

Лабораторная работа

Диски и файловые системы

Цели работы:

- изучить основные концепции и принципы функционирования дисков и файловых систем
- изучить методы управления дисками и файловыми системами в Linux
- приобрести навыки работы с утилитами управления дисками и файловыми системами

Теоретические сведения

Дисковая подсистема - совокупность аппаратных и программных компонентов, которые обеспечивают возможность доступа к физическим носителям данных и управления информацией, которая на них хранится. От работы дисковой подсистемы зависят скорость доступа к данным, надежность хранения информации и возможности её организации. Поэтому она должна быть оптимально настроена для каждой конкретной системы, с учетом типа хранилища данных, количества дисков, объема данных, которые необходимо хранить и обрабатывать, а также требований к доступности и резервированию хранилищ. Управление дисковой подсистемой возможно как через командную строку, так и с использованием графических утилит.

Получение информации об аппаратном обеспечении дисковой подсистемы

cat /proc/devices - выводит список всех доступных устройств, зарегистрированных в системе

ls -la /dev/disks/by-* - выводит список всех устройств блочного хранения (дисков) в системе, сгруппированных по различным критериям

ls -la /dev – выводит содержимое специального каталога, который содержит файлы, представляющие устройства и драйверы устройств в системе

lsblk – выводит список блочных устройств

lshw -short -C disk, hdparm -I /dev/sda, smartctl --all /dev/nvme0 – утилиты, отображающие информацию об аппаратных параметрах дисков

Разделы жесткого диска

Разделы диска представляют собой логически выделенные части накопителя данных, состоящие из смежных блоков, которые отображаются в системе как отдельные диски. Каждый раздел может иметь свою собственную файловую систему и параметры форматирования, появляется возможность установить несколько операционных систем на один жесткий диск. Использование разделов в сочетании с технологиями LVM и RAID увеличивает надежность хранения данных и снижает риск потери информации в случае сбоя в системе.

Таблица разделов диска - структура данных, которая обычно хранится в самом начале диска и содержит информацию о количестве разделов, их размере, типе файловой системы и других параметрах. Существуют несколько форматов таблиц разделов, наиболее распространенными являются MBR (Master Boot Record) и GPT (GUID Partition Table).

MBR - стандарт разметки жестких дисков, разработанный для IBM PC. Главная загрузочная запись (MBR) находится в первом секторе жесткого диска и содержит информацию о расположении разделов, а также программу начальной загрузки операционной системы. MBR поддерживает до четырех основных разделов или три основных и один расширенный раздел, в котором можно создать несколько логических дисков. Максимальный размер диска MBR ограничен 2 ТБ. MBR применяется в системах, использующих для загрузки BIOS (Basic Input/Output System).

GPT представляет собой новый формат таблицы разделов и является частью UEFI (Unified Extensible Firmware Interface) - стандарта, пришедшего на смену BIOS. UEFI использует GPT там, где BIOS использует MBR. Количество разделов не ограничено стандартом и зависит от операционной системы (техническое ограничение 2^{64} штук разделов из-за разрядности полей). GPT позволяет работать с дисками размером до 9,4 ЗБ ($9,4 \times 10^{21}$ байт) при размере сектора 512 байт. GPT обеспечивает дополнительную надежность и защиту от ошибок путем хранения резервной копии таблицы разделов в конце тома, а также наследственного MBR.

Команды Linux для работы с дисками и разделами

lsblk - выводит информацию о блочных устройствах и созданных на них разделах, их размерах, точке монтирования

blkid - выводит атрибуты блочных устройств

fdisk - утилита для модификации таблиц разделов формата MBR

gdisk - утилита для модификации таблиц разделов формата GPT

parted - утилита для разбиения дисков и создания разделов, управляющая разделами с разметкой MBR и GPT

cat /proc/partitions - выводит список разделов.

RAID

RAID (Redundant Array of Independent Disks) - технология объединения нескольких физических дисков в один логический массив для повышения отказоустойчивости и/или производительности хранилища данных. RAID может быть реализован как на аппаратном, так и на программном уровне. Отказоустойчивость RAID-массивов основана на принципе избыточности. Например, RAID 1 использует зеркалирование данных, что позволяет сохранить их в случае отказа одного из дисков, при этом стоимость хранения возрастает в 2 раза по сравнению с обычным хранением данных на. В RAID 5 данные записываются на несколько дисков, при этом один диск выделяется для хранения контрольных сумм. Если один из дисков выходит из строя, то имеется возможность восстановления данных. Степень избыточности в этом случае будет зависеть от количества дисков, объединенных в RAID. RAID 6 обеспечивает защиту от потери данных при выходе из строя сразу двух дисков в массиве, для хранения контрольной информации требуются два дополнительных диска.

RAID также может использоваться для повышения скорости выполнения дисковых операций (например, RAID 0) в первую очередь за счет

одновременной записи данных на несколько дисков. Однако, если один из дисков выйдет из строя, все данные будут утеряны.

Команды Linux для работы с RAID

sudo apt-get install mdadm - установка утилиты;

sudo mdadm --create /dev/md0 -l 1 -n 2 /dev/sd{b,c} - создание нового массива;

cat /proc/mdstat - получение текущего состояния;

/etc/mdadm.conf - файл конфигурации для утилиты mdadm в котором хранятся настройки RAID-массивов, такие как тип, устройства, используемые для создания массива и др.

LVM

LVM (Logical Volume Manager) - это технология управления логическими томами, позволяющая объединить несколько физических дисков в один или несколько логических томов для более эффективного использования дискового пространства и упрощения управления им.

Команды Linux для создания LVM

pvccreate - создает физический том на жестком диске

vgcreate - создает группу томов из физических томов

lvcreate - создает логический том в группе томов.

Команды Linux для получения состояния LVM

pvdisplay\pvs - отобразить информацию о физических томах

vgdisplay\vgs - отобразить информацию о группах томов

lvdisplay\lvs - отобразить информацию о логических томах.

Файловая система

Файловая система определяет способ организации, хранения, именования данных на носителях информации, а также правила доступа к файлам и каталогам. Кроме этого, файловая система предоставляет набор функций для работы с файлами, таких как создание, копирование, перемещение, удаление, изменение размера и т.д.

Получение информации о файловых системах

cat /proc/filesystems — выводит список файловых систем, которые поддерживаются ядром;

fsck – утилита, проверяющая файловую систему на ошибки;

file -s /dev/sda1 - выводит тип файловой системы;

df -T - выводит типы файловых систем, их размеры, используемое и доступное место, процент использования, точка монтирования файловой системы.

Монтирование файловых систем

После разбиения диска на разделы в каждом из них необходимо создать файловую систему. Затем, для того, чтобы раздел стал доступным для использования, выполняется его монтирование в основную файловую систему. Точка монтирования - каталог в файловой системе, который становится корневым каталогом для подключаемой файловой системы. В Linux все используемые разделы дисков монтируются в подкаталоги корня, а

не представляются как отдельные устройства. Монтирование раздела позволяет операционной системе обращаться к файлам и каталогам, которые находятся в этом разделе, как к обычным файлам и каталогам в файловой системе.

Для монтирования файловой системы необходимо выполнить следующие действия.

1. Создать точку монтирования. Например, для создания точки монтирования в директории `/mnt/mydisk` выполняется команда: ***sudo mkdir /mnt/mydisk***.
2. Выполнить команду монтирования, указав имя устройства и точку монтирования. Обычно при монтировании устройства, команда ***mount*** автоматически определяет тип файловой системы. При этом имеется возможность явно указать тип с помощью опции ***-t***. Например, для монтирования файловой системы `ext4` используется команда: ***sudo mount -t ext4 /dev/sda1 /mnt/mydisk***. Здесь `/dev/sda1` - это устройство, на котором расположена файловая система, а `/mnt/mydisk` - точка монтирования.
3. Проверить, что файловая система успешно примонтирована. Для этого выполняется команда: ***df -h***. В выводе должна появиться информация о примонтированной файловой системе.

Для автоматического монтирования файловой системы при каждом запуске системы необходимо добавить соответствующую запись в файл ***/etc/fstab***, в которой указываются параметры монтирования, такие, как тип файловой системы, точка монтирования, опции монтирования и другие.

Типы файловых систем

FAT (File Allocation Table) - это файловая система, которая используется для организации и хранения данных на жестких дисках, флеш-накопителях и других устройствах хранения данных. Она имеет простую структуру и хорошую совместимость с различными устройствами, но ограничена в поддержке файлов большого размера и не обеспечивает высокой степени безопасности данных.

Команды для создания FAT:

sudo apt-get install dosfstools - установка необходимых пакетов;

mkfs.vfat -F 32 -n MyDrive /dev/sd* - создает файловую систему FAT32 на устройстве `/dev/sd*` с названием `MyDrive`;

fatresize -s /dev/sd* - изменяет размер файловой системы FAT на устройстве `/dev/sd*`. Опция `-s` указывает новый размер файловой системы.

NTFS (New Technology File System) - стандартная файловая система для операционных систем семейства Windows NT.

Команды для создания NTFS:

sudo apt-get install ntfs-3g - установка необходимых пакетов;

mount -t ntfs-3g /dev/sdb1 /mnt - монтирование NTFS в Linux.

Ext2 (Extended File System 2) является предшественницей более современных файловых систем, таких как Ext3 и Ext4. Обладает хорошими показателями скорости и производительности за счет отсутствия журналирования.

Команда ***mkfs -t ext2 /dev/hdb1*** создает ext2 в разделе /dev/hdb1.

Ext4 в настоящее время является файловой системой «по умолчанию» в большинстве дистрибутивов Linux. Поддерживает журналирование и шифрование данных, имеет механизм пространственной (extent) записи файлов, уменьшающий фрагментацию и повышающий производительность.

Команда ***mkfs -t ext4 /dev/hdb1*** создает ext4 в разделе /dev/hdb1.

XFS - высокопроизводительная 64-битная файловая система. Была создана для обеспечения высокой производительности и масштабируемости при работе с дисками большого объема (более 2 Тбайт). XFS поддерживает функции журналирования, расширяемости и дефрагментации файловой системы в реальном времени, имеется возможность обнаружения и восстановления ошибок.

Команды для работы с XFS:

mkfs.xfs /dev/sda1 - создание файловой системы;

xfs_info /dev/sda1 - выводит информацию о файловой системе XFS, которая находится на устройстве /dev/sda1;

xfs_growfs / -d - расширение файловой системы XFS до максимального доступного размера, опция -d используется для вывода отладочной информации;

xfs_check /dev/sdb1, xfs_repair /dev/sdb1 - проверка и восстановление файловой системы XFS на устройстве /dev/sdb1.

Btrfs (B-tree FS) - нежурналируемая файловая система, основанная на структурах B-деревьев. Была создана с целью предоставления расширенных возможностей по управлению данными, включая копирование на лету (copy-on-write), снимки, сжатие, проверку целостности данных, возможность объединить несколько накопителей в единую файловую систему и т.д. Btrfs использует древовидную структуру для хранения данных, что позволяет быстро и эффективно обрабатывать большие объемы информации, поддерживает RAID-массивы на уровне файловой системы.

Команды для работы с Btrfs:

mkfs.btrfs /dev/sdc -L single_drive - создание файловой системы на одном диске;

mkfs.btrfs /dev/sdc /dev/sdd -L double_drive - создание файловой системы на двух дисках;

sudo mkfs.btrfs /dev/sdc /dev/sdd -d raid1 -m raid1 -L raid1_drive – создать RAID;

btrfs filesystem df / - получить информацию о файловой системе;

btrfs filesystem resize -2g /mnt - изменить размер тома в реальном времени.

Inode

Inode (индексный дескриптор) - это уникальный идентификатор, который используется в файловых системах Linux для хранения и организации информации об объектах файловой системы, таких как файлы, каталоги, символические ссылки и др. Каждый файл или каталог имеет индексный дескриптор, который содержит метаданные файла, включающие права

доступа, идентификатор владельца, размер файла, время создания и модификации, а также адресную информацию.

Команды для получения информации об индексных дескрипторах

ls -li - показывает номер индексного дескриптора для файлов и подкаталогов в указанном каталоге;

stat - показывает информацию о файле, включая номер его индексного дескриптора;

find /path/to/search -inum <inode_number> - поиск файла по номеру индексного дескриптора;

df -li - выводит информацию об использовании индексных дескрипторов в файловой системе;

du --inodes - показывает количество индексных дескрипторов в указанном каталоге и его подкаталогах.

Порядок выполнения работы

Задание 1. Разметка диска, создание файловой системы

Подключите к виртуальной машине 2 новых диска, первый диск произвольного размера, второй диск размером 3 ГБ.

1. На первом диске создайте таблицу разделов MBR. Создайте 4 раздела: первый раздел занимает 50% диска, остальные - произвольного размера. Один из разделов должен быть логическим.

2. На втором диске создайте таблицу разделов GPT. Создайте один раздел размером 2 ГБ и разметьте его как xfs. Смонтируйте раздел по пути /mnt. Создайте файл на смонтированной файловой системе. Сделайте снимок экрана вывода команды **df -hT**.

Увеличьте раздел до 3 ГБ. Расширьте файловую систему на новое свободное пространство. Убедитесь, что созданный файл остался внутри раздела и файловой системы. Сделайте снимок экрана вывода команды **df -hT**.

В отчете приведите снимки экрана, демонстрирующие разметку диска (например, командами **lsblk -a**; **fdisk -l**) и снимки экрана после выполнения команд **df**.

Задание 2. Создание RAID-массива

Создайте программный RAID 1 при помощи утилиты **mdadm**. Объем RAID выберите самостоятельно. В отчете приведите снимок экрана вывода команды **mdadm --create /dev/md0**, где md0 - название созданного RAID-массива.

Задание 3. Получение информации о файловой системе

Определите тип основной файловой системы, ее размер, общее количество и количество занятых inode.

В отчете приведите использованные команды и результаты их выполнения в виде снимков экрана.

Дополнительное задание

Сделайте снимки экрана вывода команд **df -h**, **pvs**, **lvs**, **vgs**;

- подключите 2 новых диска;
- создайте новую VG, добавьте в нее 1 диск;

- создайте два LV, распределите доступное пространство между ними поровну;
- создайте на обоих томах файловую систему xfs;
- создайте две точки монтирования и смонтируйте каждый из томов;
- сделайте снимок экрана вывода команд `df -h`;
- добавьте в VG второй оставшийся диск;
- расширьте первый LV на объём нового диска;
- расширьте файловую систему на размер нового доступного пространства;
- сделайте снимки экрана вывода команд `df -h`, `pvs`, `lvs`, `vgs`.

В отчете приведите снимки экрана, полученные при выполнении задания.

Контрольные вопросы

1. Дайте определение дисковой подсистемы операционной системы. Какие ее ключевые функции?
2. С помощью каких команд в Linux можно получить общий список блочных устройств (дисков, разделов) и информацию о них? Какие еще типы устройств существуют? Приведите примеры
3. В чем заключается основное назначение разделов на жестком диске?
4. Назовите ключевые различия между таблицами разделов MBR и GPT (количество разделов, максимальный размер диска, связь с BIOS/UEFI).
5. Какая утилита в Linux является стандартным инструментом для работы с разделами на дисках с MBR, а какая — для GPT?
6. Что такое RAID? Какова основная цель использования RAID 1 (зеркалирование) и RAID 0 (стрейпинг)? В чем их принципиальное различие с точки зрения надежности?
7. Какие виды RAID-массивов, помимо рассмотренных существуют? Что представляют собой RAID 10 и RAID 01
8. Что такое LVM (Logical Volume Manager) и каковы его основные преимущества перед работой с обычными разделами?
9. Что такое файловая система? Какие функции она предоставляет?
10. Что такое "точка монтирования" (mount point) в Linux? Опишите шаги, необходимые для ручного монтирования раздела `/dev/sdb1` с файловой системой `ext4` в каталог `/mnt/data`.
11. Какой файл конфигурации в Linux отвечает за автоматическое монтирование файловых систем при загрузке?