Защита лабораторной работы №1. Установка и конфигурация операционной системы на виртуальную машину

Бармина Ольга Константиновна

2022 Sep 5th

RUDN University, Moscow, Russian Federation

Результат выполнения лабораторной работы №1

Цель выполнения лабораторной работы

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

Результат выполнения лабораторной работы

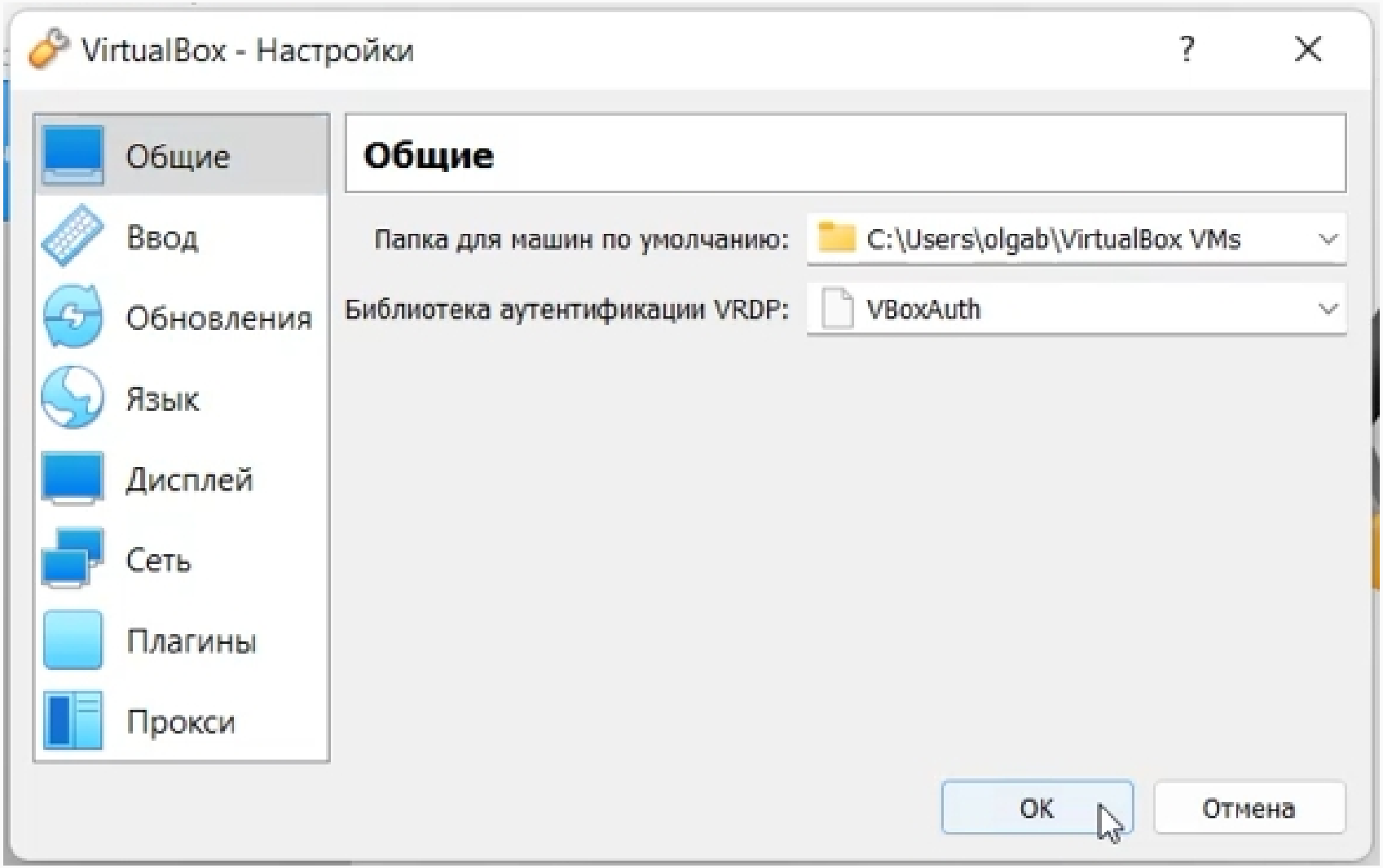
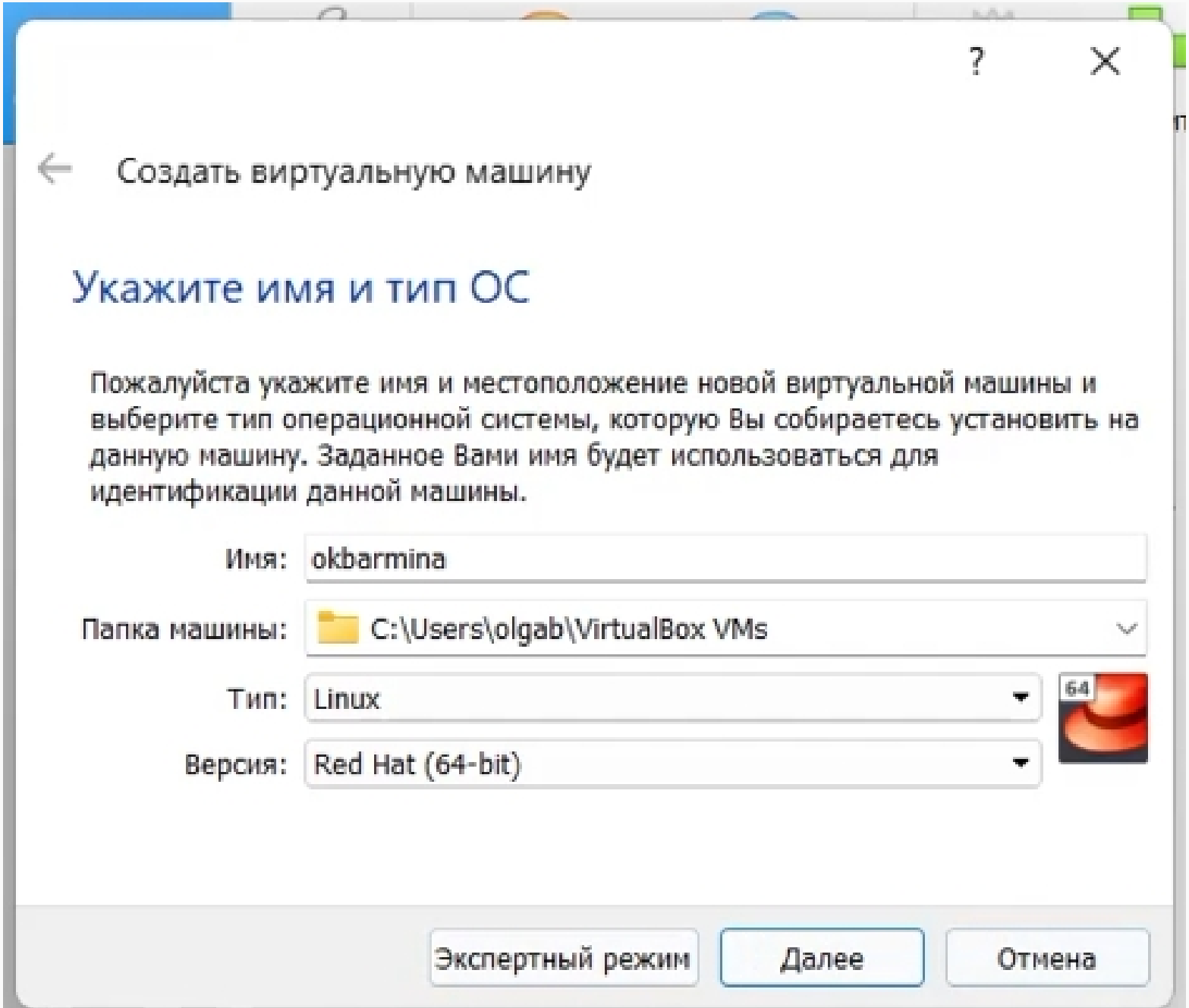


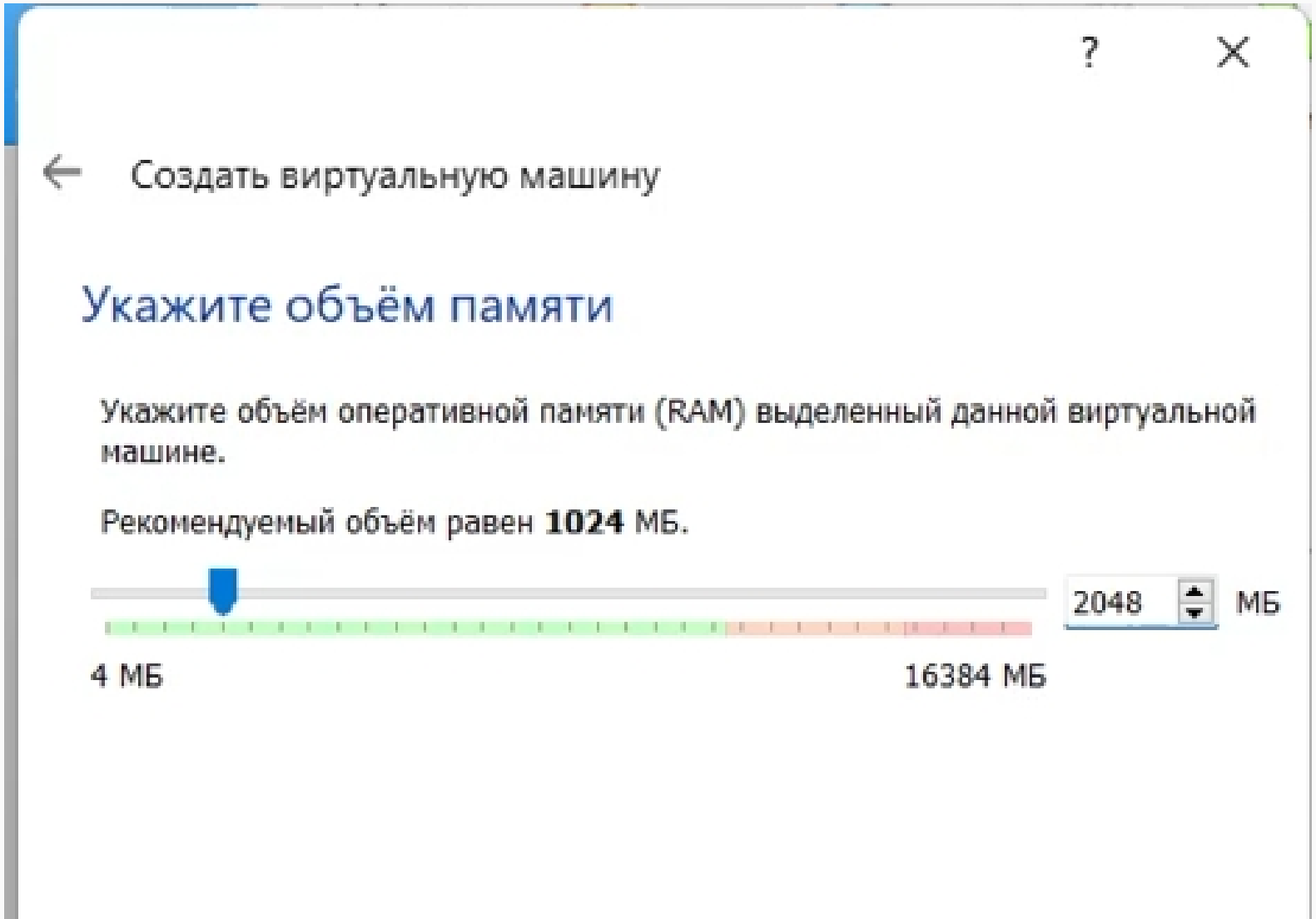
Figure 1: рис 1. Каталог для виртуальных машин

Результат выполнения лабораторной работы

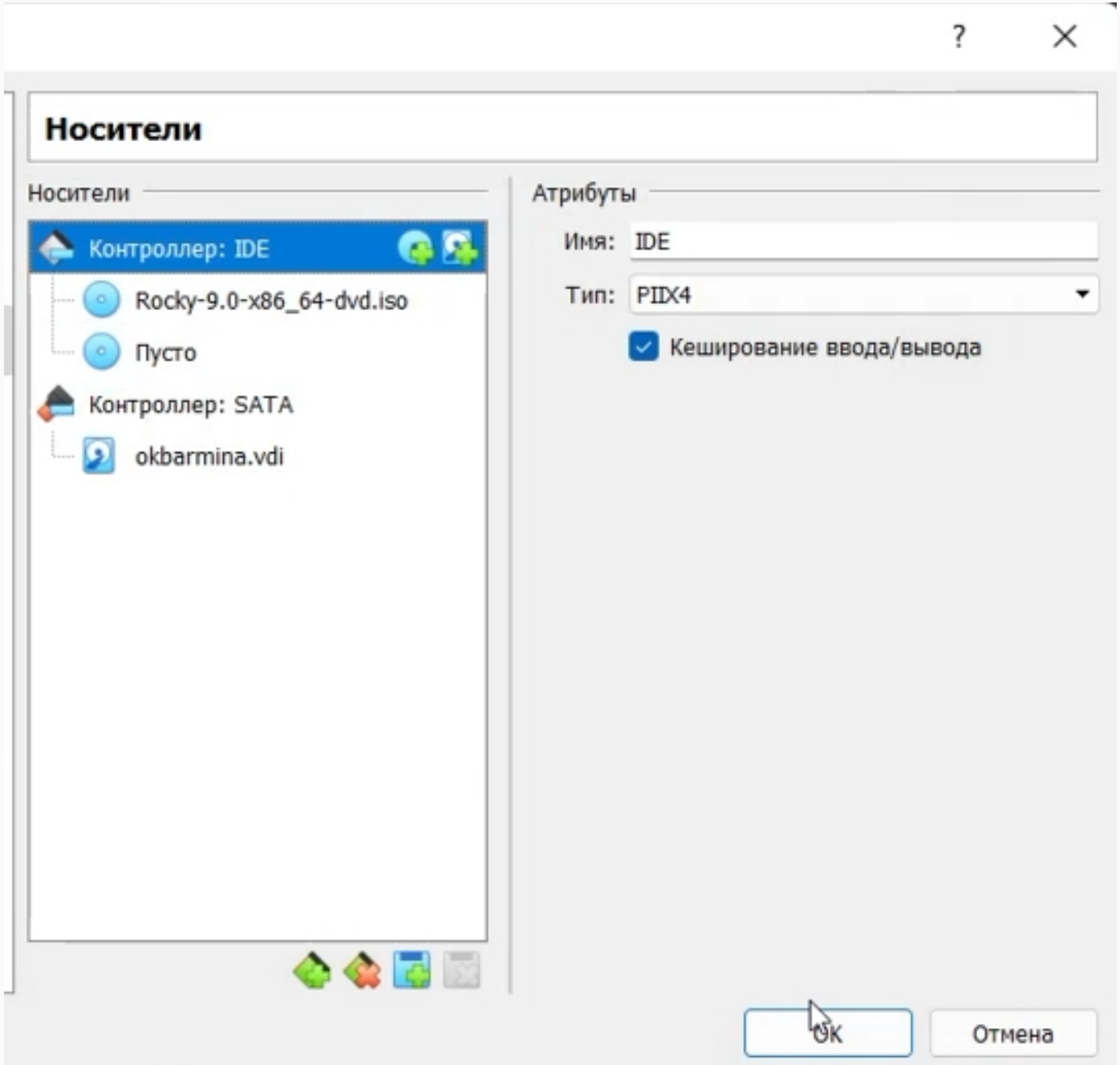


Результат выполнения лабораторной работы

Укажем размер основной памяти виртуальной машины, зададим конфигурацию жесткого диска.

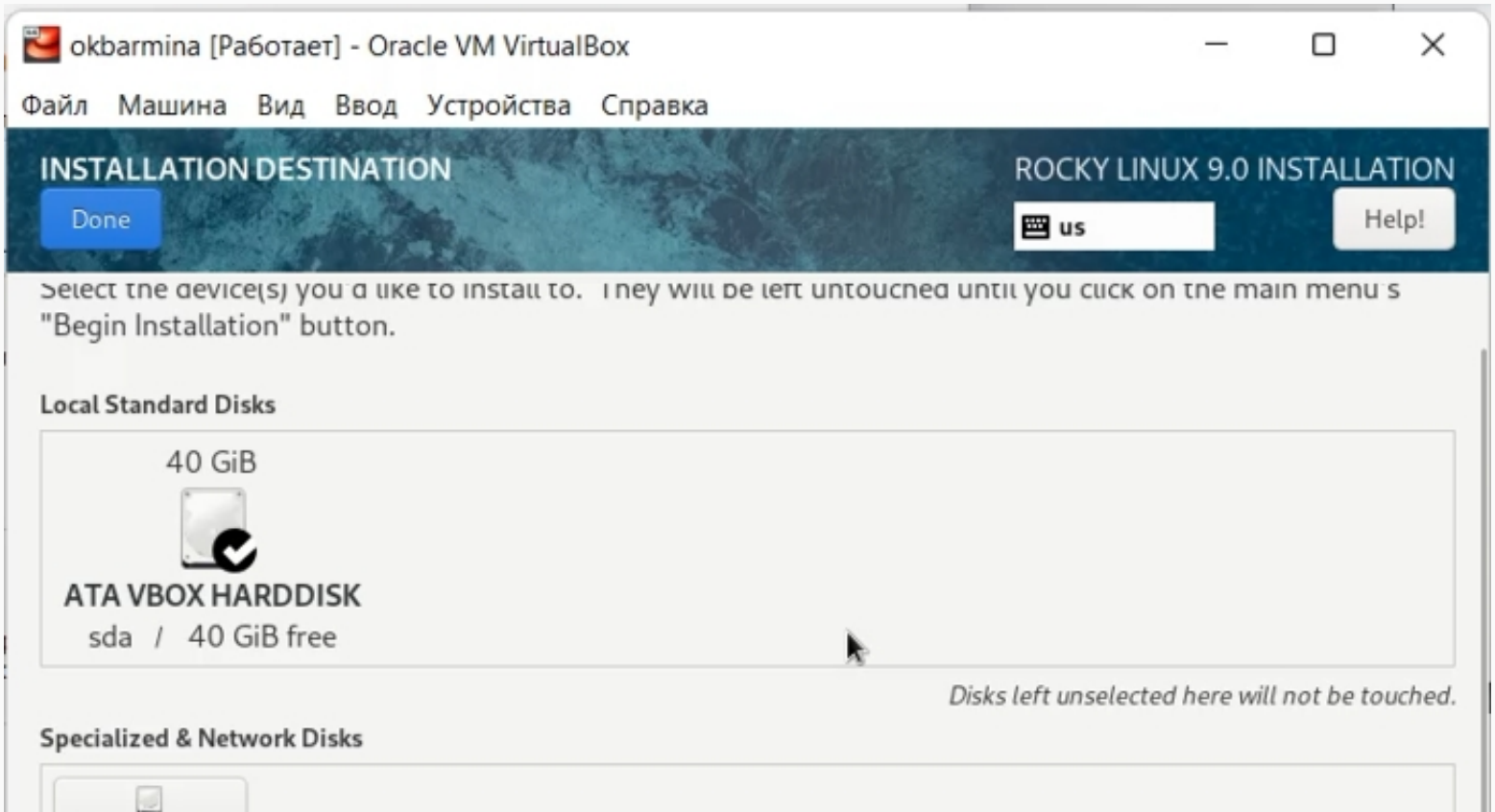


Результат выполнения лабораторной работы



Результат выполнения лабораторной работы

Запустим виртуальную машину. В разделе выбора программ укажем в качестве базового окружения Server with GUI, а в качестве дополнения — Development Tools, отключим KDUMP, место установки ОС оставляем без изменения, установливаем пароль для root и пользователя с правами администратора.



Результат выполнения лабораторной работы

```
[okbarmina@localhost ~]$ dmesg
[    0.000000] Linux version 5.14.0-70.13.1.el9_0.x86_64 (mockbuild@dal1-prod-build
er001.bld.equ.rockylinux.org) (gcc (GCC) 11.2.1 20220127 (Red Hat 11.2.1-9), GNU ld
 version 2.35.2-17.el9) #1 SMP PREEMPT Wed May 25 21:01:57 UTC 2022
[    0.000000] The list of certified hardware and cloud instances for Red Hat Enter
prise Linux 9 can be viewed at the Red Hat Ecosystem Catalog, https://catalog.redha
t.com.
[    0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.14.0-70.13.1.el9_0.x
86_64 root=/dev/mapper/rl-root ro resume=/dev/mapper/rl-swap rd.lvm.lv=rl/root rd.l
vm.lv=rl/swap rhgb quiet
[    0.000000] [Firmware Bug]: TSC doesn't count with P0 frequency!
[    0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registe
rs'
[    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, usi
ng '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 0x0000000000100000-0x000000007ffeffff] usable
[    0.000000] BIOS-e820: [mem 0x000000007fff0000-0x000000007fffffff] 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
```


Figure 6: рис 6. Последовательность загрузки системы

Результат выполнения лабораторной работы

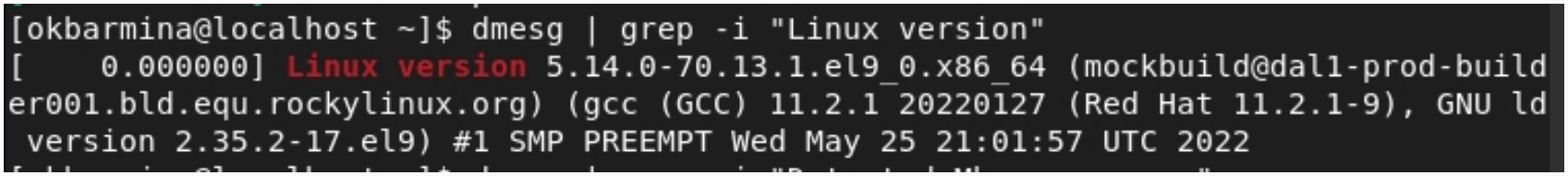


Figure 7: рис 7. Версия ядра

Результат выполнения лабораторной работы

```
[okbarmina@localhost ~]$ dmesg | grep -i "Mhz"
[    0.000009] tsc: Detected 2096.060 MHz processor
[    3.087951] e1000 0000:00:03.0 eth0: (PCI:33MHz:32-bit) 08:00:27:c8:85:e6
```

Figure 8: рис 8. Частота процессора

Результат выполнения лабораторной работы

```
[okbarmina@localhost ~]$ dmesg | grep -i "CPU0"
[    0.048237] CPU0: Hyper-Threading is disabled
[    0.156571] smpboot: CPU0: AMD Ryzen 5 5500U with Radeon Graphics (family: 0x17,
 model: 0x68, stepping: 0x1)
```

Figure 9: рис 9. Модель процессора

Результат выполнения лабораторной работы

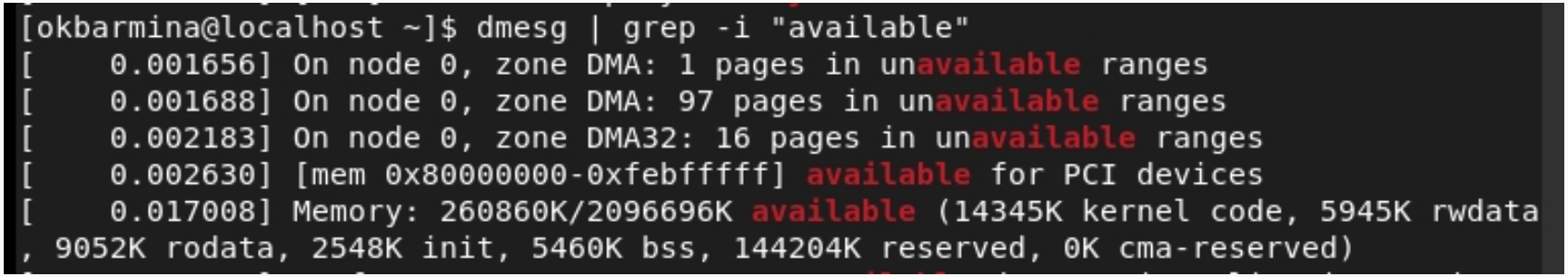


Figure 10: рис 10. Объем доступной ОП

Результат выполнения лабораторной работы

```
[okbarmina@localhost ~]$ dmesg | grep -i "Hypervisor"
[    0.000000] Hypervisor detected: KVM
```

Figure 11: рис 11. Тип гипервизора

Результат выполнения лабораторной работы

```
[okbarmina@localhost ~]$ dmesg | grep -i "file system"
[    1.253444] systemd[1]: Reached target Initrd /usr File System.
[    4.737481] systemd[1]: Set up automount Arbitrary Executable File Formats File
System Automount Point.
[    4.737687] systemd[1]: Stopped target Initrd File Systems.
[    4.737720] systemd[1]: Stopped target Initrd Root File System.
[    4.753994] systemd[1]: Mounting Huge Pages File System...
[    4.765538] systemd[1]: Mounting POSIX Message Queue File System...
[    4.776720] systemd[1]: Mounting Kernel Debug File System...
[    4.781140] systemd[1]: Mounting Kernel Trace File System...
[    4.845498] systemd[1]: Stopped File System Check on Root Device.
[    4.893997] systemd[1]: Starting Remount Root and Kernel File Systems...
```

Figure 12: рис 12. Тип файловой системы

Результат выполнения лабораторной работы

```
[okbarmina@localhost ~]$ dmesg | grep -i "mount"
[    0.047909] Mount-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[    0.047914] Mountpoint-cache hash table entries: 4096 (order: 3, 32768 bytes, li
near)
[    3.590514] XFS (dm-0): Mounting V5 Filesystem
[    3.607053] XFS (dm-0): Ending clean mount
[    4.737481] systemd[1]: Set up automount Arbitrary Executable File Formats File
System Automount Point.
[    4.753994] systemd[1]: Mounting Huge Pages File System...
[    4.765538] systemd[1]: Mounting POSIX Message Queue File System...
[    4.776720] systemd[1]: Mounting Kernel Debug File System...
[    4.781140] systemd[1]: Mounting Kernel Trace File System...
[    4.893997] systemd[1]: Starting Remount Root and Kernel File Systems...
[    7.886105] XFS (sda1): Mounting V5 Filesystem
[    7.913808] XFS (sda1): Ending clean mount
```

Figure 13: рис 13. Последовательность монтирования

Вывод

В ходе работы мы приобрели практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.