

ОП.14
ОСНОВЫ
функционирования UNIX -
СИСТЕМ

ЗАНЯТИЕ 13

Вспомним некоторые команды

Создание, удаление и изменение учетных записей может выполнять только суперпользователь `root`.

Для создания учетной записи пользователя, например, `user1` введем команду:

```
# useradd user1
```

Для создания (изменения) пароля пользователя `user1` введем команду:

```
# passwd user1
```

После ввода команды появится запись:

```
новый пароль: ****
```

```
повторите ввод нового пароля: ****
```

Вспомним некоторые команды

Пользователь самостоятельно может изменить свой пароль в любое время, выполнив команду:

```
$ passwd
```

После ввода команды появится запись:

```
новый пароль: ****
```

```
повторите ввод нового пароля: ****
```

Вспомним некоторые команды

Для создания дополнительной группы, например, `users`,
нужно выполнить команду:

```
# groupadd users
```

Для добавления пользователя (`user1`) в группу `users`
используется команда:

```
# usermod -G users user1
```

Для удаления пользователя (`user1`) используется команда:

```
# userdel user1
```

Запуск и остановка UNIX

Запуск и остановка UNIX

Этапы запуска и останова операционной системы являются начальной и конечной точками отсчета работы UNIX.

Во время запуска выполняются важнейшие процедуры инициализации, которые в дальнейшем могут оказывать решающее влияние на работоспособность системы.

Знание внутренних механизмов процесса загрузки позволяет правильно настроить системные сервисы и устройства, точно так же понимание последовательности выключения системы помогает предотвратить потерю данных.

Загрузка FreeBSD

Загрузка FreeBSD

Процесс загрузки FreeBSD предоставляет различные варианты гибкой настройки системы, позволяет выбирать одну из нескольких операционных систем, установленных на одном и том же устройстве, или даже одну из нескольких версий той же самой операционной системы.

На оборудовании архитектуры x86 за загрузку операционной системы отвечает BIOS (Basic Input/Output System, базовая система ввода/вывода).

Для этого BIOS ищет на жестком диске MBR (Master Boot Record, главная загрузочная запись), которая должна располагаться в определенном месте на диске.

Загрузка FreeBSD

BIOS может загрузить и запустить MBR (главную загрузочную запись), и предполагается, что MBR может взять на себя остальную работу, связанную с загрузкой операционной системы.

Если на вашем диске установлена только одна операционная система, то стандартной MBR будет достаточно.

Программа, находящаяся в MBR, выполняет поиск на диске первого загрузочного раздела (части диска), после чего запускает из этого раздела код загрузки операционной системы.

Загрузка FreeBSD

Если на ваших дисках установлено несколько операционных систем, то вы можете установить другую MBR, ту, что может выдать список различных операционных систем и позволит вам выбрать одну из них для загрузки.

Операционная система FreeBSD поставляется с одной из таких MBR, которую можно установить на диск.

Другие производители операционных систем также предоставляют свои MBR (главные загрузочные записи).

Загрузка FreeBSD

Далее процесс загрузки проходит по такому сценарию: вызывается вторичный загрузчик, известный как загрузчик этапов 1 и 2 операционной системы FreeBSD.

Вторичный загрузчик загружает в оперативную память ядро, которое находится в файле `/boot/kernel/kernel`.

После этого работа вторичного загрузчика заканчивается.

Загрузка FreeBSD

Затем стартует ядро, которое начинает опознавать устройства и выполняет их инициализацию.

После завершения процесса своей загрузки ядро передает управление пользовательскому процессу с именем `init`, который выполняет проверку дисков на возможность использования.

Затем `init` запускает пользовательский процесс настройки ресурсов, который монтирует файловые системы, производит настройку сетевых адаптеров для работы в сети и вообще осуществляет запуск всех процессов, обычно выполняемых в системе FreeBSD при загрузке.

Загрузка FreeBSD

MBR для FreeBSD находится в `/boot/boot0`.

Это копия MBR, т. к. настоящая MBR должна располагаться в специальном месте диска, вне области FreeBSD.

Загрузчик `boot0` очень прост, т. к. программа в MBR может иметь размер, не превышающий 512 байтов.

Если вы установили MBR FreeBSD и несколько операционных систем на ваш жесткий диск, то во время загрузки увидите приблизительно такие строки:

Загрузка FreeBSD

F1 DOS

F2 FreeBSD

F3 Linux

F4 ??

F5 Drive 1

Default: F2

Загрузка FreeBSD

Первый и второй этапы загрузки иницируются с помощью двух небольших программ, которые расположены в одной и той же области диска.

Обе программы располагаются в загрузочном секторе загрузочного раздела, т. е. там, где `boot0` или любая другая программа из MBR ожидает найти программу, которую следует запустить для продолжения процесса загрузки.

Загрузчик `boot1` имеет размер, не превышающий 512 байтов.

Его основные задачи — найти метку диска, хранящую информацию о разделе FreeBSD, и передать управление загрузчику `boot2`.

Загрузка FreeBSD

Загрузчик `boot2` устроен несколько сложнее и предназначен для работы с файловой системой FreeBSD в объеме, достаточном для нахождения в ней файлов.

Кроме того, он может предоставлять простой интерфейс для выбора и передачи управления основному загрузчику, входящему в ядро операционной системы.

Этот загрузчик более сложен, чем `boot2`, и предоставляет удобный и простой интерфейс для настройки процесса загрузки.

Загрузка FreeBSD

Загрузчик представляет собой удобный в использовании инструмент для настройки при помощи простого набора команд, управляемого более мощным интерпретатором с более сложным набором команд.

Во время инициализации загрузчик пытается произвести поиск консоли, дисков и определить, с какого диска он был запущен.

Соответствующим образом он задает значения переменных и запускает интерпретатор, которому могут передаваться пользовательские команды как из скрипта, так и в интерактивном режиме.

Загрузка FreeBSD

Затем загрузчик читает файл, устанавливающий значения по умолчанию для системных переменных.

Кроме того, загрузчик читает файл для изменения в этих переменных.

Далее с этими переменными происходит загрузка выбранных модулей и ядра.

По умолчанию загрузчик выдерживает 10-секундную паузу, ожидая нажатия клавиши <Enter>, и загружает ядро, если этого не произошло.

Загрузка FreeBSD

Если ожидание было прервано, пользователю выдается приглашение ввести одну из набора команд, с помощью которых пользователь может изменить значения переменных, выгрузить все модули, загрузить модули и продолжить процесс загрузки или перезагрузить машину.

После того как ядро завершит загрузку, оно передает управление пользовательскому процессу `init`, который расположен в файле `/sbin/init` или в файле, маршрут к которому указан в переменной `initpath` загрузчика.

Загрузка FreeBSD

Процесс автоматической перезагрузки проверяет целостность имеющихся файловых систем.

Если целостность нарушена и утилита `fsck` не может исправить положение, то процесс `init` переводит систему в однопользовательский режим для того, чтобы системный администратор сам разобрался с возникшими проблемами.

Система переходит в многопользовательский режим в одном из случаев: если процесс `init` определит, что ваши файловые системы находятся в полном порядке, или после того, как пользователь выйдет из однопользовательского режима.

Остановка FreeBSD

Остановка FreeBSD

Остановка системы может происходить посредством утилиты `shutdown`, при этом процесс `init` попытается запустить скрипт `/etc/rc.shutdown`, после чего всем процессам будет послан сигнал **SIGTERM**, а затем и сигнал **SIGKILL** тем процессам, которые еще не завершили свою работу.

Для выключения машины с FreeBSD на аппаратных платформах и системах, которые поддерживают управление электропитанием, просто воспользуйтесь командой:

```
shutdown -p now
```

для немедленного отключения электропитания.

Остановка FreeBSD

Чтобы перезагрузить систему FreeBSD, выполните команду:

```
shutdown -r now
```

Для запуска команды `shutdown` вам необходимо быть пользователем `root` или членом группы `operator`.

Кроме того, можно также воспользоваться командами `halt` и `reboot`.

Для получения дополнительной информации обратитесь к соответствующим разделам справки и к справочной странице по команде `shutdown`.

Остановка FreeBSD

Для управления электропитанием требуется наличие поддержки ACPI (Advanced System Configuration and Power Interface, усовершенствованный интерфейс конфигурирования и управления системой энергопитания) в ядре или в виде загруженного модуля при использовании FreeBSD 5-х.

Запуск Solaris 9

Запуск Solaris 9

Операционная система Solaris может быть загружена для работы в одном из нескольких режимов.

Если имеется много разделов, то загрузчик, размещенный в разделе Solaris, спросит, с какого раздела нужно загружать систему.

Этот раздел должен быть отмечен, как активный.

Если в течение нескольких секунд ответ не будет получен, то по умолчанию загрузится операционная система Solaris.

Запуск Solaris 9

Сам загрузчик работает довольно просто.

Его нельзя настроить, нельзя изменить загрузочный раздел по умолчанию на другой, с операционной системой, отличной от Solaris.

Кроме того, нельзя изменить время задержки и описание раздела.

Операционная система предполагает несколько вариантов загрузки.

Запуск Solaris 9

После инициализации ядра командой `boot` начинается загрузка системы.

В самой первой фазе загружается ядро операционной системы.

Оно расположено в корневом каталоге диска.

Само ядро состоит из небольшого базового модуля и многочисленных загружаемых модулей.

Большинство из этих модулей автоматически загружается во время начальной загрузки системы.

Запуск Solaris 9

Остальные модули, такие, например, как драйверы устройств, загружаются ядром по мере необходимости.

Когда загружается ядро, система считывает в память файл `/etc/system`.

В нем содержится список модулей, подлежащих загрузке, и параметры загрузки этих модулей.

Данный файл может редактироваться в процессе настройки системы.

После этого управление процессом загрузки передается ядру, которое выполняет инициализацию системы.

Система переходит в одно из восьми состояний запуска, называемых также состояниями инициализации

Запуск Solaris 9

Ядро операционной системы в первую очередь запускает процесс, называемый подкачкой (`swapper`).

Процесс подкачки имеет идентификатор 0 и называется `sched` (планировщиком).

Самое первое, что делает `sched`, — запускает процесс `init`.

Команда `/sbin/init` порождает процессы для запуска системы на основе записей, находящихся в файле `/etc/inittab`.

Процесс `init` является родительским процессом по отношению к остальным процессам. Он анализирует записи в файле `/etc/inittab` и определяет последовательность запуска, остановки и перезапуска остальных процессов.

Запуск Solaris 9

Если процесс `init` получает сигнал о сбое питания, он ищет в файле `/etc/inittab` специальные записи типа `powerfail` и `powerwait`.

Соответствующие этим записям процессы запускаются (если позволяет `runlevel`) перед выполнением любой дальнейшей обработки.

Процесс `init` может выполнить необходимые действия по завершению работы операционной системы и записи информации при отключении питания.

Остановка Solaris 9

Остановка Solaris 9

Работающую операционную систему можно остановить, если в режиме суперпользователя `root` выполнить команду

```
# init 0
```

После того как появится приглашение:

```
Type any key to continue
```

нужно нажать клавишу `<Enter>`.

Остановка Solaris 9

Для перезагрузки системы из режима суперпользователя нужно набрать команду:

```
#init 6
```

После того как появится приглашение:

```
Type any key to continue
```

следует нажать клавишу <Enter>.

Если система не отвечает на ввод команд, то единственный выход из этой ситуации — нажать кнопку Reset на системном блоке или использовать кнопку включения/выключения питания.

Запуск Linux

Запуск Linux

При включении компьютера BIOS производит тестирование оборудования, а затем запуск операционной системы.

Сперва выбирается устройство, с которого будет производиться запуск.

Таким устройством может быть один из жестких дисков.

Первоначально считывается самый первый сектор, который называется загрузочным (Master Boot Record, MBR), т. к. у жесткого диска может быть несколько разделов и у каждого — свой загрузочный сектор.

Запуск Linux

В загрузочном секторе находится небольшая программа, которая загружает и запускает операционную систему.

При загрузке с жесткого диска код, расположенный в MBR, проверяет таблицу разделов (также расположенную в MBR), определяет активный раздел (раздел, используемый при загрузке), считывает загрузочный сектор этого раздела и запускает считанный код.

Часто для загрузки Linux используется специальный загрузчик-**LILO**.

Запуск Linux

Существуют и другие загрузчики, подобные **LILO**, однако у него есть несколько полезных функций, которых нет в других загрузчиках, т. к. он был написан специально для Linux.

Например, имеется возможность передачи ядру параметров во время загрузки или изменения некоторых опций, встроенных в ядро.

Среди подобных загрузчиков (**bootlin**, **bootactv**) **LILO** является наилучшим выбором.

Запуск Linux

После того как ядро системы загружено в память с жесткого диска и запущено, выполняется определенная последовательность действий, описанная далее.

Поскольку ядро Linux находится в упакованном виде, то выполняется его **распаковка**.

Это осуществляет небольшая программа, расположенная в самом начале программного кода ядра.

Если на компьютере установлена нестандартная видеокарта, выдается запрос для уточнения требуемого режима.

Запуск Linux

При компиляции ядра можно сразу указать используемый режим, чтобы система не запрашивала его во время загрузки.

Режим также может быть установлен при помощи **LILO** или утилиты `rdev`.

После этого ядро тестирует аппаратную часть компьютера (жесткие диски, дисководы, сетевые адаптеры) и конфигурирует соответствующие драйверы устройств.

Запуск Linux

Далее ядро пытается смонтировать файловую систему для суперпользователя `root`.

Место, куда она будет смонтирована, задается во время компиляции с помощью утилиты `rdev` или **LILO**.

Тип файловой системы определяется автоматически.

Если система `root` не монтируется, например, из-за того, что ядро не содержит драйвер соответствующей файловой системы, то система зависает.

Запуск Linux

Файловая система `root` обычно монтируется в режиме `read-only`, причем это устанавливается таким же образом, как и узел монтирования.

Это делает возможным проверку файловой системы, когда она смонтирована, хотя проверка файловой системы, установленной в режиме `read-write`, не рекомендуется.

Затем ядро запускает программу `init` в фоновом режиме.

Программа расположена в каталоге `/sbin` и именно она становится главным процессом.

Процесс `init` выполняет различные функции, требуемые при установке системы.

Запуск Linux

Наконец, `init` запускает программу `getty` для виртуальных консолей и сериальных устройств.

Эта программа позволяет подключаться к системе посредством виртуальных консолей и терминалов, подключенных через последовательные порты.

Программа `init` может быть сконфигурирована также для запуска и других программ.

После этого процесс запуска системы считается завершенным и система готова к работе.

Остановка Linux

Остановка Linux

При выключении системы Linux необходимо выполнить некоторые процедуры.

Если этого не сделать, то файловые системы и файлы могут быть повреждены.

Это происходит по причине наличия в Linux дискового кэша.

Информация из кэша записывается на диск только через некоторые промежутки времени.

Остановка Linux

Такой подход в работе операционной системы Linux значительно повышает её производительность.

При этом не позволяет просто выключить питание, поскольку в дисковом кэше может находиться большое количество информации.

При внезапной остановке файловая система может быть частично повреждена, т. к. на диск обычно сбрасывается только часть информации.

Остановка Linux

Другой причиной является мультизадачность системы, где одновременно способно выполняться несколько процессов.

Внезапное выключение питания может быть губительным для системы.

Особенно это касается компьютеров, на которых одновременно работает несколько пользователей.

Существуют команды, предназначенные для правильного выключения системы.

Остановка Linux

К ним относятся команды `shutdown` и `halt`, расположенные в каталоге `/sbin`.

Есть два обычных способа их применения.

Если система установлена на компьютере, где работает один пользователь, то для правильного выключения выполняют такие действия:

1. Завершают работу всех программ.
2. Завершают работу всех виртуальных консолей.

Остановка Linux

3. Входят в систему как суперпользователь `root` (или остаются подключенными под этим пользователем).

4. Выполняется команда:

```
halt
```

или

```
shutdown -h now
```

При желании можно определить задержку, которая устанавливается заменой параметра `now` на знак `+` и число минут, по истечении которых будет завершена работа системы.

Остановка Linux

Если на компьютере, на котором установлена система, работает одновременно несколько пользователей, то возможно использование команды `shutdown` в следующем формате:

```
shutdown -h +время сообщение
```

Здесь *время* — это время, по истечении которого работа системы будет завершена, а *сообщение* — информация, объясняющая причина выключения.

Выполнение этой команды предупредит каждого пользователя, работающего в системе, что она будет выключена, например, через 10 минут.

Остановка Linux

Выдача сообщения автоматически повторяется несколько раз перед прекращением работы системы, и каждый раз с более коротким интервалом.

При использовании команды `halt` нельзя установить задержку, поэтому данная программа редко применяется в многопользовательских системах.

Остановка Linux

После запуска процесса прекращения работы системы демонтируются все файловые системы (кроме системы `root`), завершается выполнение всех процессов и программ-демонов, затем демонтируется файловая система `root`, и вся работа завершается.

Далее выдается сообщение, в котором говорится, что можно отключить питание.

Только после этого питание компьютера может быть отключено.

Остановка Linux

В некоторых случаях невозможно завершить этот процесс соответствующим образом.

Например, при повреждении кода ядра в памяти нарушается его работа или система зависает, и нет возможности ввести новую команду.

В этом случае нужно выключить питание, но после перезагрузки проверить операционную систему.

Остановка Linux

Если же неполадки не такие серьезные, например, вышла из строя клавиатура, а ядро работает нормально, то наилучшим вариантом будет подождать несколько минут, пока программа `update` не сохранит на диске информацию, хранящуюся в кэш-буфере.

Только после этого выключить питание.

Остановка Linux

Иногда **выключают** компьютер принудительно после трехкратного выполнения команды `sync`, которая сбрасывает на диск содержимое буфера.

Если в момент выключения работа всех программ была завершена, то эта процедура почти идентична выполнению команды `shutdown`.

Однако файловые системы ещё не демонтированы.

Это может привести к некоторым проблемам.

В любом случае использование этого способа не рекомендуется.

Остановка Linux

Процесс **перезагрузки** может выполняться путем прекращения работы системы, выключения питания и включения снова.

Более простой способ — это выполнить команду `shutdown` с опцией `-r`, например:

```
shutdown -r now
```

Также можно использовать команду:

```
reboot.
```


Остановка Linux

Команда `shutdown` может также применяться для перевода системы в однопользовательский режим, в котором к системе никто не может подключиться кроме пользователя `root`, использующего для работы главную консоль.

Иногда это применяется для административных целей, для выполнения которых не может быть использована нормально работающая система.

Вспомним некоторые команды

`su` — выполняет команду с заменой идентификатора пользователя и группы; `su (Set UID)` – утилита командной строки, которая позволяет вам временно стать другим пользователем (изменить "владельца" текущего сеанса) без необходимости завершать сеанс и открывать новый.

Если имя временного пользователя не указано, то им становится суперпользователь `root`.

Если при этом использовать опцию `-l`, то получим доступ к переменным окружения администратора в полном объеме.

При этом текущий каталог, в котором была отдана команда, сменится на домашний каталог пользователя `root`.

Вспомним некоторые команды

Например, для смены текущего пользователя на `alex`, введём команду:

```
$ su alex
```

```
Password:
```

```
[*****]$
```

При этом надо знать пароль пользователя `alex`.

Для выхода из команды `su`, то есть вернуться к первоначальному пользователю, можно множеством различных способов, например, можно набрать в командной строке `exit` и нажать ENTER:

```
$ exit
```

нажать клавишу < ENTER >.

Вспомним некоторые команды

Для смены владельца на суперпользователя `root` выполним команду:

```
$ su -l
```

```
Password:
```

```
[*****] #
```

При этом необходимо знать пароль суперпользователя `root`.

Суперпользователь может создать нового пользователя, назначить ему пароль и выйти из команды `su`, набрав в командной строке `exit` и нажать ENTER:

```
# exit
```

нажать клавишу < ENTER >.

Контрольные вопросы

1. Пользуясь командой `su` сменить пользователя на `root`.
2. Создать нового пользователя.
3. Назначить ему пароль.
4. Создать дополнительную группу.
5. Добавить нового пользователя в созданную группу.
6. Выйти из программы `su`.

Вспомним некоторые команды

Для быстрого поиска используются очень эффективные средства, одним из которых является команда `find`.

Для поиска файлов, принадлежащих пользователю `alex` выполним команду:

```
# find . -user alex
```

Для отображения списка файлов текущего каталога программы достаточно выполнить команду:

```
# find . -print
```

Для получения содержимого произвольного каталога, например, `/home/developer` нужно выполнить команду:

```
# find /home/developer -print
```

Вспомним некоторые команды

Для поиска файлов, которые были созданы n дней назад выполним команду:

```
# find . -ctime n
```

Для поиска файлов, которые были изменены n дней назад выполним команду:

```
# find . -mtime n
```

Для поиска файлов в текущем каталоге с именами, которые заканчиваются на `tmp`, нужно выполнить команду:

```
# find . -name '*tmp' -print
```

Контрольные вопросы

1. Пользуясь командой `find` найти все файлы, принадлежащие пользователю...
2. Пользуясь командой `find` найти содержимое произвольного каталога...
3. Пользуясь командой `find` найти все файлы созданные... дней назад.
4. Пользуясь командой `find` найти все файлы измененные... дней назад.

Список литературы:

1. Юрий Магда. UNIX для студентов, Санкт-Петербург «БХВ-Петербург», 2007.
2. Unix и Linux: руководство системного администратора, 4-е издание, 2012, Э. Немет, Г. Снайдер, Т. Хейн, Б. Уэйли
3. Организация UNIX систем и ОС Solaris 9, Торчинский Ф.И., Ильин Е.С., 2-е издание, исправленное, 2016.

Спасибо за внимание!

Преподаватель: Солодухин Андрей Геннадьевич

Электронная почта: asoloduhin@kait20.ru