

Впервые опубликовано на [developerWorks](#) 04.04.2006

Учебник для экзамена LPI 102, Тема 106: Загрузка, инициализация, остановка и уровни выполнения

Администрирование Linux для начинающих (LPIC-1), тема 106

Загрузка системы

Обзор загрузки

Вкратце, процесс загрузки персональных компьютеров заключается в следующем:

При включении компьютера BIOS (Basic Input/Output System - базовая система ввода-вывода) проводит самотестирование.

1. Когда компьютер проходит самотестирование, BIOS загружает главную загрузочную запись (или MBR, обычно из первого 512-байтового сектора загрузочного устройства). Обычно это первый жесткий диск системы, но также может быть и дискета, CD-диск или USB-ключ.
2. Для жесткого диска MBR загружает первичный загрузчик, которым обычно является загрузчик LILO или загрузчик GRUB для системы Linux. Это другая односекторная запись размером 512 байт.
3. Первичный загрузчик обычно загружает последовательность записей, называемую вторичным загрузчиком (или иногда "полуторным" ("stage 1.5") загрузчиком).
4. Вторичный загрузчик загружает операционную систему. Для Linux это ядро и, возможно, начальный RAM-диск (initrd).

К этому моменту ваша система должна быть готова к установке одного из двух популярных загрузчиков: LILO (Linux LOader) или GRUB (GRand Unified Boot loader). Вы должны быть способны использовать выбранный вами загрузчик для обычной загрузки, описанной выше. Отсылаем вас к учебнику "учебник для экзамена LPI 101 (тема 102): установка Linux и управление пакетами", если вам нужно просмотреть установку загрузчика или основную загрузку.

Для воздействия на процесс загрузки вашей системы, вы можете:

1. Изменять устройство, с которого вы загружаетесь. Обычно вы загружаетесь с жесткого диска, но иногда вам может потребоваться загрузиться с дискеты, ключа памяти USB, CD или DVD привода или через сеть. Установка таких альтернативных загрузочных устройств требует, чтобы ваша BIOS была соответственно сконфигурирована. Способ, как сделать это, зависит от вашей системы и ее BIOS. Это выходит за рамки этого учебника или требований этого объекта LPI, так что обращайтесь к вашей системной документации.

2. Вы можете взаимодействовать с загрузчиком, выбирая, какую из нескольких возможных конфигураций вы хотите загрузить. В этом учебнике вы научитесь делать это для загрузчиков LILO и GRUB.
3. Вы можете использовать GRUB или LILO, как только загрузчик его загрузит, для передачи параметров ядру, чтобы управлять тем, как ядро запускает систему.

LILO

По умолчанию конфигурационный файл LILO находится в `/etc/lilo.conf`. Листинг 1 демонстрирует пример из системы, на которой работает Red Hat Enterprise Linux 3. Система имеет раздел Red Hat 9 с корневой файловой системой, монтированной на `/mnt/hda7`, и Windows® XP на `/dev/hda1`.

Листинг 1. Пример LILO конфигурации

```
[root@lyrebird root]# cat /etc/lilo.conf
prompt
timeout=50
compact
default=latest-EL
boot=/dev/fd0
map=/boot/map
install=/boot/boot.b
message=/boot/message2
lba32
password=mypassword
restricted
image=/mnt/hda7/boot/vmlinuz-2.4.20-31.9
    label=redhat9
    alias=shrike
    initrd=/mnt/hda7/boot/initrd-2.4.20-31.9.img
    read-only
    append="hdd=ide-scsi root=LABEL=RH9"
image=/boot/vmlinuz-2.4.21-40.EL
    label=2.4.21-40.EL
    alias=latest-EL
    initrd=/boot/initrd-2.4.21-40.EL.img
    read-only
    append="hdd=ide-scsi root=LABEL=RHEL3"
image=/boot/vmlinuz-2.4.21-37.0.1.EL
    label=2.4.21-37a.EL
```

```

        initrd=/boot/initrd-2.4.21-37.0.1.EL.img
        read-only
        append="hdd=ide-scsi root=LABEL=RHEL3"
image=/boot/vmlinuz-2.4.21-37.EL
        label=2.4.21-37.EL
        initrd=/boot/initrd-2.4.21-37.EL.img
        read-only
        append="hdd=ide-scsi root=LABEL=RHEL3"
image=/boot/vmlinuz-2.4.21-32.0.1.EL
        label=2.4.21-32.EL
        alias=early
        initrd=/boot/initrd-2.4.21-32.0.1.EL.img
        read-only
        append="hdd=ide-scsi root=LABEL=RHEL3"
other=/dev/hda1
        loader=/boot/chain.b
        label=WIN-XP
        alias=xp

```

Помните, что каждый раз, когда вы вносите изменения в `/etc/lilo.conf` или устанавливаете новое ядро, вы должны запустить `lilo`. Программа `lilo` переписывает MBR или загрузочную запись раздела, чтобы отразить внесенные вами изменения, включая занесение абсолютного адреса ядра на диске. Если ваш конфигурационный файл включает образы Linux из нескольких разделов, вы должны смонтировать разделы, потому что команда `lilo` требует доступа к разделу, чтобы найти образ.

Параметр `message` в конфигурационном файле LILO может ссылаться на текстовый файл или на специально созданный файл в формате PCX. Дистрибутив Red Hat Enterprise Linux включает графический файл `/boot/message`, который содержит заставку Red Hat. С иллюстративной целью конфигурация из Листинга 1 использует текстовый файл, представленный в Листинге 2. Изменение графических сообщений LILO выходит за пределы этого учебника. Имейте в виду, что это также не очень хорошо задокументировано.

Листинг 2. Текстовое загрузочное сообщение LILO

```

[root@lyrebird root]# cat /boot/message2
Booting lyrebird

```

Если ваш файл конфигурации не включает параметра сообщения, то вы увидите очень простую строку приглашения: `LILO boot:`. В противном случае вы увидите или текстовое сообщение или графический фон и меню. Возможно, вам потребуется

удерживать клавишу Shift в процессе загрузки, чтобы видеть приглашение, так как система может быть сконфигурирована таким образом, чтобы пропускать его.

Если вы видите пользовательскую текстовую строку приглашения или строку по умолчанию, вы можете нажать клавишу Tab, чтобы отобразить список доступных образов для загрузки. Вы можете или ввести имя образа, как показано в Листинге 3, или нажать Enter чтобы выбрать первый вариант. Если вам доступно графическое меню, используйте клавиши управления курсором, чтобы выделить вариант, который вы хотите загрузить.

Листинг 3. Пример строки приглашения LILO

```
LILO
Booting lyrebird
boot:
latest-EL      shrike      redhat9      2.4.21-40.EL
2.4.21-37a.EL  2.4.21-37.EL  early      2.4.21-32.EL
xp             WIN-XP
boot: latest-EL
```

Помимо отображения конфигурационного файла LILO, можно указать ключ -q команды lilo для отображения информации о вариантах загрузки LILO. Добавьте ключ -v для более содержательного вывода. Два примера, использующих конфигурационный файл из Листинга 1, приведены в Листинге 4.

Листинг 4. Отображение конфигурации LILO

```
[root@lyrebird root]# lilo -q
latest-EL      *
shrike
redhat9
2.4.21-40.EL
2.4.21-37a.EL
2.4.21-37.EL
early
2.4.21-32.EL
xp
WIN-XP
[root@lyrebird root]# lilo -q -v | tail +22 | head -n 9
shrike
    Password is required for specifying options
    Boot command-line won't be locked
    No single-key activation
```

```
VGA mode is taken from boot image
Kernel is loaded "high", at 0x00100000
Initial RAM disk is 149789 bytes
No fallback

Options: "ro BOOT_FILE=/mnt/hda7/boot/vmlinuz-2.4.20-31.9 hdd=ide-
scsi root=LABEL=RH9"
```

GRUB

Файл конфигурации GRUB по умолчанию находится в `/boot/grub/grub.conf` или `/boot/grub/menu.lst`. Если присутствуют оба файла, то один обычно является символьной ссылкой на другой. Листинг 5 содержит пример из той же системы, которую вы рассматривали для LILO, хотя здесь иллюстрируется только несколько вариантов.

Листинг 5. Пример конфигурации GRUB

```
default=1
timeout=10
splashimage=(hd0,2)/boot/grub/fig1x.xpm.gz
foreground=23334c
background=82a6bc
password --md5 $1$H8LlM1$cI0Lfs5.C06xFJYPQ8Ixz/
title Red Hat Linux (2.4.20-31.9)
    root (hd0,6)
    kernel /boot/vmlinuz-2.4.20-31.9 ro root=LABEL=RH9 hdd=ide-scsi
    initrd /boot/initrd-2.4.20-31.9.img
    savedefault
    boot
title Red Hat Enterprise Linux WS A (2.4.21-40.EL)
    root (hd0,10)
    kernel /boot/vmlinuz-2.4.21-40.EL ro root=LABEL=RHEL3 hdd=ide-
scsi
    initrd /boot/initrd-2.4.21-40.EL.img
title Win/XP
    rootnoverify (hd0,0)
    chainloader +1
```

GRUB предоставляет интерфейс меню вместо строки приглашения LILO. Он также может использовать пароль, зашифрованный алгоритмом MD5, в противоположность простому текстовому паролю LILO. И, что, может быть, наиболее

важно, изменения, сделанные в конфигурационном файле GRUB, не требуют переустановки GRUB в MBR. Заметьте, что многие дистрибутивы автоматически обновляют конфигурационный файл GRUB (или LILO) при переходе на новую версию ядра, но если вы устанавливаете новое ядро самостоятельно или создаете новый начальный RAM-диск, вам, возможно, потребуется отредактировать конфигурационный файл.

GRUB также не требует монтирования раздела, чтобы сконфигурировать для него загрузочный образ. Вы заметите строки типа `root (hd0,6)` и `splashimage=(hd0,2)/boot/grub/fig1x.xpm.gz`. GRUB обращается к жестким дискам как `hdn`, где `n` - целое число от 0 и выше. Подобным образом нумеруются, начиная с нуля, и разделы на диске.

Итак, в этой системе `(hd0,2)` представляет основной раздел `/dev/hda3`, а `(hd0,6)` - логический раздел `/dev/hda7`. Флоппи-дисковод - это обычно `(fd0)`. Не забывайте брать это в кавычки, если вы вызываете GRUB с параметрами из оболочки `bash`, например, при установке GRUB на флоппи-диск или MBR.

Если вам хочется изменить фоновое изображение для GRUB, то вы ограничены 14 цветами. Ваше любимое JPEG-изображение может выглядеть несколько иначе, когда будет сведено к 14 цветам. Вы можете видеть результат на Рисунке 1, где показано изображение из конфигурационного файла выше, использующее фото, сделанное мной в Гласьер Бэй на Аляске. Вам может также захотеться выбрать подходящие основной и фоновый цвета для вашей текстовой командной строки из цветов на изображении; Рисунок 2 иллюстрирует измененный цвет текста и фона.

Рисунок 1. Фото, сведенное к 14 цветам, для фонового изображения GRUB



Рисунок 2. Цвет текста и его фона, выбранные из цветов фотографии

Red Hat Enterprise Linux WS A (2.4.21-40.EL)

Когда отображается меню GRUB, вы выбираете загрузочный образ, используя клавиши управления курсором для перемещения по списку вверх и вниз.

В отличие от LILO, GRUB ведет себя как маленькая оболочка с несколькими

командами, которые позволяют вам делать такие вещи, как редактирование команд перед их выполнением, или нахождение и загрузка конфигурационного файла, или отображение файлов с использованием команды `cat`. Находясь в меню, вы можете нажать `e` на строке, чтобы изменить ее, `c` - чтобы переключиться на командную строку GRUB, `b` - чтобы загрузить систему, `p` - чтобы ввести пароль и `Esc` - чтобы вернуться в меню или на предыдущий шаг. Существует также команда `grub`, которая создает эмуляцию оболочки, в которой вы можете протестировать вашу конфигурацию GRUB или ваши навыки использования команд GRUB. Внутри оболочки GRUB команда `help` выводит список команд. Использование `help commandname` предоставляет справку по команде с именем `commandname`. Листинг 6 иллюстрирует вызов справки и доступные команды.

Листинг 6. Использование оболочки GRUB

```
[root@lyrebird root]# grub
Probing devices to guess BIOS drives. This may take a long time.
find FILENAME                                geometry DRIVE [CYLINDER HEAD
SECTOR [
halt [--no-apm]                               help [--all] [PATTERN ...]
hide PARTITION                               initrd FILE [ARG ...]
kernel [--no-mem-option] [--type=TYPE] makeactive
map TO_DRIVE FROM_DRIVE                     md5crypt
module FILE [ARG ...]                       modulenounzip FILE [ARG ...]
pager [FLAG]                                partnew PART TYPE START LEN
parttype PART TYPE                           quit
reboot                                       root [DEVICE [HDBIAS]]
rootnoverify [DEVICE [HDBIAS]]               serial [--unit=UNIT] [--port=PORT]
[--
setkey [TO_KEY FROM_KEY]                    setup [--prefix=DIR] [--
stage2=STAGE2_
terminal [--dumb] [--no-echo] [--no-ed terminfo [--name=NAME --cursor-
address
testvbe MODE                                unhide PARTITION
uppermem KBYTES                             vbeprobe [MODE]
grub> help rootnoverify
rootnoverify: rootnoverify [DEVICE [HDBIAS]]
    Similar to `root`, but don't attempt to mount the partition. This
    is useful for when an OS is outside of the area of the disk that
    GRUB can read, but setting the correct root device is still
    desired. Note that the items mentioned in `root` which derived
    from attempting the mount will NOT work correctly.

grub>
```

В качестве практического упражнения вы можете продолжить предыдущий пример и использовать команду GRUB find для поиска файлов конфигурации. Далее вы можете загрузить файл конфигурации из (hd0,2), то есть из /dev/hda3, как показано в Листинге 7.

Листинг 7. Использование GRUB для нахождения и загрузки конфигурационного файла GRUB

```
grub> find /boot/grub/menu.lst
(hd0,2)
(hd0,6)
(hd0,7)
(hd0,8)
(hd0,9)
(hd0,10)
grub> configfile (hd0,2)/boot/grub/menu.lst
```

Когда вы загружаете конфигурационный файл, вы можете увидеть меню, подобное тому, которое показано в Листинге 8. Помните, что это было выполнено из оболочки GRUB, которая эмулирует реальное окружение GRUB и не отображает фоновое изображение. Однако, это по сути дела то, что вы увидите наложенным на ваше фоновое изображение, когда действительно будете загружать систему, используя GRUB.

Листинг 8. Меню GRUB

```
GRUB version 0.93 (640K lower / 3072K upper memory)
+-----+
| Red Hat Linux (2.4.20-31.9) |
| Red Hat Linux (2.4.20-6) |
| Red Hat Enterprise Linux WS A (2.4.21-40.EL) |
| Red Hat Enterprise Linux WS A (2.4.21-37.0.1.EL) |
| Red Hat Enterprise Linux WS A (2.4.21-37.EL) |
| Red Hat Enterprise Linux WS A (2.4.21-32.0.1.EL) |
| Red Hat Enterprise Linux WS A (2.4.21-27.0.4.EL) |
| Red Hat Enterprise Linux WS A (2.4.21-27.0.2.EL) |
| Red Hat Enterprise Linux WS A (2.4.21-27.0.1.EL) |
| Red Hat Enterprise Linux WS A (2.4.21-20.EL) |
| Red Hat Enterprise Linux WS (2.4.21-27.0.1.EL) |
| Red Hat Enterprise Linux WS (2.4.21-15.0.2.EL) |
+-----+
```


Use the ^ and v keys to select which entry is highlighted.

Press enter to boot the selected OS or 'p' to enter a password to unlock the next set of features.

The highlighted entry will be booted automatically in 5 seconds.

Предположим, вы выделили третью строчку, то есть Red Hat Enterprise Linux WS A (2.4.21-40.EL), и нажали e, чтобы редактировать ее. Вы увидите что-то подобное Листингу 9.

Листинг 9. Редактирование строки конфигурации GRUB

```
GRUB version 0.93 (640K lower / 3072K upper memory)
```

```
+-----+
| root (hd0,10)                                     |
| kernel /boot/vmlinuz-2.4.21-40.EL ro root=LABEL=RHEL3 hdd=ide-scsi |
| initrd /boot/initrd-2.4.21-40.EL.img             |
|                                                    |
|                                                    |
|                                                    |
+-----+
```

Use the ^ and v keys to select which entry is highlighted.

Press 'b' to boot, 'e' to edit the selected command in the boot sequence, 'c' for a command-line, 'o' to open a new line after ('O' for before) the selected line, 'd' to remove the selected line, or escape to go back to the main menu.

Вы снова с помощью клавиш управления курсором выбираете строку для изменения и затем нажимаете e чтобы ее редактировать. Например, если вы удалили раздел /dev/hda5, то вашим корневым разделом станет /dev/hda10 или (hd0,9) вместо /dev/hda11 или (hd0,10). Сохраните и измените значение. После этого нажмите Enter, чтобы подтвердить изменения, или Esc для отмены. Наконец, нажмите b, чтобы загрузить систему.

GRUB имеет достаточно возможностей, чтобы отображать файлы вашей файловой системы, и он запускается с правами суперпользователя, так что вам действительно нужно защищать вашу систему паролем GRUB. Однако, помните, что если пользователь может загрузиться с переносного носителя, то он может использовать свою собственную конфигурацию GRUB. Смотрите часть, связанную с безопасностью, в руководстве GRUB (см. Ресурсы) для большей информации о безопасности и других аспектах GRUB. Вы также можете просмотреть эту информацию на вашей системе, используя команду info grub.

Параметры ядра

Параметры ядра (иногда называемые параметрами загрузки) снабжают ядро информацией о параметрах оборудования, которые оно не может определить самостоятельно, чтобы переназначить значения, которые оно может в противном случае неверно определить, или избежать задания неподходящих значений. Например, вы можете захотеть загрузить симметричную многопроцессорную систему (SMP) в однопроцессорном режиме или указать альтернативную корневую файловую систему. Некоторые версии ядер требуют параметр для разрешения поддержки больших объемов памяти на системах с количеством памяти выше определенного.

Если вы используете LILO, вы указываете дополнительные (или перекрывающие) параметры после того, как вводите имя загружаемого ядра. Например, если вы только что добавили новое ядро с названием `/boot/vmlinuz-2.4.21-40.EL-prep`, вам следует ввести команду, чтобы сказать LILO использовать его, как показано в Листинге 10.

Листинг 10. Задание параметров загрузки с помощью LILO

```
boot: latest-EL image=/boot/vmlinuz-2.4.21-40.EL-prep
```

Используя GRUB, вы можете вводить другой набор команд для ядра и операторы `initrd` или, что предпочтительней, использовать средства редактирования, которые вы только что изучили, чтобы изменить существующую строку, добавив `-prep` после имени существующего образа ядра.

Когда ядро заканчивает загрузку, оно обычно запускает `/sbin/init`. Эта программа продолжает работать, пока система не будет остановлена. Ей обычно присвоен идентификатор процесса (PID) 1, как это видно из Листинга 11.

Листинг 11. Процесс init

```
[root@lyrebird root]# ps --pid 1
  PID TTY          TIME CMD
    1 ?            00:00:04 init
```

Программа `init` загружает оставшуюся часть вашей системы, запуская серии скриптов. Эти скрипты обычно находятся в `/etc/rc.d/init.d` или `/etc/init.d`, они реализуют такие службы, как установка имени хоста системы, проверка файловой системы на ошибки, монтирование дополнительных файловых систем, включение сети, запуск служб печати, и так далее. После выполнения скриптов `init` запускает программу, называемую `getty`, которая выводит приглашение для ввода логина на консоль. Графические экраны входа в систему обрабатываются по-другому, как вы изучали в учебнике "Учебник для экзамена LPI 102, тема 105: Ядро."

Если ваша система загрузит ядро, но не сможет успешно запустить `init`, вы можете попробовать восстановить систему, указав альтернативную программу

инициализации. Например, указание `init=/bin/sh` загрузит вашу систему в оболочку с правами суперпользователя, из которой вы получите возможность восстановить систему.

Вы можете узнать больше про доступные параметры загрузки, используя страницы руководства `man` для `bootparam` или просматривая файл `/usr/src/linux/Documentation/ramdisk.txt`, который может быть назван `/usr/src/linux-$(uname -r)/Documentation/kernel-parameters.txt` на некоторых системах.

Не приходится и говорить, что, если каждый раз при загрузке вы вынуждены использовать один и тот же набор дополнительных параметров, то вам следует добавить их в конфигурационный файл. Не забудьте при этом перезапустить `lilo`, если вы используете `LILO`.

События загрузки

Во время загрузки Linux, на консоль выводится большое количество сообщений, описывающих загружающееся ядро, аппаратные средства вашей системы и другие вещи, связанные с ядром. Эти сообщения обычно быстро проскакивают, и вы, вероятно, не сможете прочитать их, несмотря на то что при загрузке есть задержки, когда процесс загрузки ожидает чего-то, например, при невозможности найти сервер времени или при необходимости проверки файловой системы. С появлением проекта Linux Bootsplash (см. Ресурсы) эти сообщения могут накладываться на графический фон, или они могут быть скрыты и заменены простой строкой состояния. Если ваш дистрибутив поддерживает скрытый режим, у вас обычно есть возможность переключиться обратно на отображение загрузочных сообщений, нажав какую-нибудь клавишу, например, `F2`.

dmesg

Приятно иметь возможность вернуться и просмотреть сообщения ядра. Поскольку стандартный вывод связан с процессом, а ядро не имеет идентификатора процесса, оно помещает сообщения ядра (и модулей) в буфер кольца ядра. Вы можете отобразить буфер кольца ядра, используя команду `dmesg`, отображающую эти сообщения в стандартный вывод. Конечно, вы можете перенаправить этот вывод в файл для последующего анализа или направить его разработчику ядра для отладочных целей. Листинг 12 показывает некоторые варианты вывода, которые вы можете увидеть.

Листинг 12. Неполный вывод `dmesg`

```
[root@lyrebird root]# dmesg | head -n 30
Linux version 2.4.21-40.EL (bhcompile@hs20-bc1-7.build.redhat.com) (gcc
version 3.2.3
20030502 (Red Hat Linux 3.2.3-54)) #1 Thu Feb 2 22:32:00 EST 2006
BIOS-provided physical RAM map:
```

```
BIOS-e820: 0000000000000000 - 000000000009f800 (usable)
BIOS-e820: 000000000009f800 - 00000000000a0000 (reserved)
BIOS-e820: 00000000000e0000 - 0000000000100000 (reserved)
BIOS-e820: 0000000000100000 - 0000000005f6f0000 (usable)
BIOS-e820: 0000000005f6f0000 - 0000000005f6fb000 (ACPI data)
BIOS-e820: 0000000005f6fb000 - 0000000005f700000 (ACPI NVS)
BIOS-e820: 0000000005f700000 - 0000000005f780000 (usable)
BIOS-e820: 0000000005f780000 - 00000000060000000 (reserved)
BIOS-e820: 00000000fec00000 - 00000000fec10000 (reserved)
BIOS-e820: 00000000fee00000 - 00000000fee01000 (reserved)
BIOS-e820: 00000000ffc00000 - 00000000ffc00000 (reserved)
BIOS-e820: 00000000fffffc00 - 0000000100000000 (reserved)
```

631MB HIGHMEM available.

896MB LOWMEM available.

NX protection not present; using segment protection

On node 0 totalpages: 391040

zone(0): 4096 pages.

zone(1): 225280 pages.

zone(2): 161664 pages.

IBM machine detected. Enabling interrupts during APM calls.

Kernel command line: ro root=LABEL=RHEL3 hdd=ide-scsi

ide_setup: hdd=ide-scsi

Initializing CPU#0

Detected 2392.059 MHz processor.

Console: colour VGA+ 80x25

Calibrating delay loop... 4771.02 BogoMIPS

Page-cache hash table entries: 524288 (order: 9, 2048 KB)

Page-pin hash table entries: 131072 (order: 7, 512 KB)

Буфер кольца ядра также используется для некоторых событий после того, как система загружена. Это определенные программные сбои и события "горячего" подключения. Листинг 13 показывает строку для программы, которая вызвала ошибку сегментации, и несколько строк, связанных с подключением ключа памяти USB.

Листинг 13. Позднейшие события в буфере кольца ядра

```
[root@attic4 ~]# dmesg |tail -n 19
main[15961]: segfault at 0000000000529000 rip 0000000000403b5d rsp
00007fffffd15d00
error 6
```

```
usb 1-4.3: new high speed USB device using ehci_hcd and address 4
scsi5 : SCSI emulation for USB Mass Storage devices
usb-storage: device found at 4
usb-storage: waiting for device to settle before scanning
  Vendor: Sony      Model: Storage Media      Rev: 0100
  Type:   Direct-Access                      ANSI SCSI revision: 00
SCSI device sdb: 1014784 512-byte hdwr sectors (520 MB)
sdb: Write Protect is off
sdb: Mode Sense: 43 00 00 00
sdb: assuming drive cache: write through
SCSI device sdb: 1014784 512-byte hdwr sectors (520 MB)
sdb: Write Protect is off
sdb: Mode Sense: 43 00 00 00
sdb: assuming drive cache: write through
  sdb: sdb1
sd 5:0:0:0: Attached scsi removable disk sdb
usb-storage: device scan complete
SELinux: initialized (dev sdb1, type vfat), uses genfs_contexts
/var/log/messages
```

Как только ваша система запустилась из точки /sbin/init, ядро начинает записывать события в буфер кольца, как вы только что видели, а процессы используют демон syslog для журналирования сообщений, обычно в /var/log/messages. В отличие от буфера кольца, каждая строка syslog имеет отметку времени, а файл продолжает существовать между перезапусками системы. Этот файл - то место, куда вам следует в первую очередь заглянуть при ошибках, возникших на стадии выполнения скриптов init при загрузке.

Большинство демонов имеют имена, заканчивающиеся на 'd'. Листинг 14 показывает, как посмотреть несколько последних сообщений о состоянии демонов после перезагрузки.

Листинг 14. Сообщения демонов из /var/log/messages

```
[root@lyrebird root]# grep "^Apr.*d\:" /var/log/messages|tail -n 14
Apr  2 15:36:50 lyrebird kernel: hdd: attached ide-scsi driver.
Apr  2 15:36:52 lyrebird apmd: apmd startup succeeded
Apr  2 15:36:26 lyrebird rc.sysinit: Setting hostname lyrebird:
succeeded
Apr  2 15:36:26 lyrebird rc.sysinit: Initializing USB keyboard:
succeeded
Apr  2 15:36:55 lyrebird sshd:  succeeded
Apr  2 15:36:55 lyrebird xinetd: xinetd startup succeeded
```

```
Apr  2 15:36:56 lyrebird ntpd:  succeeded
Apr  2 15:36:56 lyrebird ntpd:  succeeded
Apr  2 15:36:56 lyrebird ntpd:  succeeded
Apr  2 15:36:56 lyrebird ntpd: ntpd startup succeeded
Apr  2 15:36:57 lyrebird crond: crond startup succeeded
Apr  2 15:36:58 lyrebird atd:  atd startup succeeded
Apr  2 15:36:58 lyrebird snastart: insmod: streams: no module by that
name found
Apr  2 15:36:58 lyrebird rhnsd: rhnsd startup succeeded
```

Вы также найдете журналы для многих других системных программ в `/var/log`. Например, вы можете посмотреть журнал запуска для вашей системы X Window, которую вы изучали в учебнике "Учебник для экзамена LPI 101, Тема 110: Система X Window ."

Уровни выполнения, остановка и перезагрузка

Уровни выполнения

Уровни выполнения определяют, какие задачи могут быть выполнены в текущем состоянии (или на текущем уровне выполнения) системы Linux. Каждая система Linux поддерживает три базовых уровня выполнения и один или более уровней для обычной работы. Базовые уровни выполнения показаны в Таблице 3.

Таблица 3. Базовые уровни выполнения Linux

Уровень	Назначение
0	Выключение (или остановка) системы
1	Однопользовательский режим; обычно имеет псевдоним <code>s</code> или <code>S</code>
6	Перезагрузка системы

Использование уровней выполнения, за исключением базовых, различается среди дистрибутивов. Один из распространенных наборов показан в Таблице 4.

Таблица 4. Другие распространенные уровни выполнения Linux

Уровень	Назначение
2	Многопользовательский режим без поддержки сети
3	Многопользовательский режим с поддержкой сети
5	Многопользовательский режим с поддержкой сети и системы X Window

Дистрибутив Slackware использует уровень выполнения 4 вместо 5 для полного запуска системы X Window. Debian использует один уровень выполнения для любого многопользовательского режима, обычно это уровень 2. Обращайтесь к документации для вашего дистрибутива.

Уровень выполнения по умолчанию

При запуске Linux уровень выполнения по умолчанию определяется из строки `id:` файла `/etc/inittab`. Листинг 15 показывает типичную строку для такой системы, как Red Hat Enterprise Linux, использующей уровень выполнения 5 для системы X Window.

Листинг 15. Уровень выполнения по умолчанию из `/etc/inittab`

```
[root@lyrebird root]# grep "^id:" /etc/inittab
id:5:initdefault:
```

Изменение уровня выполнения

Есть несколько способов изменить уровень выполнения. Чтобы сделать постоянное изменение, можно отредактировать `/etc/inittab` и изменить уровень выполнения по умолчанию, как вы только что видели.

Если вам нужно только перевести систему на другой уровень выполнения, у вас есть пара способов сделать это. Например, представьте, что вы только что установили новое ядро, и вам нужно добавить некоторые модули после того, как система загрузилась с новым ядром, но до загрузки системы X Window. Вы, возможно,

захотите перевести систему на уровень выполнения 3, чтобы выполнить это. Вы делаете это во время загрузки, редактируя строку ядра (GRUB) или добавляя параметр после имени выбранной системы (LILO). Используйте одну цифру для указания желаемого уровня выполнения (в данном случае 3). Например, вы можете изменить строку из Листинга 5, который вы видели в предыдущем разделе, как показано в Листинге 16.

Листинг 16. Установка уровня выполнения по умолчанию во время загрузки

```
kernel /boot/vmlinuz-2.4.21-40.EL ro root=LABEL=RHEL3 hdd=ide-scsi 3
```

Как только вы закончили вашу работу на уровне выполнения 3, вы, возможно, захотели переключиться на уровень выполнения 5. К счастью, вам не надо перезагружать систему. Вы можете использовать команду `telinit`, чтобы сообщить процессу `init`, на какой уровень выполнения он должен переключиться.

Вы можете определить текущий уровень выполнения, используя команду `runlevel`, которая помимо текущего показывает и предыдущий уровень выполнения. Если первый выведенный символ - 'N', то уровень выполнения не менялся с того момента, как система была загружена. Листинг 17 иллюстрирует проверку и изменение уровня выполнения.

Листинг 17. Проверка и изменение уровня выполнения

```
[root@lyrebird root]# runlevel
N 3
[root@lyrebird root]# telinit 5
[root@lyrebird root]# runlevel
3 5
```

Если вы используете команду `ls`, чтобы отобразить длинный листинг команды `telinit`, вы увидите, что в действительности она является символьной ссылкой на команду `init`. Команда `init` знает, каким образом она была вызвана, и ведет себя соответственно. Поскольку `init` обычно имеет PID 1, она также достаточно интеллектуальна, чтобы знать, если вы запустили ее с помощью `init` а не `telinit`. В этом случае она предполагает, что вы хотите, чтобы она вела себя так, как будто вы вызвали `telinit` вместо нее. Например, вы можете использовать `int 5` вместо `telinit 5`, чтобы переключиться на уровень выполнения 5.

Однопользовательский режим

В отличие от таких операционных систем для персональных компьютеров, как DOS или Windows, Linux по своей сути является многопользовательской системой. Однако, бывают ситуации, когда это становится проблемой, например, когда вам нужно восстановить основную файловую систему или базу данных, или установить и протестировать какое-нибудь новое оборудование. Уровень выполнения 1, или

однопользовательский режим, - это ваше решение для таких ситуаций. Фактическая реализация зависит от дистрибутива, но обычно вы попадаете в оболочку с минимальными возможностями. Обычно отсутствует поддержка сети, нет (или очень мало) запущенных демонов. На некоторых системах вы должны пройти аутентификацию, но на других вы заходите прямо в оболочку как суперпользователь. Однопользовательский режим может быть вашим спасителем, но вы можете также разрушить систему, поэтому будьте осторожны всякий раз, когда заходите в систему с правами суперпользователя.

Также как и с переключением на обычные многопользовательские уровни выполнения, вы можете переключиться на однопользовательский режим, используя `telinit 1`. Как видно из Таблицы 3, 's' и 'S' - это псевдонимы для уровня выполнения 1, так что вместо этого вы можете, например, использовать `telinit s`.

Команды остановки

Помимо использования `telinit` или `init` для остановки работы пользователей и перехода в однопользовательский режим, вы также можете вызвать команду `shutdown`. Команда `shutdown` отсылает сообщения с предупреждением всем зарегистрированным пользователям и блокирует дальнейшие входы в систему. Затем она посылает сигнал `init` о переключении уровней выполнения. После этого процесс `init` посылает всем запущенным процессам сигнал `SIGTERM`, давая им шанс сохранить данные или корректно завершить работу. После 5 секунд или другой задержки, если она указана, `init` посылает сигнал `SIGKILL`, чтобы насильно завершить все оставшиеся процессы.

По умолчанию `shutdown` переключается на уровень выполнения 1 (однопользовательский режим). Вы можете указать ключ `-h`, чтобы остановить систему, или ключ `-r`, чтобы перезагрузиться. Стандартное сообщение выводится вместе с любым указанным вами сообщением. Время можно задать как абсолютное время в формате `hh:mm` или как относительное время `n`, где `n` - количество минут до выключения. Для немедленной остановки используйте `now`, что эквивалентно `+0`.

Если вы вызвали остановку с задержкой, и время еще не прошло, у вас есть возможность отменить остановку, нажав `Ctrl-c`, если команда работает в настоящий момент, или вызвав `shutdown` с ключом `-c` для отмены отложенной остановки. Листинг 18 содержит несколько примеров использования `shutdown`, а также отмены этой команды.

Листинг 18. Примеры остановки

```
[root@lyrebird root]# shutdown 5 File system recovery needed
Broadcast message from root (pts/0) (Mon Apr  3 22:44:29 2006):
File system recovery needed
The system is going DOWN to maintenance mode in 5 minutes!
```

```
Shutdown cancelled.
[root@lyrebird root]# shutdown -r 10 Reloading updated kernel&
[1] 5388
Broadcast message from root (pts/0) (Mon Apr  3 22:45:15 2006):
Reloading updated kernel
The system is going DOWN for reboot in 10 minutes!
[root@lyrebird root]# fg
shutdown -r 10 Reloading updated kernel
Shutdown cancelled.
[root@lyrebird root]# shutdown -h 23:59&
[1] 5390
[root@lyrebird root]# shutdown -c
Shutdown cancelled.
[1]+  Done                  shutdown -h 23:59
```

Вы, возможно, заметили, что наш последний пример не вызвал сообщения с предупреждением. Если время до остановки превышает 15 минут, то сообщение не посылается раньше, чем за 15 минут до этого события, как показано в Листинге 19. Листинг 19 также показывает использование ключа `-t` для увеличения задержки между сигналами SIGTERM и SIGKILL от 5 секунд до 60.

Листинг 19. Другой пример остановки

```
[root@lyrebird root]# date;shutdown -t60 17 Time to do backups
Mon Apr  3 22:51:45 EDT 2006
Broadcast message from root (pts/0) (Mon Apr  3 22:53:45 2006):
Time to do backups
The system is going DOWN to maintenance mode in 15 minutes!
```

Листинг 20. Перезагрузка системы

```
[root@lyrebird root]# reboot
Broadcast message from root (pts/0) (Mon Apr  3 22:58:27 2006):
The system is going down for reboot NOW!
```

Убедимся, что система перезагрузилась обратно на уровень выполнения 3, что видно из использования команд `runlevel` и `uptime` в Листинге 21.

Листинг 21. Другой пример перезагрузки системы

```
[ian@lyrebird ian]$ /sbin/runlevel
N 3
[ian@lyrebird ian]$ uptime
```

```
23:05:51 up 6 min, 1 user, load average: 0.00, 0.06, 0.03
```

Также возможно использовать telinit (или init) для остановки или перезагрузки системы. Также как и в других случаях использования telinit, пользователям не будет послано никаких сообщений, и команда выполнится немедленно, хотя между сигналами SIGTERM и SIGKILL и существует задержка. Про дополнительные ключи telinit, init и shutdown смотрите в соответствующих страницах руководства man.

Остановка, перезагрузка и отключение питания

Вам следует знать еще несколько команд, связанных с остановкой и перезагрузкой.

Команда halt останавливает систему.

Команда poweroff является символьной ссылкой на команду halt, останавливающей систему и делающей затем попытку отключения питания.

Команда reboot - это другая символьная ссылка на команду halt, останавливающая систему и затем перезагружающая ее.

Если какая-то из этих команд вызывается, когда система не находится на уровне выполнения 0 или 6, то вместо нее выполняется соответствующая команда shutdown.

Для получения информации о ключах, которые можно использовать с этими командами, а также более подробной информации об их работе, обращайтесь к страницам руководства man.

Конфигурация уровня выполнения

Теперь вы, возможно, будете удивлены, почему нажатие Ctrl-Alt-Delete на некоторых системах вызывает перезагрузку, и как вообще конфигурируется поведение уровня выполнения. Помните поле

```
id
```

в /etc/inittab? Что ж, существует еще несколько полей в /etc/inittab, а также набор инициализирующих скриптов в таких каталогах, как rc1.d или rc5.d, где цифра обозначает уровень выполнения, к которому применяются скрипты из этого каталога. Листинг 22 содержит строку для Ctrl-Alt-Delete, таким образом, вы видите, почему это вызывает перезагрузку системы.

Листинг 22. Отлавливание ctrl-alt-delete

```
[root@lyrebird root]# grep -i ctrl /etc/inittab
# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

Скрипты, используемые `init` при запуске системы, изменении уровня выполнения или остановке системы, обычно хранятся в каталогах `/etc/init.d` или `/etc/rc.d/init.d`. Наборы символьных ссылок в каталогах `rcn.d`, по одному каталогу для каждого уровня выполнения `n`, определяют запускается ли скрипт при входе на уровень выполнения, или останавливается при выходе из него. Эти ссылки начинаются с "K" или "S", за которыми следует число из двух цифр и имя службы, как показано в Листинге 23.

Листинг 23. Скрипты `init`

```
[root@lyrebird root]# find /etc -path "*rc[0-9]*.d/???au*"
/etc/rc.d/rc0.d/K95audit
/etc/rc.d/rc0.d/K72autofs
/etc/rc.d/rc1.d/K95audit
/etc/rc.d/rc1.d/K72autofs
/etc/rc.d/rc2.d/S20audit
/etc/rc.d/rc2.d/K72autofs
/etc/rc.d/rc3.d/S20audit
/etc/rc.d/rc3.d/S28autofs
/etc/rc.d/rc4.d/K95audit
/etc/rc.d/rc4.d/S28autofs
/etc/rc.d/rc5.d/S20audit
/etc/rc.d/rc5.d/S28autofs
/etc/rc.d/rc6.d/K95audit
/etc/rc.d/rc6.d/K72autofs
[root@lyrebird root]# cd /etc/rc.d/rc5.d
[root@lyrebird rc5.d]# ls -l ???a*
lrwxr-xr-x 1 root root 15 Jan 11 2005 S20audit ->
../init.d/audit
lrwxr-xr-x 1 root root 14 Jan 11 2005 S26apmd ->
../init.d/apmd
lrwxr-xr-x 1 root root 16 Jan 11 2005 S28autofs ->
../init.d/autofs
lrwxr-xr-x 1 root root 13 Jan 11 2005 S95atd ->
../init.d/atd
```

Здесь вы видите, что службы `audit` и `autofs` имеют Knn вхождений во всех уровнях выполнения и Snn вхождений для обеих в уровнях выполнения 3 и 5. "S" означает, что служба запускается при входе на уровень выполнения, в то время как "K" означает, что она должна быть остановлена. Компонент `nn` в имени ссылки определяет приоритетный порядок, по которому служба должна быть запущена или остановлена. В этом примере `audit` запускается до `autofs` и останавливается после.

Для большей информации о командах `init` и `inittab` обращайтесь к страницам руководства `man`.