

Лабораторная работа №1

Отчёт

Сергеев Д. О.

07 марта 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Сергеев Даниил Олегович
- Студент
- Направление: Прикладная информатика
- Российский университет дружбы народов
- 1132246837@pfur.ru

Цель работы

Приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

Задание

- Установить операционную систему.
- Обновить или установить необходимое программное обеспечение.
- Повысить комфорт работы с операционной системой.
- Настроить раскладку клавиатуры
- Установить программное обеспечение для создания документации.
- Создать локальный каталог для выполнения заданий по предмету.

Ход выполнения лабораторной работы

Создание виртуальной машины

Для начала откроем менеджер виртуальных машин Oracle VirtualBox и нажмем на кнопку создать в графическом интерфейсе. Выберем тип машины Linux, подтип Fedora. Зададим имя, удовлетворяющее соглашению о наименовании.

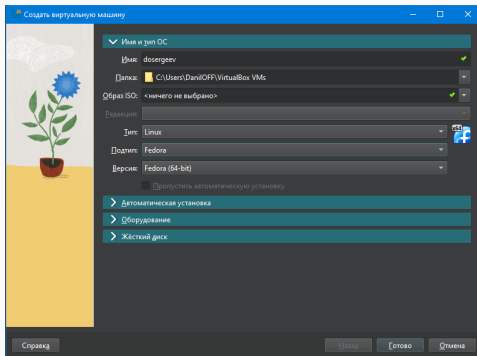


Рис. 1: Окно создания ВМ.

Создание виртуальной машины

Выделим размер основной памяти виртуальной машины до 4096 МБ и 4 процессора. Включим поддержку UEFI(EFI).

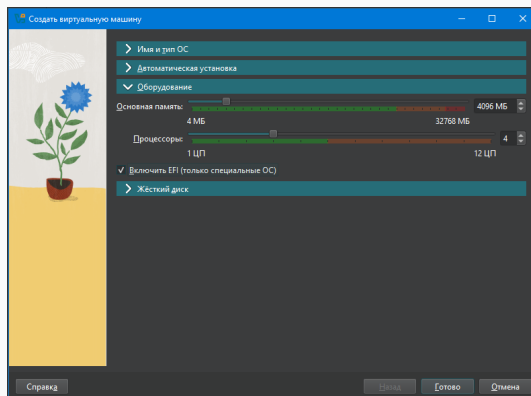


Рис. 2: Окно оборудования ВМ.

Зададим жёсткий диск VDI с размером 80 ГБ.

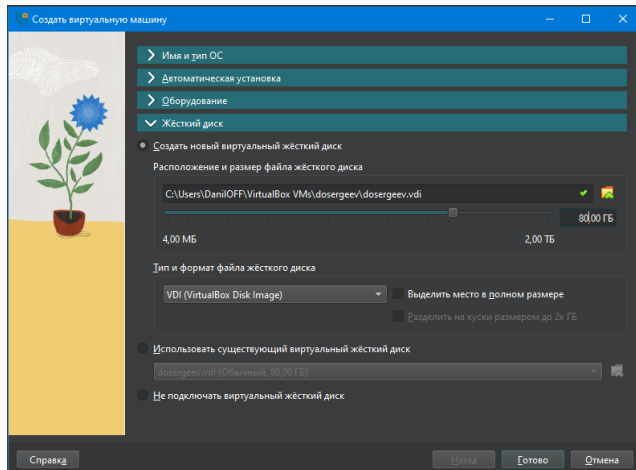


Рис. 3: Окно настройки жёсткого диска.

Создание виртуальной машины

В качестве графического контроллера поставим VMSVGA, включим 3D ускорение, выделим 256 МБ видеопамати.

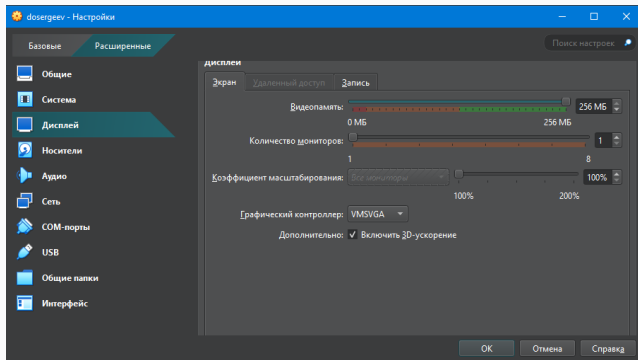


Рис. 4: Настройка дисплея ВМ.

Включим общий буфер обмена и перетаскивание объектов между хостом и гостевой ОС.

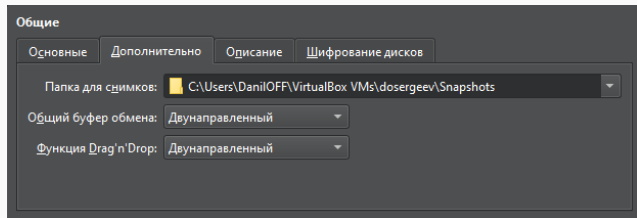


Рис. 5: Окно дополнительных настроек.

Запустим виртуальную машину. После вставим оптический диск с образом Fedora-Sway-Live-x86_64-41-1.4.iso и перезагрузим её.

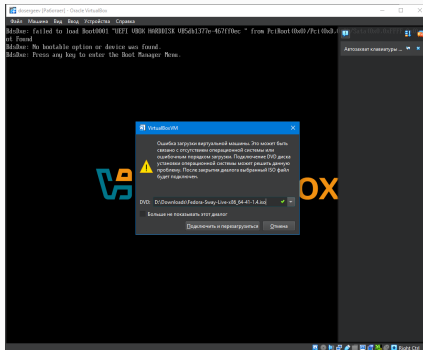


Рис. 6: Установка live CD.

Запустим Fedora, перейдем в режим базовой графики в меню boot, чтобы не было проблем при отображении.



Рис. 7: Меню GRUB.



Рис. 8: Запуск ОС в режиме базовой графики.

После запуска системы, нажмем Win+d и запустим установщик Anaconda командой liveinst.

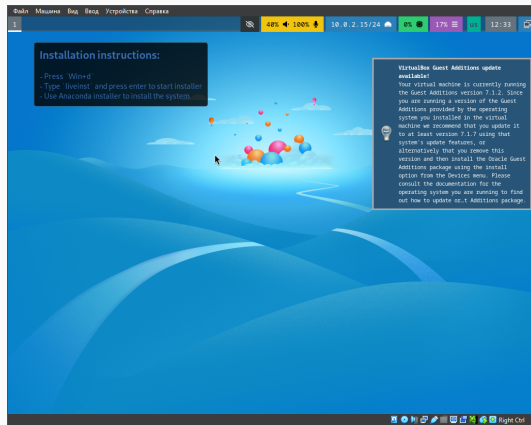


Рис. 9: Интерфейс ОС Fedora sway.

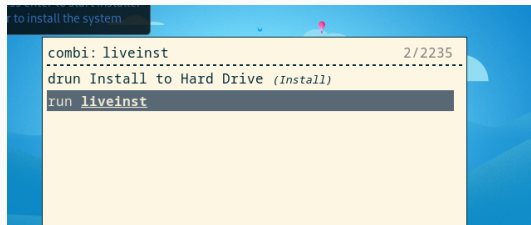


Рис. 10: Запуск liveinst.

Выберем язык интерфейса Русский

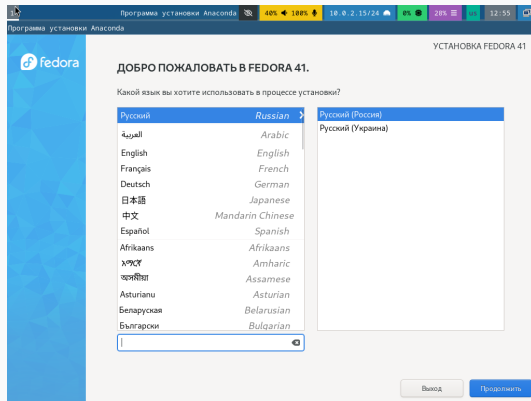


Рис. 11: Приветствие загрузчика Anaconda, выбор языка.

Выберем стандартное место установки ОС.

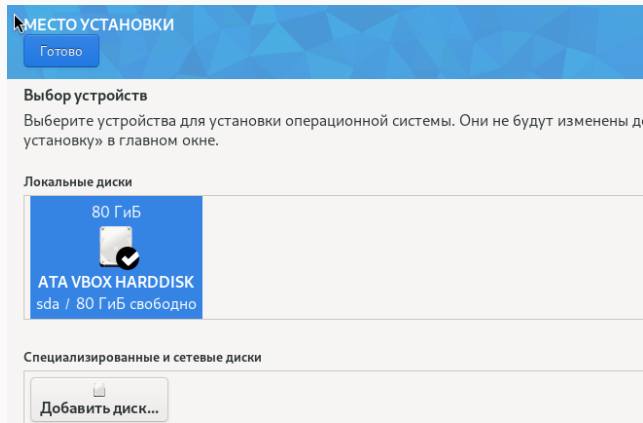
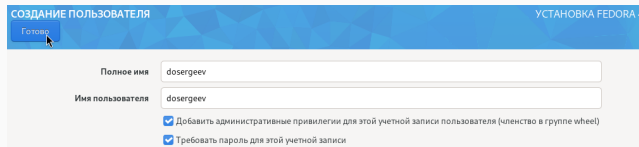


Рис. 12: Выбор устройства для установки ОС.

Установим имя и пароль для пользователя.



The screenshot shows the 'СОЗДАНИЕ ПОЛЬЗОВАТЕЛЯ' (Create User) window in the Fedora 40 installer. The window has a blue header with the title and 'УСТАНОВКА FEDORA 40' on the right. A 'Готово' (Done) button is in the top left. Below the header, there are two text input fields: 'Полное имя' (Full name) and 'Имя пользователя' (Username), both containing the text 'dosergeev'. At the bottom, there are two checked checkboxes: 'Добавить административные привилегии для этой учетной записи пользователя (членство в группе wheel)' and 'Требовать пароль для этой учетной записи'.

Рис. 13: Создание пользователя.

Начнем установку ОС Fedora sway на жёсткий диск.

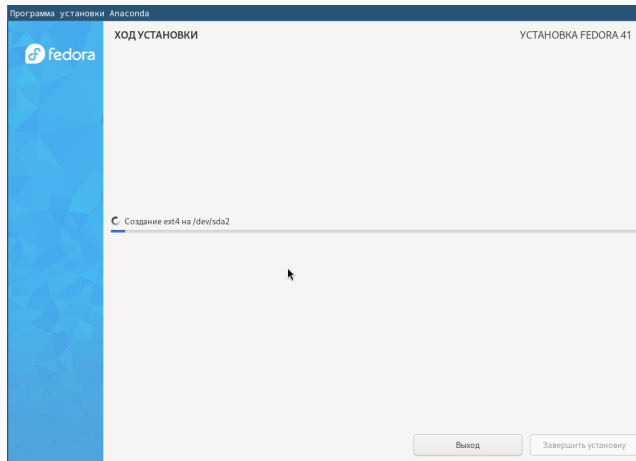


Рис. 14: Процесс установки ОС.

Перезапустим виртуальную машину и войдем в качестве созданного пользователя.

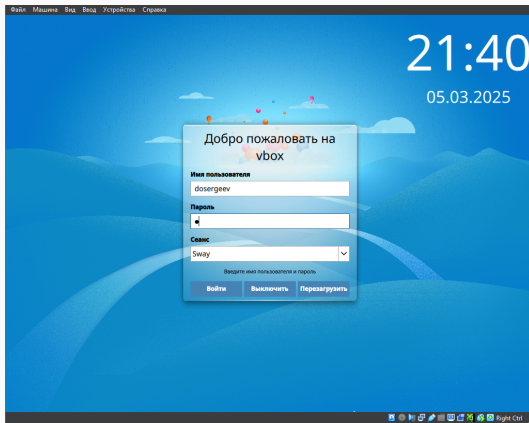


Рис. 15: Окно входа в сеанс.

Запустим терминальный мультиплексор `tmux`, переключимся на роль супер-пользователя, установим средства разработки.

```
dosergeev@vbox: ~$ sudo -i
Мы полагаем, что ваш системный администратор изложил вам основы
безопасности. Как правило, всё сводится к трём следующим правилам:

#1) Уважайте частную жизнь других.
#2) Думайте, прежде чем что-то вводить.
#3) С большой властью приходит большая ответственность.

По соображениям безопасности пароль, который вы введёте, не будет виден.

[sudo] пароль для dosergeev:
root@vbox:~# dnf -y group install development-tools
Updating and loading repositories:
Fedora 41 openh264 (From Cisco) - x86_64
Fedora 41 - x86_64 - Updates
Fedora 41 - x86_64
```

Рис. 16: Установка development-tools.

Также установим пакет DKMS.

```
root@rhel8:~# dnf -y install dnsm
Updating and loading repositories:
Repositories loaded.
```

Package	Arch	Version	Repository	Size
Installing:				
dnsm	noarch	3.1.5-1.fc41	updates	188.2 KiB
Installing dependencies:				
kernel-core	x86_64	6.13.5-200.fc41	updates	73.9 MiB
kernel-devel-matched	x86_64	6.13.5-200.fc41	updates	0.0 B
kernel-modules-core	x86_64	6.13.5-200.fc41	updates	37.3 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).
```

^[[A][A][A][A][A][A][A][A][A][A][A][A][A][A][A][A] dnsm-0:3.1.5-1.fc41.noarch		%	[<=>
0.0 B/s	0.0 B	00m00s	
[1/5] kernel-devel-matched-0:6.13.5-200.fc41.x86_64	100%	2.5 MiB/s 196.0 KiB	00m00s

Рис. 17: Установка пакета DKMS.

Установка драйверов для VirtualBox

Подмонтируем диск и запустим установку драйверов. В конце установки перезагрузим виртуальную машину.

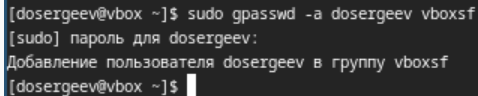
```
root@vbox:~# mount /dev/sr0 /media
mount: /media: WARNING: source write-protected, mounted read-only.
root@vbox:~# /media/VBoxLinuxAdditions.run
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.1.7 Guest Additions for Linux 100%
VirtualBox Guest Additions installer
This system appears to have a version of the VirtualBox Guest Additions
already installed. If it is part of the operating system and kept up-to-date,
there is most likely no need to replace it. If it is not up-to-date, you
should get a notification when you start the system. If you wish to replace
it with this version, please do not continue with this installation now, but
instead remove the current version first, following the instructions for the
operating system.

If your system simply has the remains of a version of the Additions you could
not remove you should probably continue now, and these will be removed during
installation.

Do you wish to continue? [yes or no]
```

Рис. 18: Установка дополнений гостевой ОС.

Внутри виртуальной машины добавим своего пользователя в группу vboxsf.

A terminal window with a dark background and light-colored text. The text shows a user named dosergeev at a machine named vbox, in the home directory (~), running the command 'sudo gpasswd -a dosergeev vboxsf'. The prompt changes to '[sudo] пароль для dosergeev:', followed by the message 'Добавление пользователя dosergeev в группу vboxsf' and the return of the prompt '[dosergeev@vbox ~]\$' with a cursor.

```
[dosergeev@vbox ~]$ sudo gpasswd -a dosergeev vboxsf
[sudo] пароль для dosergeev:
Добавление пользователя dosergeev в группу vboxsf
[dosergeev@vbox ~]$
```

Рис. 19: Добавление пользователя в vboxsf.

Подключение общей папки

В хостовой системе подключим общую папку с помощью графического интерфейса.

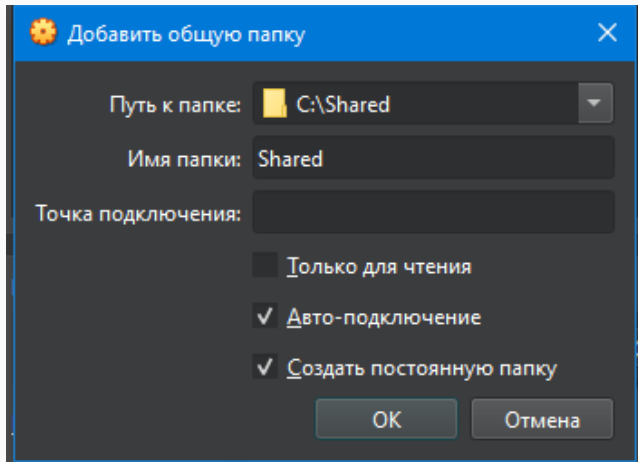


Рис. 20: Добавление общей папки через настройки VirtualBox.

Средства установки уже были установлены в предыдущих пунктах, поэтому начнем с обновления всех пакетов.

```
[dosergeev@vbox ~]$ sudo dnf -y update
Updating and loading repositories:
Repositories loaded.
Package Arch Version Repository Size
Upgrading:
ImageMagick x86_64 1:7.1.1-43-1.fc41 updates 88.4 KiB
replacing ImageMagick x86_64 1:7.1.1-38-1.fc41 anaconda 88.4 KiB
ImageMagick-libs x86_64 1:7.1.1-43-1.fc41 updates 9.1 MiB
replacing ImageMagick-libs x86_64 1:7.1.1-38-1.fc41 anaconda 9.1 MiB
NetworkManager x86_64 1:1.50-3-1.fc41 updates 5.7 MiB
replacing NetworkManager x86_64 1:1.50-0-1.fc41 anaconda 5.7 MiB
NetworkManager-adsl x86_64 1:1.50-3-1.fc41 updates 40.0 KiB
replacing NetworkManager-adsl x86_64 1:1.50-0-1.fc41 anaconda 40.0 KiB
NetworkManager-bluetooth x86_64 1:1.50-3-1.fc41 updates 105.2 KiB
replacing NetworkManager-bluetooth x86_64 1:1.50-0-1.fc41 anaconda 105.2 KiB
NetworkManager-12tp x86_64 1:20-20-1.fc41 updates 681.9 KiB
```

Рис. 21: Обновление пакетов.

Установим программу для удобства работы в консоли: Midnight commander.

```
[dosergeev@vbox ~]$ sudo dnf -y install tmux mc
[sudo] пароль для dosergeev:
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет "tmux-3.5a-2.fc41.x86_64" уже установлен.
```

Пакет	Арх.	Версия
Установка:		
mc	x86_64	1:4.8.32-1.fc41
Установка зависимостей:		
gpm-libs	x86_64	1.20.7-48.fc41

Рис. 22: Установка mc.

Теперь установим другой вариант консоли.

```
[dosergeev@vbox ~]$ sudo dnf -y install kitty
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет                               Арх.      Версия
Установка:
  kitty                             x86_64    0.39.1-1.fc41
Установка зависимостей:
  kitty-kitten                      x86_64    0.39.1-1.fc41
  kitty-shell-integration           noarch    0.39.1-1.fc41
  kitty-terminfo                    noarch    0.39.1-1.fc41
Установка слабых зависимостей:
  ripgrep                           x86_64    14.1.1-1.fc41
```

Рис. 23: Установка kitty.

Установка и обновление программного обеспечения

Подключим автоматическое обновление. Для этого установим dnf-automatic и запустим таймер.

```
[dosergeev@vbox ~]$ sudo dnf -y install dnf-automatic
[sudo] пароль для dosergeev:
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет                                Арх.      Версия
Установка:
dnf5-plugin-automatic                x86_64     5.2.10.0-2.fc41

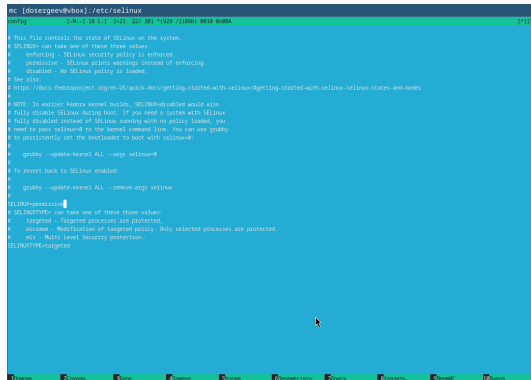
Сводка транзакции:
Установка:      1 пакета

Общий размер входящих пакетов составляет 141 KiB. Необходимо загрузить 141 KiB.
После этой операции будут использоваться дополнительные 179 KiB (установка 179 KiB, удаление 0 B).
[1/1] dnf5-plugin-automatic-0:5.2.10.0-2.fc41.x86_64
-----
[1/1] Total
Выполнение транзакции
[1/3] Проверить файлы пакета                100% | 100.0  B/s
[2/3] Подготовить транзакцию                100% |  3.0  B/s |
[3/3] Установка dnf5-plugin-automatic-0:5.2.10.0-2.fc41.x86_64          100% |
Завершено!
[dosergeev@vbox ~]$ sudo systemctl enable --now dnf-automatic.timer
Created symlink '/etc/systemd/system/timers.target.wants/dnf5-automatic.timer' -> '/usr/lib/systemd/system/dnf5-automatic.timer'.
[dosergeev@vbox ~]$
```

Рис. 24: Подключение dnf-automatic.

Установка и обновление программного обеспечения

Отключим SELinux. В файле `/etc/selinux/config` заменим значение `selinux` с `enforcing` на `permissive`. Перезагрузим виртуальную машину.



```
mc [dosergeev@vbox]: /etc/selinux
config [1M] [38 L] [1=21 227 38] *1929 /110Mb 8038 8x8086 [110K]

# This file controls the state of SELinux on the system.
# SELinux can take one of three values:
# enforcing - SELinux security policy is enforced.
# permissive - SELinux prints warnings instead of enforcing.
# disabled - No SELinux policy is loaded.
# See also:
# https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-selinux/#getting-started-with-selinux-selinux-states-and-modes
#
# NOTE: In earlier Fedora kernel builds, SELINUX=disabled would also
# fully disable SELinux during boot. If you need a system with SELinux
# fully disabled instead of SELinux running with no policy loaded, you
# need to pass selinux=0 to the kernel command line. You can use grubby
# to persistently set the bootloader to boot with selinux=0.
#
# 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 three values:
# targeted - Targeted processes are protected.
# minimum - Modification of targeted policy. Only selected processes are protected.
# mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

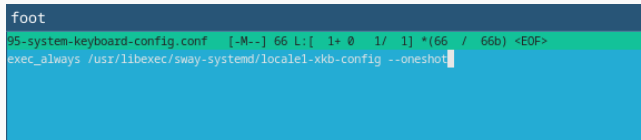
Рис. 25: Настройка SELinux.

Запустим терминальный мультиплексор tmux, создадим конфигурационный файл.

```
dosergeev@vbox:~$ mkdir -p ~/.config/sway
dosergeev@vbox:~$ mkdir ~/.c
.cache/ .config/
dosergeev@vbox:~$ mkdir ~/.config/sway/config.d
dosergeev@vbox:~$ touch ~/.config/sway/config.d/95-system-keyboard-config.conf
```

Рис. 26: Создание 95-system-keyboard-config.conf.

Отредактируем созданный файл.



```
foot
95-system-keyboard-config.conf [-M--] 66 L:[ 1+ 0 1/ 1] *(66 / 66b) <EOF>
exec_always /usr/libexec/sway-systemd/locale1-xkb-config --oneshot
```

Рис. 27: Редактирование 95-system-keyboard-config.conf.

Настройка раскладки клавиатуры

Переключимся на роль супер-пользователя и отредактируем 00-keyboard.conf. Перезапустим виртуальную машину.

```
foot
00-keyboard.conf [----] 83 L:[ 1+ 9 10/ 12] *(438 / 450b) 0010 0x00A
# Written by systemd-localed(8), read by systemd-localed and Xorg. It's
# probably wise not to edit this file manually. Use localectl(1) to
# update this file.
Section "InputClass"
    Identifier "system-keyboard"
    MatchIsKeyboard "on"
    Option "XkbLayout" "us,ru"
    Option "XkbModel" "pc105"
    Option "XkbVariant" ",winkeys"
    Option "XkbOptions" "grp:rctrl_toggle,compose:ralt,terminate:ctrl_alt_bksp"
EndSection
```

Рис. 28: Редактирование 00-keyboard.conf.

Запустим терминальный мультиплексор `tmux`, переключимся на роль супер-пользователя. Установим `pandoc` с помощью менеджера пакетов `dnf`.

```
dosergeev@vbox:~$ sudo -i
[sudo] пароль для dosergeev:
root@vbox:~# sudo dnf -y install pandoc
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет
```

	Арх.	Версия
Установка:		
<code>pandoc</code>	x86_64	3.1.11.1-32.fc41
Установка зависимостей:		
<code>pandoc-common</code>	noarch	3.1.11.1-31.fc41

Рис. 29: Установка `pandoc`.

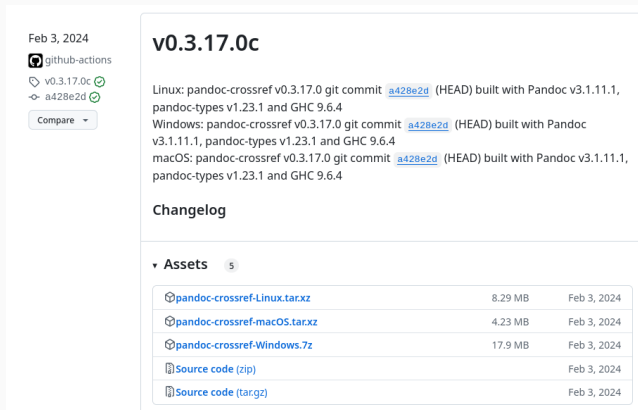
Установим pandoc-crossref. Для начала проверим версию pandoc:

```
[dosergeev@vbox ~]$ pandoc --version
pandoc 3.1.11.1
Features: -server +lua
Scripting engine: Lua 5.4
User data directory: /home/dosergeev/.local/share/pandoc
Copyright (C) 2006-2023 John MacFarlane. Web: https://pandoc.org
This is free software; see the source for copying conditions. There is no
warranty, not even for merchantability or fitness for a particular purpose.
[dosergeev@vbox ~]$
```

Рис. 30: Проверка версии pandoc.

Установка ПО для создания документации

Получается, что версия pandoc - 3.1.11.1. Зайдем на github и найдем соответствующий релиз. Скачаем его и распакуем в /tmp.



Feb 3, 2024

github-actions

v0.3.17.0c ✓

a428e2d ✓

Compare

v0.3.17.0c

Linux: pandoc-crossref v0.3.17.0 git commit [a428e2d](#) (HEAD) built with Pandoc v3.1.11.1, pandoc-types v1.23.1 and GHC 9.6.4

Windows: pandoc-crossref v0.3.17.0 git commit [a428e2d](#) (HEAD) built with Pandoc v3.1.11.1, pandoc-types v1.23.1 and GHC 9.6.4

macOS: pandoc-crossref v0.3.17.0 git commit [a428e2d](#) (HEAD) built with Pandoc v3.1.11.1, pandoc-types v1.23.1 and GHC 9.6.4

Changelog

Assets 5

pandoc-crossref-Linux.tar.xz	8.29 MB	Feb 3, 2024
pandoc-crossref-macOS.tar.xz	4.23 MB	Feb 3, 2024
pandoc-crossref-Windows.7z	17.9 MB	Feb 3, 2024
Source code (zip)		Feb 3, 2024
Source code (tar.gz)		Feb 3, 2024

Рис. 31: Скачивание нужной версии pandoc-crossref.

Скопируем все файлы из архива в каталог `/usr/local/bin`.

```
[dosergeev@vbox ~]$ cd /tmp
[dosergeev@vbox tmp]$ ls
2bad953c-c3e6-4704-8648-a19968d2c158.zip
pandoc-crossref-Linux
sddm-auth-b7be0a20-5e65-435a-ba2f-936aa48f1fa8
sddm--nKzIOs
systemd-private-6c476d479bbf4450a6587d02b1fb8970-abrt.service-3PyQ09
systemd-private-6c476d479bbf4450a6587d02b1fb8970-chronyd.service-4t3KGI
systemd-private-6c476d479bbf4450a6587d02b1fb8970-dbus-broker.service-OFotAS
systemd-private-6c476d479bbf4450a6587d02b1fb8970-irqbalance.service-yg06B1
systemd-private-6c476d479bbf4450a6587d02b1fb8970-ModemManager.service-4aeD06
systemd-private-6c476d479bbf4450a6587d02b1fb8970-polkit.service-gRjaVw
systemd-private-6c476d479bbf4450a6587d02b1fb8970-rtkit-daemon.service-JagwqH
systemd-private-6c476d479bbf4450a6587d02b1fb8970-systemd-logind.service-BadgZX
systemd-private-6c476d479bbf4450a6587d02b1fb8970-systemd-oomd.service-zfNMJv
systemd-private-6c476d479bbf4450a6587d02b1fb8970-systemd-resolved.service-ydGkAY
systemd-private-6c476d479bbf4450a6587d02b1fb8970-upower.service-YR9hnd
xa-GlYW22
[dosergeev@vbox tmp]$ cd pandoc-crossref-Linux/
[dosergeev@vbox pandoc-crossref-Linux]$ ls
pandoc-crossref  pandoc-crossref.1
[dosergeev@vbox pandoc-crossref-Linux]$ cp * /usr/local/bin
cp: невозможно создать обычный файл '/usr/local/bin/pandoc-crossref': Отказано в доступе
cp: невозможно создать обычный файл '/usr/local/bin/pandoc-crossref.1': Отказано в доступе
[dosergeev@vbox pandoc-crossref-Linux]$ sudo cp * /usr/local/bin
[sudo] пароль для dosergeev:
[dosergeev@vbox pandoc-crossref-Linux]$ ls /usr/local/bin/
lib/  lib64/  libexec/  local/
[dosergeev@vbox pandoc-crossref-Linux]$ ls /usr/local/bin/
pandoc-crossref  pandoc-crossref.1
[dosergeev@vbox pandoc-crossref-Linux]$
```

Рис. 33: Копирование pandoc-crossref в нужный каталог.

Ход выполнения домашнего задания

Ход выполнения домашнего задания

1. Дождемся загрузки графического окружения и откроем терминал. Пропишем команду `dmesg` и узнаем последовательность загрузки системы.

```
root
[ 0.000000] Linux version 6.13.5-200.fc41.x86_64 (mockbuildd00e03du54f8364b379359fe70f52a8f23) (gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7), GNU ld version 2.43.1-5.fc41) #1 SMP PREEMPT_DYNAMIC Thu Feb 27 15:07:31 UTC 2025
[ 0.000000] Command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.13.5-200.fc41.x86_64 root=UUID=e76d5148-8cd9-453d-931a-f42f53bcd8f1 ro rootflags=subvol=root nomodeset rhgb quiet
[ 0.000000] [Firmware Bug]: TSC doesn't count with P8 frequency!
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009ffff] usable
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x000000000015ffff] usable
[ 0.000000] BIOS-e820: [mem 0x0000000000160000-0x00000000001a8fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000001a9000-0x00000000001dcfff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] ACPI NVS
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000001de000-0x00000000001dcfff] usable
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] APIC: Static calls initialized
[ 0.000000] efi: EFI v2.7 by EDK II
[ 0.000000] efi: ACPI=0xde97e000 ACPI 2.0=0xde97e014 SMBIOS=0xde9fd000 MOKvar=0xde9f6000 RNG=0xde975018
[ 0.000000] random: crng init done
[ 0.000000] efi: Remove mem129: MMIO range=[0xffc00000-0xffffffff] (4MB) from e820 map
[ 0.000000] e820: remove [mem 0xffc00000-0xffffffff] reserved
[ 0.000000] secureboot: Secure boot disabled
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: Intel(R) Core(TM) i7-10700K CPU @ 3.80GHz VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] DMI: Memory slots populated: 8/0
```

Рис. 35: Вывод команды `dmesg`

2. Получим информацию о:

- Версии ядра Linux -> 6.13.5-200.fc41.x86_64
- Частоте процессора -> 3400 MHz
- Модели процессора -> AMD Ryzen 5 2600
- Объёме доступной ОЗУ -> ~4 GB
- Типе гипервизора -> KVM
- Типе файловой системы корневого раздела -> EXT4-fs
- Последовательности монтирования файловых систем -> BTRFS, EXT4-fs

Ход выполнения домашнего задания

```
[root@vbox ~]# dmesg | grep -i "linux version"
[ 0.000000] Linux version 6.13.5-200.fc41.x86_64 (mockbuild@be03da54f8364b379359fe70f52a8f23) (gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7), GNU ld version 2.43.1-5.fc41) #1 SMP PREEMPT_DYNAMIC Thu Feb 27 15:07:31 UTC 2025
[root@vbox ~]# dmesg | grep -i mhz
[ 0.000010] tsc: Detected 3399.998 MHz processor
[ 10.350687] e1000 0000:00:03:0 eth0: (PCI:33MHz:32-bit) 08:00:27:5a:e3:4a
[root@vbox ~]# dmesg | grep -i cpu0
[ 0.872844] smpboot: CPU0: AMD Ryzen 5 2600 Six-Core Processor (family: 0x17, model: 0x8, stepping: 0x2)
[root@vbox ~]# dmesg | grep -i "memory available"
[root@vbox ~]# dmesg | grep -i "available"
[ 0.006767] On node 0, zone DMA: 1 pages in unavailable ranges
[ 0.007857] On node 0, zone DMA: 96 pages in unavailable ranges
[ 0.250632] On node 0, zone DMA32: 73 pages in unavailable ranges
[ 0.251445] On node 0, zone DMA32: 786 pages in unavailable ranges
[ 0.287451] On node 0, zone Normal: 3730 pages in unavailable ranges
[ 0.290619] [mem 0xdfff0000-0xffffffff] available for PCI devices
[ 0.305826] Booted with the nomodeset parameter. Only the system framebuffer will be available
[ 0.875348] Performance Events: PMU not available due to virtualization, using software events only.
[ 0.903677] Memory: 3937260K/4175560K available (22528K kernel code, 4456K rwdata, 16892K rodata, 4924K init, 4632K bss, 231552K reserved, 0K cma-reserved)
[root@vbox ~]# dmesg | grep -i hypervisor
[ 0.000000] Hypervisor detected: KVM
[root@vbox ~]# dmesg | grep -i filesystem
[ 4.305077] BTRFS info (device sda3): first mount of filesystem e76d5148-8cd9-453d-931a-f42f53bcdbf1
[ 9.775802] EXT4-fs (sda2): mounted filesystem d438a579-5100-4f51-a135-e71b831df7c6 r/w with ordered data mode. Quota mode: none.
[root@vbox ~]#
```

Рис. 36: Нахождение информации о системе

Вывод

В результате выполнения лабораторной работы я приобрел навыки установки операционной системы на виртуальную машину и научился минимально настраивать систему для дальнейшей работы сервисов.