

# **Отчёт по лабораторной работе №1**

**Операционные системы**

Екатерина Павловна Канева

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Задание</b>	<b>6</b>
<b>3 Выполнение лабораторной работы</b>	<b>7</b>
3.1 Лабораторная работа . . . . .	7
3.2 Дополнительное задание . . . . .	18
3.3 Контрольные вопросы . . . . .	21
<b>4 Выводы</b>	<b>24</b>

# Список иллюстраций

3.1	Запуск Oracle VM VirtualBox. . . . .	8
3.2	Смена хост-комбинации. . . . .	8
3.3	Создание виртуальной машины. . . . .	9
3.4	Выбор объёма оперативной памяти. . . . .	9
3.5	Выбор создания виртуального жёсткого диска. . . . .	10
3.6	Выбор формата жёсткого диска. . . . .	10
3.7	Выбор динамического типа жёсткого диска. . . . .	11
3.8	Выбор имени и размера виртуального жёсткого диска. . . . .	11
3.9	Увеличение объёма видеопамяти. . . . .	12
3.10	Выбор образа жёсткого диска. . . . .	12
3.11	Выбор образа жёсткого диска. . . . .	13
3.12	Запуск установки образа ОС. . . . .	13
3.13	Выбор языка интерфейса при установке. . . . .	14
3.14	Выбор даты и времени. . . . .	15
3.15	Выбор раскладки клавиатуры. . . . .	15
3.16	Проверка состояния виртуального жёсткого диска. . . . .	16
3.17	Извлечение образа жёсткого диска. . . . .	16
3.18	Скачивание архива. . . . .	17
3.19	Распаковка архива. . . . .	17
3.20	Проверка имён пользователя и хоста виртуальной машины. . . . .	18
3.21	Проверка версии Linux. . . . .	18
3.22	Проверка частоты процессора. . . . .	19
3.23	Проверка модели процессора. . . . .	19
3.24	Проверка объёма доступной памяти. . . . .	19
3.25	Тип обнаруженного гипервизора. . . . .	20
3.26	Тип файловой системы корневого раздела. . . . .	20
3.27	Последовательность монтирования файловых систем. . . . .	21

# **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 Задание**

- Установка операционной системы на виртуальную машину.
- Настройка виртуальной машины.
- Получить следующую информацию:
  - Версия ядра Linux (Linux version).
  - Частота процессора (Detected Mhz processor).
  - Модель процессора (CPU0).
  - Объём доступной оперативной памяти (Memory available).
  - Тип обнаруженного гипервизора (Hypervisor detected).
  - Тип файловой системы корневого раздела.
  - Последовательность монтирования файловых систем.

# **3 Выполнение лабораторной работы**

## **3.1 Лабораторная работа**

Виртуальная машина была установлена и настроена в предыдущем семестре в рамках дисциплины “Архитектура компьютера”, поэтому отчёт об установке операционной системы на виртуальную машину был уже готов (как и сама работа). Ниже приведены описание и скриншоты из того отчёта.

Работа выполнялась на персональном ноутбуке.

Предварительно было установлено дополнительно программное обеспечение – виртуальная машина Oracle VM VirtualBox (пакет Windows hosts с сайта в сети Интернет: здесь) и образ необходимый образ операционной системы (Fedora 36: x86\_64 Live ISO-образ с сайта в сети Интернет: здесь). При установке виртуальной машины были выбраны предложенные системой параметры, поэтому снимки экрана приложены не будут.

Следующим шагом была запущена виртуальная машина Oracle VM VirtualBox (рис. 3.1):

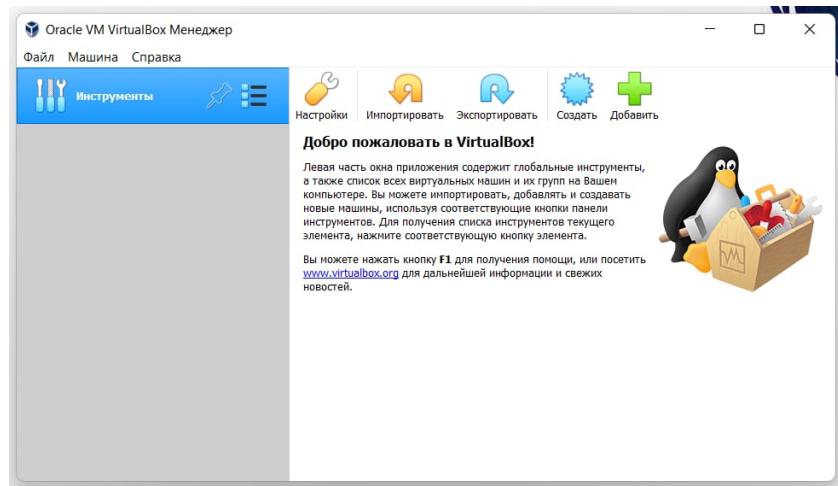


Рис. 3.1: Запуск Oracle VM VirtualBox.

Далее была сменена хост-комбинация на Ctrl+Alt (рис. 3.2):

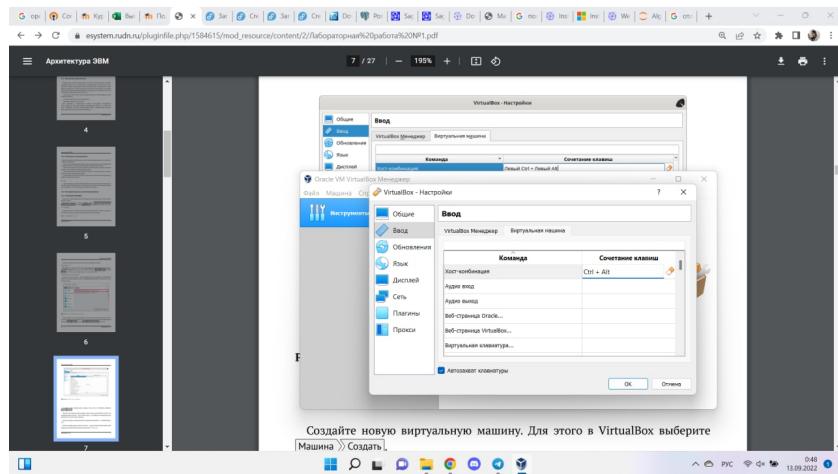


Рис. 3.2: Смена хост-комбинации.

После этого была создана виртуальная машина, началась её настройка. В графе Имя был указан логин в дисплейном классе – ерканева; в графе Папка машины путь к заранее созданному каталогу, соответствующему нужному пути в дисплейном классе – \var\tmp\ерканева; в графе Тип – Linux; в графе Версия - Fedora 64-bit (рис. 3.3):

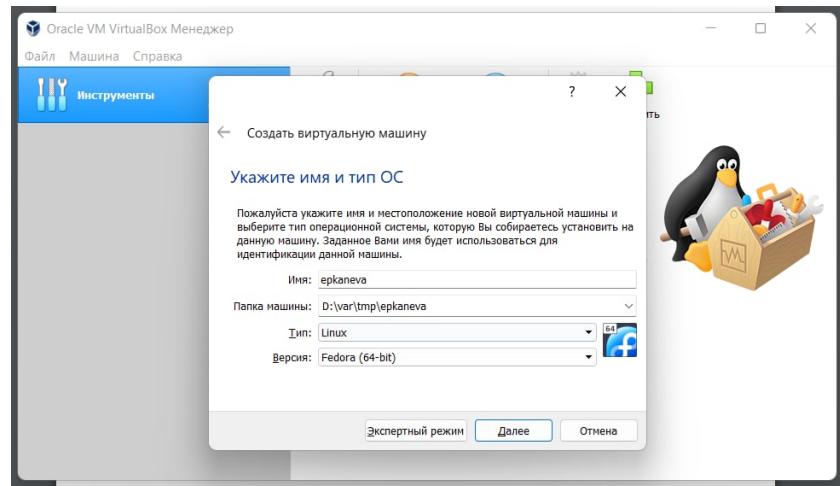


Рис. 3.3: Создание виртуальной машины.

Далее был выбран объём памяти – 4096 Мб (рис. 3.4):

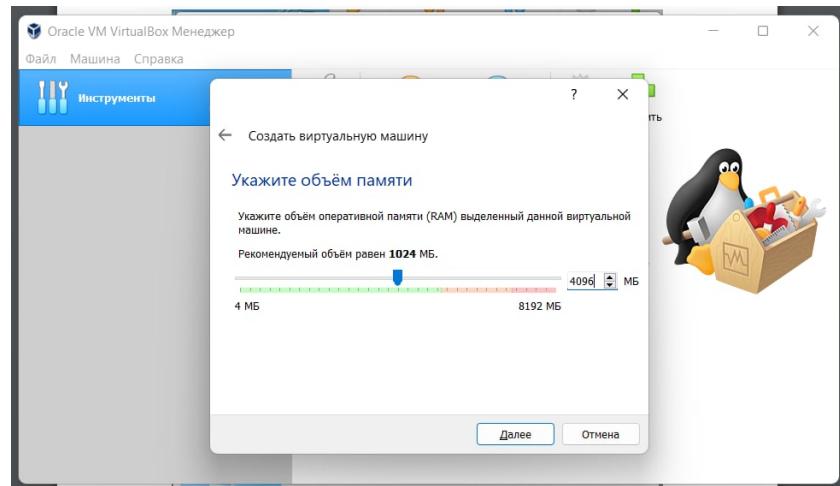


Рис. 3.4: Выбор объёма оперативной памяти.

Далее был создан виртуальный жёсткий диск (рис. 3.5), выбран тип VDI (VirtualBox Disk Image) (рис. 3.6), выбран динамический тип жёсткого диска (рис. 3.7):

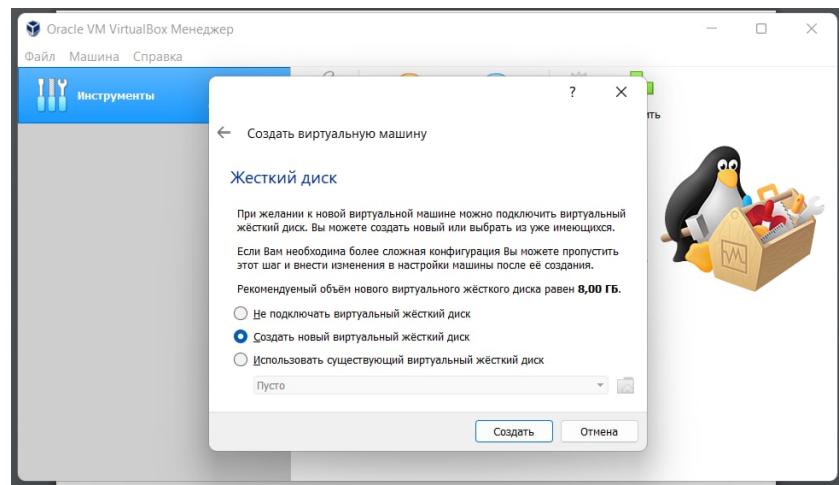


Рис. 3.5: Выбор создания виртуального жёсткого диска.

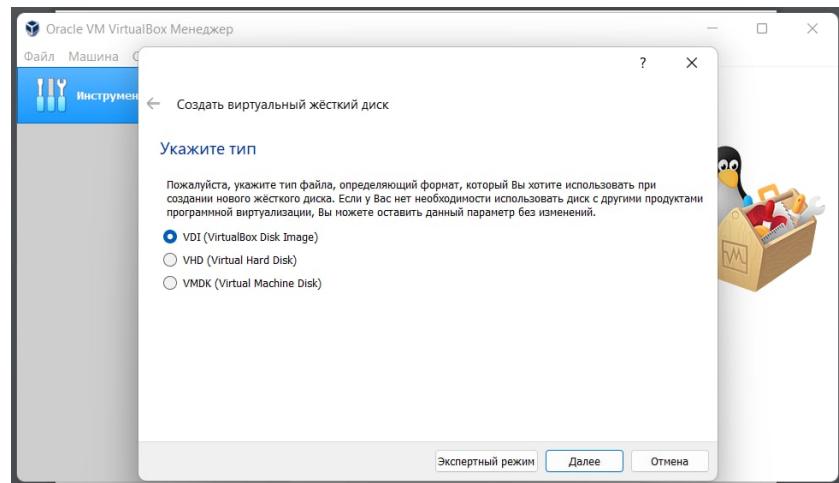


Рис. 3.6: Выбор формата жёсткого диска.

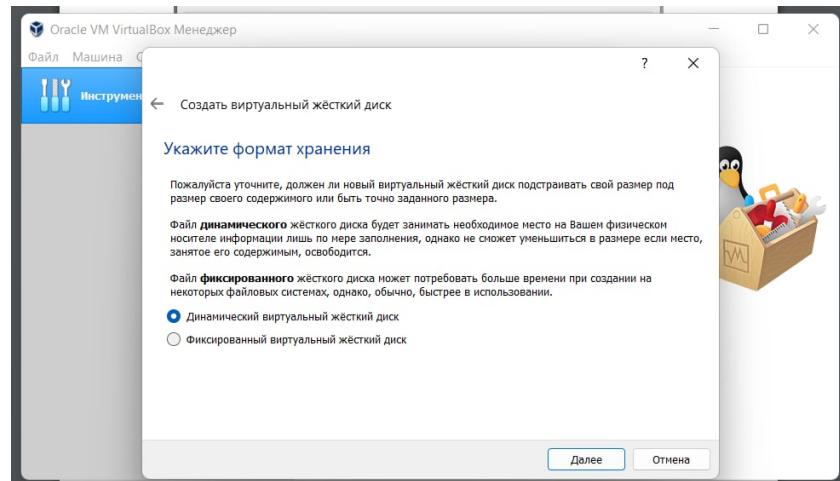


Рис. 3.7: Выбор динамического типа жёсткого диска.

Далее для виртуального жёсткого диска был задан объём 80 Гб и выбран путь /var/tmp/epkaneva/epkaneva.vdi (рис. 3.8):

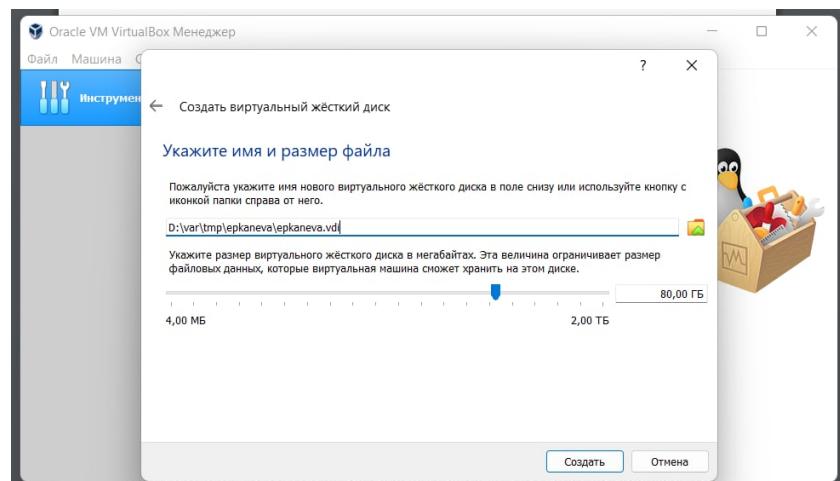


Рис. 3.8: Выбор имени и размера виртуального жёсткого диска.

Далее был увеличен до 128 Мб доступный объём видеопамяти (рис. 3.9):

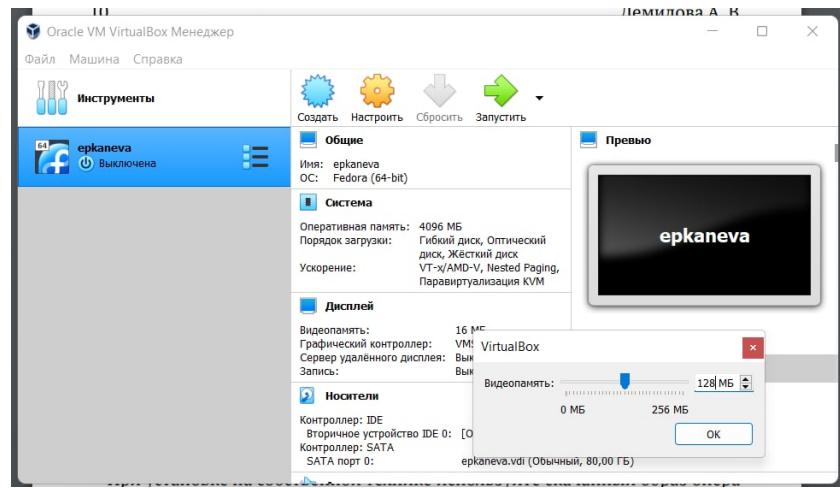


Рис. 3.9: Увеличение объёма видеопамяти.

После этого был выбран образ жёсткого диска (рис. 3.10 и 3.11):

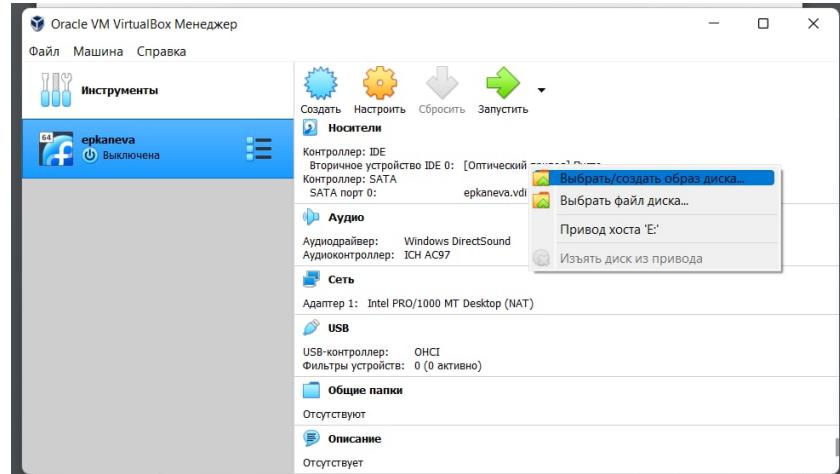


Рис. 3.10: Выбор образа жёсткого диска.

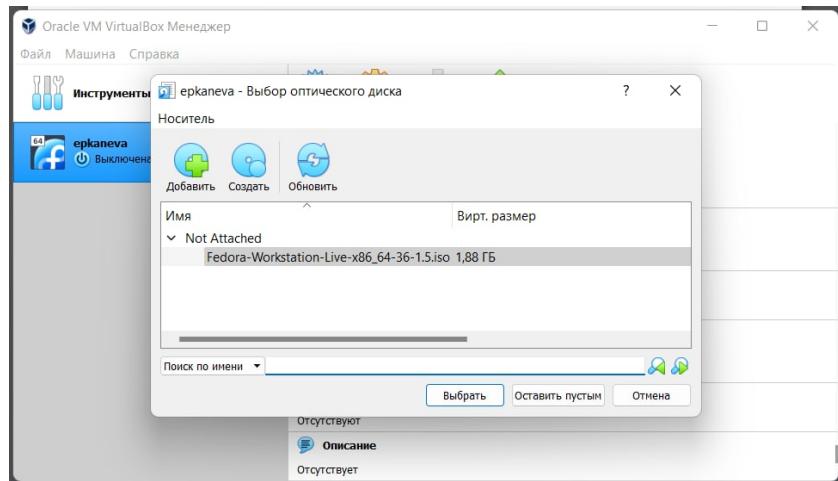


Рис. 3.11: Выбор образа жёсткого диска.

После завершения настройки виртуальная машина была запущена. Далее была запущена установка образа ОС - `Install to Hard Drive` (рис. 3.12):

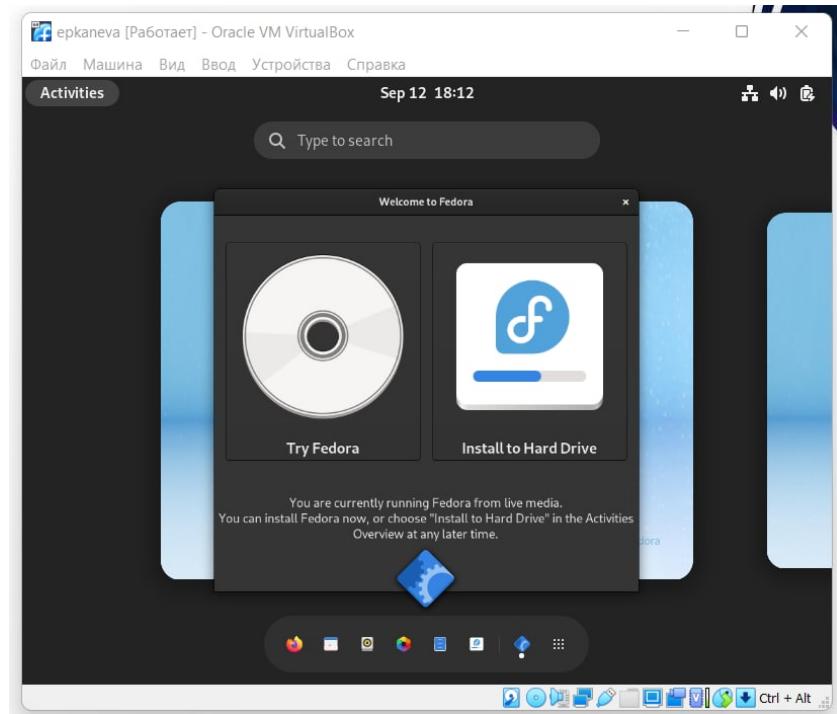


Рис. 3.12: Запуск установки образа ОС.

Далее запустился процесс установки. В качестве языка интерфейса был выбран английский язык (рис. 3.13). На этапе Обзора установки дата и время были

установлены в соответствии с часовым поясом (рис. 3.14), в качестве языков клавиатуры установлены английский и русский (рис. 3.15), было проверено состояние жёсткого диска, на который выполняется установка (рис. 3.16).

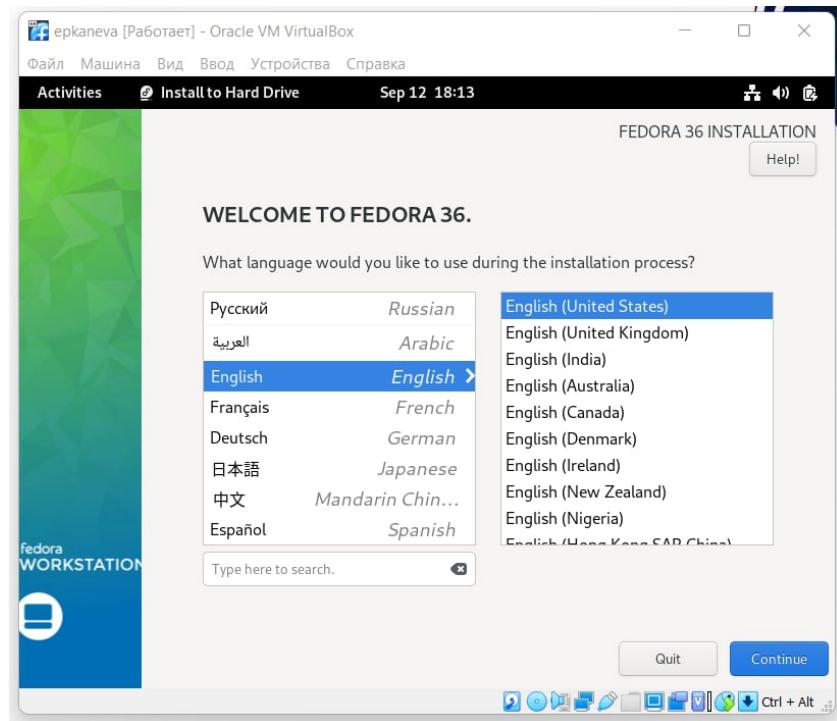


Рис. 3.13: Выбор языка интерфейса при установке.

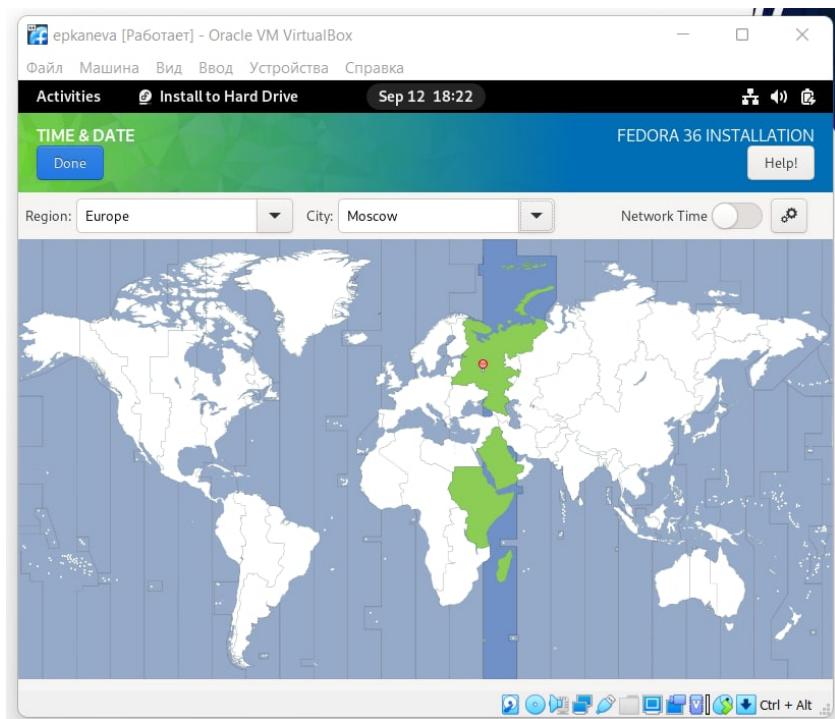


Рис. 3.14: Выбор даты и времени.

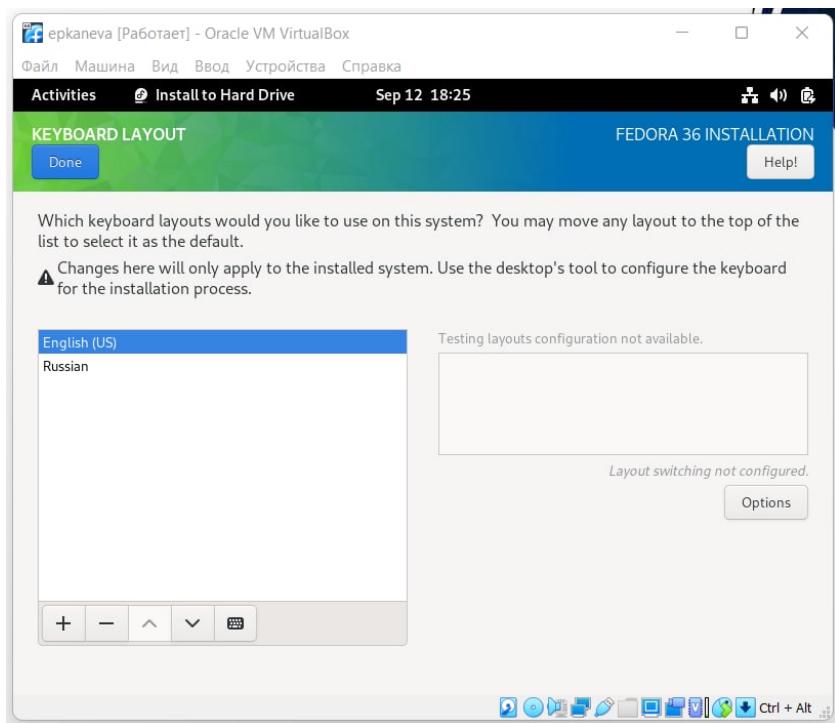


Рис. 3.15: Выбор раскладки клавиатуры.

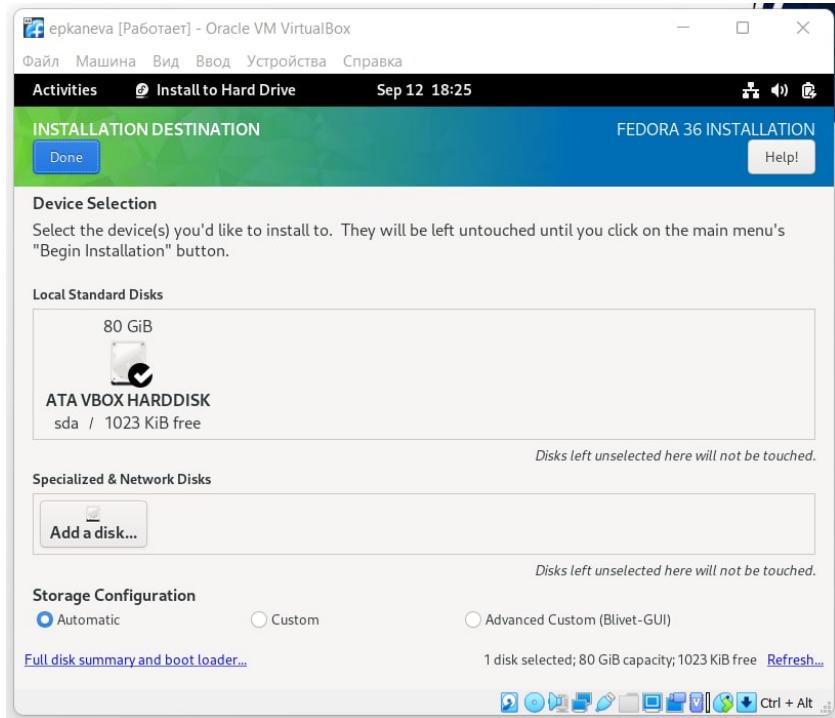


Рис. 3.16: Проверка состояния виртуального жёсткого диска.

Далее была запущена установка. По завершении виртуальная машина была выключена, был извлечён образ жёсткого диска (рис. 3.17). После повторного включения были заданы имя, соответствующее логину в дисплейном классе (ерканева), и пароль пользователя.

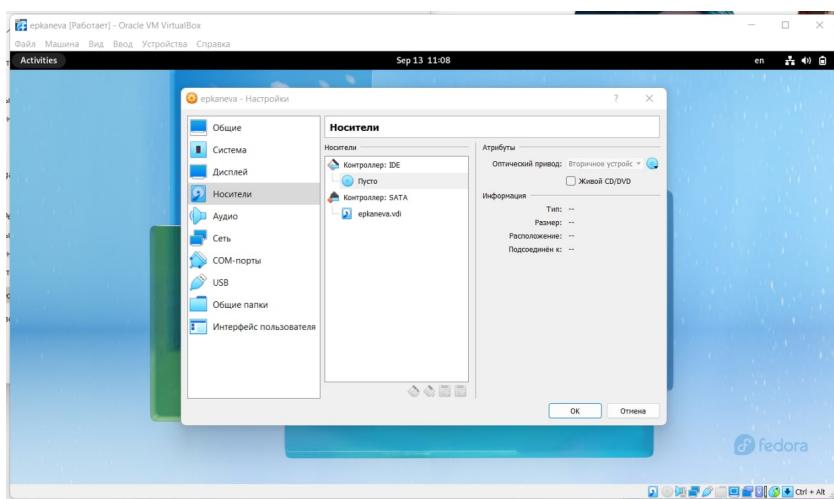


Рис. 3.17: Извлечение образа жёсткого диска.

Образ ОС был успешно установлен на виртуальную машину, виртуальная машина запускается и работает корректно.

В ходе выполнения более поздних лабораторных работ были установлены pandoc (нет снимков экрана, т.к. тогда ПК выключился в какой-то момент) и texlive (рис. 3.18 и 3.19):

```
[epkaneva@fedora tmp]$ wget https://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz
--2022-10-18 02:50:26-- https://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz
Resolving mirror.ctan.org (mirror.ctan.org)... 5.35.249.60
Connecting to mirror.ctan.org (mirror.ctan.org)|5.35.249.60|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://mirror.macomnet.net/pub/CTAN/systems/texlive/tlnet/install-tl-unx.tar.gz [following]
--2022-10-18 02:50:27-- https://mirror.macomnet.net/pub/CTAN/systems/texlive/tlnet/install-tl-unx.tar.gz
Resolving mirror.macomnet.net (mirror.macomnet.net)... 195.128.64.25
Connecting to mirror.macomnet.net (mirror.macomnet.net)|195.128.64.25|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5834586 (5.6M) [application/octet-stream]
Saving to: 'install-tl-unx.tar.gz'

install-tl-unx.tar.gz 100%[=====] 5.56M 7.79MB/s in 0.7s
2022-10-18 02:50:28 (7.79 MB/s) - 'install-tl-unx.tar.gz' saved [5834586/5834586]
```

Рис. 3.18: Скачивание архива.

```
[epkaneva@fedora tmp]$ zcat install-tl-unx.tar.gz | tar xf -
```

Рис. 3.19: Распаковка архива.

Были также проверены имена пользователя, хоста виртуальной машины (рис. 3.20):

```
[epkaneva@fedora ~]$ sudo -i
[sudo] password for epkaneva:
[root@fedora ~]# adduser -G wheel epkaneva
adduser: user 'epkaneva' already exists
[root@fedora ~]# hostnamectl set-hostname epkaneva
[root@fedora ~]# hostnamectl
  Static hostname: epkaneva
    Icon name: computer-vm
      Chassis: vm
    Machine ID: e0806ae2194f45998f5946893c27281d
        Boot ID: e1218c661d364551a31a5582a19a6421
  Virtualization: oracle
Operating System: Fedora Linux 36 (Workstation Edition)
      CPE OS Name: cpe:/o:fedoraproject:fedora:36
            Kernel: Linux 6.1.10-100.fc36.x86_64
      Architecture: x86-64
  Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
```

Рис. 3.20: Проверка имён пользователя и хоста виртуальной машины.

## 3.2 Дополнительное задание

Далее была начата работа по выполнению “дополнительных заданий” (или “домашнего задания”).

С помощью различных команд (в основном, dmesg) была получена следующая информация:

- Версия ядра Linux, т.е. Linux version (рис. 3.21):

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

```
[root@epkaneva ~]# dmesg | grep -i "Linux version"
[    0.000000] Linux version 6.1.10-100.fc36.x86_64 (mockbuild@bkernel01.iad2.fedoraproject.org) (gcc (GCC) 12.2.1 20221121 (Red Hat 12.2.1-4), GNU ld version 2
.37-37.fc36) #1 SMP PREEMPT_DYNAMIC Mon Feb  6 19:58:39 UTC 2023
```

Рис. 3.21: Проверка версии Linux.

Видим, что ядро Linux ver. 6.1.10-100.fc36.x86\_64 (86 архитектура, 64 бит).

- Частота процессора, т.е. Detected Mhz processor (рис. 3.22):

```
dmesg | grep -i "processor"
```

```
[root@epkaneva ~]# dmesg | grep -i "processor"
[    0.000014] tsc: Detected 2591.998 MHz processor
[    0.217832] smpboot: Total of 1 processors activated (5183.99 BogoMIPS)
[    0.260121] ACPI: Added _OSI(Processor Device)
[    0.260123] ACPI: Added _OSI(Processor Aggregator Device)
```

Рис. 3.22: Проверка частоты процессора.

Частота процессора: 2591.998 (~2592) MHz.

- Модель процессора, т.е. CPU0 (рис. 3.23):

```
dmesg | grep -i "CPU0"
```

```
[root@epkaneva ~]# dmesg | grep -i "CPU0"
[    0.217832] smpboot: CPU0: Intel(R) Core(TM) i3-10110U CPU @ 2.10GHz (family:
0x6, model: 0x8e, stepping: 0xc)
```

Рис. 3.23: Проверка модели процессора.

Модель процессора: Intel(R) Core(TM) i1-10110U.

- Объём доступной оперативной памяти, т.е. Memory available (рис. 3.24):

```
dmesg | grep -i "Memory"
```

```
[root@epkaneva ~]# dmesg | grep -i "Memory"
[    0.022057] ACPI: Reserving FACP table memory at [mem 0xfffff0f0-0xfffff01e3]
[    0.022058] ACPI: Reserving DSDT table memory at [mem 0xfffff0470-0xfffff2794]
[    0.022059] ACPI: Reserving FACS table memory at [mem 0xfffff0200-0xfffff023f]
[    0.022060] ACPI: Reserving FACS table memory at [mem 0xfffff0200-0xfffff023f]
[    0.022060] ACPI: Reserving APIC table memory at [mem 0xfffff0240-0xfffff0293]
[    0.022061] ACPI: Reserving SSDT table memory at [mem 0xfffff02a0-0xfffff046b]
[    0.021687] Early memory node ranges
[    0.028425] PM: hibernation: Registered nosave memory: [mem 0x000000000-0x00000fff]
[    0.028427] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009ffff]
[    0.028428] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000fffff]
[    0.028428] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[    0.028429] PM: hibernation: Registered nosave memory: [mem 0xfffff0000-0xdffffff]
[    0.028430] PM: hibernation: Registered nosave memory: [mem 0xe00000000-0xebfffff]
[    0.028430] PM: hibernation: Registered nosave memory: [mem 0xfc000000-0xfe0fffff]
[    0.028431] PM: hibernation: Registered nosave memory: [mem 0xfc1000-0xfedfffff]
[    0.028431] PM: hibernation: Registered nosave memory: [mem 0xfe000000-0xfe0fffff]
[    0.028432] PM: hibernation: Registered nosave memory: [mem 0xfe000000-0xffffffff]
[    0.028432] PM: hibernation: Registered nosave memory: [mem 0xffffc0000-0xffffffff]
[    0.066647] Memory: 3971004K/4193848K available (16393 kernel code, 3265K rdata, 12468K rodata, 3032K init, 4596K bss, 222584K reserved, 0K cma-reserved)
```

Рис. 3.24: Проверка объёма доступной памяти.

Памяти доступно: 3971004K (из 4193848K).

- Тип обнаруженного гипервизора, т.е. Hypervisor detected (рис. 3.25):

```
dmesg | grep -i "Hypervisor detected"
```

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

Рис. 3.25: Тип обнаруженного гипервизора.

Тип гипервизора: KVM.

- Тип файловой системы корневого раздела (рис. 3.26). Для этого заходим в приложение Disks:

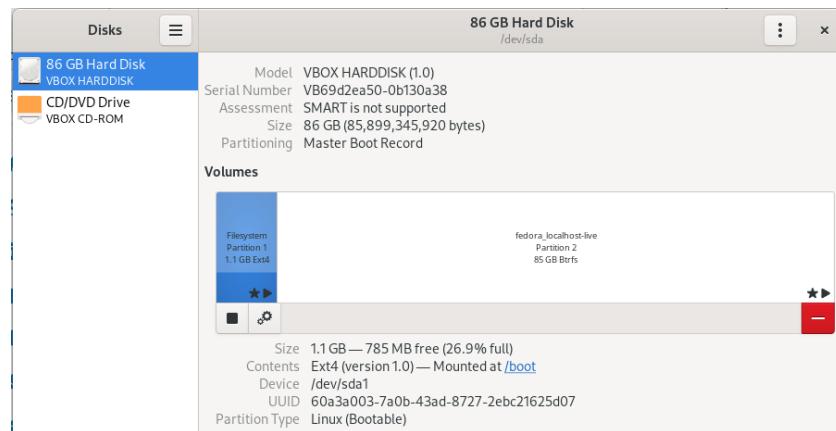


Рис. 3.26: Тип файловой системы корневого раздела.

Тип: ext4.

- Последовательность монтирования файловых систем (рис. 3.27):

```
mount
```

```
[root@epkaneva ~]# 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,nodev,seclabel,size=4896K,nr_inodes=1048576,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)
tmpfs on /dev/pts type tmpfs (rw,nosuid,nodev,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=800)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=882406K,nr_inodes=819200,mode=755,inode64)
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)
/dev/sda1 on / type btrfs (rw,relatime,seclabel,compress=zstd),space_cache=v2,subvolid=257,subvol=/root)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc type systemd-sysusers.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=780)
tmpfs on /tmp type tmpfs (rw,nosuid,nodev,seclabel,size=2086240K,nr_inodes=1048576,inode64)
/dev/sda1 on /boot type ext4 (rw,relatime,seclabel)
/dev/sda2 on /home type btrfs (rw,relatime,seclabel,compress=zstd:1,space_cache=v2,subvolid=256,subvol=/home)
binfmt_misc on /proc/sys/fs/binfmt_misc type binfmt_misc (rw,nosuid,nodev,noexec,relatime,seclabel)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=401244K,nr_inodes=100311,mode=700,uid=1000,gid=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)
```

Рис. 3.27: Последовательность монтирования файловых систем.

### 3.3 Контрольные вопросы

1. Какую информацию содержит учётная запись пользователя?

Имя пользователя, зашифрованный пароль пользователя, идентификационный номер пользователя, идентификационный номер группы пользователя, домашний каталог, командный интерпретатор пользователя.

2. Укажите команды терминала и приведите примеры:

- для получения справки по команде: `man <command>` (`man cd`)
- для перемещения по файловой системе: `cd <path>` (`cd work/study`)
- для просмотра содержимого каталога: `ls` or `dir`
- для определения объёма каталога: `du`
- для создания каталогов: `mkdir <name>` (`mkdir lab01`)
- для удаления каталогов: `rm <name>` (`rm -r lab01`)
- для создания файлов: `touch <name>` (`touch lab01.md`)
- для удаления файлов: `rm -r <name>` (`rm lab01.md`)

- для задания определённых прав на файл / каталог: `chmod + x <name/path>`  
(`chmod + x lab01`)
- для просмотра истории команд: `history`.

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

Файловая система – это порядок, определяющий способ организации, хранения и именования данных на носителях информации в компьютерах, а также в другом электронном оборудовании.

Примеры файловых систем:

- Журналируемая файловая система (JFS). Журналируемые файловые системы позволяют быстро восстанавливать данные в случае сбоя. Это достигается за счет ведения журнала изменений файлов.
- Файловая система на компакт-диске. Это файловая система, которая хранится на компакт-диске и доступна только для чтения.
- Файловая система RAM. Диск RAM – это виртуальный жесткий диск, хранящийся в оперативной памяти.
- Сетевая файловая система (NFS). Сетевая файловая система, или NFS, – это распределенная файловая система, предоставляющая доступ к файлам и каталогам, хранящимся в удаленных системах, обычными средствами для работы с локальными файлами.
- Система имен файлов (NameFS). Система имен файлов содержит функции монтирования файл-на-файл и каталог-на-каталог (также называемое слабое монтирование), которые позволяют монтировать подкаталог или файловую систему в другом месте в области имен файлов, что позволяет иметь доступ к файлу с помощью двух различных путей.

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

С помощью команды `mount`.

5. Как удалить зависший процесс?

с помощью команды `kill`.

## **4 Выводы**

Установили ОС на виртуальную машину и настроили минимально необходимые для дальнейшей работы программы и сервисы. Узнали дополнительную информацию о машине.