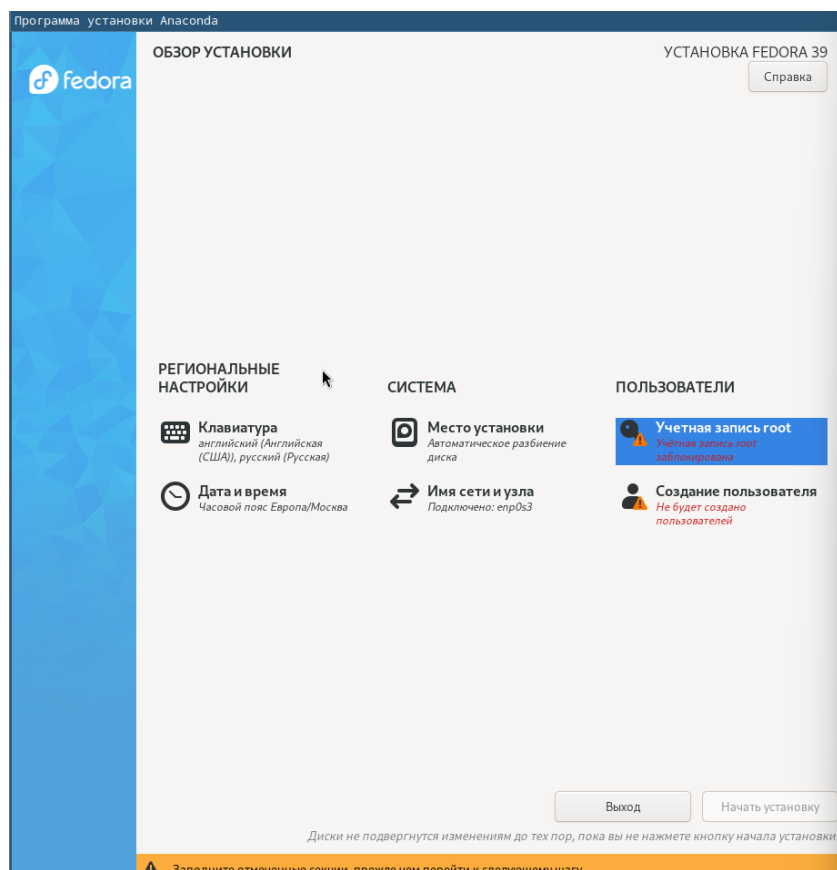


Рис. 2.7: уч запись

8 Настраиваю дату и время(рис. [2.8])

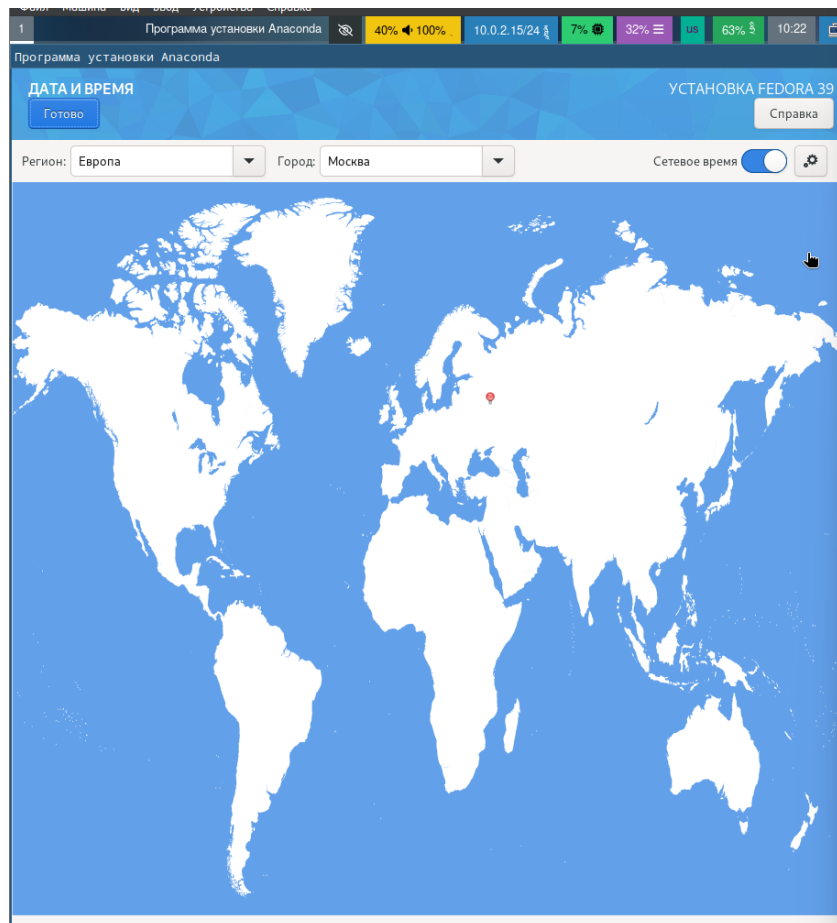


Рис. 2.8: дата и время

9

Устанавливаю автоматическую конфигурацию.(рис. [2.9])

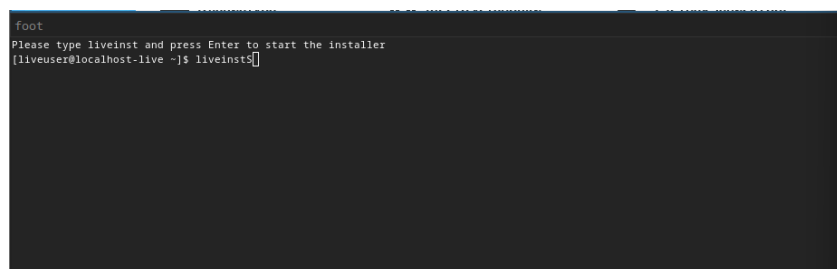


Рис. 2.9: Конфигурация

10

Ждем пока все установится для работы.(рис. [2.10])

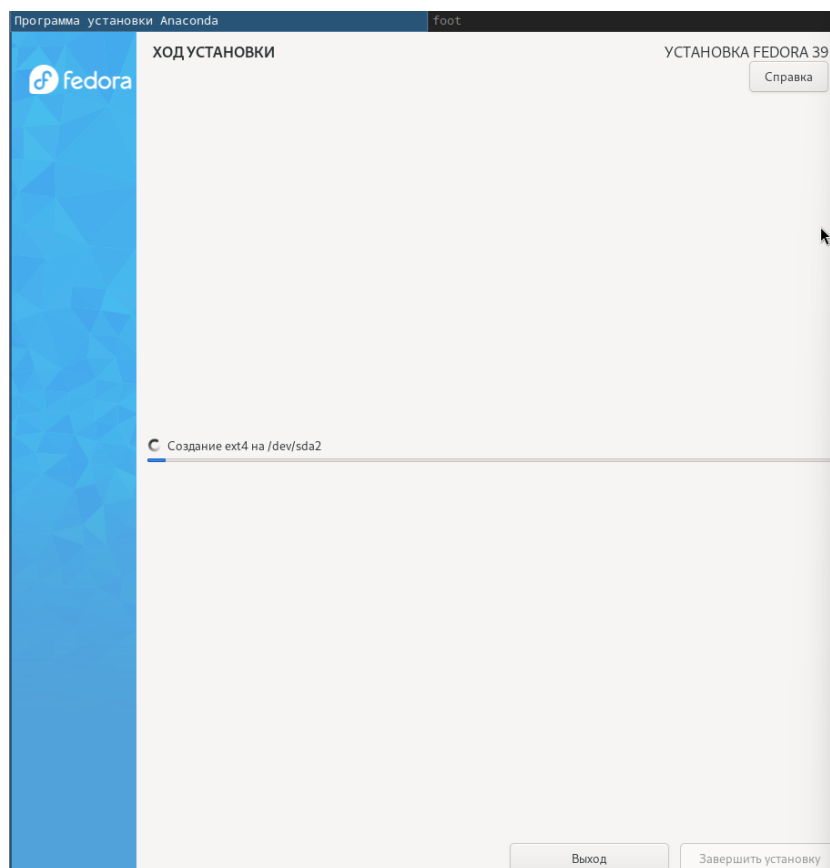


Рис. 2.10: установка

11

Добавляю новый привод оптических дисков.(рис. [2.11])

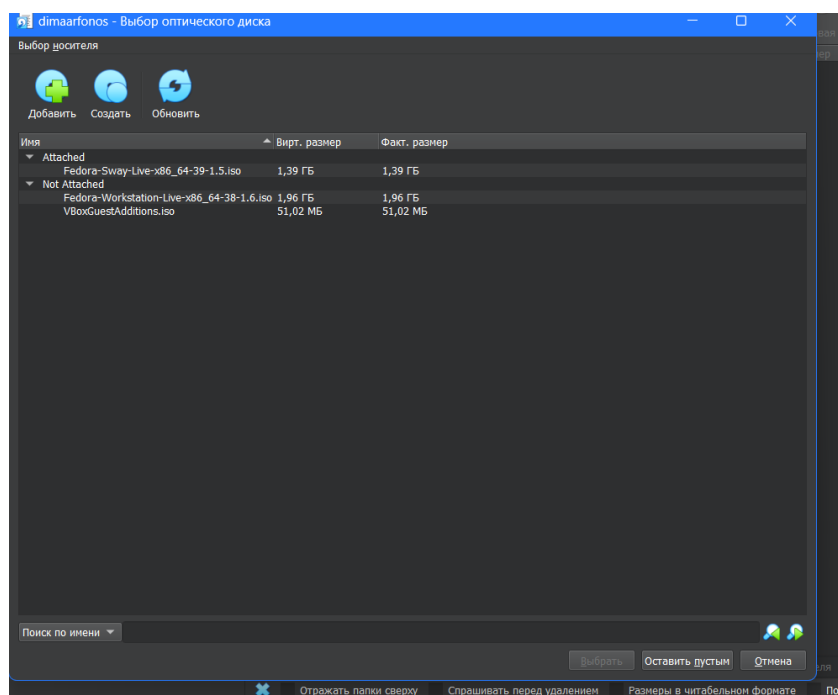


Рис. 2.11: образ опт диска

12

Запускаю виртуальную машину.(рис. [2.12])

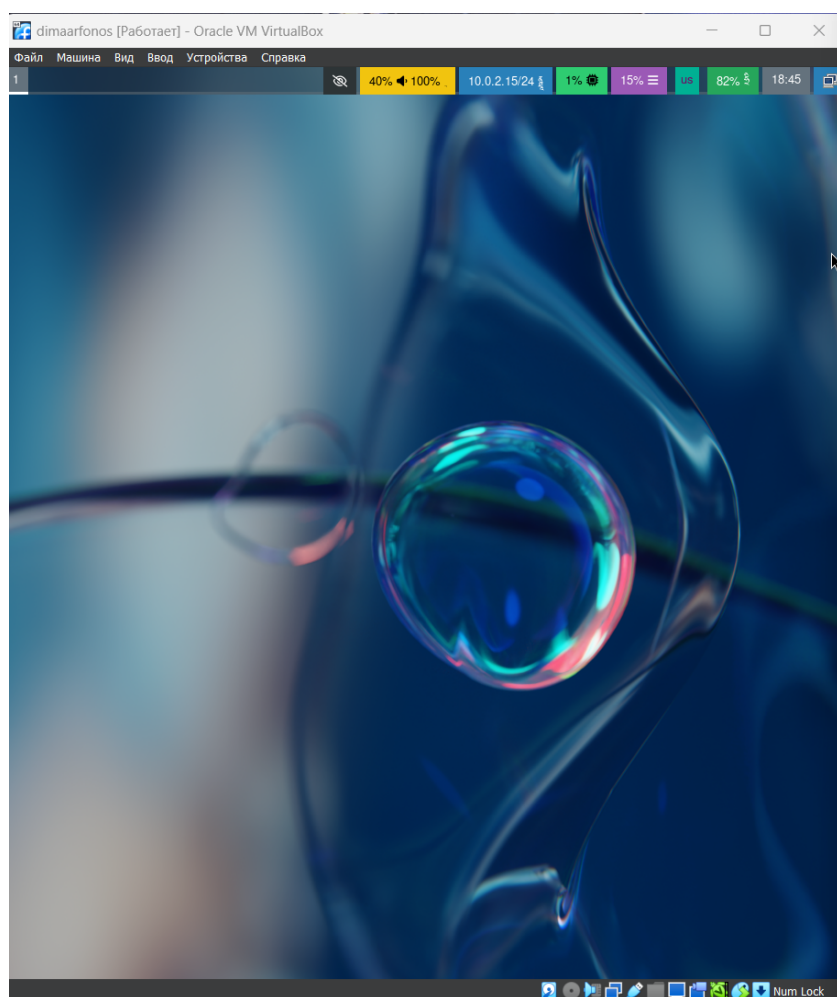


Рис. 2.12: Запуск

13

Запускаю режим суперпользователя.(рис. [2.13])

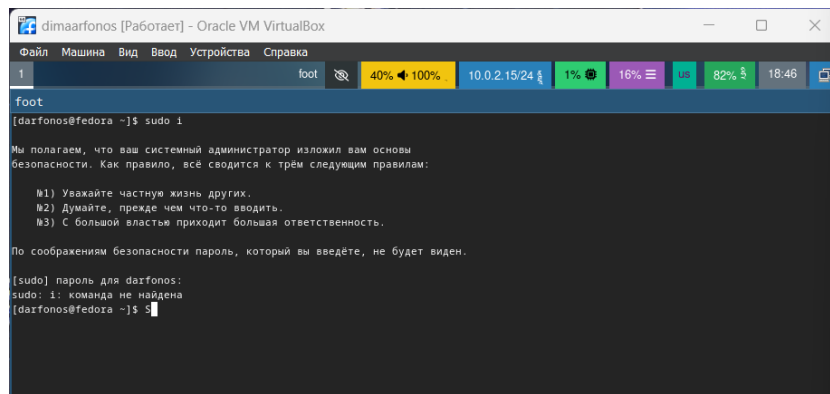


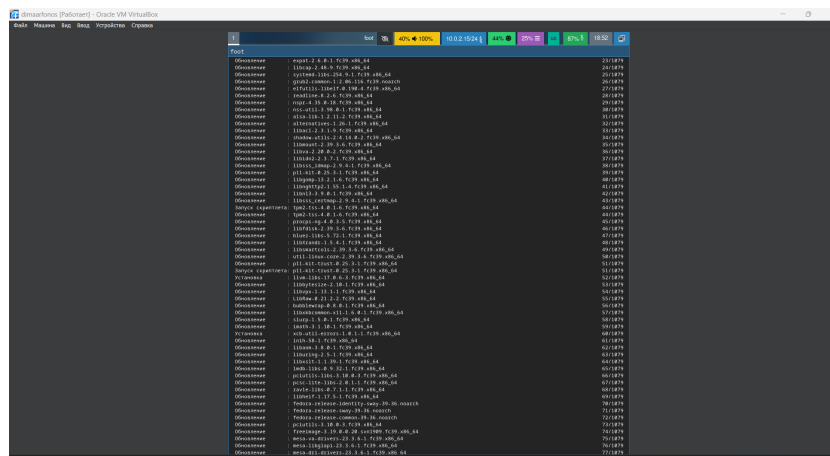
Рис. 2.13: Запуск root

14

Обновляю все пакеты.(рис. [2.15])

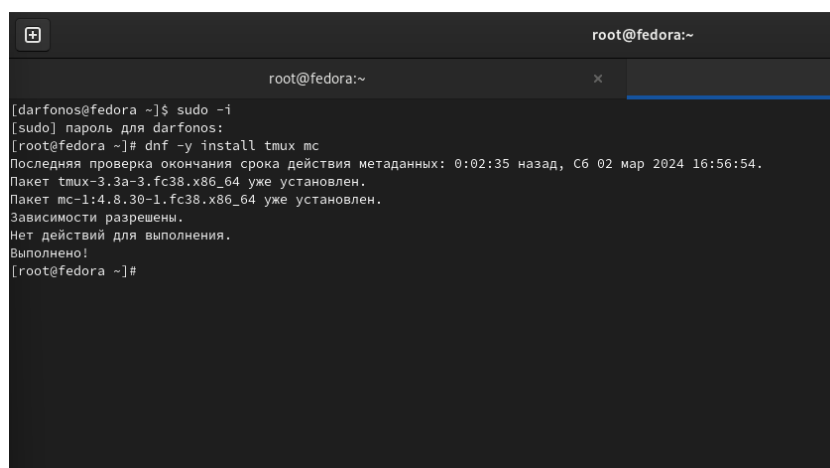


Рис. 2.14: обновление



15

Программа для удобства работы в консоли.(рис. [2.16])



16

Ввожу команду для автообновления системы.(рис. [2.17])


```
[root@fedora ~]# dnf install dnf-automatic
Последняя проверка окончания срока действия метаданных: 0:06:14 назад, Сб 02 мар 2024 16:56:54.
Зависимости разрешены.
=====
Пакет                Архитектура          Версия                Р
=====
Установка:
dnf-automatic         noarch                4.19.0-1.fc38         up
Обновление:
dnf                   noarch                4.19.0-1.fc38         up
dnf-data              noarch                4.19.0-1.fc38         up
libdnf                x86_64               0.73.0-1.fc38         up
замена libdnf.x86_64 0.72.0-1.fc38
python3-dnf           noarch                4.19.0-1.fc38         up
python3-hawkey        x86_64               0.73.0-1.fc38         up
замена python3-hawkey.x86_64 0.72.0-1.fc38
python3-libdnf        x86_64               0.73.0-1.fc38         up
замена python3-libdnf.x86_64 0.72.0-1.fc38
yum                   noarch                4.19.0-1.fc38         up
=====
Результат транзакции
=====
Установка 1 Пакет
Обновление 7 Пакетов
Общий размер: 2.8 М
Объем загрузки: 1.6 М
```

Рис. 2.17: установка программ

17

Устанавливаю таймер(рис. [2.18])

```
[root@fedora ~]# systemctl enable --now dnf-automatic.timer
Created symlink /etc/systemd/system/timers.target.wants/dnf-automatic.timer → /usr/lib/systemd/system/dnf-automatic.timer
[root@fedora ~]#
```

Рис. 2.18: установка таймера

18

Отключение SELinux изменяя конфигурацию нужной папки (рис. [2.19])

```
Открыть ▾ + config /etc/selinux
#
# grubby --update-kernel ALL --remove-args selinux
#
SELINUX=permissive
# SELINUXTYPE= can take one of these three values:
# targeted - Targeted processes are protected,
# minimum - Modification of targeted policy. Only selected processes are
protected.
# mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Рис. 2.19: Отключение SELinux

19

Установите средства разработки для драйверов виртуал бокс.(рис. [2.21])

```
darfonos@fedora ~]$ sudo -i
[sudo] пароль для darfonos:
Попробуйте ещё раз.
[sudo] пароль для darfonos:
[root@fedora ~]# dnf -y group install "Development Toolset"
```

Рис. 2.20: Установите средства разработки:

```
xapian-core-libs                x86_64                1.4.23-1.fc38
xz-devel                       x86_64                5.4.1-1.fc38
Установка слабых зависимостей:
elfutils-debuginfod-client-devel x86_64                0.190-2.fc38
kernel-devel                   x86_64                6.7.6-100.fc38
Установка групп:
Development Tools

Результат транзакции
=====
Установка   35 Пакетов
Обновление   3 Пакета

Общий размер: 128 М
Объем загрузки: 126 М
Загрузка пакетов:
[SKIPPED] gettext-0.21.1-2.fc38.x86_64.rpm: Already downloaded
[SKIPPED] elfutils-0.190-2.fc38.x86_64.rpm: Already downloaded
[SKIPPED] elfutils-debuginfod-client-0.190-2.fc38.x86_64.rpm: Already downloaded
[SKIPPED] elfutils-libs-0.190-2.fc38.x86_64.rpm: Already downloaded
(5/38): diffstat-1.65-2.fc38.x86_64.rpm
(6/38): bison-3.8.2-4.fc38.x86_64.rpm
(7/38): ed-1.10-2.fc38.x86_64.rpm
(8/38): flex-2.6.4-12.fc38.x86_64.rpm
(9/38): dyninst-12.2.0-2.fc38.x86_64.rpm
(10/38): md4-1.4.10-5.fc38.x86_64.rpm
(11/38): patch-2.7.6-19.fc38.x86_64.rpm
(12/38): doxygen-1.9.6-7.fc38.x86_64.rpm
(13/38): patchutils-0.4.2-9.fc38.x86_64.rpm
(14/38): tbb-2020.3-16.fc38.x86_64.rpm
```

Рис. 2.21: Установите средства разработки:

20

Установка dkms(рис. [2.22])

```
Установлен:
dkms-3.0.12-1.fc38.noarch                kernel-devel-matched-6.7.6-100.fc38.x86_64                openssl-1.1.1-1.fc38.x86_64

Выполнено!
[root@fedora ~]# dnf -y install texlive texlive-l
[1] 0:bash*
```

Рис. 2.22: установка программ

21

Подмонтирую диск и запускаю установку драйвера(рис. [2.23])

```
[root@fedora ~]# mount /dev/sr0 /media
mount: /media: /dev/sr0 already mounted on /run/media/darfonos/VBox_GAs_7.0.10.
dmesg(1) may have more information after failed mount system call.
[root@fedora ~]# /media/VBoxLinuxAdditions.run
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.0.10 Guest Additions for Linux 100%
VirtualBox Guest Additions installer
Removing installed version 7.0.10 of VirtualBox Guest Additions...
```

Рис. 2.23: установка пдрайвера

22

Перезагружаю машину(рис. [2.24])

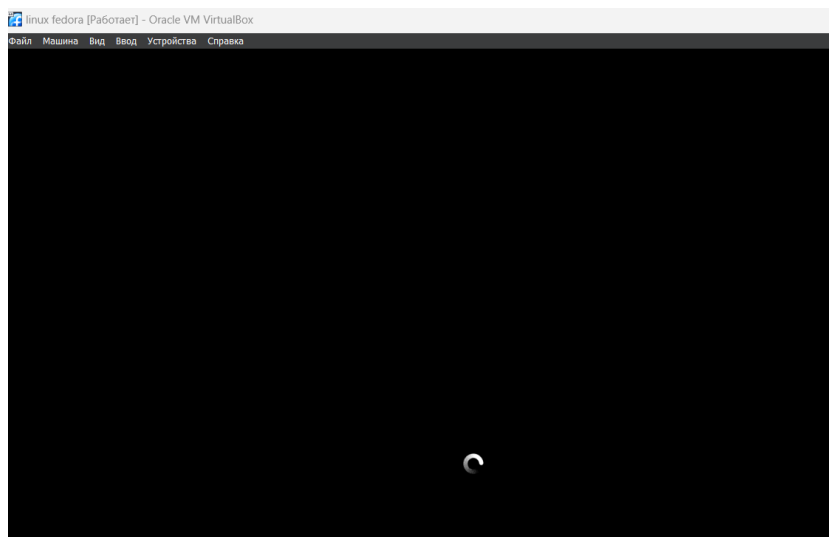


Рис. 2.24: перезагрузка

23

Установите и проверка имени хоста(рис. [2.25])

```
[root@fedora ~]# hostnamectl set-hostname darfonos
[root@fedora ~]# hostnamectl
      Static hostname: darfonos
            Icon name: computer-vm
          Chassis: vm
        Machine ID: 9a19a9e90e10496e84c752ab028e53ce
          Boot ID: a78bc88e91bb43a69b6497f9547a4b6e
    Virtualization: oracle
  Operating System: Fedora Linux 38 (Workstation Edition)
        CPE OS Name: cpe:/o:fedoraproject:fedora:38
      OS Support End: Tue 2024-05-14
OS Support Remaining: 2month 1w 4d
          Kernel: Linux 6.7.6-100.fc38.x86_64
[
  Architecture: x86-64
  Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
  Firmware Version: VirtualBox
    Firmware Date: Fri 2006-12-01
[root@fedora ~]#
```

Рис. 2.25: установка имени хоста

24

Внутри виртуальной машины добавляю своего пользователя в группу vboxsf(рис. [2.26])

```
Firmware Date: Fri 2006-12-01
[root@fedora ~]# gpasswd -a darfonos vboxsf
Добавление пользователя darfonos в группу vboxsf
[root@fedora ~]#
```

Рис. 2.26: создание общей папки

25

Устанавливаю пакет pandoc для редактирование текстов(рис. [2.27])

```
[darfonos@darfonos ~]$ sudo -i
[sudo] пароль для darfonos:
[root@darfonos ~]# dnf -y install pandoc
```

Рис. 2.27: установка программ

26

Устанавливаю из внешних источников репозиторий с pandoc-crossref(рис. [2.28])

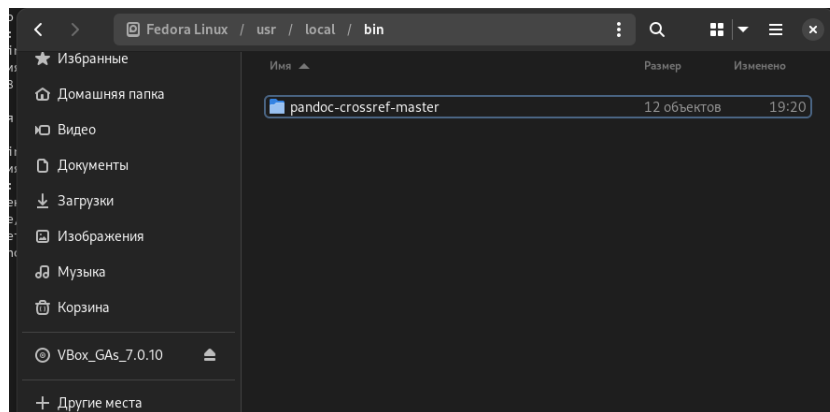


Рис. 2.28: установка программ

27

Устанавливаю полную версию пакетов texlive(рис. [??])

```
[root@darfonos ~]# dnf -y install texlive
Последняя проверка окончания срока действия метаданных: 2:28:40 назад, Сб 02 мар 2024 16:56:54.
Пакет texlive-10:2022-65.fc38.noarch уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
[root@darfonos ~]# ss
```

{#fig:27 width=70% }

3 Самостоятельная работа

29

В окне терминала проанализирую последовательность загрузки системы, выполнив команду `dmesg`(рис. [3.1])

```
[ 0.000000] Linux version 6.7.6-100.fc38.x86_64 (mockbuild@cle7225e7d254ae18cc5108caa4c0fa8) (gcc (GCC) 13.2.1 20231011 (Red Hat 13.2.1-4), GNU ld version 2.39-16.fc38) #1 SMP PREEMPT_DYNAMIC Fri Feb 23 18:29:24 UTC 2024
[ 0.000000] Command line: BOOT_IMAGE=(hdb,gpt2)/vmlinuz-6.7.6-100.fc38.x86_64 root=UUID=1fd41b0b-fdd9-4c27-a869-f872a029033c ro rootflags=subvol=root rhgb quiet
[ 0.000000] [Firmware Bug]: TSC doesn't count with P0 frequency!
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbfff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000001000000-0x0000000000fffff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000ffff000-0x0000000000fffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000ffffc000-0x00000000fffffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000010000000-0x000000001000ffff] usable
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] APIC: static calls initialized
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: Imnotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using mrs 4b564d01 and 4b564d00
[ 0.000000] kvm-clock: using sched offset of 594447676125 cycles
[ 0.000000] clocksource: kvm-clock: mask: 0xffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881590591483 ns
[ 0.000015] tsc: Detected 2995.994 MHz processor
[ 0.002694] e820: update [mem 0x00000000-0x0000ffff] usable => reserved
[ 0.002702] e820: remove [mem 0x00000000-0x0000ffff] usable
[ 0.002708] last_pfn = 0x120000 max_arch_pfn = 0x400000000
[ 0.002736] MTRRs disabled by BIOS
[ 0.002739] x86 PAT: Configuration [0-7]: WB WC UC- UC WB WP UC- WT
[ 0.002759] last_pfn = 0xdffff max_arch_pfn = 0x400000000
[ 0.002789] found SMP MP-table at [mem 0x0009ffff-0x0009ffff]
[ 0.002951] RAMDISK: [mem 0x3f6b000-0x3f6dffff]
[ 0.002958] ACPI: Early table checksum verification disabled
[ 0.002962] ACPI: RSDP 0x0000000000000000 000024 (v02 VBOX )
```

Рис. 3.1: последовательность загрузки

28

Получаю информацию о Версия ядра Linux(рис. [3.2])

```
[root@darfonos ~]# dmesg | grep -i "Linux version"
[ 0.000000] Linux version 6.7.6-100.fc38.x86_64 (mockbuild@cle7225e7d254ae18cc5108caa4c0fa8) (gcc (GCC) 13.2.1 20231011 (Red Hat 13.2.1-4), GNU ld version 2.39-16.fc38) #1 SMP PREEMPT_DYNAMIC Fri Feb 23 18:29:24 UTC 2024
[root@darfonos ~]#
```

Рис. 3.2: Версия ядра Linux

29

Узнаю частоту процессора(рис. [3.3])

```
[root@darfonos ~]# dmesg | grep -i "Mhz processor"
[ 0.000015] tsc: Detected 2095.994 MHz processor
[root@darfonos ~]#
```

Рис. 3.3: Частота процессора

30

Узнаю Модель процессора(рис. [3.4])

```
[root@darfonos ~]# dmesg | grep -i "CPU0"
[ 0.283978] smpboot: CPU0: AMD Ryzen 5 5500U with Radeon Graphics (family: 0x17, model: 0x68, stepping: 0x1)
[root@darfonos ~]#
```

Рис. 3.4: Модель процессора

31

Вывожу на экран объём доступной оперативной памяти(рис. [3.5])

```
[root@darfonos ~]# dmesg | grep -i "available"
[ 0.004643] On node 0, zone DMA: 1 pages in unavailable ranges
[ 0.004672] On node 0, zone DMA: 97 pages in unavailable ranges
[ 0.013541] On node 0, zone Normal: 16 pages in unavailable ranges
[ 0.017287] [mem 0xe0000000-0xfebfffff] available for PCI devices
[ 0.055417] Memory: 3963180K/4193848K available (20480K kernel code, 3276K rwddata, 14748K rodata, 4588K init, 4892K b
[ 0.284355] Performance Events: PMU not available due to virtualization, using software events only.
[ 5.232909] vmwgfx 0000:00:02.0: [drm] Available shader model: SM_5.
[root@darfonos ~]#
```

Рис. 3.5: Объём доступной оперативной памяти

32

Тип обнаруженного гипервизора(рис. [3.6])

```
[root@darfonos ~]# dmesg | grep -i "Hypervisor detected"
[ 0.000000] Hypervisor detected: KVM
[root@darfonos ~]#
```

Рис. 3.6: гипервизор

33

Вывожу Тип файловой системы корневого раздела. (CPU0).(рис. [3.7])

```
[root@darfonos ~]# df -T
```

| Файловая система | Тип | 1K-блоков | Использовано | Доступно | Использовано% | Монтировано в |
|------------------|----------|-----------|--------------|----------|---------------|-------------------------------------|
| devtmpfs | devtmpfs | 4096 | 0 | 4096 | 0% | /dev |
| tmpfs | tmpfs | 2002260 | 0 | 2002260 | 0% | /dev/shm |
| tmpfs | tmpfs | 800904 | 1388 | 799516 | 1% | /run |
| /dev/sda3 | btrfs | 91358208 | 27084216 | 61634120 | 31% | / |
| tmpfs | tmpfs | 2002260 | 88 | 2002172 | 1% | /tmp |
| /dev/sda3 | btrfs | 91358208 | 27084216 | 61634120 | 31% | /home |
| /dev/sda2 | ext4 | 906780 | 264748 | 663320 | 29% | /boot |
| tmpfs | tmpfs | 400452 | 156 | 400296 | 1% | /run/user/1000 |
| /dev/sr0 | iso9660 | 52244 | 52244 | 0 | 100% | /run/media/darfonos/VBox_GAs_7.0.10 |

```
[root@darfonos ~]#
```

Рис. 3.7: тип файловой системы

34

вывожу на экран последовательность монтирования файловых систем.(рис. [3.8])

```
[root@darfonos ~]# mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=495436,mode=755,inode64)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
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)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=800904k,nr_inodes=819200,mode=755,inode64)
cgrou2 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)
ramfs on /run/credentials/systemd-vconsole-setup.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
/dev/sda3 on / type btrfs (rw,relatime,seclabel,compress=zstd:1,space_cache=v2,subvol=257,subvol=/root)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=33,prgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe)
queue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
```

Рис. 3.8: Последовательность монтирования файловых систем.

4 Вывод

В данной лабораторной работе я приобрел практические навыки установки операционной системы на виртуальную машину на Windows 11, и научился настраивать минимально необходимые для дальнейшей работы сервисы.