

Управление загрузкой системы

Майоров Дмитрий Андреевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	10

Список иллюстраций

2.1	6
2.2	6
2.3	6
2.4	7
2.5	7
2.6	7
2.7	8
2.8	8
2.9	8
2.10	9
2.11	9
2.12	9

Список таблиц

1 Цель работы

Получить навыки работы с загрузчиком системы GRUB2

2 Выполнение лабораторной работы

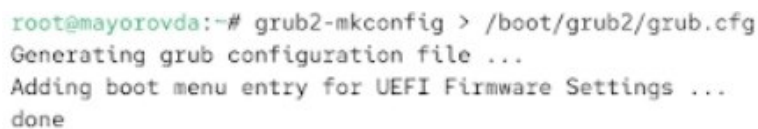
Получаем полномочия администратора. Открываем файл `/etc/default/grub` для редактирования. И устанавливаем параметр `GRUB_TIMEOUT=10`



```
GNU nano 8.1 /etc/default/grub Modified
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR="$(sed 's, release .*$,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="resume=UUID=442631fb-88ba-415e-bf2a-8ce2b74a7c01 rd."
GRUB_DISABLE_RECOVERY="true"
GRUB_ENABLE_BLSCFG=true
```

Рисунок 2.1

Записываем изменения в GRUB2, введя в командной строке следующую команду. Перезагружаем систему и убеждаемся, что при загрузке мы видим прокрутку загрузочных сообщений



```
root@mayorovda:~# grub2-mkconfig > /boot/grub2/grub.cfg
Generating grub configuration file ...
Adding boot menu entry for UEFI Firmware Settings ...
done
```

Рисунок 2.2



```
done
root@mayorovda:~# reboot
```

Рисунок 2.3

```
Stopping systemd-udevd.service - Rule-based Manager for Device Events and Files...
OK | Finished initrd-cleanup.service - Cleaning Up and Shutting Down Daemons.
OK | Stopped systemd-udevd.service - Rule-based Manager for Device Events and Files.
OK | Closed systemd-udevd-control.socket - udev Control Socket.
OK | Closed systemd-udevd-kernel.socket - udev Kernel Socket.
OK | Stopped dracut-pre-udev.service - dracut pre-udev hook.
OK | Stopped dracut-cmdline.service - dracut cmdline hook.
Starting initrd-udevadm-cleanup-db.service - Cleanup udev Database...
OK | Stopped systemd-tmpfiles-setup-dev.service - Create Static Device Nodes in /dev.
OK | Stopped systemd-sysusers.service - Create System Users.
OK | Stopped systemd-tmpfiles-setup-dev-early.service - Create Static Device Nodes in /dev.
OK | Stopped kmod-static-nodes.service - Create List of Static Device Nodes.
OK | Finished initrd-udevadm-cleanup-db.service - Cleanup udev Database.
OK | Reached target initrd-switch-root.target - Switch Root.
OK | Finished plymouth-switch-root.service - Plymouth switch root service.
Starting initrd-switch-root.service - Switch Root...
```

Рисунок 2.4

При перезагрузке системы, при появлении меню GRUB выбираем строку с текущей версией ядра и нажимаем клавишу E для редактирования. В конце строки, загружающей ядро, вводим `systemd.unit=rescue.target`. И удаляем опции `rhgb` и `quit`

```
load_video
set gfxpayload=keep
insmod gzio
linux ($root)/vmlinuz-6.12.0-124.29.1.el10_1.x86_64 root=/dev/mapper/r1_may\
prowda-root ro resume=UUID=442631fb-88ba-415e-bf2a-8ce2b74a7c01 rd.lvm.lv=r\
l_mayorovda/root rd.lvm.lv=r1_mayorovda/swap crashkernel=2G-64G:256M,64G-:5\
12M systemd.unit=rescue.target_
initrd ($root)/initramfs-6.12.0-124.29.1.el10_1.x86_64.img $tuned_initrd
```

Рисунок 2.5

Смотрим список всех файлов модулей, которые загружены в настоящее время. Загружена базовая системная среда

```
root@mayorovda:~# systemctl list-units_
```

Рисунок 2.6

Смотрим задействованные переменные среды оболочки. Перезагружаем систему

```

root@mayorovda:~# systemctl show-environment
LANG=en_US.UTF-8
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
XDG_DATA_DIRS=/var/lib/flatpak/exports/share:/usr/local/share:/usr/share/
root@mayorovda:~# systemctl reboot

```

Рисунок 2.7

При перезагрузке системы, при появлении меню GRUB выбираем строку с текущей версией ядра и нажимаем клавишу E для редактирования. В конце строки, загружающей ядро, вводим `systemd.unit=emergency.target`. И удаляем опции `rhgb` и `quit`

```

load_video
set gfxpayload=keep
insmod gzio
linux ($root)/vmlinuz-6.12.0-124.29.1.el10_1.x86_64 root=/dev/mapper/r1_may\
orovda-root ro resume=UUID=442631fb-88ba-415e-bf2a-8ce2b74a7c01 rd.lvm.lv=r\
l_mayorovda/root rd.lvm.lv=r1_mayorovda/swap crashkernel=2G-64G:256M,64G-5\
12M systemd.unit=emergency.target
initrd ($root)/initramfs-6.12.0-124.29.1.el10_1.x86_64.img $tuned_initrd

```

Рисунок 2.8

Смотрим список всех файлов модулей, которые загружены в настоящее время. Видим, что количество загружаемых файлов модулей уменьшилось до минимума

```

(or press Control-D to continue):
root@mayorovda:~# systemctl list-units_

```

Рисунок 2.9

Снова запускаем виртуальную машину. При появлении меню GRUB выбираем строку с текущей версией ядра и нажимаем клавишу E для редактирования. В конце строки, загружающей ядро, вводим `rd.break` И удаляем опции `rhgb` и `quit`


```
load_video
set gfxpayload=keep
insmod gzio
linux ($root)/vmlinuz-6.12.0-124.29.1.el10_1.x86_64 root=/dev/mapper/r1_may\
orovda-root ro resume=UUID=442631fb-88ba-415e-bf2a-8ce2b74a7c01 rd.lvm.lv=r\
l_mayorovda/root rd.lvm.lv=r1_mayorovda/swap crashkernel=2G-64G:256M,64G-:5\
12M rd.break
initrd ($root)/initramfs-6.12.0-124.29.1.el10_1.x86_64.img $tuned_initrd
```

Рисунок 2.10

Вводим команду `mount -o remount,rw /sysroot` для получения доступа к системному образу для чтения и записи. Делаем содержимое каталога `/sysimage` новым корневым каталогом. Устанавливаем новый пароль

```
sh-5.2# mount -o remount,rw /sysroot
sh-5.2# chroot /sysroot
sh-5.2# passwd
New password:
Retype new password:
passwd: password updated successfully
```

Рисунок 2.11

Загружаем политику SELinux, чтобы убедиться, что тип контекста установлен правильно

```
sh-5.2# load_policy -i
[ 137.651182] audit: type=1404 aud
[ 137.747763] SELinux: policy cap
```

Рисунок 2.12

Устанавливаем правильный тип контекста для `/etc/shadow`. Перезагружаем систему

3 Выводы

Получены навыки работы с загрузчиком системы GRUB2