# 同步

网络路径的表示方法：

ssh

rsync 同步

rsync [选项] 源目录 目标目录

复制：完全拷贝源到目标

同步：增量拷贝，值传输变化过的数据

选项

-n：测试同步过程不做实际修改

--delete：删除目标文件夹内多余的文档

-a：归档模式，

-v：显示详细信息

-z：传输过程中启用压缩解压

rsync -avz /nsd01 /test01 同步，将/nsd01/的所有新的文件同步的到/test01

rsync --delete -avz /nsd01 /test01 使源目录与目标目录内容保持一致

rsync + ssh （远程同步）

rsync --delete -avz root@192.168.4.7:/mnt/ /test01/ 下行同步

rsync --delete -avz /test01/ [root@192.168.4.7:/mnt/](mailto:root@192.168.4.7:/mnt/) 上行同步

***实时同步***（数据传输主要是ssh负责）

1.ssh管理验证机制

1. 验证公钥与私钥的配对，没有用到加密解密
2. 验证用户密码
3. 采用公钥与私钥，实现ssh无密码验证

ssh-keygen生成公钥与私钥

ls /root/.ssh/

id\_rsa私钥 id\_rsa.pub公钥 known\_hosts 记录远程管理过的所有机器

ssh-copy-id [root@192.168.4.207](mailto:root@192.168.4.207) 传输公钥给指定主机的指定位置

2.安装源码包inotify-tools工具

inotifywait 选项 监控目录

-m，持续监控

-r，递归监控、包括子目录及文件

-q，减少屏幕输出信息

-e，指定监视的modify、move、create、delete、attrib(属性)等事件类别

例：inotifywait -rmq /mnt/

1. 书写同步脚本

while [条件判断] # [条件判断]可以是一条命令，运行成功为真，反之为假；

do

循环执行的语句

done & # “&” 在后台运行脚本

例：

#!/bin/bash

while inotifywait -rqq /opt/

do

rsync --delete -az /opt/ root@192.168.4.207:/opt/

done &