

1. 简述/etc/fstab里面各字段的含义

文件系统（File System）：

- 指定设备或分区的路径，可以是设备文件（如 `/dev/sda1`）、UUID（如 `UUID=xxxx-xxxx`）、LABEL（如 `LABEL=home`），也可以是网络文件系统的路径（如 NFS 路径 `server:/export/path`）。

挂载点（Mount Point）：

- 指定该文件系统挂载到的目录（如 `/`，`/home`，`/data`）。如果是交换分区，通常会填 `swap`。

文件系统类型（File System Type）：

- 指定文件系统的类型（如 `ext4`，`xfs`，`swap`，`nfs`）。系统会根据此字段选择合适的驱动程序进行挂载。

挂载选项（Mount Options）：

指定挂载选项，用逗号分隔（如 `defaults`，`noatime`，`ro`）。常见的选项包括：

- `defaults`：使用默认选项。
- `ro`：只读挂载。
- `rw`：读写挂载。
- `noatime`：不更新文件的访问时间，提升性能。
- `user`：允许非 `root` 用户挂载。

转储频率（Dump Frequency）：

- 此字段设置是否使用 `dump` 命令备份文件系统。
- `0` 表示不备份，`1` 表示备份。通常非关键文件系统会设为 `0`。

文件系统自检次序（Pass Number）：

- 指定启动时 `fsck` 的检查顺序。
- `0` 表示不检查，`1` 表示优先级最高（通常是根分区），`2` 表示其他分区。其他非关键分区一般设为 `2`。

2. 简述RAID0,RAID1,RAID5三种工作模式的工作原理及特点

RAID 0：带区卷，连续以位或字节为单位分割数据，并行读/写于多个磁盘上，因此具有很高的数据传输率，但它没有数据冗余，**RAID 0** 只是单纯地提高性能，并没有为数据的可靠性提供保证，而且其中的一个磁盘失效将影响到所有数据。因此，**RAID 0** 不能应用于数据安全性要求高的场合

RAID 1：镜像卷，它是通过磁盘数据镜像实现数据冗余，在成对的独立磁盘上产生互为备份的数据，不能提升写数据效率。当原始数据繁忙时，可直接从镜像拷贝中读取数据，因此**RAID1** 可以提高读取性能，**RAID 1** 是磁盘阵列中单位成本最高的，镜像卷可用容量为总容量的1/2，但提供了很高的数据安全性和可用性，当一个磁盘失效时，系统可以自动切换到镜像磁盘上读写，而不需要重组失效的数据

RAID5：至少由3块硬盘组成，分布式奇偶校验的独立磁盘结构，它的奇偶校验码存在于所有磁盘上，任何一个硬盘损坏，都可以根据其它硬盘上的校验位来重建损坏的数据（最多允许1块硬盘损坏），所以**raid5**可以实现数据冗余，确保数据的安全性，同时**raid5**也可以提升数据的读写性能

3. 如果买了个比较新的硬盘，容量大于2T，linux需要什么命令对磁盘分区

在 Linux 中可以使用 `gdisk` 或 `parted` 命令来对大于 2TB 的硬盘进行分区

`sudo gdisk /dev/sdb`

输入 `n` 创建一个新分区，`gdisk` 会提示您选择分区号、起始扇区和结束扇区。可以按默认值或根据需要调整。

输入 `w` 写入分区表并退出

使用 `parted` 进行分区

使用 `parted` 命令创建 GPT 分区表：

`parted /dev/sdb`

```
# 设置分区表类型为 GPT:
parted mklabel gpt

# 创建分区
parted mkpart primary 0GB 1000GB

# 退出
parted quit
```

4. 怎么在不影响应用运行的情况下，将数据迁移到新盘（使用我们学过的知识）
提示（逻辑卷）

```
# 分区
fdisk /dev/sdc

# 创建物理卷
pvcreate /dev/sdc1

# 创建卷组
vgcreate mytest /dev/sdc1

# 创建逻辑卷
lvcreate -L 20G -n mylvs mytest

# 创建文件系统
mkfs.ext4 /dev/mytest/mylvs

# 挂载
mount /dev/mytest/mylvs /data

# 在里面添加文件
cp /etc/passwd /data

# 先卸载
umount /data

# 新盘分区
fdisk /dev/sdb

# 将新盘加入到卷组:
pvcreate /dev/sdb # 将新盘初始化为物理卷
vgextend mytest /dev/sdb1 # 将新盘加入到现有的卷组

# 迁移数据到新盘: 使用 pvmove 命令将数据从旧盘迁移到新盘上
pvmove /dev/sdc1 /dev/sdb1 # 将 /dev/sdc1（旧盘）上的数据迁移到卷组的/dev/sdb1

# 重新挂载
mount /dev/mytest/mylvs /data

# 查看数据还在
ls /data

# lsblk, 发现逻辑卷所在磁盘成功移到新盘
lsblk

# 移除旧盘: 数据迁移完成后, 您可以将旧盘从卷组中移除:
vgreduce mytest /dev/sdc1
```

5. apt remove 和 apt purge, apt autoremove有什么区别, 分别的使用场景是什么

```
apt remove
```

功能: `apt remove` 命令会删除指定的软件包, 但会保留与该软件包相关的配置文件。这些配置文件通常位于 `/etc` 目录中

使用场景: 如果您只是暂时不需要该软件, 但可能将来还会重新安装, 使用 `apt remove` 可以保留配置文件, 以便重新安装后仍然保留之前的设置。

`apt purge`

功能: `apt purge` 命令会删除指定的软件包及其所有的配置文件。这将完全清理与该软件包相关的所有数据。

使用场景: 如果您不再需要该软件, 且打算再次使用, 可以选择 `apt purge` 来彻底删除该软件及其配置文件, 释放更多的磁盘空间。

`apt autoremove`

功能: 卸载所有自动安装且不再使用的软件包

使用场景: 无论使用 `apt remove` 还是 `apt purge`, 卸载之后的孤立依赖包不会自动清理。可以使用 `autoremove` 清理不再需要的依赖:

6. 实现私有yum仓库

- 该仓库内的安装包, 部分是光盘的BaseOS
- 部分同步阿里云上的EPEL源
- 剩下的将光盘的AppStream中的字母o开头的包同步到私有仓, 并自行创建元数据

```
# 安装webserver
yum install -y nginx

# 关闭防火墙
systemctl disable --now firewalld

# 关闭SELinux
setenforce 0

# 启动nginx
systemctl start nginx

# 挂载光盘
mount /dev/sr0 /mnt
cd /usr/share/nginx/html/
mkdir yum

# 配置BaseOS
cp -r /mnt/BaseOS .

# 配置epel
yum install -y epel-release
# 查看repoid
yum repolist
# 同步epel源
dnf reposync --repoid=epel --download-metadata -p .
```

```
# 配置AppStream/o
mkdir -p /usr/share/nginx/html/yum/AppStream/Packages
cp -r /usr/share/nginx/html/yum/AppStream/Package/o
/usr/share/nginx/html/yum/AppStream/Packages/

# 安装createrepo
yum install -y createrepo
createrepo /usr/share/nginx/html/yum/AppStream/

# 配置客户端
vim self-baseos.repo

[BaseOS]
name=self-BaseOS
baseurl=http://10.0.0.166/yum/BaseOS/
#pgpcheck=1
enabled=1
#pgpkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-Rocky-9

vim self-appstream.repo

[AppStream]
name=self-AppStream
baseurl=http://10.0.0.166/yum/AppStream/
#pgpcheck=1
enabled=1
#pgpkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-Rocky-9

vim self-epel.repo

[epel]
name=self-epel
baseurl=http://10.0.0.166/yum/epel/
pgpcheck=0
enabled=1
#pgpkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-Rocky-9
```

7. 简述集线器，交换机，路由器分别是几层设备，为什么

集线器：一层设备

集线器仅处理物理信号而不涉及数据链路层（Layer 2）或更高层次的操作

交换机：二层设备

由于交换机在数据链路层工作，能够根据MAC地址进行智能转发和管理，因此被称为二层设备。

路由器：三层设备

把一个数据包从一个设备发送到不同网络里的另一个设备中（跨网域）。路由器只关心网络得状态和决定网络中的最佳路径。路由的实现依靠路由器中的路由表来完成

8. 简述TCP/IP网络协议栈，（应用层，表示层，会话层统一算应用层）每层的作用是什么？

物理层：

对于物理层来说，上层交下来的帧**iframe**，把帧中的每个**bit**，变成物理信号，在介质中传递给对方，这是发送端做的事，接收端负责把物理媒体上承载的物理信号，电磁波信号，光信号，把它还原回原来的数字数据**0101**

链路层：

链路层的作用：在物理层提供服务的基础上，在相邻的两点之间，传输以帧为单位的数据，

网络层

在链路层提供的相邻两点数据传输的服务的基础上，传输以分组为单位的端到端的数据传输

传输层

其次，网络层提供的传输本身是不可靠的，可能会发生数据包丢了，错了，乱序，重复等等，传输层另一个可能得服务是需要将网络层提供的不可靠的数据传输，把它变成可靠的数据传输

应用层

应用进程在传输层所提供的服务的基础上，完成应用报文之间的交互，从而实现各种各样的功能，比如订单查询，文件上传，文件下载，邮件通信等等