**Файловая система ОС Linux**

Одним из наиболее важных компонентов в ОС Linux является файловая система. В данном разделе будут рассмотрены понятие файловой системы и ее предназначение; иерархия файловой системы ОС Linux; типы файлов ОС Linux. Остальные вопросы, касающиеся администрирования файловой системы, более подробно будут рассмотрены в модуле 3.

**Предназначение файловой системы**

В ОС Linux, как и в любой другой ОС семейства UNIX, любой объект является файлом, хранящимся на файловой системе. Файловая система представляет собой некоторое устройство (например, жесткий диск), отформатированное для хранения файлов. Файловые системы могут находиться на жестких дисках, гибких дисках, CD-ROM или других носителях, которые позволяют осуществлять произвольный или последовательный доступ к данным.

Условно файловую систему ОС Linux можно разделить на следующие компоненты.

Пространство имен - методы именования объектов файловой системы и их иерархическая организация.

Программный интерфейс приложения (API) - набор системных вызовов и библиотек, предназначенных для управления объектами файловой системы.

Модель безопасности - содержит в себе общую схему защиты, разделения прав доступа к объектам и совместного использования объектов.

Техническая реализация - программный код, связывающий логические модели файловой системы с аппаратным обеспечением.

Основными задачами файловой системы являются:

упорядочивание хранимых данных;

простой и быстрый доступ к хранимым данным;

обеспечение целостности хранимых данных.

Точный формат и способы хранения файлов в ОС Linux не имеют значения, так как система обеспечивает общий интерфейс для всех распознаваемых ею типов файловых систем. В ОС Linux файловой системой, устанавливаемой по умолчанию, является ext3fs. При доступе к любой файловой системе из ОС Linux данные представляются в виде иерархии каталогов с находящимися в них файлами вместе с идентификаторами владельцев и групп, битами прав доступа и прочими атрибутами.

**Иерархия файловой системы**

Иерархия каталогов файловой системы ОС Linux соответствует общепринятому в мире UNIX стандарту Filesystem Hierarchy System (FHS). Основное преимущество данного стандарта заключается в том, что определенные типы файлов расположены в соответствующих каталогах.

Например, большинство конфигурационных файлов располагается в каталоге /etc, а файлы журналов различных сервисов располагаются в каталоге /var/log.

Описание каталогов файловой системы ОС Linux

Каталоги /bin, /usr/bin, /usr/locl/bin, /sbin, /usr/sbin и /usr/local/sbin содержат установленные в системе команды. При работе в системе под обычным пользователем, вам будут доступны только команды каталогов /bin, /usr/bin и /usr/locl/bin, так как в стандарте FHS определено, что в каталогах sbin должны содержаться только административные команды.

Рисунок 1.2.Структура каталогов ОС Linux

Основным каталогом файловой системы ОС Linux является корневой каталог /. Ниже корневого каталога располагаются все остальные каталоги, созданные на локальной дисковой подсистеме или подмонтированные с внешних устройств. Процедура монтирования файловой системы означает связывание каталога существующей файловой системы, называемого точкой монтирования, с корневым каталогом новой файловой системы.

Монтирование файловой системы к точке монтирования осуществляется при помощи команды mount. В следующем листинге приведен пример монтирования DVD привода, содержащего дистрибутив ОС Red Hat Enterprise Linux 5.

[root@rhel5 ~]# mount /dev/hdc /mnt/

mount: block device /dev/hdc is write-protected, mounting read-only

В результате выполнения команды mount система вывела информацию о том, что блочное устройство (в данном случае DVD привод) было смонтировано в режиме чтения.

Список смонтированных файловых систем хранится в файле /etc/fstab. Благодаря этому возможны автоматическая проверка целостности файловой системы при помощи команды fsck и монтирование файловых систем на этапе начальной загрузки, а также выполнение сокращенных команд наподобие mount /var/spool. Информация, содержащаяся в этом файле, отражает расположения файловых систем на диске. Подробнее файл /etc/fstab будет рассмотрен в модуле 3.

Размонтирование файловых систем осуществляется командой umount. «Заблокированную» файловую систему размонтировать невозможно. В ней не должно быть ни открытых файлов, ни текущих каталогов выполняющихся процессов. Если размонтируемая файловая система содержит исполняемые программы, они не должны быть запущены. В следующем листинге приводится пример размонтирования ранее подмонтированной файловой системы в каталоге /mnt.

[root@rhe!5 ~]# umount /mnt [root@rhel5 ~]# Is -1 /mnt total 0

Для того, чтобы узнать какие устройства подмонтированы к системе в данный момент необходимо выполнить команду mount без параметров. В следующем листинге приведен пример определения подмонтированных устройств.

[root@rhel5 ~]# mount /dev/sda2 on / type ext3 (rw)

proc on /ргос type proc (rw)

sunrpc on /var/lib/nfs/rpc\_pipefs type rpc\_pipefs (rw) /dev/hdc on /mnt type iso9660 (ro)

Как видим, в выводе команды mount последовательно отображается подмонтированное устройство, точка монтирования, тип файловой системы и дополнительные опции монтирования.

**Типы файлов**

При работе с ОС Linux важно понимать, что любой объект ОС является файлом. Это ключевая особенность ОС Linux по сравнению с операционными системами семейства Windows.

Файлы различаются как по своей структуре, так и по своему предназначению. В ОС Linux определены семь типов файлов:

регулярные файлы (обычные файлы);

каталоги;

ссылки;

советы;

именованные каналы;

файлы блочных устройств;

файлы символьных устройств.

Определить тип файла можно с помощью команды Is -Id. Первый символ в строке вывода обозначает тип файла. В следующем листинге выдается информация о файле

/dev/hdc. [root@rhel5 ~]# Is -Id /dev/hdc

brw-rw------ 1 root disk 22, 0 Dec 22 13:30 /dev/hdc

Регулярные файлы

К регулярным файлам относятся бинарные файлы, библиотеки, текстовые файлы и файлы различных приложений. ОС Linux не накладывает никаких ограничений на структуру данных файлов. К их содержимому возможен как последовательный, так и прямой доступ.

Каталоги

Файлы данного типа хранят именованные ссылки на другие файлы. Имя файла, находящегося в каталоге, в действительности хранится в родительском каталоге, а не в самом файле.

В ОС Linux имеются так называемые специальные каталоги. Специальные каталоги, такие как «.» и «..», обозначают, соответственно, текущий рабочий каталог и его родительский каталог.

Ссылки

В ОС Linux принято различать символические и жесткие ссылки, каждая из которых имеет особенное значение.

Символическая ссылка позволяет вместо имени файла указывать его псевдоним. В процессе поиска файла по символическим ссылкам, ядро извлекает хранящиеся в них имена. Жёсткая ссылка является прямой, т.е. указывает непосредственно на индексный дескриптор файла, тогда как символическая ссылка указывает на файл по его имени. Файл, адресуемый символической ссылкой, и сама ссылка представляют собой разные объекты файловой системы.

Символические ссылки могут содержать произвольное имя, т.е. в них разрешается указывать на файлы, хранящиеся в других файловых системах и даже на несуществующие файлы. Жесткие ссылки не могут указывать на файл, находящийся в другой файловой системе.

ОС Linux подсчитывает количество ссылок на каждый файл и при удалении файла не освобождает блоки данных до тех пор, пока не будет удалена последняя ссылка на него.

Сокеты

Сокет - это специальный тип файла, используемый процессами для взаимодействия друг с другом. Установленные посредством сокетов соединения позволяют процессам взаимодействовать, не подвергаясь влиянию других процессов. В ОС Linux выделяется несколько видов сокетов, использование которых предполагает наличие сетевой инфраструктуры. Локальные сокеты доступны только на локальном компьютере, обращение к ним осуществляется через специальные объекты файловой системы, а не через сетевые порты. Такие сокеты принято называть сокетами домена UNIX (UNIX domain socket). Помимо локальных сокетов существуют сетевые сокеты, позволяющие процессам обмениваться данными по сети.

Несмотря на то, что другие процессы распознают файлы сокетов как элементы каталога, только процессы, между которыми установлено соответствующее соединение, могут осуществлять чтение и запись файлов сокета. С локальными сокетами работают различные сервисы ОС Linux - CUPS, X Window и Syslog.

Именованные каналы (FIFO)

Файлы данного типа подобны сокетам, поскольку тоже используются для взаимодействия между процессами, однако в отличие от сокетов в именованных каналах данные передаются только в одном направлении.

Файлы блочных и символьных устройств

Файлы блочных и символьных устройств позволяют приложениям получать доступ к аппаратным средствам и периферийному оборудованию системы. На этапе конфигурирования к ядру ОС Linux динамически подгружаются необходимые модули, предназначенные для управления аппаратными средствами системы. За управление конкретным устройством отвечает специальный модуль, называемый драйвером устройства.

Драйверы устройств образуют стандартный интерфейс взаимодействия, который воспринимается пользователем как набор обычных файлов. Получив запрос к файлу символьного или блочного устройства, ядро передает данный запрос соответствующему драйверу. Файлы блочных и символьных устройств сами по себе не являются драйверами. Их можно рассматривать как шлюзы, через которые драйвер принимает запросы на выполнение предусмотренных операций.

Файлы символьных устройств не используют буферизацию в процессе операций ввода-вывода. Все операции ввода-вывода осуществляются немедленно по мере поступления. К символьным устройствам относятся виртуальные терминалы, модемы и другие устройства, не поддерживающие произвольный доступ к данным.

Файлы блочных устройств обрабатываются драйверами, которые осуществляют ввод-вывод данных цельными блоками. В данном случае, на уровне ядра выполняется буферизация данных ввода-вывода. К блочным устройствам относятся такие устройства как жесткие диски, DVD приводы, модули памяти, а так же все остальные устройства, поддерживающие произвольный доступ к данным.

Файлы блочных и символьных устройств характеризуются двумя номерами: старшим (major) и младшим (minor). Старший номер устройства позволяет ядру определить, к какому драйверу относится файл, а младший номер идентифицирует конкретное физическое устройство.

Просмотреть номера файлов устройств можно, используя команду Is:

[root@rhe!5 ~]# Is -1 /dev/

brw-г-------- 1 root disk 8, 0 Dec 22 13:30 sda

brw-r-------- 1 root disk 8, 2 Dec 22 13:30 sda2

brw-rw------- 1 root disk 22, 0 Dec 22 13:30 hdc

crw-rw--------- 1 root root 4, 0 Dec 22 16:29 ttyO

crw--w--------- 1 root tty 4, 1 Dec 23 00:05 ttyl

crw-------------- 1 root root 5, 1 Dec 22 16:43 console

Из данного листинга видно, что блочному устройству sda2 соответствует старший номер 8 (номер драйвера) и младший номер 2 (номер устройства). Номер драйвера 8 в ОС Linux соответствует драйверу SCSI дисков sd. Младший номер устройства 2 является порядковым номером данного устройства, т.е, по сути, файл /dev/sda2 представляет второй раздел, созданный на жестком диске SCSI sda.

Информацию о старших и младших номерах, используемых конкретным драйвером устройства, можно найти в четвертом разделе справочного руководства man для данного драйвера.

**Основные команды управления файлами**

В ОС Linux имена файлов могут быть представлены как в нижнем, так и в верхнем регистре, причем регистр символов имеет значение. Два одинаковых имени файла, записанных в разных регистрах, будут являться физически разными файлами. В имени файла допускается использовать символы «.», «-», «\_», «~», а также ряд других символов, однако их использование может иметь специальное значение для ОС Linux.

К специальным относятся символам «\*», «?», «/», «\», «"», «~». С помощью символов «\*» и «?» можно указывать шаблоны" подстановки имен, которые могут иметь достаточно сложную структуру. Символ «/» недопустимо использовать в качестве имени файла, поскольку он предназначен для разделения каталогов при указании пути к файлу. Символ «\» в основном используется для переноса ввода текста на следующую строку. Использования данного символа в имени файла, как правило, заключается в экранировании пробелов между отдельными словами имени файла. В ОС Linux имена файлы могут содержать пробелы, однако их необходимо экранировать при помощи символа «\» или записывать имя файла в кавычки. Единичный символ «~» имеет особое значение и используется как ссылка на домашний каталог пользователя.

В предыдущем разделе уже упоминались специальные каталоги «.» и «..», которые используются, соответственно, для обращения к текущему рабочему каталогу и родительскому каталогу. В случае использования символа «.» в качестве первого символа имени файла, данный файл будет скрыт для обычного просмотра, а так же для команды удаления, если при запуске данной команды не была указана соответствующая опция.

Длина имени файла в ОС Linux зависит от типа файловой системы, на которой располагается данный файл. На файловой системе ext3, используемой в ОС Linux по умолчанию, максимальная длина имени файла не может превышать 255 символов.

Следующие команды являются базовыми для проведения повседневных операций администрирования ОС. К таким командам относятся команды, позволяющие просматривать, копировать, перемещать, переименовывать и удалять файлы.