

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

дисциплина: Операционные системы

Студент: Кондратьев Арсений
Вячеславович

Группа: НПИбд-01-21

Ст. билет №: 1132210645

Москва

2022

Цель работы: Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

Ход работы:

1. Скачиваем Oracle VM VirtualBox и Fedora Linux (Рис. 1 и 2)



Рис. 1 (скачиваем VirtualBox)

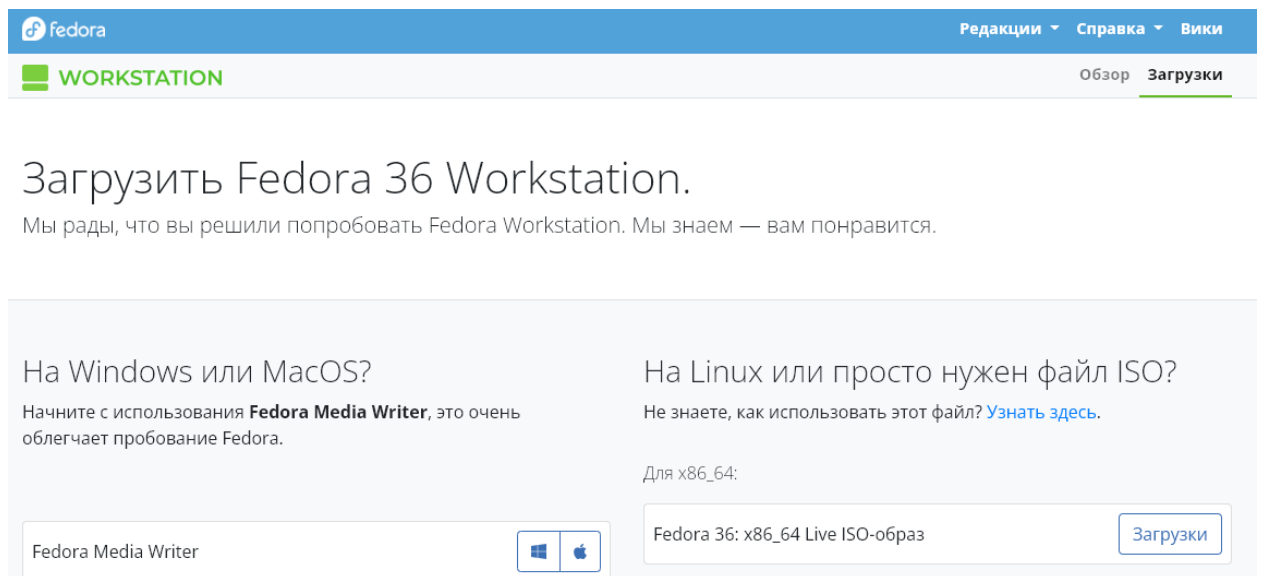


Рис. 2 (скачиваем Fedora)

2. Устанавливаем виртуальную машину выбирая необходимый объем памяти и делая ее динамической. (рис. 3 и 4)

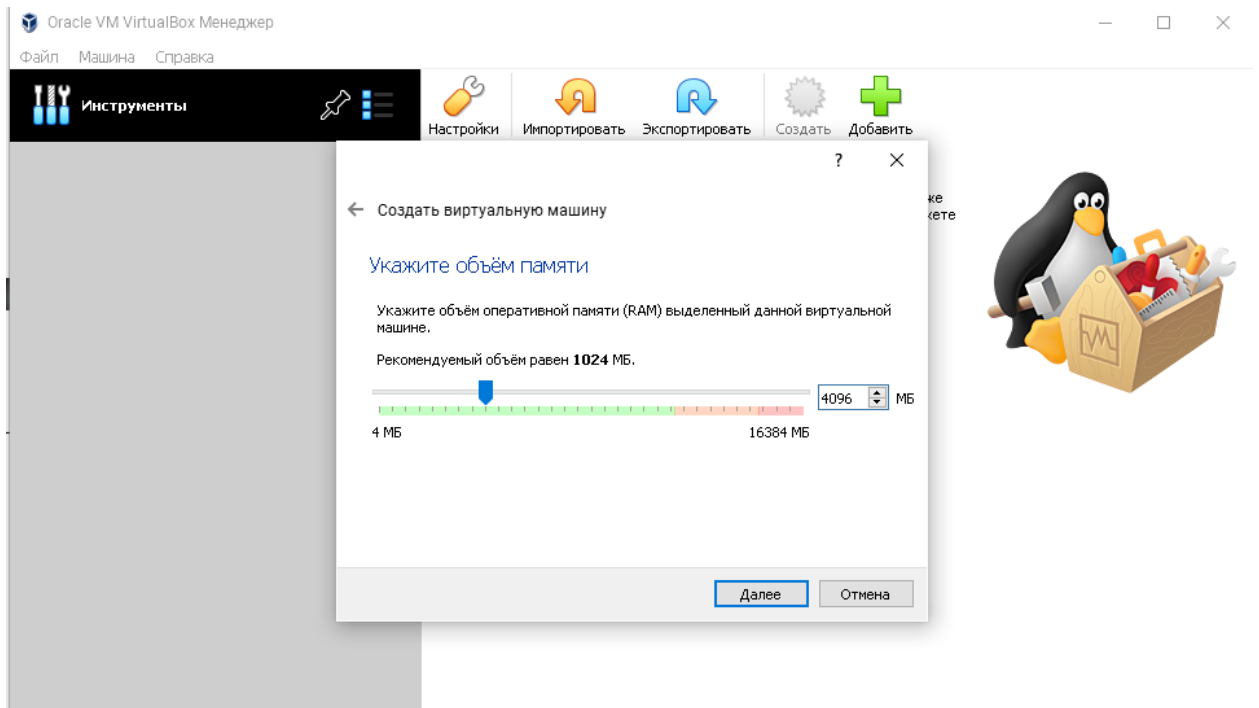


Рис. 3 (указываем объем оперативной памяти)

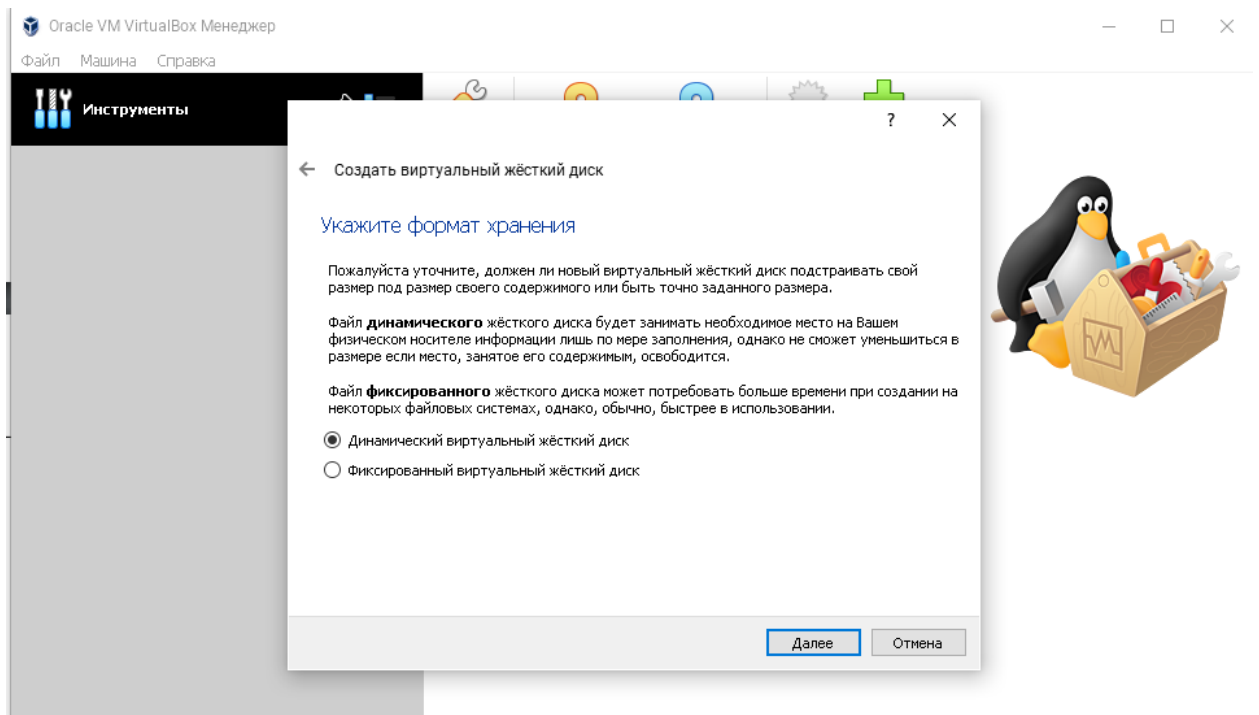


Рис. 4 (выбираем динамический тип памяти)

3. Устанавливаем Fedora на VirtualBox (Рис. 5)

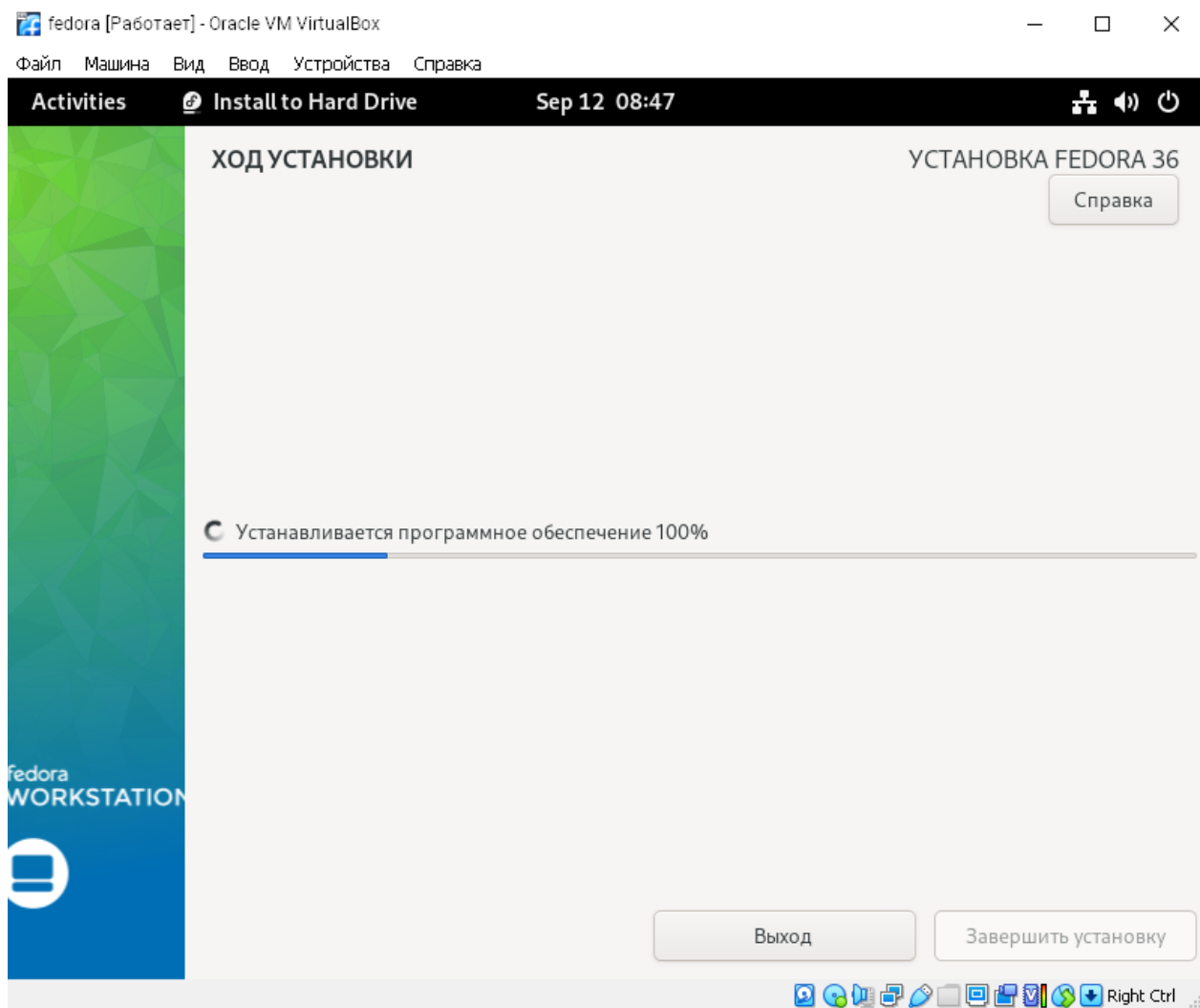


Рис. 5 (процесс установки Fedora)

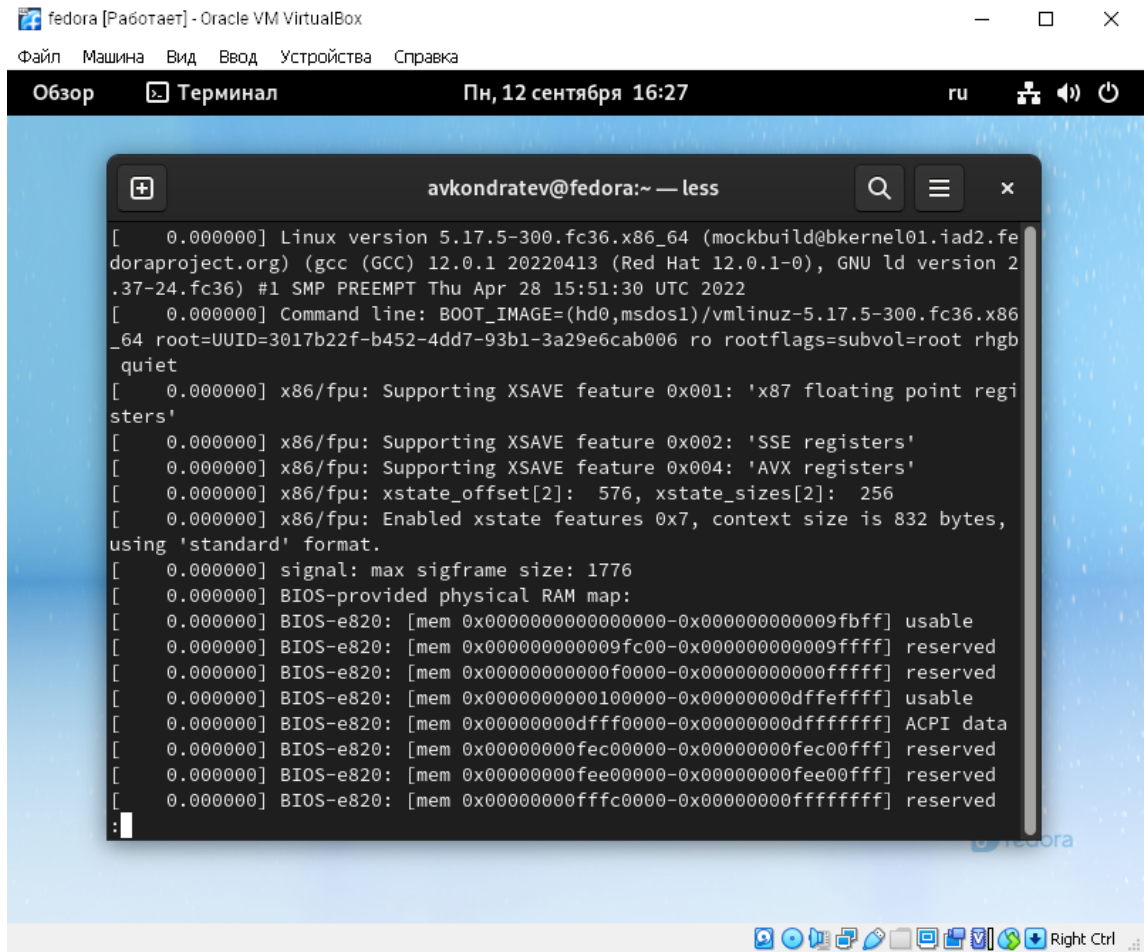
4. Выбираем имя пользователя, логин и пароль

(Arseny V. Kondratev; avkondratev)

Домашнее задание.

Просматриваем вывод команды `"dmesg | less"` или используем поиск с помощью `"dmesg | grep -i "то, что ищем"`

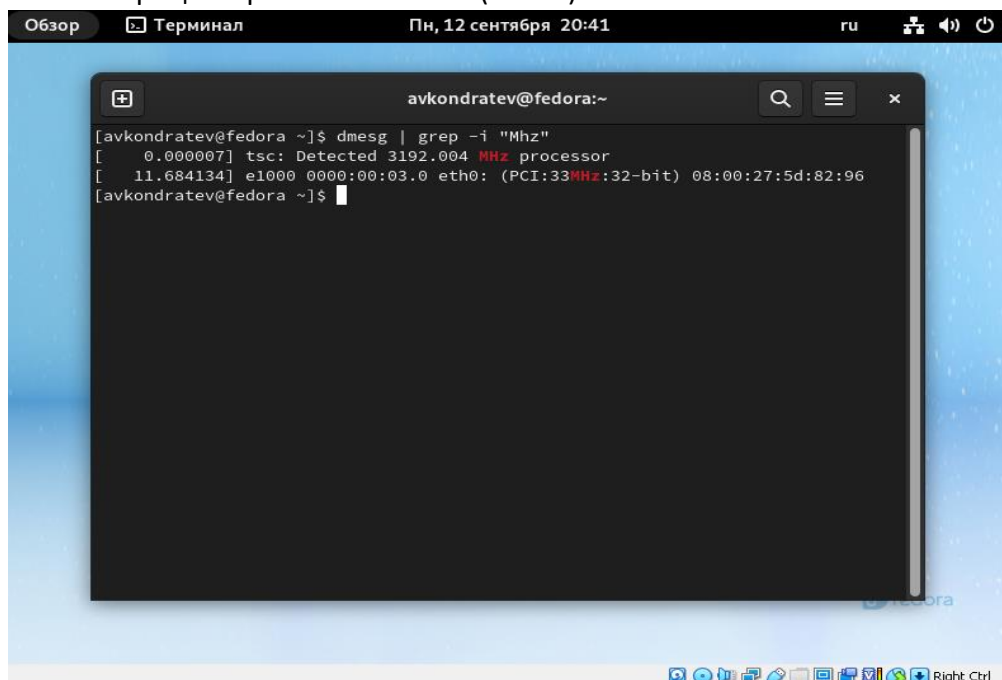
1. Версия ядра 5.17.5-300.fc36.x86_64 (Рис. 6)



```
avkondratev@fedora:~ — less
[ 0.000000] Linux version 5.17.5-300.fc36.x86_64 (mockbuild@bkernel01.iad2.fedoraproject.org) (gcc (GCC) 12.0.1 20220413 (Red Hat 12.0.1-0), GNU ld version 2.37-24.fc36) #1 SMP PREEMPT Thu Apr 28 15:51:30 UTC 2022
[ 0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.17.5-300.fc36.x86_64 root=UUID=3017b22f-b452-4dd7-93b1-3a29e6cab006 ro rootflags=subvol=root rhgb quiet
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
[ 0.000000] x86/fpu: xstate_offset[2]: 576, xstate_sizes[2]: 256
[ 0.000000] x86/fpu: Enabled xstate features 0x7, context size is 832 bytes, using 'standard' format.
[ 0.000000] signal: max sigframe size: 1776
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x0000000000dfffff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000dfff000-0x000000000dffffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee00fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved
:
```

Рис. 6 (результат команды dmesg | less)

2. Частота процессора 3192.004 Mhz (Рис. 7)



```
avkondratev@fedora:~
[avkondratev@fedora ~]$ dmesg | grep -i "Mhz"
[ 0.000007] tsc: Detected 3192.004 MHz processor
[ 11.684134] e1000 0000:00:03:00 eth0: (PCI:33MHz:32-bit) 08:00:27:5d:82:96
[avkondratev@fedora ~]$
```

Рис. 7(ищем частоту процессора)

3. Модель процессора Intel core i5-6500 (рис. 8)

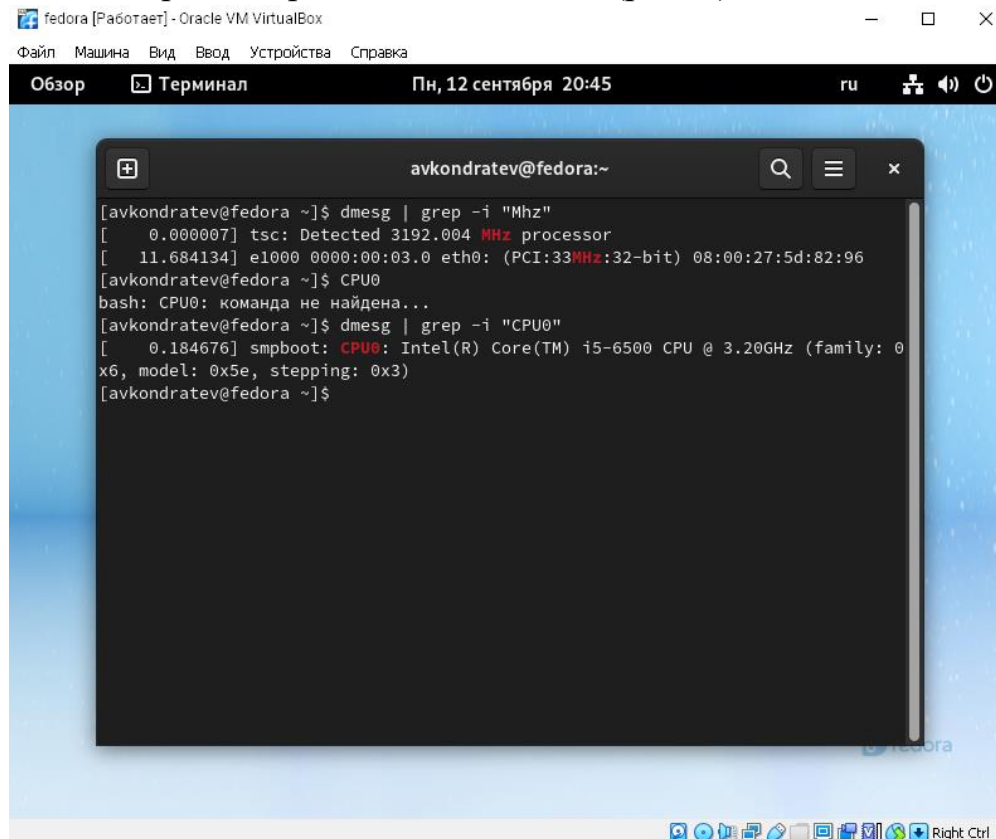


Рис. 8 (ищем модель процессора)

4. Объем доступной оперативной памяти 3986140K (рис. 9)

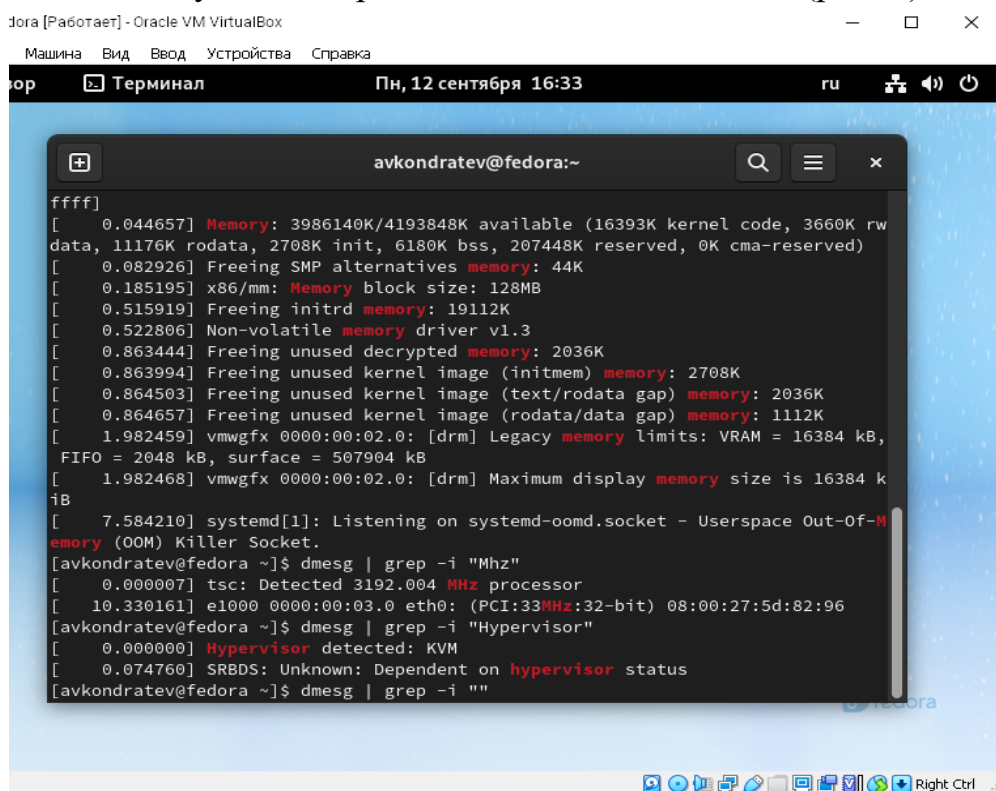
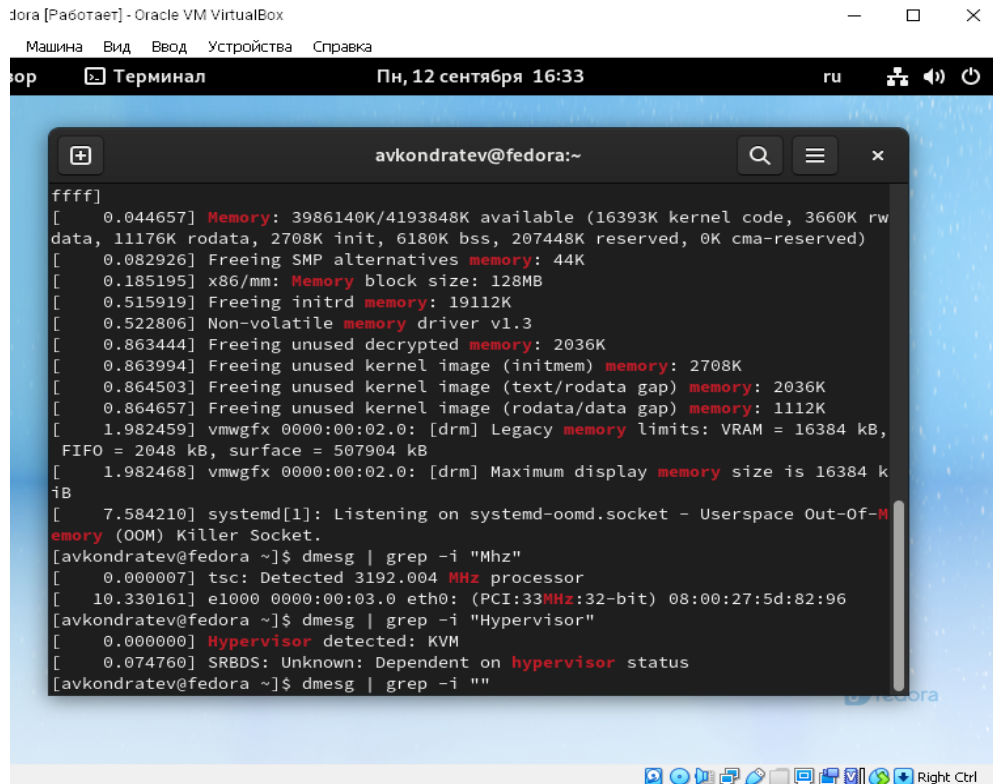


Рис. 9 (Ищем объем доступной оперативной памяти)

5. Тип обнаруженного гипервизора KVM (Рис. 10)

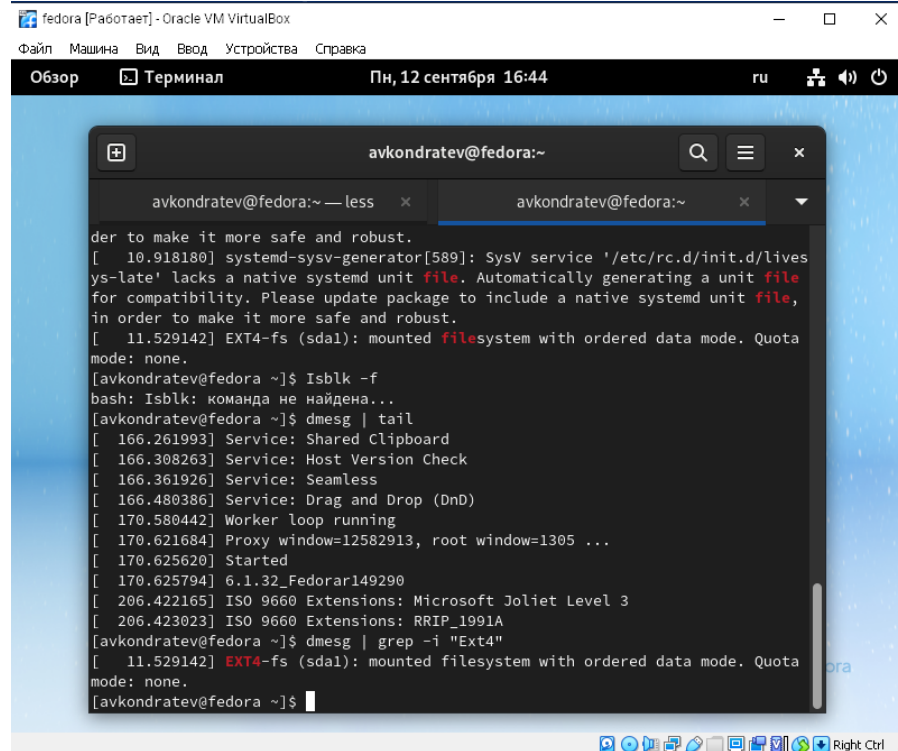


The screenshot shows a terminal window titled "avkondratev@fedora:~" with a search bar and menu icons. The terminal output displays system boot logs, including memory allocation and kernel initialization. Key lines include: "Hypervisor detected: KVM" and "SRBDS: Unknown: Dependent on hypervisor status". The user has executed the command "dmesg | grep -i 'Mhz'" and "dmesg | grep -i 'Hypervisor'".

```
ffff]
[ 0.044657] Memory: 3986140K/4193848K available (16393K kernel code, 3660K rw
data, 11176K rodata, 2708K init, 6180K bss, 207448K reserved, 0K cma-reserved)
[ 0.082926] Freeing SMP alternatives memory: 44K
[ 0.185195] x86/mm: Memory block size: 128MB
[ 0.515919] Freeing initrd memory: 19112K
[ 0.522806] Non-volatile memory driver v1.3
[ 0.863444] Freeing unused decrypted memory: 2036K
[ 0.863994] Freeing unused kernel image (initmem) memory: 2708K
[ 0.864503] Freeing unused kernel image (text/rodata gap) memory: 2036K
[ 0.864657] Freeing unused kernel image (rodata/data gap) memory: 1112K
[ 1.982459] vmwgfx 0000:00:02.0: [drm] Legacy memory limits: VRAM = 16384 kB,
FIFO = 2048 kB, surface = 507904 kB
[ 1.982468] vmwgfx 0000:00:02.0: [drm] Maximum display memory size is 16384 k
iB
[ 7.584210] systemd[1]: Listening on systemd-oomd.socket - Userspace Out-Of-M
emory (OOM) Killer Socket.
[avkondratev@fedora ~]$ dmesg | grep -i "Mhz"
[ 0.000007] tsc: Detected 3192.004 MHz processor
[ 10.330161] e1000 0000:00:03:0 eth0: (PCI:33MHz:32-bit) 08:00:27:5d:82:96
[avkondratev@fedora ~]$ dmesg | grep -i "Hypervisor"
[ 0.000000] Hypervisor detected: KVM
[ 0.074760] SRBDS: Unknown: Dependent on hypervisor status
[avkondratev@fedora ~]$ dmesg | grep -i ""
```

Рис. 10 (ищем тип гипервизора)

6. Тип файловой системы корневого раздела EXT4 (Рис. 11)

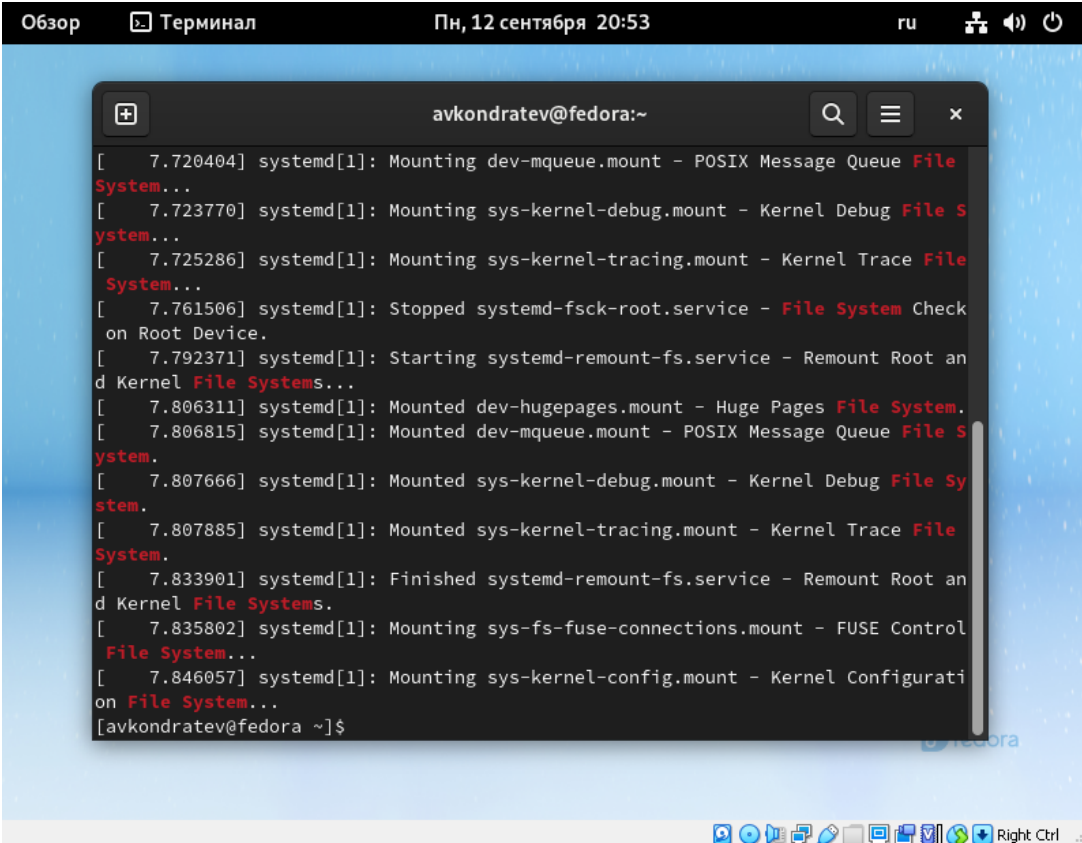


The screenshot shows a terminal window titled "avkondratev@fedora:~" with a search bar and menu icons. The terminal output displays system boot logs, including the detection of the EXT4 filesystem. Key lines include: "EXT4-fs (sda1): mounted filesystem with ordered data mode. Quota mode: none." and "EXT4-fs (sda1): mounted filesystem with ordered data mode. Quota mode: none." The user has executed the command "dmesg | tail" and "dmesg | grep -i 'Ext4'".

```
der to make it more safe and robust.
[ 10.918180] systemd-sysv-generator[589]: SysV service '/etc/rc.d/init.d/lives
ys-late' lacks a native systemd unit file. Automatically generating a unit file
for compatibility. Please update package to include a native systemd unit file,
in order to make it more safe and robust.
[ 11.529142] EXT4-fs (sda1): mounted filesystem with ordered data mode. Quota
mode: none.
[avkondratev@fedora ~]$ lsblk -f
bash: lsblk: команда не найдена...
[avkondratev@fedora ~]$ dmesg | tail
[ 166.261993] Service: Shared Clipboard
[ 166.308263] Service: Host Version Check
[ 166.361926] Service: Seamless
[ 166.480386] Service: Drag and Drop (DnD)
[ 170.580442] Worker loop running
[ 170.621684] Proxy window=12582913, root window=1305 ...
[ 170.625620] Started
[ 170.625794] 6.1.32_Fedorar149290
[ 206.422165] ISO 9660 Extensions: Microsoft Joliet Level 3
[ 206.423023] ISO 9660 Extensions: RRIP_1991A
[avkondratev@fedora ~]$ dmesg | grep -i "Ext4"
[ 11.529142] EXT4-fs (sda1): mounted filesystem with ordered data mode. Quota
mode: none.
[avkondratev@fedora ~]$
```

Рис. 11 (ищем тип файловой системы)

7. Ищем последовательность монтирования файловых систем с помощью команды `dmesg | grep -i "file system"`. (Рис. 12)



The screenshot shows a terminal window titled 'avkondratev@fedora:~' with a search icon, menu icon, and close button. The terminal displays the output of the command `dmesg | grep -i "file system"`. The output shows a sequence of messages from systemd[1] regarding the mounting of various file systems. The messages are as follows:

```
[ 7.720404] systemd[1]: Mounting dev-mqueue.mount - POSIX Message Queue File System...
[ 7.723770] systemd[1]: Mounting sys-kernel-debug.mount - Kernel Debug File System...
[ 7.725286] systemd[1]: Mounting sys-kernel-tracing.mount - Kernel Trace File System...
[ 7.761506] systemd[1]: Stopped systemd-fsck-root.service - File System Check on Root Device.
[ 7.792371] systemd[1]: Starting systemd-remount-fs.service - Remount Root and Kernel File Systems...
[ 7.806311] systemd[1]: Mounted dev-hugepages.mount - Huge Pages File System.
[ 7.806815] systemd[1]: Mounted dev-mqueue.mount - POSIX Message Queue File System.
[ 7.807666] systemd[1]: Mounted sys-kernel-debug.mount - Kernel Debug File System.
[ 7.807885] systemd[1]: Mounted sys-kernel-tracing.mount - Kernel Trace File System.
[ 7.833901] systemd[1]: Finished systemd-remount-fs.service - Remount Root and Kernel File Systems.
[ 7.835802] systemd[1]: Mounting sys-fs-fuse-connections.mount - FUSE Control File System...
[ 7.846057] systemd[1]: Mounting sys-kernel-config.mount - Kernel Configuration File System...
[avkondratev@fedora ~]$
```

The terminal window has a top bar with 'Обзор' (Overview), 'Терминал' (Terminal), and the date/time 'Пн, 12 сентября 20:53'. The bottom bar shows system icons and 'Right Ctrl'.

Рис. 12 (ищем последовательность монтирования файловых систем.)

Вывод: Я приобрел практические навыки установки операционной системы на виртуальную машину и настройки минимально необходимых для дальнейшей работы сервисов.

Контрольные вопросы:

1. Идентификатор пользователя, Идентификатор группы, Полное имя, Домашний каталог, Командная оболочка
2. 1-man
2-cd
3-ls
4-du
5-mkdir(создание), rm(удаление)\
6- chmod
7-history
3. Файловая система – это инструмент, позволяющий операционной системе и программам обращаться к нужным файлам и работать с ними. При этом программы оперируют только названием файла, его размером и датой создания. Все остальные функции по поиску необходимого файла в хранилище и работе с ним берет на себя файловая система накопителя.

FAT32 - Современная версия FAT32 вышла в 1995 году. Она может работать с томами размером до 32 ГБ и файлами размером до 4 ГБ. При этом система не работает с накопителями объемом более 8 Тб

NTFS - Структура системы хранения данных имеет вид бинарного дерева. В отличие от иерархической, как у FAT32, доступ к информации осуществляется по запросу, а поиск ведется по названию файла. При этом система имеет каталог, отсортированный по названиям. Массив делится на 2 части и отсекается та, в которой данного файла не будет, оставшаяся часть также делиться на 2, и так далее до тех пор, пока не будет найден нужный файл.

ReFS - Последняя разработка Microsoft, доступная для серверов Windows 8 и 10. Архитектура файловой системы в основном организована в виде B + -tree. Файловая система ReFS обладает высокой отказоустойчивостью.

Ext2, Ext3, Ext4 или Extended Filesystem– стандартная файловая система, первоначально разработанная еще для Minix. Содержит максимальное количество функций и является наиболее стабильной в связи с редкими изменениями кодовой базы. Начиная с ext3 в системе используется функция журналирования. Сегодня версия ext4 присутствует во всех дистрибутивах Linux.

4. С помощью команды `finfmnt`
5. С помощью команды `kill`