

# **Отчёт по лабораторной работе №1**

Барбакова Алиса Саяновна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Создание виртуальной машины . . . . .	8
4.2	После установки . . . . .	9
4.2.1	Обновления . . . . .	9
4.2.2	Повышение комфорта работы. Отключение SELinux . . . .	9
4.3	Настройка раскладки клавиатуры . . . . .	10
4.4	Автоматическое обновление . . . . .	10
4.5	Установка программного обеспечения для создания документации	11
4.6	Домашнее задание . . . . .	11
<b>5</b>	<b>Выводы</b>	<b>13</b>
	<b>Список литературы</b>	<b>14</b>

## Список иллюстраций

4.1	Конфигурации новой виртуальной машины . . . . .	8
4.2	Установка ОС . . . . .	8
4.3	Обновление пакетов . . . . .	9
4.4	Отключение SELinux . . . . .	9
4.5	Редактирование файла . . . . .	10
4.6	Установка ПО . . . . .	10
4.7	Редактирование файла . . . . .	11
4.8	Установка pandoc и texlive . . . . .	11
4.9	Домашнее задание . . . . .	12
4.10	Домашнее задание . . . . .	12

## **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

- 1)Запуск VirtualBox и создание новой виртуальной машины (операционная система Linux, Fedora).
- 2)Настройка установки ОС.
- 3)Перезапуск виртуальной машины и установка драйверов для VirtualBox.
- 4)Подключение образа диска дополнений гостевой ОС.
- 5)Установка необходимого ПО для создания документации.
- 6)Выполнение домашнего задания.

## 3 Теоретическое введение

Что такое виртуальная машина?

Виртуальная машина функционирует внутри вашего компьютера как отдельная физическая машина. Вы можете установить и запустить операционную систему так же, как и на реальном компьютере, поскольку она содержит собственное виртуальное оборудование, включая центральный процессор, графический процессор, память и хранилище. Самое приятное то, что вы можете запускать Linux, Windows и даже другие операционные системы одновременно на одном физическом компьютере, настроив более одной виртуальной машины. Вы можете запустить операционную систему Linux внутри операционной системы Windows, а затем запустить любую другую операционную систему в другой вкладке или даже запустить Windows в Linux, а затем запустить Linux поверх этого, если хотите. Возможности практически безграничны.

## 4 Выполнение лабораторной работы

### 4.1 Создание виртуальной машины

Я создаю новую виртуальную машину, указываем имя ASBarbakova. Делаю базовые настройки. (рис. ??1).

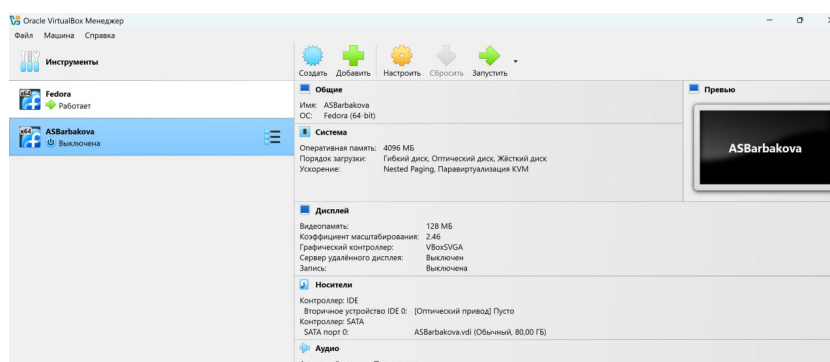


Рис. 4.1: Конфигурации новой виртуальной машины

Производим установку операционной системы. (рис. 4.2).

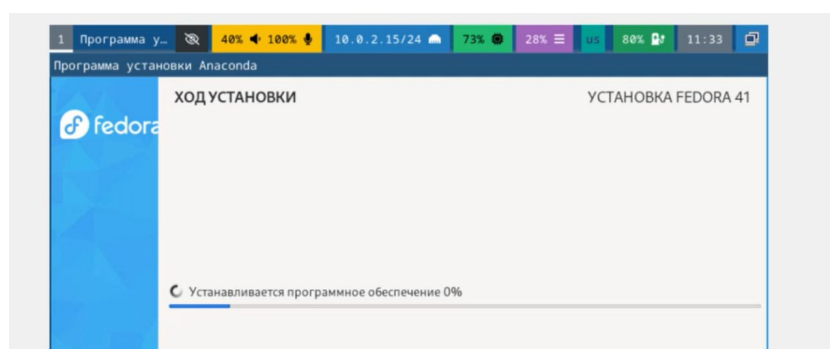


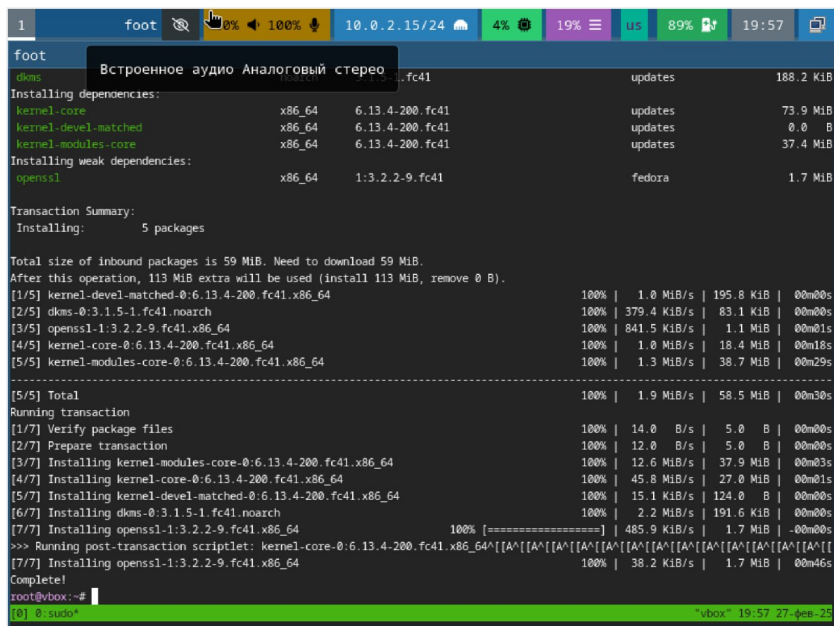
Рис. 4.2: Установка ОС



## 4.2 После установки

### 4.2.1 Обновления

Вхожу в ОС под своей учетной записью. Открываю терминал, произвожу установку обновлений. (рис. 4.3).



```
foot
dms
Installing dependencies:
kernel-core x86_64 6.13.4-200.fc41 updates 188.2 KiB
kernel-devel-matched x86_64 6.13.4-200.fc41 updates 73.9 MiB
kernel-modules-core x86_64 6.13.4-200.fc41 updates 0.0 B
kernel-modules-core x86_64 6.13.4-200.fc41 updates 37.4 MiB
Installing weak dependencies:
openssl x86_64 1:3.2.2-9.fc41 fedora 1.7 MiB

Transaction Summary:
Installing: 5 packages

Total size of inbound packages is 59 MiB. Need to download 59 MiB.
After this operation, 113 MiB extra will be used (install 113 MiB, remove 0 B).
[1/5] kernel-devel-matched-0:6.13.4-200.fc41.x86_64 100% | 1.0 MiB/s | 195.8 KiB | 00m00s
[2/5] dms-0:3.1.5-1.fc41.noarch 100% | 379.4 KiB/s | 83.1 KiB | 00m00s
[3/5] openssl-1:3.2.2-9.fc41.x86_64 100% | 841.5 KiB/s | 1.1 MiB | 00m01s
[4/5] kernel-core-0:6.13.4-200.fc41.x86_64 100% | 1.0 MiB/s | 18.4 MiB | 00m18s
[5/5] kernel-modules-core-0:6.13.4-200.fc41.x86_64 100% | 1.3 MiB/s | 38.7 MiB | 00m29s
-----
[5/5] Total 100% | 1.9 MiB/s | 58.5 MiB | 00m30s
Running transaction
[1/7] Verify package files 100% | 14.0 B/s | 5.0 B | 00m00s
[2/7] Prepare transaction 100% | 12.0 B/s | 5.0 B | 00m00s
[3/7] Installing kernel-modules-core-0:6.13.4-200.fc41.x86_64 100% | 12.6 MiB/s | 37.9 MiB | 00m33s
[4/7] Installing kernel-core-0:6.13.4-200.fc41.x86_64 100% | 45.8 MiB/s | 27.0 MiB | 00m11s
[5/7] Installing kernel-devel-matched-0:6.13.4-200.fc41.x86_64 100% | 15.1 KiB/s | 124.0 B | 00m00s
[6/7] Installing dms-0:3.1.5-1.fc41.noarch 100% | 2.2 MiB/s | 191.6 KiB | 00m00s
[7/7] Installing openssl-1:3.2.2-9.fc41.x86_64 100% [=====] | 485.9 KiB/s | 1.7 MiB | 00m00s
>>> Running post-transaction scriptlet: kernel-core-0:6.13.4-200.fc41.x86_64
[7/7] Installing openssl-1:3.2.2-9.fc41.x86_64 100% | 38.2 KiB/s | 1.7 MiB | 00m46s
Complete!
root@vbox:~#
```

Рис. 4.3: Обновление пакетов

### 4.2.2 Повышение комфорта работы. Отключение SELinux

Устанавливаю программу tmux. (рис. fig. 4.4). Запускаю tmux, открываю midnight commander, в файле /etc/selinux/config заменяю значение SELINUX=enforcing на SELINUX=permissive (рис. 4.4).



```
grubby --update-kernel ALL --args selinux=0
To revert back to SELinux enabled:
grubby --update-kernel ALL --remove-args selinux
SELINUX=permissive
SELINUXTYPE can take one of these three values:
targeted - Targeted processes are protected,
minimum - Minimum protection is provided,
disabled - No SELinux protection is provided
```

Рис. 4.4: Отключение SELinux

Перегружаю виртуальную машину.

## 4.3 Настройка раскладки клавиатуры

Создаю конфигурационный файл `~/.config/sway/config.d/95-system-keyboard-config` и редактирую его, добавляя строчку `exec_always /usr/libexec/sway-systemd/locale1-xkb-config --oneshot`

Также редактирую конфигурационный файл `/etc/X11/xorg.conf.d/00-keyboard.conf`. Перегружаю виртуальную машину. (рис. 4.5).

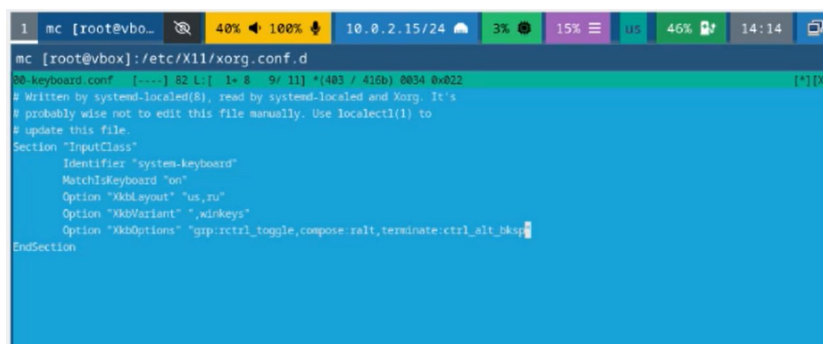


Рис. 4.5: Редактирование файла

## 4.4 Автоматическое обновление

Устанавливаю ПО для автоматического обновления (рис. 4.6).

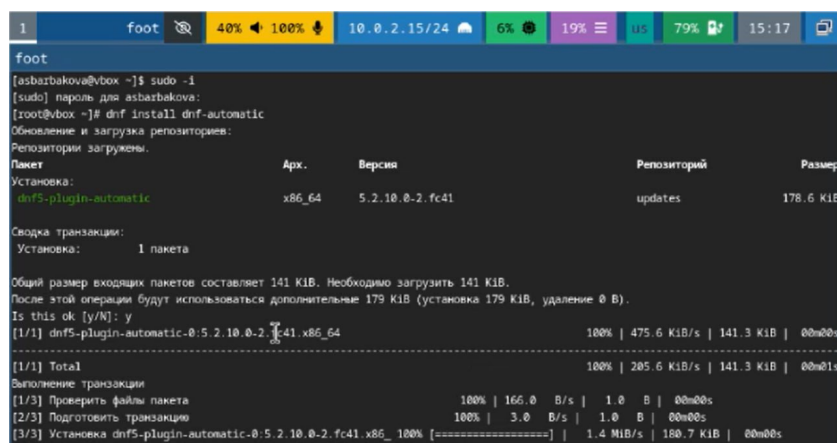


Рис. 4.6: Установка ПО

Редактирую необходимый конфигурационный файл, запускаю таймер. (рис.

4.7).



Рис. 4.7: Редактирование файла

## 4.5 Установка программного обеспечения для создания документации

Скачиваю pandoc и texlive на виртуальную машину (рис. 4.8).

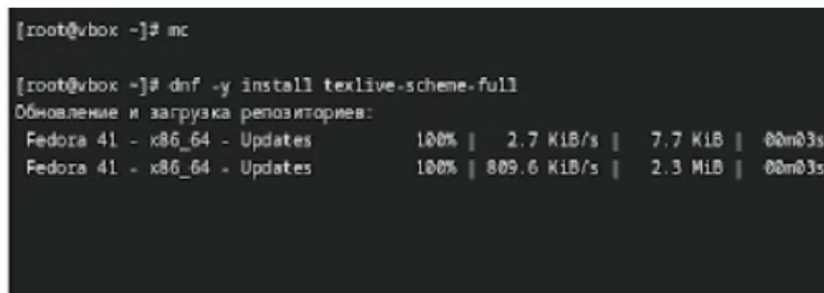


Рис. 4.8: Установка pandoc и texlive

## 4.6 Домашнее задание

С помощью ввода в терминал команды `dmesg | less`, отвечаю на вопросы и выполняю домашнее задание (рис. 4.9). (рис. 4.10).

```
1
root
[ashtabakov@vbox ~]$ sudo -i
[ashtabakov@vbox ~]$ ls
[ashtabakov@vbox ~]$ dmesg | less
[ashtabakov@vbox ~]$ dmesg | grep -i "linux version"
[ 0.000000] Linux version 6.13.5-200.fc41.x86_64 (mockbuild@b0e83da54f8364379359fe70f52a8f23) (gcc (GCC) 14.2.1 20250118 (Red Hat 14.2.1-7), GNU ld version 2.43.1-5.fc41) #1 SMP PREEMPT_
VMware Thu Feb 27 15:47:31 UTC 2025
[ashtabakov@vbox ~]$ dmesg | grep -i "Detected Mhz processor"
[ashtabakov@vbox ~]$ dmesg | grep -i "processor"
[ 0.000025] tsc: Detected 2995.204 MHz processor
[ 0.240286] upboot: Total of 1 processors activated (5990.40 BogoMIPS)
[ 0.253163] ACPI: Added _GSI(processor Device)
[ 0.253165] ACPI: Added _GSI(processor Aggregator Device)
[ashtabakov@vbox ~]$ dmesg | grep -i "CPU"
[ 0.239731] upboot: CPU: Intel(R) Core(TM) Ultra 5 125H (family: 0x6, model: 0x4a, stepping: 0x4)
[ashtabakov@vbox ~]$ dmesg | grep -i "Memory available"
[ashtabakov@vbox ~]$ dmesg | grep -i "Memory"
[ 0.000000] DMI: Memory slots populated: 0/8
[ 0.000722] ACPI: Reserving FACP table memory at [mem 0xfffff000-0xfffff01c]
[ 0.000725] ACPI: Reserving SSDT table memory at [mem 0xfffff010-0xfffff020]
[ 0.000726] ACPI: Reserving FACS table memory at [mem 0xfffff020-0xfffff02f]
[ 0.000727] ACPI: Reserving FACS table memory at [mem 0xfffff020-0xfffff02f]
[ 0.000728] ACPI: Reserving APIC table memory at [mem 0xfffff020-0xfffff020]
[ 0.000729] ACPI: Reserving SSDT table memory at [mem 0xfffff020-0xfffff040]
[ 0.009382] Early memory node ranges
[ 0.020763] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.020765] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
```

Рис. 4.9: Домашнее задание

```
[root@vbox ~]$ dmesg | grep -i "Hypervisor detected"
[ 0.000000] Hypervisor detected: KVM
[root@vbox ~]$ dmesg | grep -i "root filesystem"
[root@vbox ~]$ dmesg | grep -i "mount"
[ 0.137207] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 0.137218] Mount-point-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 2.452505] BTRFS: device label fedora devid 1 transid 548 /dev/sda3 (8:3) scanned by mount (452)
[ 2.453968] BTRFS info (device sda3): first mount of filesystem 758feba-61d8-4a8e-95cd-4a320e3b7561
[ 3.899930] systemd[1]: run-credentials-systemd-v2djournal.service: Deactivated successfully.
[ 3.905820] systemd[1]: Set up automount proc-sys-fs-binfmt-misc.automount - Arbitrary Executable File Formats File System Automount.
[ 3.927021] systemd[1]: Listening on systemd-mountfsd.socket - DDI File System Mounter Socket.
[ 3.935146] systemd[1]: Mounting dev-hugepages.mount - Huge Pages File System...
[ 3.942084] systemd[1]: Mounting dev-mqueue.mount - POSIX Message Queue File System...
[ 3.949015] systemd[1]: Mounting sys-kernel-debug.mount - Kernel Debug File System...
[ 3.955191] systemd[1]: Mounting sys-kernel-tracing.mount - Kernel Trace File System...
[ 4.128936] systemd[1]: Starting systemd-remount-fs.service - Remount Root and Kernel File Systems...
[ 6.487082] EXT4-fs (sda2): mounted filesystem a2ae1a7-9d07-4b18-a953-12a0997c0e1d r/w with ordered data mode. Quota mode: none.
```

Рис. 4.10: Домашнее задание

## 5 Выводы

В результате выполнения лабораторной работы я приобрела навыки установки операционной системы на виртуальную машину, а также настройки минимально необходимых для дальнейшей работы сервисов.

## Список литературы

- 1) Кулябов Д. С. Введение в операционную систему UNIX - Лекция.
- 2) Таненбаум Э., Бос Х. Современные операционные системы. - 4-е изд. -СПб. : Питер, 2015. - 1120 с.