

rsync

1.为什么要做备份?

- > 数据非常的重要
- > 保证数据不丢失
- > 便于快速的恢复

2.能不能不做备份?

- > 可以, 对于不是特别重要的数据可以不考虑

3.备份怎么做?

- > 完全备份, 全备, 效率低下。
- > 增量备份, 增备, 效率较高。

4.用什么工具做备份?

- > scp 网络之间的拷贝, 全量拷贝的方式 (ssh协议)
- > rsync 远程同步 (增量)

rsync是一款开源的备份工具

可以在不同主机之间进行同步 (windows和Linux之间 Mac和Linux Linux和Linux)

可实现全量备份与增量备份

因此非常适合用于架构集中式备份或异地备份等应用。

备份的方式

- > 完全备份, 每次都进行全部备份 (效率低下, 占用空间)
- > 增量备份, 仅备份客户端与服务端差异的部分 (提高备份效率, 节省空间, 适合异地备份)

rsync数据的同步模式

- > 推送: 本地将数据上传至备份服务器上 (上传)
- > 拉取: 备份服务器获取本地服务器的数据 (下载)

rsync的数据传输方式

- > 本地传输（类似于使用cp命令）
- > 远程传输（通过网络传输 a-->b）
- > 守护进程（运行一个服务一直在后台）

```
1      命令      选项      源文件      目标文件
2 Local:  rsync [OPTION...] SRC...  [DEST]
3 rsync -avz ./anaconda-ks.cfg /opt/
4
5 Access via remote shell:（走的ssh协议）
6 拉取 Pull: rsync [OPTION...] [USER@]HOST:SRC...
  [DEST]
7 推送 Push: rsync [OPTION...] SRC... [USER@]HOST:DEST
8
9 推送方式:
10     ##nfs----->推送方
11     ##backup-->接收方
12
13     [root@nfs ~]# rsync -avz /etc/passwd
  root@172.16.1.41:/tmp
14     将本地的/etc/passwd文件，推送到172.16.1.41服务器的tmp目
  录下，使用的是41的root用户身份
15
16 拉取方式:
17     ##nfs----->拉取方
18     ##backup-->备份源
19     [root@nfs ~]# rsync -avz
  root@172.16.1.41:/etc/services ./
20     本地nfs服务器下载172.16.1.41服务器/etc/services这个文
  件，至本地当前目录下，使用的是41的系统用户root身份
```

1.安装

```
1 yum install rsync -y
```

2.配置

```
1 [root@backup ~]# rpm -qc rsync
2 /etc/rsyncd.conf          # 主配置文件
```

```

3 /etc/sysconfig/rsyncd          # 选项
4
5 [root@backup ~]# cat /etc/rsyncd.conf
6 uid = rsync
7 gid = rsync
8 port = 873
9 fake super = yes
10 use chroot = no
11 max connections = 200
12 timeout = 600
13 ignore errors
14 read only = false
15 list = false
16 auth users = rsync_backup
17 secrets file = /etc/rsync.passwd
18 log file = /var/log/rsyncd.log
19 #####
20 [backup]
21 comment = welcome to database backup!
22 path = /backup

```

3.配置详解

```

1 [root@backup ~]# vim /etc/rsyncd.conf
2 uid = rsync                      # 运行进程的用户
3 gid = rsync                      # 运行进程的用户组
4 port = 873                       # 监听端口
5 fake super = yes                 # 无需让rsync以root身份
    运行，允许接收文件的完整属性
6 use chroot = no                  # 禁锢推送的数据至某个目
    录，不允许跳出该目录
7 max connections = 200            # 最大连接数
8 timeout = 600                   # 超时时间
9 ignore errors                    # 忽略错误信息
10 read only = false               # 对备份数据可读写
11 list = false                    # 不允许查看模块信息
12 auth users = rsync_backup        # 定义虚拟用户，作为连接认
    证用户
13 secrets file = /etc/rsync.passwd # 定义rsync服务用户连接认
    证密码文件路径

```

```

14
15 [backup]                # 定义模块信息
16 comment = commit       # 模块注释信息
17 path = /backup          # 定义接收备份数据目录

```

4.创建rsync进程启动时需要使用的用户

```

1 [root@backup ~]# useradd rsync -M -s /sbin/nologin
2 [root@backup ~]# id rsync
3 uid=1000(rsync) gid=1000(rsync) groups=1000(rsync)

```

5.创建密码文件，在密码文件中写入对应的虚拟用户以及虚拟用户的密码

```

1 /etc/rsync.passwd---》rsync虚拟用户以及rsync虚拟用户的密码
2 [root@backup ~]# echo "rsync_backup:123456" >
  /etc/rsync.passwd
3 [root@backup ~]# chmod 600 /etc/rsync.passwd

```

6.创建存储备份数据的目录，并进行授权

```

1 [root@backup ~]# mkdir /backup
2 [root@backup ~]# chown -R rsync.rsync /backup/

```

7.启动rsync服务并加入开机自启动

```

1 [root@backup ~]# systemctl start rsyncd.service
2 [root@backup ~]# systemctl enable rsyncd

```

8.检查rsync的873端口是否存在

```

1 [root@backup ~]# netstat -lntp
2 Active Internet connections (only servers)
3 Proto Recv-Q Send-Q Local Address           Foreign
  Address      State        PID/Program name
4 tcp        0      0 0.0.0.0:873             0.0.0.0:*
                LISTEN      11129/rsync

```

客户端测试

Access via rsync daemon:

```
1 下载 Pull: rsync [OPTION...] [USER@]HOST::SRC... [DEST]
2
3 上传 Push: rsync [OPTION...] SRC... [USER@]HOST::DEST
4
5 客户端nfs推送（注意加/ 和不加/的区别）
6
7 [root@nfs ~]# rsync -avz /etc/
rsync_backup@172.16.1.41::backup #推送/etc/下面的内容
8 [root@nfs ~]# rsync -avz /etc
rsync_backup@172.16.1.41::backup #推送/etc这个目录
```

```
1 客户端拉取
2
3 [root@nfs ~]# rsync -avz
rsync_backup@172.16.1.41::backup /opt
4 拉取服务端172.16.1.41的backup目录下面的所有内容至本地的 /opt目
录
```

注意事项:

rsync_backup: 客户端通过该虚拟用户连接rsync服务

- > 是一个虚拟用户，由服务端的配置文件中定义【auth users】
- > 通常存放在指定的一个文件中，该文件也是有服务端配置文件定

义【secrets file】

rsync: 【模块对应的目录，必须授权为配置文件中定义的uid和gid的用户】

- > 用于运行rsync服务时需要使用到的系统用户
- > 用于将接收到的数据以自己的身份写入到对应的目录中

实现客户端与服务端无差异同步: --delete

```
1 #推送方式实现无差异，以客户端为准，客户端有什么服务端就有什么
2 [root@nfs ~]# rsync -avz --delete /root
  rsync_backup@172.16.1.41::backup
3
4 #拉取方式实现无差异，以服务端为准，服务端有什么客户端就有什么
5 [root@nfs ~]# rsync -avz --delete
  rsync_backup@172.16.1.41::backup /opt/
```

对传输时候进行限速：

```
1 [root@nfs ~]# dd if=/dev/zero of=./size.disk bs=1M
  count=500  生成大文件
2
3 限制传输的速率为网速的1%
4 [root@nfs ~]# rsync -avzP --bwlimit=1 ./size.disk
  rsync_backup@172.16.1.41::backup
5 Password:
6 sending incremental file list
7 size.disk
8      118,358,016  22%    1.01MB/s    0:06:33
```

取消每次传输时需要输入密码【客户端才会配置】

方式一：

```
1 [root@nfs ~]# echo "123456" > /etc/rsync.pass
2 [root@nfs ~]# chmod 600 /etc/rsync.pass
  上该文件找123456
3 [root@nfs ~]# rsync -avzP --bwlimit=1 ./size.disk
  rsync_backup@172.16.1.41::backup --password-
  file=/etc/rsync.pass
```

方式二：写Shell脚本

```
1 [root@nfs ~]# export RSYNC_PASSWORD=123456
2 [root@nfs ~]# rsync -avzP ./size.disk
  rsync_backup@172.16.1.41::backup
```

rsync参数: -avz

1	-a	#归档模式传输, 等于-tropgd1
2	-v	#详细模式输出, 打印速率, 文件数量等
3	-z	#传输时进行压缩以提高效率
4	-r	#递归传输目录及子目录, 即目录下得所有目录都同样传输。
5	-t	#保持文件时间信息
6	-o	#保持文件属主信息
7	-p	#保持文件权限
8	-g	#保持文件属组信息
9	-l	#保留软连接
10	-P	#显示同步的过程及传输时的进度等信息
11	-D	#保持设备文件信息
12	-L	#保留软连接指向的目标文件
13	-e	#使用的信道协议, 指定替代rsh的shell程序 ssh
14	--exclude=PATTERN	#指定排除不需要传输的文件模式
15	--exclude-from=file	#文件名所在的目录文件
16	--bwlimit=100	#限速传输
17	--partial	#断点续传
18	--delete	#让目标目录和源目录数据保持一致