

# Лабораторная работа №1. Установка ОС Linux

[Сунь Шэнцзе]

2026-02-24

number: "1132254527"

## **1. Цель работы**

- Получите практические навыки установки операционной системы (Linux Fedora) на виртуальную машину.
- Научитесь настраивать основные службы, необходимые после запуска системы.
- Ознакомьтесь с основными функциями и операциями командной строки VirtualBox.
- Научитесь анализировать информацию о запуске системы для получения данных об оборудовании и ядре.

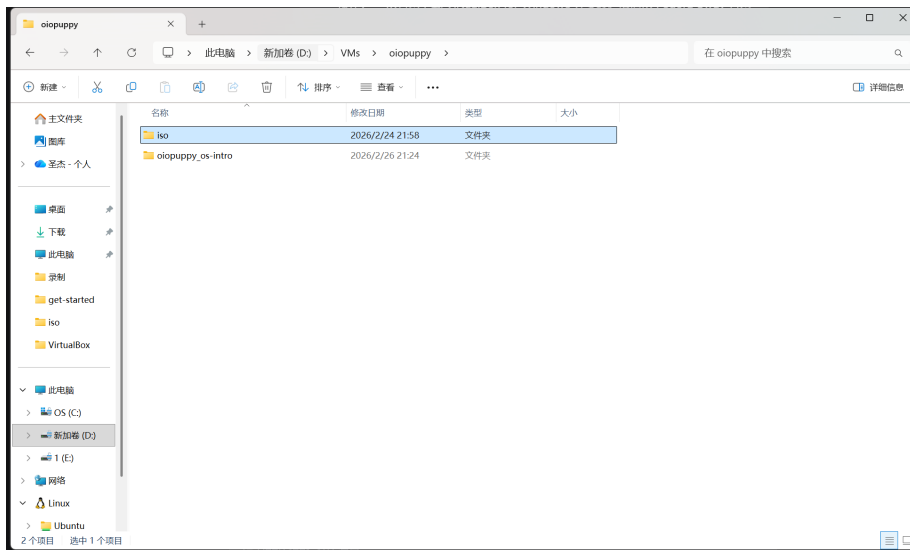
## 2. Порядок выполнения работы и результаты

### 2.1 Шаг 1: Установите VirtualBox и скачайте ISO-образ Fedora Sway.



\* **Инструкция:** Загрузите и установите VirtualBox для Windows с официального сайта. Затем загрузите последнюю версию ISO-образа с сайта Fedora Sway (например, Fedora-Sway-Live-x86\_64-41-1.4.iso). \* **Результат:** VirtualBox успешно установлен, и ISO-файл сохранен локально.

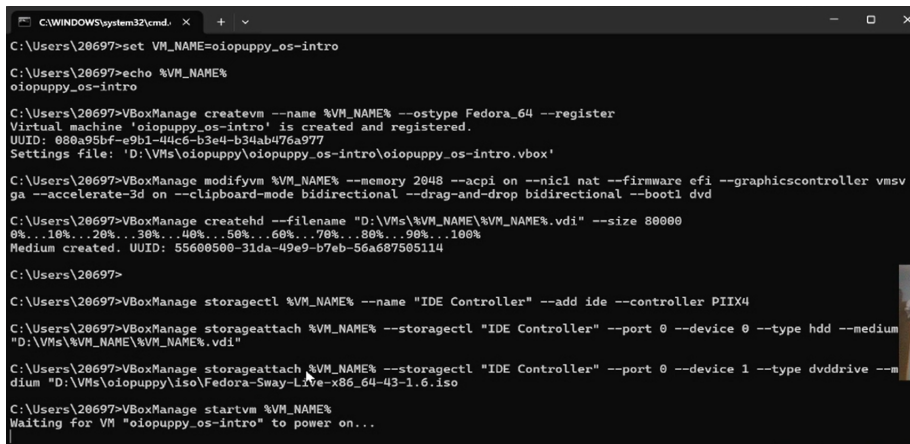
## 2.2 Шаг 2: Создайте рабочую директорию и настройте VirtualBox.



\* **Заказ** `cmd` `d: mkdir D:\VMs\oioruppy mkdir D:\VMs\oioruppy\iso` Переместите ISO-образ в папку D:\VMs\oioruppy\iso\.

Затем настройте глобальные параметры VirtualBox в командной строке: `cmd VBoxManage setproperty language C VBoxManage setproperty machinefolder D:\VMs\oioruppy VBoxManage setextradata global GUI/Input/HostKeyCombination 65383 VBoxManage list systemproperties | find "Default machine folder:"` **Результат:** Папка виртуальной машины по умолчанию установлена на D:\VMs\oioruppy, а ключ Host изменен на ключ Menu во избежание конфликтов.

## 2.3 Шаг 3: Создайте виртуальную машину



```
C:\WINDOWS\system32\cmd. x + v
C:\Users\20697>set VM_NAME=oiopuppy_os-intro
C:\Users\20697>echo %VM_NAME%
oiopuppy_os-intro
C:\Users\20697>VBoxManage createvm --name %VM_NAME% --ostype Fedora_64 --register
Virtual machine 'oiopuppy_os-intro' is created and registered.
UUID: 080a95bf-e9b1-44c6-b3e4-b34ab476a977
Settings file: 'D:\VMs\oiopuppy\oiopuppy_os-intro\oiopuppy_os-intro.vbox'
C:\Users\20697>VBoxManage modifyvm %VM_NAME% --memory 2048 --acpi on --nic1 nat --firmware efi --graphicscontroller vmsvga --accelerate-3d on --clipboard-mode bidirectional --drag-and-drop bidirectional --boot1 dvd
C:\Users\20697>VBoxManage createhd --filename "D:\VMs\%VM_NAME%\%VM_NAME%.vdi" --size 80000
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
Medium created. UUID: 55600500-31da-49e9-b7eb-56a687505114
C:\Users\20697>
C:\Users\20697>VBoxManage storagectl %VM_NAME% --name "IDE Controller" --add ide --controller PIIX4
C:\Users\20697>VBoxManage storageattach %VM_NAME% --storagectl "IDE Controller" --port 0 --device 0 --type hdd --medium "D:\VMs\%VM_NAME%\%VM_NAME%.vdi"
C:\Users\20697>VBoxManage storageattach %VM_NAME% --storagectl "IDE Controller" --port 0 --device 1 --type dvddrive --medium "D:\VMs\oiopuppy\iso\Fedora-Sway-Live-x86_64-41-1.4.iso"
C:\Users\20697>VBoxManage startvm %VM_NAME%
Waiting for VM "oiopuppy_os-intro" to power on...
```

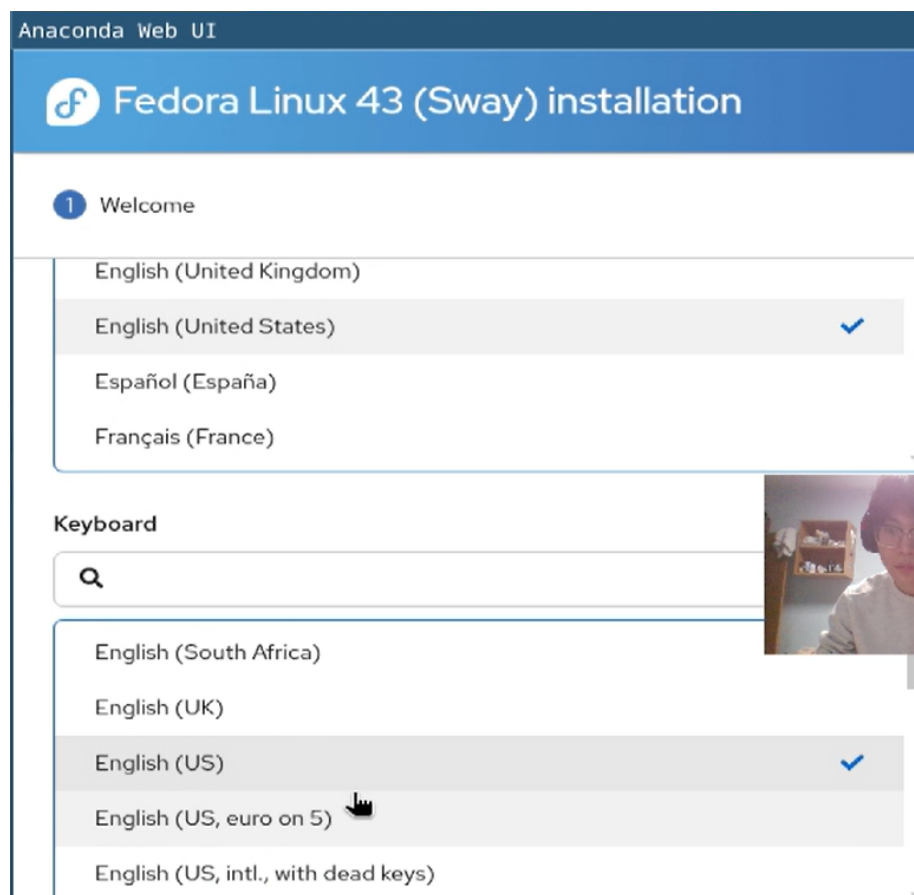
\* **Заказ** cmd set VM\_NAME=oiopuppy\_os-intro echo %VM\_NAME% VBoxManage createvm --name %VM\_NAME% --ostype Fedora\_64 --register VBoxManage modifyvm %VM\_NAME% --memory 2048 --acpi on --nic1 nat --firmware efi --graphicscontroller vmsvga --accelerate-3d on --clipboard-mode bidirectional --drag-and-drop bidirectional --boot1 dvd VBoxManage createhd --filename "D:\VMs\%VM\_NAME%\%VM\_NAME%.vdi" --size 80000 VBoxManage storagectl %VM\_NAME% --name "IDE Controller" --add ide --controller PIIX4 VBoxManage storageattach %VM\_NAME% --storagectl "IDE Controller" --port 0 --device 0 --type hdd --medium "D:\VMs\%VM\_NAME%\%VM\_NAME%.vdi" VBoxManage storageattach %VM\_NAME% --storagectl "IDE Controller" --port 0 --device 1 --type dvddrive --medium "D:\VMs\oiopuppy\iso\Fedora-Sway-Live-x86\_64-41-1.4.iso" **Результат:** Виртуальная машина oio puppy\_os-intro была успешно создана и настроена с 2 ГБ оперативной памяти, виртуальным жестким диском объемом 80 ГБ и смонтированным образом Fedora ISO.

## 2.4 Шаг 4: Запустите виртуальную машину и войдите в рабочую среду.



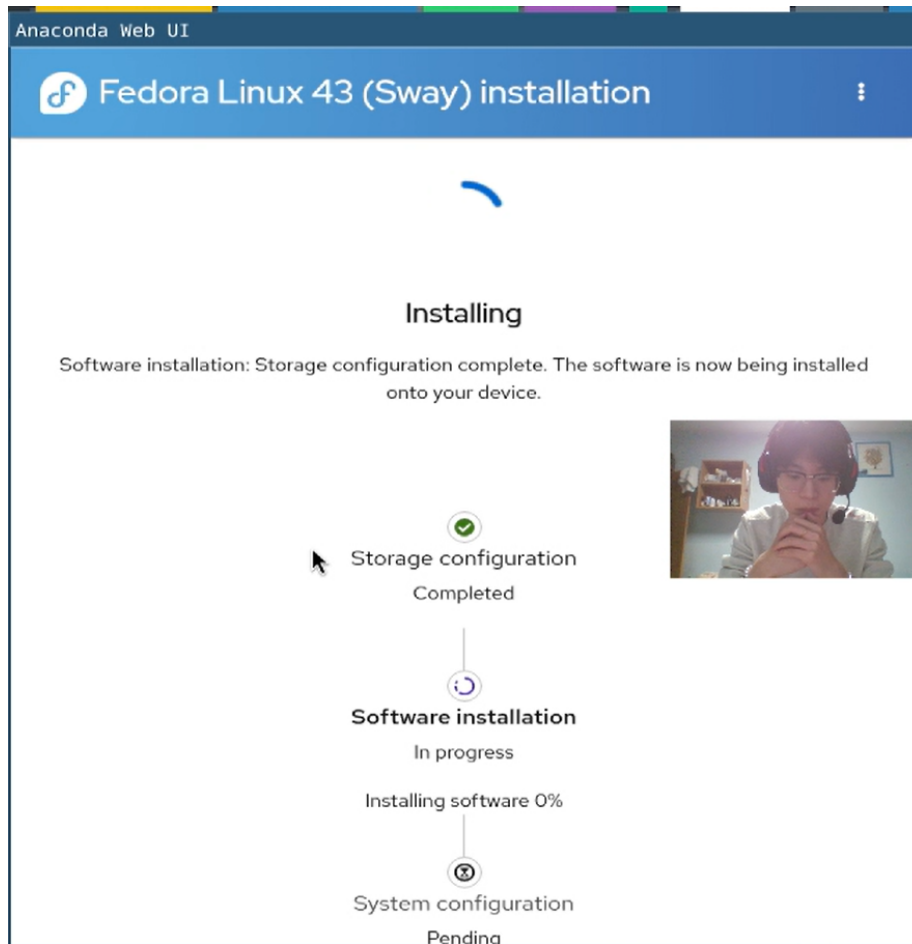
- \* **Команда:** `VBoxManage startvm %VM_NAME%` \* **Инструкции:** В окне виртуальной машины выберите `Start Fedora-Sway-Live 41` и дождитесь запуска системы. Нажмите `Win+Enter`, чтобы открыть терминал, и введите `liveinst`, чтобы запустить установщик.
- \* **Результат:** Успешно загружен рабочий стол Fedora Live и запущен графический установщик.

## 2.5 Шаг 5: Настройка параметров установки



\* **Инструкции:** Во время установки настройте следующие параметры: - Язык: Английский (США) - Часовой пояс: Выберите свой регион (например, Азия/Шанхай) - Раскладка клавиатуры: Добавьте русский язык и установите клавишу переключения на **Правая клавиша Ctrl** - Место установки: Автоматическое разбиение диска - Сеть и имя хоста: Установите имя хоста на `oioruppu` и включите сетевой доступ - Настройки пользователя: Установите пароль `root`, создайте обычного пользователя `oioruppu` и предоставьте ему права администратора \* **Результат:** После настройки всех параметров установки нажмите «Начать установку», чтобы начать установку.

## 2.6 Шаг 6: Процесс установки и перезапуск



\* **Инструкции:** Дождитесь завершения установки (примерно 5-10 минут), затем нажмите «Перезагрузить систему». После перезагрузки, если вы снова войдете в ISO-образ, отмонтируйте CD и перезапустите виртуальную машину в меню VirtualBox. \*  
**Результат:** Успешно загружен с жесткого диска, вошел в меню Grub, выбрал Fedora Linux и вышел на рабочий стол.

## 2.7 Шаг 7: Настройка после установки

### 2.7.1 Установите дополнения для гостевой системы VirtualBox.

```

sunshengjie@fedora:~$ sudo /media/cdrom/VBoxLinuxAdditions.run
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.0.26 Guest Additions for Linux 100%
VirtualBox Guest Additions installer
Removing installed version 7.0.26 of VirtualBox Guest Additions...
Copying additional installer modules ...
Installing additional modules ...
VirtualBox Guest Additions: Starting.
VirtualBox Guest Additions: Setting up modules
VirtualBox Guest Additions: Building the VirtualBox Guest Additions kernel
modules. This may take a while.
VirtualBox Guest Additions: To build modules for other installed kernels, run
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup <version>
VirtualBox Guest Additions: or
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup all
VirtualBox Guest Additions: Building the modules for kernel
6.18.12-200.fc43.x86_64.

VirtualBox Guest Additions: Look at /var/log/vboxadd-setup.log to find out what
went wrong
File context for /opt/VBoxGuestAdditions-7.0.26/other/mount.vboxsf already defined, modifying instead
VirtualBox Guest Additions: Running kernel modules will not be replaced until
the system is restarted or 'rcvboxadd reload' triggered
VirtualBox Guest Additions: reloading kernel modules and services
VirtualBox Guest Additions: cannot reload kernel modules: one or more module(s)
is still in use
VirtualBox Guest Additions: kernel modules and services were not reloaded
The log file /var/log/vboxadd-setup.log may contain further information.

```

\* **Команда (выполняется в терминале виртуальной машины)** `bash sudo -i dnf -y group install "Development Tools" dnf -y install dkms kernel-devel mount /dev/sr0 /media /media/VBoxLinuxAdditions.run reboot` **Результат:** Приложение Guest Additions успешно установлено, и такие функции, как интеграция с мышью и обмен буфером обмена, работают корректно.

## 2.7.2 Обновите систему и установите часто используемые инструменты.



```

foot
100% 10.0.2.15/24 2% 28% US 100% 00:31

kernel x86_64 6.18.12-200.fc43 updates 0.0 MiB
kernel-modules x86_64 6.18.12-200.fc43 updates 96.3 MiB
kernel-modules-extra x86_64 6.18.12-200.fc43 updates 4.2 MiB
Installing group/module packages:
gnupg2-g13 x86_64 2.4.9-5.fc43 updates 199.7 KiB
replacing gnupg2 x86_64 2.4.9-5.fc43 koji-override-0 6.5 MiB
gnupg2-scdemon x86_64 2.4.9-5.fc43 updates 689.6 KiB
replacing gnupg2 x86_64 2.4.8-4.fc43 koji-override-0 6.5 MiB
gnupg2-utils x86_64 2.4.9-5.fc43 updates 342.7 KiB
replacing gnupg2 x86_64 2.4.8-4.fc43 koji-override-0 6.5 MiB
gnupg2-wks x86_64 2.4.9-5.fc43 updates 269.4 KiB
replacing gnupg2 x86_64 2.4.8-4.fc43 koji-override-0 6.5 MiB
Installing dependencies:
aml x86_64 0.3.0-8.fc43 fedora updates 28.9 KiB
dns-root-data noarch 2026260100-2.fc43 updates 18.9 KiB
mozilla-openh264 x86_64 2.6.0-2.fc43 updates 1.1 MiB
ngtcp2-crypto-gnutls x86_64 1.19.0-1.fc43 updates 39.6 KiB
ngtcp2-crypto-openssl x86_64 1.19.0-1.fc43 updates 51.7 KiB
openh264 x86_64 2.6.0-2.fc43 fedora-cisco-openh264 koji-override-0 6.5 MiB
replacing noopenh264 x86_64 2.6.0-2.fc43 updates 6.5 MiB
valli x86_64 0.1.1-1.fc43 updates updates
Installing weak dependencies:
scurl noarch 2026.01.05-1.fc43 updates updates
Transaction Summary:
Installing: 15 packages
Upgrading: 547 packages
Replacing: 549 packages
Total size of inbound packages is 1 GiB. Need to download 1 GiB.
After this operation, 200 MiB extra will be used (install 3 GiB, remove 2 GiB).
[ 1/562] gnupg2-util-0:2.4.9-5.fc43.x86_64 100% | 255.1 KiB/s | 129.6 KiB | 00m01s
[ 2/562] gnupg2-g13-0:2.4.9-5.fc43.x86_64 100% | 153.2 KiB/s | 82.6 KiB | 00m01s
[ 3/562] gnupg2-wks-0:2.4.9-5.fc43.x86_64 100% | 1.8 MiB/s | 111.4 KiB | 00m00s
[ 4/562] valli-0:0.1.1-1.fc43.x86_64 100% | 370.1 KiB/s | 22.6 KiB | 00m00s
[ 5/562] gnupg2-scdemon-0:2.4.9-5.fc43.x86_64 100% | 498.7 KiB/s | 308.7 KiB | 00m01s
[ 6/562] aml-0:0.3.0-8.fc43.x86_64 100% | 214.5 KiB/s | 28.2 KiB | 00m00s
[ 7/562] kernel-0:6.18.12-200.fc43.x86_64 100% | 1.1 MiB/s | 224.6 KiB | 00m00s
[ 8/562] ngtcp2-crypto-gnutls-0:1.19.0-1.fc43.x86_64 100% | 290.8 KiB/s | 23.8 KiB | 00m00s
[ 9/562] kernel-modules-extra-0:6.18.12-200.fc43.x86_64 1% [ ] | 1.2 MiB/s | 1.8 MiB | 01m22s
[10/562] kernel-modules-extra-0:6.18.12-200.fc43.x86_64 30% [=====] | 627.5 KiB/s | 1.4 MiB | 00m00s
[11/562] openh264-0:2.6.0-2.fc43.x86_64 0% [<==] | 1.0 B/s | 0.0 B | 05s00h

```

\* **Заказ** `bash sudo -i dnf -y update dnf -y install tmux mc kitty` **Результат:** Система обновлена до последней версии, и установлены часто используемые инструменты.

## 2.7.3 Отключить SELinux

```

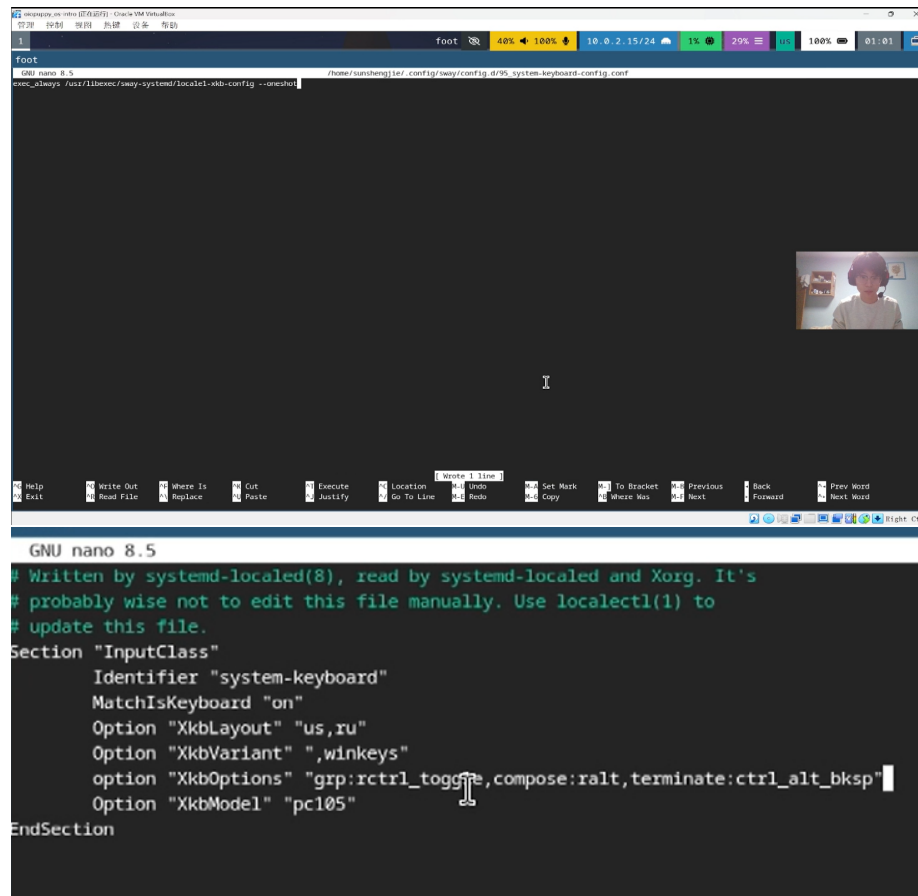
foot
GNU nano 8.5 /etc/selinux/config Modified
# This file controls the state of SELinux on the system.
# SELinux can take one of three values:
# enforcing - SELinux security policy is enforced.
# permissive - SELinux prints warnings instead of enforcing.
# disabled - No SELinux policy is loaded.
# See also:
# https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-selinux/getting-started-with-selinux-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=
SELINUXTYPE can take one of these three values:
# targeted - Targeted processes are protected.
# minimum - Modification of targeted policy. Only selected processes are protected.
# strict - Multi-level Security protection.
SELINUXTYPE=targeted
Write to File: /etc/selinux/config
[Help] [Cancel] [DOS Format] [Mac Format] [Append] [Prepend] [Backup File] [Discard Buffer] [Browse]

```

\* **Заказ** `bash nano /etc/selinux/config` Измените SELINUX=enforcing на SELINUX=permissive, сохраните изменения и перезапустите систему. **Результат:** SELinux переключается в разрешающий

режим.

## 2.7.4 Настройка сохранения раскладки клавиатуры



- \* **Заказ** `bash mkdir -p ~/.config/sway/config.d nano ~/.config/sway/config.d/95-system-keyboard-config.conf` Писать `exec_always /usr/libexec/sway-systemd/locale1-xkb-config --oneshot bash sudo -i mkdir -p /etc/X11/xorg.conf.d nano /etc/X11/xorg.conf.d/00-keyboard.conf` Запишите конфигурацию (включая `grp:rctrl_toggle`), сохраните и перезапустите.
- \* **Результат:** После перезапуска вы можете использовать клавишу `Ctrl` для переключения между английской и русской раскладками.

## 2.8 Глава 8: Домашнее задание - Анализ dmesg

```
sunshengjie@fedora:~$ sudo dmesg | grep "Linux version"
[sudo] password for sunshengjie:
[ 0.000000] Linux version 6.18.12-200.fc43.x86_64 (mockbuild@953c9dc84bd7c460fb40cc5d15480f8a1) (gcc (GCC) 15.2.1 20260123 (Red Hat 15.2.1-7), GNU ld version 2.45.1-4.43) #1 SMP PREEMPT_DYNAMIC Mon Feb 16 18:58:26 UTC 2026
sunshengjie@fedora:~$ sudo dmesg | grep -i "detected.*MHz processor"
[ 0.000005] tsc: Detected 2803.196 MHz processor
sunshengjie@fedora:~$ sudo dmesg | grep "CPU0"
[ 0.180626] smpboot: CPU0: 13th Gen Intel(R) Core(TM) i7-13650HX (family: 0x6, model: 0xb7, stepping: 0x1)
sunshengjie@fedora:~$ sudo dmesg | grep -i "memory" | tail -5
[ 0.383537] Freeing initrd memory: 37336K
[ 0.481420] Freeing unused decrypted memory: 2028K
[ 0.481990] Freeing unused kernel image (initmem) memory: 5156K
[ 0.482103] Freeing unused kernel image (text/rodata gap) memory: 260K
[ 0.482244] Freeing unused kernel image (rodata/data gap) memory: 892K
sunshengjie@fedora:~$ sudo dmesg | grep -i "Hypervisor detected"
[ 0.000000] Hypervisor detected: KVM
sunshengjie@fedora:~$ sudo dmesg | grep -i "root"
[ 0.000000] Command line: BOOT_IMAGE=(hdb,gpt2)/vmlinuz-6.18.12-200.fc43.x86_64 root=UUID=f80b4c47-2bf6-4d65-ad46-2206fa9f28ef ro rootflags=subvol=root rhgb quiet
[ 0.056914] Kernel command line: BOOT_IMAGE=(hdb,gpt2)/vmlinuz-6.18.12-200.fc43.x86_64 root=UUID=f80b4c47-2bf6-4d65-ad46-2206fa9f28ef ro rootflags=subvol=root rhgb quiet
[ 0.203364] ACPI: PCI Root Bridge [PCI0] (domain 0000 [bus 00-ff])
[ 0.203755] pci_bus 0000:00: root bus resource [io 0x0000-0x0cf7 window]
[ 0.203756] pci_bus 0000:00: root bus resource [io 0x0000-0xffff window]
[ 0.203760] pci_bus 0000:00: root bus resource [mem 0x000a0000-0x000bffff window]
[ 0.203761] pci_bus 0000:00: root bus resource [mem 0x00000000-0xfdf00000 window]
[ 0.203762] pci_bus 0000:00: root bus resource [bus 00-ff]
[ 0.251550] Trying to unpack rootfs image as initramfs...
[ 3.009909] systemd[1]: initrd-switch-root.service: Deactivated successfully.
[ 3.040037] systemd[1]: Stopped initrd-switch-root.service - Switch Root.
[ 3.044010] systemd[1]: Stopped target initrd-switch-root.target - Switch Root.
[ 3.044036] systemd[1]: Stopped target initrd-root-fs.target - Initrd Root File System.
[ 3.092230] systemd[1]: plymouth-switch-root.service: Deactivated successfully.
[ 3.100581] systemd[1]: Stopped plymouth-switch-root.service - Plymouth switch root service.
[ 3.100858] systemd[1]: system-fsck-root.service: Deactivated successfully.
[ 3.100892] systemd[1]: Stopped system-fsck-root.service - File System Check on Root Device.
[ 3.153874] systemd[1]: Starting system-re mount-fs.service - Remount Root and Kernel File Systems...
sunshengjie@fedora:~$ sudo dmesg | grep "mount"
[ 1.771260] BTRFS: device label fedora devid 1 transid 664 /dev/sda3 (8:3) scanned by mount (344)
[ 1.771860] BTRFS: info (device sda3): first mount of filesystem f80b4c47-2bf6-4d65-ad46-2206fa9f28ef
[ 3.042853] systemd[1]: Set up automount proc-sys-fs-binfmt_misc.automount - Arbitrary Executable File Formats File System Automount Point.
[ 3.052175] systemd[1]: Mounting dev-hugepages.mount - Huge Pages File System...
[ 3.056222] systemd[1]: Mounting dev-queue.mount - POSIX Message Queue File System...
[ 3.062434] systemd[1]: Mounting sys-kernel-debug.mount - Kernel Debug File System...
[ 3.066117] systemd[1]: Mounting sys-kernel-tracing.mount - Kernel Trace File System...
[ 3.066343] systemd[1]: fips-crypto-policy-overlay.service - Bind-mount FIPS crypto-policy in FIPS mode skipped, unmet condition check ConditionKernelCommandLine=fips=1
[ 3.089296] systemd[1]: Mounting sys-fs-fuse-connections.mount - FUSE Control File System...
[ 3.153874] systemd[1]: Starting system-re mount-fs.service - Remount Root and Kernel File Systems...
[ 3.214865] audit: type=1130 audit(1772226366.419:10): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=systemd-re mount-fs comm="systemd" exec="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
[ 5.073109] EXT4-fs (sda2): mounted filesystem dc55725c-089f-4744-a6da-97907ccb93aa r/w with ordered data mode. Quota mode: none.
```

\* **Команды и вывод:** `bash dmesg | grep "Linux version"`  
`dmesg | grep -i "detected.*MHz processor"` `dmesg | grep "CPU0"` `dmesg | grep -i "Memory available"` `dmesg | grep -i "Hypervisor detected"` `dmesg | grep "VFS: Mounted root"` `dmesg | grep "mount"` **Результат:** Информация, включающая версию ядра, частоту процессора, модель процессора, доступную память, тип гипервизора, тип корневой файловой системы и порядок монтирования файловых систем, получена успешно. В выводе указано, что система работает в среде виртуальной машины VirtualBox.

### 3. Ответы на вопросы для самопроверки

1. **Учётная запись пользователя содержит какую информацию?**

Учетные записи пользователей обычно содержат следующую информацию: имя пользователя, идентификатор пользователя (UID), идентификатор группы (GID), путь к домашнему каталогу, оболочка по умолчанию, заполнитель пароля (фактически зашифрованный и хранящийся в `/etc/shadow`) и необязательное полное имя. Эта информация в основном хранится в файле `/etc/passwd`.

2. **Укажите команды терминала и приведите примеры:**

- **Получить справку:** `man ls` (Отображает руководство по команде `ls`)
- **Переместить:** `cd /tmp` (Перейти в каталог `/tmp`)
- **Просмотреть содержимое каталога:** `ls -la` (Выводит список всех файлов и их подробную информацию)
- **Просмотреть размер каталога:** `du -sh ~` (Отображает общий размер домашнего каталога)
- **Создать/Удалить:** `mkdir testdir` (Создает каталог), `rmdir testdir` (Удаляет пустой каталог), `touch file.txt` (Создает файл), `rm file.txt` (Удаляет файл)
- **Изменить права доступа:** `chmod 755 script.sh` (Устанавливает права на чтение, запись и выполнение для владельца, а также на чтение и выполнение для группы и других пользователей)
- **Просмотреть историю:** `history` (Отображает историю команд)

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

Файловая система — это метод и структура данных, используемые операционной системой для определения файлов на устройстве хранения (например, жестком диске) или разделе; другими словами, это способ организации файлов. Распространенные примеры:

- **ext4** Встроенная в Linux файловая система с журналированием поддерживает большие файлы, обеспечивает высокую производительность и обладает хорошей совместимостью.
- **XFS** Высокопроизводительная 64-битная файловая система с журналированием, отлично справляющаяся с обработкой больших файлов и высокой параллельной обработкой.
- **btrfs** Современные файловые системы с механизмом

копирования при записи поддерживают расширенные функции, такие как моментальные снимки, сжатие и контрольные суммы.

- **NTFS** В Windows файловая система по умолчанию поддерживает большие файлы, контроль доступа и журналирование.
- **FAT32/exFAT** В основном используется для мобильных устройств, таких как USB-флеш-накопители, и обладает широкой совместимостью, но exFAT поддерживает файлы больших размеров.

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

Команды `mount` или `df -hT` позволяют просмотреть информацию о текущих смонтированных файловых системах, их точках монтирования, типах и объемах.

#### 5. Как удалить зависший процесс?

Сначала используйте команду `ps aux | grep process_name`, чтобы найти PID процесса, а затем отправьте сигнал на его завершение

- изящное завершение `kill PID`
- Принудительное прекращение (применимо к приостановленным процессам) `kill -9 PID`

## Выводы

В ходе этого практического занятия я успешно завершил весь процесс установки виртуальной машины Fedora Sway на хост-систему Windows с помощью VirtualBox. Мои основные выводы следующие

1. **Освоил основные принципы работы технологии виртуализации:** Я научился создавать, настраивать и управлять виртуальными машинами, включая параметры памяти, жесткого диска, сети и EFI.
2. **Знаком с процессом установки системы Fedora** От загрузки в Live-среду до использования графического установщика (Anaconda), включая разметку диска, создание пользователей, настройку раскладки клавиатуры и т. д.
3. **Разберитесь в базовой конфигурации после запуска системы** Для расширения функциональности виртуальной машины была установлена программа VirtualBox Guest Additions, система была обновлена, установлены часто используемые инструменты, а SELinux был отключен для упрощения последующих операций.

4. **Научился сохранять настройки раскладки клавиатуры на постоянной основе**□Изменение конфигурационных файлов sway и X11 позволяет сохранить действие переключения раскладки клавиатуры (правая клавиша Ctrl) после перезагрузки.
5. **Освойте использование команды dmesg**□Анализируя информацию о запуске системы, мы получили ключевые данные об оборудовании, такие как версия ядра, процессор, память и гипервизор, что углубило наше понимание процесса запуска Linux.
6. **Улучшена поддержка основных команд Linux.**□При ответе на вопросы, касающиеся управления, были рассмотрены часто используемые команды, такие как операции с файлами, управление правами доступа и управление процессами.

Все экспериментальные задачи выполнены. Я не только освоил метод установки Linux в виртуальной машине, но и заложил прочный фундамент для дальнейшего углубленного изучения операционных систем и системного администрирования.