

Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)

Факультет «Информатика и системы управления»
Кафедра «Компьютерные системы и сети»

В.Ю. Мельников

Исследование процесса загрузки ОС Linux

Электронное учебное издание

Методические указания по выполнению лабораторных работ
по дисциплине "Операционные системы"

2022

Введение

Цель работы - исследование процесса загрузки Linux. Освоение работы с интерпретатором командной строки «bash».

Продолжительность работы - 1 час.

Задание

Согласно данному пособию:

- Ознакомиться с этапами загрузки компьютера
- Выполнить загрузку без ввода пароля и смену пароля суперпользователя
- Научиться пользоваться интерпретатором командной строки «bash»

НЕ включать в отчёт:

- Не надо приводить команды, выполняемые в процессе освоения «bash»

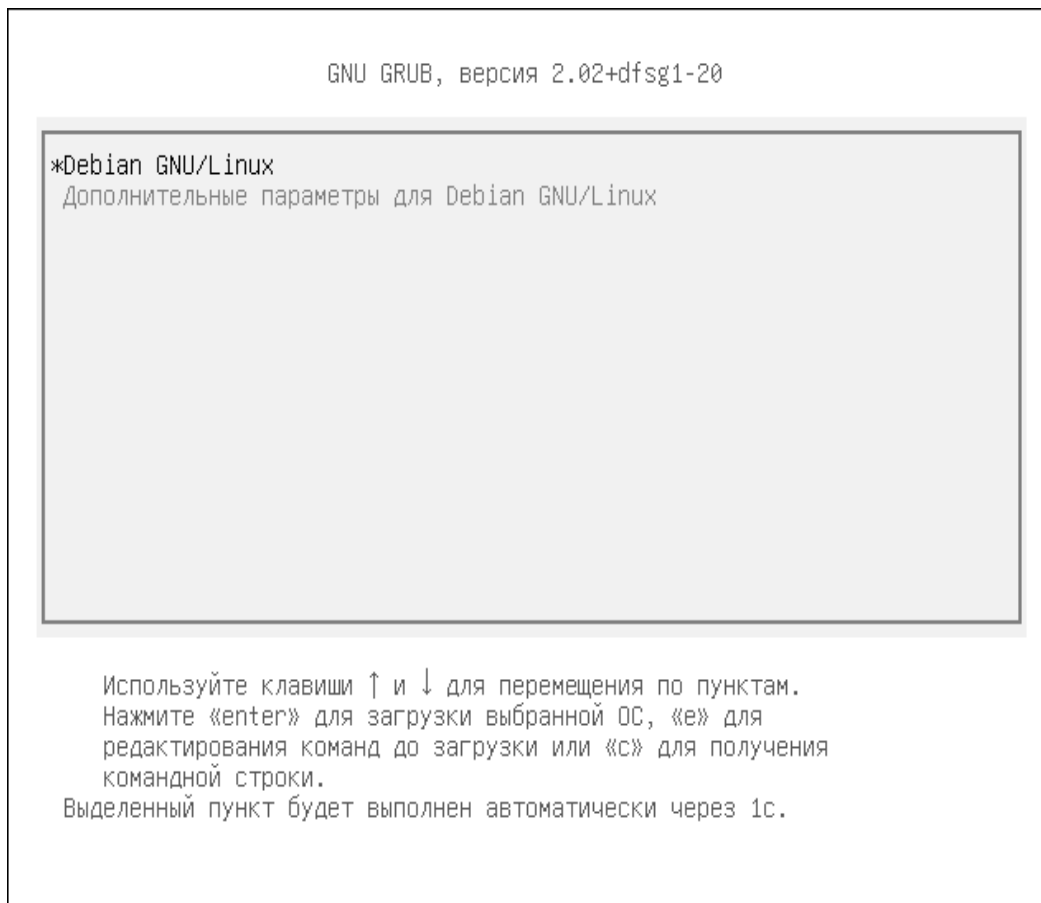
Этапы загрузки компьютера

BIOS

Сначала из ПЗУ копируется в оперативную память BIOS — базовая система ввода вывода. BIOS проверяет исправность процессора, памяти и видеоадаптера. На экран выводится информация об этом процессе.

Первичный загрузчик

Затем, с заданного в BIOS физического жёсткого диска из 0-сектора считывается Master Boot Record (MBR). В MBR хранится таблица разделов, на которые разбит жесткий диск и первичный загрузчик. Помните, при установке Debian установил первичный загрузчик GRUB в MBR.



Если бы не был установлен GRUB, а более простой первичный загрузчик, он нашёл бы в таблице разделов раздел, помеченный как активный, считал из него 0-сектор и передал управление на вторичный загрузчик.

GRUB устроен сложнее. Первичный загрузчик GRUB считывает несколько секторов, лежащих сразу за MBR — полуторный загрузчик.

Полуторный загрузчик GRUB уже умеет работать с файловой системой. Он читает вторичный загрузчик из определённого файла и передаёт управление ему.

GRUB при установке или настройке подготовил меню выбора операционных систем, установленных в разные разделы. При загрузке он выводит это меню и предлагает пользователю выбрать операционную систему, которую надо загрузить.

По умолчанию будет выбрана заданная в настройках ОС и через несколько секунд она будет автоматически загружена. Для ускорения процесса можно нажать «Enter»

Вторичный загрузчик

Загружает из каталога «/boot» файловой системы образ ядра и образ «initrd» (о нём позже). Затем передаёт управление ядру операционной системы.

После изменения параметров в конфигурационных файлах ядра, необходимо дать команду, которая обновит образ ядра.

Ядро Linux

Содержит основные функции, необходимые большинству программ. В том числе функции запуска программ, обеспечение многозадачности.

При загрузке ядро монтирует «initrd» (Initial RAM Disk). Это диск в оперативной памяти, на котором расположена временная файловая система, используемая ядром Linux при начальной загрузке. На диске «initrd» расположен минимальный набор утилит и драйверы используемых файловых систем.

Далее ядро linux монтирует разделы реального диска, демонтирует «initrd» и запускает программу /sbin/init

Менеджер управления службами init (systemd)

/sbin/init это первая программа, запускаемая при загрузке операционной системы. Она в свою очередь запускает в несколько этапов службы (сервисы) и переходит в режим ожидания. Она снова активируется при завершении работы компьютера и останавливает службы.

В современных версиях Linux устанавливается менеджер управления службами «systemd». В этом случае, «/sbin/init» представляет собой ссылку на программу «systemd». Эта система ускоряет загрузку системы за счёт параллельного запуска сервисов. Порядок запуска сервисов и команды для их запуска заданы в текстовых файлах описания сервисов, которые расположены в каталоге «/etc/systemd/system».

К сожалению, инсталлятор нашего стартового набора установил старый менеджер управления службами «System V init». Сервисы запускаются сценариями на языке «bash», расположенными в каталоге /etc/init.d. Псевдокомментарии в этих сценариях определяют порядок загрузки этих сервисов.

После запуска сервисов менеджер управления службами запускает программу «login», которая запрашивает имя пользователя и пароль а затем запускает интерпретатор команд «bash».

Когда потеряли пароль суперпользователя

Время от времени случается, что пользователь возвращается из отпуска и не может вспомнить свой пароль. Пароль не хранится в системе даже в зашифрованном виде. Хранится только хеш-код пароля по которому восстановить пароль невозможно. Если забыл пароль простой пользователь, надо войти в систему от имени суперпользователя и задать пользователю новый пароль. Хуже когда забыли пароль суперпользователя. К счастью в Debian, если есть физический доступ к компьютеру, можно сменить пароль

суперпользователя, даже если старый пароль утерян.

- Перегрузите компьютер
- Когда появится меню GRUB нажмите клавишу «е»

GNU GRUB, версия 2.02+dfsg1-20

```
setparams 'Debian GNU/Linux'

    load_video
    insmod gzio
    if [ x$grub_platform = xxen ]; then insmod xzio; insmod lzopio; \
fi
    insmod part_msdos
    insmod ext2
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-ieee1275='ieee1\
275//disk@0,msdos1' --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-\
baremetal=ahci0,msdos1 60ff9034-a6c3-4a4a-8a5f-9ef3218b4654
    else
        search --no-floppy --fs-uuid --set=root 60ff9034-a6c3-4a4a-8a5\ ↓
```

Поддерживается несколько Emacs-подобных команд редактирования на экране. Есть списки дополнений по TAB. Нажмите Ctrl-x или F10 для загрузки, Ctrl-c или F2 для получения командной строки или ESC для отмены изменений и возврата в меню GRUB.

- клавишами «↓» «↑» опустите курсор вниз текста и продолжаем опускаться дальше. При этом прокручивается текст сценария загрузки. Ближе к концу текста, конце строки, начинающийся с "linux /boot/vmlinuz-...", поставьте пробел и допишите «init=/bin/bash». Строка длинная, чтобы переместить курсор в конец строки используйте клавишу «End». Теперь, вместо запуска служб linux будет запущен интерпретатор командной строки (bash).
- Нажмите ctrl+x

```
/dev/sda1: clean, 31340/589824 files, 242744/2359040 blocks
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
root@(none):/# _
```

Ну вот, мы вошли в linux без пароля. Можем давать команды!

- Но надо сначала перемонтировать файловую систему с правами на запись.

```
mount -rw -o remount /
```

Теперь можно сменить пароль.

- Дайте команду «passwd» и введите новый пароль (2 раза)

```
root@(none):/# mount -rw -o remount /
root@(none):/# passwd root
New password:
Retype new password:
passwd: password updated successfully
root@(none):/# _
```

- Осталось перезагрузиться. Для этого дайте из меню «Машина» виртуальной машины команду «Перезапустить».

Запуск программ, командная оболочка «bash»

Когда linux загрузился запускается программа «login» которая запрашивает имя пользователя и его пароль. Когда вы ввели правильный пароль, «login» запускает интерпретатор команд (или командную оболочку) «**bash**». Именно этот процесс обеспечивает ввод команды и её выполнение. Рассмотрим подробнее его возможности.

Нажмите клавишу «↑» на клавиатуре. Высвечивается предыдущая набранная команда. Повторные нажатия «↑» высвечивают предыдущие команды. Всегда можно повторить введенную ранее команду, при необходимости отредактировав её. Историю команд можно получить командой «history»

```
root@debian:~# history 5
549 free
550 cat /proc/meminfo
551 pstree
552 pstree -h
553 history 5
```

История набранных вами команд хранится в файле истории в открытом виде поэтому, даже если команда позволяет ввести пароль в командной строке, не стоит пользоваться этой возможностью, а использовать форму, которая запрашивает пароль в процессе выполнения команды.

Команду не обязательно набирать полностью. Наберите «hi» и нажмите клавишу «Tab». Интерпретатор команд допишет за вас команду «history». А вот «h», для авто дополнения недостаточно, поскольку несколько команд начинается на эту букву. Нажмите клавишу «Tab» дважды. Высветится список команд, начинающихся на введенную строку:

```
root@debian:~# h
h2ph      hash      help      history   hostname
h2xs      hd        helpztags host       hostnamectl
halt      head      hexdump   hostid     hwclock
root@debian:~# h
```

Даже, если вам не трудно ввести команду целиком, лучше пользоваться авто дополнением, во избежание ошибок ввода. Поможет оно и если вы забыли, как точно пишется команда. Имена команд представляют собой либо английские слова, либо аббревиатуры, так что первые пару букв можно угадать. Особенно авто дополнение полезно

при вводе в команде имени файла.

А ещё, интерпретатор команд «bash» выполняет командные сценарии. Когда администратору надоедает выполнять ежедневно одни и те же команды, он создаёт файл, пишет в него сценарий, и впредь выполняет эти действия одной командой. Сценарий для выполнения несложной задачи написать проще, чем программу на языке «С» и не надо устанавливать пакеты компилятора и сборщика. Язык сценариев имеет все конструкции языка программирования (циклы, условия, переменные):

```
#!/bin/sh
#этот сценарий ничего не делает. Просто фрагменты кода
s1='строка 1'
s2='строка 2'
if [[ "$s1" == "$s2" ]]
then
    echo 'условие выполняется'
else
    echo 'условие не выполняется'
fi

i=0
while [ $i -lt 5 ]
do
    echo "$i"
    i=$(( $i+1 ))
done
```

Но отсутствуют возможности обращения к системным функциям, да и синтаксис неудобный. Так, что если не удалось написать сценарий за 15 минут, бросайте и пишите программу на Вашем любимом языке программирования.

Но вернёмся к выполнению команды. Мы ввели команду и нажали клавишу «ENTER». Интерпретатор команд разбирает введённый текст. Первое слово (до пробела) это имя программы. Может быть задан путь к программе (например, «/usr/bin/ps») или только имя программы, (например, «ps»).

Программа это файл, помеченный признаком «выполняемый». Если команда начинается с «/», значит задан полный путь к программе, начиная с корневого каталога. Интерпретатор команд запускает соответствующую программу.

Если путь к программе не указан, интерпретатор команд ищет программу сначала в текущем каталоге, а затем, в каталогах, перечисленных в переменной окружения «PATH». Пути разделяются символом «:».

```
root@debian:~# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
```

Это был суперпользователь. А вот, пути для поиска программ пользователя «user»

```
user@debian:~$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
```

Обратите внимание, на отсутствие путей «/sbin», «/usr/sbin» и «/usr/local/sbin». В этих каталогах лежат инструменты системного администратора, часть из них обычный пользователь не имеет прав запускать, но многие команды и обычный пользователь может использовать для получения информации, надо только указать полный путь к команде.

При необходимости, в эту переменную можно добавить нужные Вам пути. Например, команда «`export PATH=$PATH:/home/user/my_programs`» добавляет путь «/home/user/my_programs» к имеющимся на время работы данного интерпретатора команд. Если надо добавить этот путь постоянно, добавьте эту команду в конец файла «`.bashrc`». Мы ещё заглянем в этот файл в работе «Исследование методов защиты»

После имени программы, через пробел следуют параметры, которые уточняют, что должна сделать команда. Например, «`cat /proc/meminfo`» выводит на экран содержимое файла «/proc/meminfo».

Необязательные параметры называют опциями. В команде «`free -h`» опция «-h» говорит команде, что надо выводить объём памяти в килобайтах и мегабайтах. Без этого параметра команда «`free`» выводит объём памяти в байтах.

Параметров может быть несколько, и разделяются они пробелами. Если параметр (например, имя файла) содержит пробел, параметр следует заключить в апострофы или кавычки. Например, команда «`mkdir 'Операционные системы'`» создаст каталог «Операционные системы». Если апострофы не поставить, интерпретатор команд передаст программе 2 параметра и «`mkdir`» создаст 2 каталога «Операционные» и «системы».

Если имя файла содержит служебные символы, «`& | > < ; $ ()`» следует использовать апострофы, в этом случае служебные символы рассматриваются, как обычные. И наоборот, если вам надо подставить в команду значение переменной окружения, следует использовать кавычки. Сравните:

```
root@debian:~# echo "*** $USER ***"
*** root ***
root@debian:~# echo '*** $USER ***'
*** $USER ***
```

Справка по командам

Параметры у каждой программы свои. Можно получить краткую справку по большинству программ, указав параметр «`--help`». Например, команда «`free --help`» выдаёт справку по команде «`free`». Часто даже краткая справка не умещается на экране. Можно просмотреть её постранично, добавив «`|more`». Например, «`cat --help|more`». Для перелистывания страниц используется клавиша «пробел», для выхода клавишу «q».

Более подробную информацию по параметрам команд можно найти в Интернете или используя команду «man КОМАНДА» (от слова «manual» - руководство). Например, «man cat».

Для листания страниц используйте «PgUp», «PgDown».

Для поиска, наберите «/СТРОКА» и нажмите клавишу «Enter». Для поиска следующего места заданной строки в тексте нажмите «n» (next)

Для выхода, как в большинстве интерактивных приложений, используйте клавишу «q» (quit).

После возврата в «bash» хочется очистить экран от результатов предыдущей команды. Для этого нажмите «Ctrl»+«L».

Загрузка файлов из интернета

Бывает, нужно на сервере без графической подсистемы скачать файл из интернета.

Как это сделать без браузера?

```
apt-get install wget  
cd /home  
wget АДРЕС
```

Скачайте из интернета какуюнибудь картинку, для установки в качестве фона рабочего стола на следующем занятии. Она будет дожидаться Вас в каталоге «/root».

Контрольные вопросы

- Откуда загружается BIOS?
- Откуда у нас загружается первичный загрузчик?
- Что хранится в initrd?
- Что такое bash?
- Для чего нужен bash?
- Как повторить предыдущую команду?
- Как выполнить автодополнение команды?
- Как выполнить автодополнение имени файла?
- Что означает «/» в начале имени файла
- Где bash ищет исполняемый файл, соответствующий команде?
- Как задать в команде файл с именем, содержащим пробелы?
- Как задать в команде файл с именем, содержащим «\$»?

Литература

Вывод команды «man»

<https://www.altlinux.org>

Терминал Linux. Горячие сочетания клавиш для Bash (<https://linuxrussia.com/keyboard-shortcuts-for-bash.html>)