

Лабораторная работа №1

Операционные системы

Башиянц А. К.

05 марта 2025

Российский университет дружбы народов, Москва, Россия

Вводная часть

Цель данной работы — приобрести практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

- Устанавливать виртуальную машину;
- Устанавливать необходимые пакеты;
- Получать информацию о системе.

Выполнение лабораторной работы

После скачивания образа Fedora установим виртуальную машину.

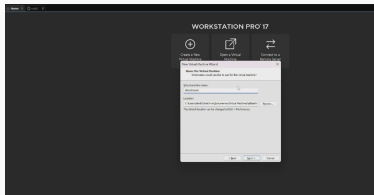


Рис. 1: Установка Fedora (VMWare)

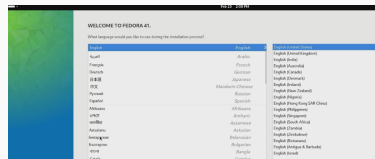


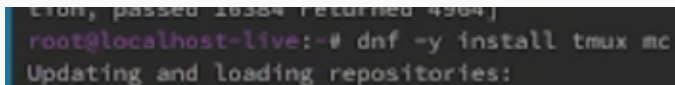
Рис. 2: Установка Fedora

Установим средства разработки, пакет DKMS, подмонтируем диск и установим драйвера.

```
liveuser@localhost-live:~$ sudo -i
root@localhost-live:~# sudo dnf -y group install development-tools
Updating and loading repositories:
Fedora 41 - x86_64 - Updates          100% | 3.0 MiB/s | 11.1 MiB | 00m04s
Fedora 41 openh264 (From Cisco) - x86_ 100% | 3.1 KiB/s | 6.0 KiB | 00m02s
Fedora 41 - x86_64                   38% | 137.6 KiB/s | 14.5 MiB | ~02m53s
```

Рис. 3: Установка средства разработки

Установим средства разработки и программы для удобства работы в консоли.

A terminal window with a dark background. The prompt is 'root@localhost-live:~#'. The command 'dnf -y install tmux mc' has been entered. Below the command, the text 'Updating and loading repositories:' is displayed. A white cursor is visible at the end of the line.

```
tion, passed 16384 returned 4964]  
root@localhost-live:~# dnf -y install tmux mc  
Updating and loading repositories:  
█
```

Рис. 4: Установка средства разработки

Сделаем автоматическое обновление.

```
) provided by libcurl-8.9.1-3.fc41.x86_64 from updates
- package libcurl-minimal-8.9.1-2.fc41.x86_64 from fedora conflicts with libcurl(x86_64) provided by libcurl-8.9.1-3.fc41.x86_64 from updates
- package libcurl-minimal-8.9.1-3.fc41.x86_64 from updates conflicts with libcurl(x86_64) provided by libcurl-8.9.1-3.fc41.x86_64 from updates
- cannot install the best candidate for the job
```

Package	Arch	Version	Repository	Size
Installing:				
dnf-automatic	noarch	4.21.1-1.fc41	fedora	76.6 KiB
Skipping packages with conflicts:				
libcurl	x86_64	8.9.1-2.fc41	fedora	818.1 KiB
libcurl	x86_64	8.9.1-3.fc41	updates	809.3 KiB
libcurl-minimal	x86_64	8.9.1-3.fc41	updates	641.2 KiB
Skipping packages with broken dependencies:				
dnf5-plugin-automatic	x86_64	5.2.10.0-2.fc41	updates	178.6 KiB

Transaction Summary:

Installing: 1 package

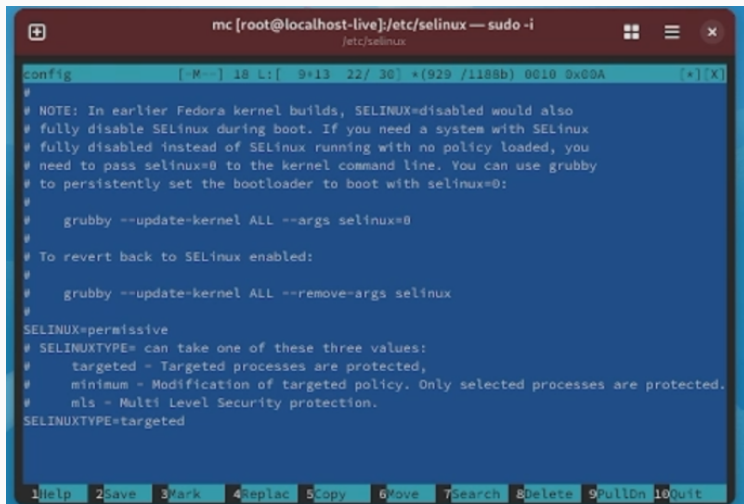
Skipping: 4 packages

Total size of inbound packages is 41 KiB. Need to download 41 KiB.

After this operation, 77 KiB extra will be used (install 77 KiB, remove 0 B).

Отключение SELinux

Отключим SELinux.



```
mc [root@localhost-live]:/etc/selinux — sudo -i
/etc/selinux

config [-M--] 18 L:[ 9*13 22/ 30] *(929 /1188b) 0010 0x00A [*][X]
#
# 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 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

1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn 10Quit
```

Установим имя пользователя и название хоста. Проверим, что все правильно.

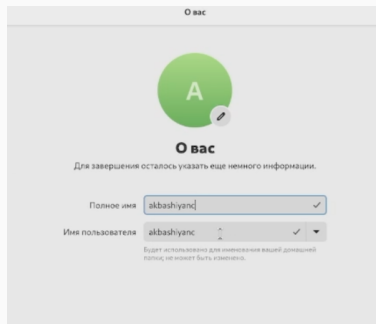


Рис. 6: Установка имени пользователя

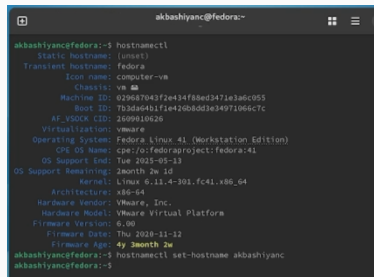


Рис. 7: Проверка

Установим Pandoc

```
root@akbashiyanc:~# sudo dnf -y install pandoc
Updating and loading repositories:
  Fedora 41 - x86_64 - Updates          100% | 1.3 MiB/s | 11.1 MiB | 00m09s
  Fedora 41 openh264 (From Cisco) - x86_ 100% | 2.4 KiB/s | 6.0 KiB | 00m02s
  Fedora 41 - x86_64                   100% | 4.6 MiB/s | 35.4 MiB | 00m08s
Repositories loaded.
Package Arch Version Repository Size
Installing:
pandoc x86_64 3.1.11.1-32.fc41 fedora 185.0 MiB
Installing dependencies:
pandoc-common noarch 3.1.11.1-31.fc41 fedora 1.9 MiB

Transaction Summary:
Installing: 2 packages

Total size of inbound packages is 27 MiB. Need to download 27 MiB.
After this operation, 187 MiB extra will be used (install 187 MiB, remove 0 B).
[1/2] pandoc-common-0:3.1.11.1-31.fc41. 100% | 419.6 KiB/s | 537.1 KiB | 00m01s
```

Рис. 8: Установка Pandoc

Установим дистрибутив TeXlive

```
root@akbashiyanc:~# dnf -y install texlive-scheme-full
Updating and loading repositories:
  Fedora 41 - x86_64 - Updates      100% | 17.9 KiB/s | 19.1 KiB | 00m01s
  Fedora 41 - x86_64 - Updates      100% | 1.2 MiB/s | 3.3 MiB | 00m03s
Repositories loaded.
```

Рис. 9: Установка TeXlive

Домашнее задание

С помощью `grep` выясним следующую информацию:

```
root@akbashiyanc:~# dmesg | grep -i "Linux version"
[    0.000000] Linux version 6.11.4-301.fc41.x86_64 (mockbuild@9b6b61418589428cb
880a7020233b56f) (gcc (GCC) 14.2.1 20240912 (Red Hat 14.2.1-3), GNU ld version 2
.43.1-2.fc41) #1 SMP PREEMPT_DYNAMIC Sun Oct 20 15:02:33 UTC 2024
root@akbashiyanc:~# dmesg | grep -i "Detected Mhz processor"
root@akbashiyanc:~# dmesg | grep -i "Mhz"
[    0.000000] vmware: TSC freq read from hypervisor : 2694.936 MHz
[    0.000025] tsc: Detected 2694.936 MHz processor
[    1.314226] hpet0: 16 comparators, 64-bit 14.318180 MHz counter
root@akbashiyanc:~# dmesg | grep -i "CPU0"
[    0.780206] smpboot: CPU0: AMD Ryzen 3 4300U with Radeon Graphics (family: 0x
17, model: 0x60, stepping: 0x1)
```

Рис. 10: Версия ядра Linux

Частота процессора (Detected Mhz processor)

```
root@akbashiyanc:~# dmesg | grep -i "Detected Mhz processor"
root@akbashiyanc:~# dmesg | grep -i "Mhz"
[ 0.000000] vmware: TSC freq read from hypervisor : 2694.936 MHz
[ 0.000025] tsc: Detected 2694.936 MHz processor
[ 1.314226] hpet0: 16 comparators, 64-bit 14.318180 MHz counter
root@akbashiyanc:~#
```

Рис. 11: Частота процессора

Модель процессора (CPU0)

```
root@akbashiyan:~# dmesg | grep -i "CPU0"  
[ 0.780206] smpboot: CPU0: AMD Ryzen 3 4300U with Radeon Graphics (family: 0x  
17, model: 0x60, stepping: 0x1)  
root@akbashiyan:~#
```

Рис. 12: Модель процессора

Объём доступной оперативной памяти (Memory available)

```
[ 0.140289] Booting paravirtialized kernel on platform type: x86_64
root@akbashiyanc:~# dmesg | grep -i "RAM"
[ 0.000000] BIOS-provided physical RAM map:
[ 0.003092] total RAM covered: 3072M
[ 0.003339] gran_size: 64K chunk_size: 64K          num_reg: 2          lose cov
er RAM: 0G
[ 0.008417] RAMDISK: [mem 0x34182000-0x360b8fff]
[ 0.565979] Unknown kernel command line parameters "rhgb BOOT_IMAGE=(hd0,gpt2
)/vmlinuz-6.11.4-301.fc41.x86_64", will be passed to user space.
[ 0.686790] Trampoline variant of Tasks RCU enabled.
[ 0.784529] signal: max sigframe size: 1776
[ 1.313428] e820: reserve RAM buffer [mem 0x00097c00-0x0009ffff]
[ 1.313436] e820: reserve RAM buffer [mem 0xbfee0000-0xbfffffff]
```

Рис. 13: Объём доступной оперативной памяти

Тип обнаруженного гипервизора (Hypervisor detected)

```
root@akbashiyan:~# dmesg | grep -i "Hypervisor"  
[ 0.000000] Hypervisor detected: VMware  
[ 0.000000] vmware: TSC freq read from hypervisor : 2694.936 MHz  
[ 0.000000] vmware: Host bus clock speed read from hypervisor : 66000000 Hz  
[ 0.140133] Booting paravirtualized kernel on VMware hypervisor
```

Рис. 14: Тип обнаруженного гипервизора

Тип файловой системы корневого раздела

```
root@akbashiyanc:~# mount
/dev/nvme0n1p3 on / type btrfs (rw,relatime,seclabel,compress=zstd:1,ssd,space_cache=v2,subvolid=257,subvol=/root)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=368402,mode=755,inode64)
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)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
```

Рис. 15: Тип файловой системы корневого раздела

Последовательность монтирования файловых систем

```
root@akbashiyanc:~# dmesg | grep -i "mount"
[ 0.772293] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 0.772977] Mountpoint-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 6.461185] BTRFS: device label fedora devid 1 transid 533 /dev/nvme0n1p3 (259:3) scanned by mount (579)
[ 6.462277] BTRFS info (device nvme0n1p3): first mount of filesystem blad6a61-418c-4522-b989-51ee6aba2bb8
[ 8.198326] systemd[1]: run-credentials-systemd\x2djournald.service.mount: Deactivated successfully.
[ 8.208226] systemd[1]: Set up automount proc-sys-fs-binfmt_misc.automount -
```

Рис. 16: Последовательность монтирования файловых систем

Выводы

- В этой лабораторной работе мы изучили работу виртуальной машины и ее настройки.