

Информационная безопасность. Отчет по лабораторной работе №1

**Установка и конфигурация операционной системы на виртуальную
машину**

Попов Дмитрий Павлович

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	9
4	Список литературы	10

List of Figures

2.1	Последовательность загрузки ОС	6
2.2	Версия ядра Linux	7
2.3	Частота процессора	7
2.4	Модель процессора	7
2.5	Объем доступной оперативной памяти	7
2.6	Тип обнаруженного гипервизора	7
2.7	Тип файловой системы корневого раздела	8
2.8	Последовательность монтирования файловых систем	8

List of Tables

1 Цель работы

Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину. Анализ последовательности загрузки системы через консоль [1].

2 Выполнение лабораторной работы

Домашнее задание
Дождитесь загрузки графического окружения и откройте терминал. В окне терминала проанализируйте последовательность загрузки системы, выполнив команду `dmesg`. Можно просто посмотреть вывод этой команды: `dmesg | less` fig. 2.1 [2].

```
[dpopov@dpopov ~]$ dmesg
[ 0.000000] Linux version 5.14.0-70.13.1.el9_0.x86_64 (mockbuild@dal1-prod-builder001.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 Enterprise Linux 9 can be viewed a
t the Red Hat Ecosystem Catalog, https://catalog.redhat.com.
[ 0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.14.0-70.13.1.el9_0.x86_64 root=/dev/mapper/rl-roo
t ro resume=/dev/mapper/rl-swap rd.lvm.lv=rl/root rd.lvm.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 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 0x0000000001000000-0x000000000bffff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000bfff0000-0x00000000bfffffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using msrs 4b564d01 and 4b564d00
[ 0.000000] kvm-clock: cpu 0, msr dc01001, primary cpu clock
[ 0.000003] kvm-clock: using sched offset of 9796089454 cycles
[ 0.000005] clocksource: kvm-clock: mask: 0xffffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 88159059
1483 ns
[ 0.000009] tsc: Detected 2295.684 MHz processor
[ 0.001296] e820: update [mem 0x00000000-0x0000ffff] usable ==> reserved
[ 0.001307] e820: remove [mem 0x000a0000-0x0000ffff] usable
[ 0.001315] last_pfn = 0xbfff0 max_arch_pfn = 0x400000000
[ 0.001349] Disabled
```

Figure 2.1: Последовательность загрузки ОС

Можно использовать поиск с помощью `grep`: `dmesg | grep -i "то, что ищем"`. По-
лучите следующую информацию. 1. Версия ядра Linux (Linux version) fig. 2.2. 2.

Частота процессора (Detected Mhz processor) fig. 2.3. 3. Модель процессора (CPU0) fig. 2.4. 4. Объем доступной оперативной памяти (Memory available) fig. 2.5. 5. Тип обнаруженного гипервизора (Hypervisor detected) fig. 2.6. 6. Тип файловой системы корневого раздела fig. 2.7. 7. Последовательность монтирования файловых систем fig. 2.8.

```
[dpopov@dpopov ~]$ dmesg | grep -i "Linux version"
[ 0.000000] Linux version 5.14.0-70.13.1.el9_0.x86_64 (mockbuild@dal1-prod-builder001.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
```

Figure 2.2: Версия ядра Linux

```
[dpopov@dpopov ~]$ dmesg | grep -i "Mhz"
[ 0.000009] tsc: Detected 2295.684 MHz processor
[ 2.737554] e1000 0000:00:03:0 eth0: (PCI:33MHz:32-bit) 08:00:27:8d:11:48
```

Figure 2.3: Частота процессора

```
[dpopov@dpopov ~]$ dmesg | grep -i "CPU0"
[ 0.057034] CPU0: Hyper-Threading is disabled
[ 0.155974] smpboot: CPU0: AMD Ryzen 7 3700U with Radeon Vega Mobile Gfx (family: 0x17, model: 0x18, stepping
: 0x1)
```

Figure 2.4: Модель процессора

```
[dpopov@dpopov ~]$ dmesg | grep -i "Memory"
[ 0.001675] ACPI: Reserving FACP table memory at [mem 0xbfff00f0-0xbfff01e3]
[ 0.001676] ACPI: Reserving DSDT table memory at [mem 0xbfff0470-0xbfff2794]
[ 0.001677] ACPI: Reserving FACS table memory at [mem 0xbfff0200-0xbfff023f]
[ 0.001678] ACPI: Reserving FACS table memory at [mem 0xbfff0200-0xbfff023f]
[ 0.001679] ACPI: Reserving APIC table memory at [mem 0xbfff0240-0xbfff0293]
[ 0.001680] ACPI: Reserving SSDT table memory at [mem 0xbfff02a0-0xbfff046b]
[ 0.002205] Early memory node ranges
[ 0.004501] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.004504] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009ffff]
[ 0.004505] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000effff]
[ 0.004506] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[ 0.019895] Memory: 215736K/3145272K available (14345K kernel code, 5945K rwdata, 9052K rodata, 2548K init, 5
460K bss, 163664K reserved, 0K cma-reserved)
[ 0.061573] Freeing SMP alternatives memory: 36K
[ 0.170860] x86/mm: Memory block size: 128MB
[ 0.288365] Non-volatile memory driver v1.3
[ 0.911033] Freeing initrd memory: 53964K
[ 1.058298] Freeing unused decrypted memory: 2036K
[ 1.058798] Freeing unused kernel image (initmem) memory: 2548K
[ 1.061285] Freeing unused kernel image (text/rodata gap) memory: 2036K
[ 1.061539] Freeing unused kernel image (rodata/data gap) memory: 1188K
[ 2.110894] [TTM] Zone kernel: Available graphics memory: 1521708 KiB
[ 2.111002] [drm] Max dedicated hypervisor surface memory is 507904 KiB
[ 2.111002] [drm] Maximum display memory size is 16384 KiB
```

Figure 2.5: Объем доступной оперативной памяти

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

Figure 2.6: Тип обнаруженного гипервизора

```
[dpopov@dpopov ~]$ dmesg | grep -i "file.*system"
[ 1.190260] systemd[1]: Reached target Initrd /usr File System.
[ 3.201335] XFS (dm-0): Mounting V5 Filesystem
[ 4.119918] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 4.120133] systemd[1]: Stopped target Initrd File Systems.
[ 4.120168] systemd[1]: Stopped target Initrd Root File System.
[ 4.129609] systemd[1]: Mounting Huge Pages File System...
[ 4.132331] systemd[1]: Mounting POSIX Message Queue File System...
[ 4.143367] systemd[1]: Mounting Kernel Debug File System...
[ 4.151111] systemd[1]: Mounting Kernel Trace File System...
[ 4.193082] systemd[1]: Stopped File System Check on Root Device.
[ 4.238910] systemd[1]: Starting Remount Root and Kernel File Systems...
[ 6.467661] XFS (sda1): Mounting V5 Filesystem
```

Figure 2.7: Тип файловой системы корневого раздела

```
[dpopov@dpopov ~]$ dmesg | grep -i "mount"
[ 0.055915] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 0.055927] Mountpoint-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
[ 3.201335] XFS (dm-0): Mounting V5 Filesystem
[ 3.217823] XFS (dm-0): Ending clean mount
[ 4.119918] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 4.129609] systemd[1]: Mounting Huge Pages File System...
[ 4.132331] systemd[1]: Mounting POSIX Message Queue File System...
[ 4.143367] systemd[1]: Mounting Kernel Debug File System...
[ 4.151111] systemd[1]: Mounting Kernel Trace File System...
[ 4.238910] systemd[1]: Starting Remount Root and Kernel File Systems...
[ 6.467661] XFS (sda1): Mounting V5 Filesystem
[ 6.702807] XFS (sda1): Ending clean mount
```

Figure 2.8: Последовательность монтирования файловых систем

3 Выводы

Приобретены практические навыки установки операционной системы на виртуальную машину, навыки анализа последовательности загрузки системы через консоль.

4 Список литературы

1. Методические материалы курса
2. Задание к лабораторной работе № 1