

25. rsync

笔记本： 优秀笔记

创建时间： 2018/12/29 15:20

更新时间： 2018/12/29 15:26

作者： 306798658@qq.com

1. rsync工具介绍

Linux系统下有很多数据备份工具，常用的是rsync，从字面意思理解为remote sync（远程同步）。rsync不仅可以远程同步数据（类似于scp），而且可以本地同步数据（类似于cp），但不同于cp或者scp的一点是，它不会覆盖以前的数据（如果数据已经存在），而是先判断已经存在的数据和新数据的差异，只有数据不同时才会把不相同的部分覆盖。

Rsync是一款快速且功能非常多的文件复制工具。它可以在本地复制，通过任何远程shell或从另一台主机复制到/从远程rsync守护进程。它提供了大量的选项控制其行为的每个方面并允许非常灵活指定要复制的文件集。它是著名的其增量转换算法，可减少发送的数据量通过发送只有来源之间的差异在网络上文件和目标中的现有文件。Rsync很广泛用于备份和镜像以及作为改进的复制命令日常使用。

安装rsync (yum install -y rsync)

```
[root@damozhiying ~]# rsync -av /etc/passwd /tmp/1.txt //把/etc/passwd文件拷贝到/tmp下为1.txt
sending incremental file list
passwd

sent 1,747 bytes received 35 bytes 3,564.00 bytes/sec
total size is 1,655 speedup is 0.93
```

上面命令中：

- a参数中包含了很多选项，后面会详细介绍
- v查看到可视化过程
 - 查看到发送了多少个字节
 - 多少字节每秒
 - 文件一共有多大
 - 速度是多少

如果要改成远程复制，数据备份的格式是： 用户名@IP: path,比如192.168.112.136: /root/.具体用法如下：

```
[root@damozhiying ~]# rsync -av /etc/passwd root@192.168.112.136:/tmp/1.txt
The authenticity of host '192.168.112.136 (192.168.112.136)' can't be established.
ECDSA key fingerprint is SHA256:ZQLXi+kierWi2t64Yc5vUhPPWkMub8f0CBjnYRlX2Iw.
ECDSA key fingerprint is MD5:ff:9f:37:87:81:89:fc:ed:af:c6:62:c6:32:53:7a:ad.
Are you sure you want to continue connecting (yes/no)? yes //确实是否连接
Warning: Permanently added '192.168.112.136' (ECDSA) to the list of known hosts.
root@192.168.112.136's password: //输入密码
sending incremental file list
```

```
sent 45 bytes received 12 bytes 3.08 bytes/sec //此时出现详细同步的信息
total size is 1,655 speedup is 29.04
[root@damozhiying ~]#
```

注意：这里要求远程的机器也必须安装有rsync,需要验证密码是因为这里没有做两机互联.

一些常用的命令语法格式：

- rsync [OPTION]... SRC DEST
- rsync [OPTION]... SRC [USER@]host:DEST // 第一例不加user@host, 默认的就是root
- rsync [OPTION]... [USER@]HOST:SRC DEST//从远程目录同步数据到本地
- rsync [OPTION]... [USER@]HOST::SRC DEST
- rsync [OPTION]... SRC [USER@]HOST::DEST

里面SRC是源文件，DEST是目标目录或者文件；[USER@]HOST是远程服务器，host代表IP，方括号可以省略；

2. rsync常用选项

rsync是一个功能非常强大的工具，其命令也有很多功能特色选项。

- -a 包含-rtplgoD参数选项；
- -r 同步目录时要加上，类似cp时的-r选项；
- -v 同步时显示一些信息，让我们知道同步的过程；
- -l 保留软连接；若是拷贝的原目录里面有一个软链接文件，那这个软链接文件指向到了另外一个目录下。在加上-l，它会把软链接文件本身拷贝到目标目录里面去
- -L 加上该选项后，同步软链接时会把源文件给同步；
- -p 保持文件的权限属性；
- -o 保持文件的属主；
- -g 保持文件的属组；
- -D 保持设备文件信息；/dev/sdb1 这样的设备文件有它的特殊性，如果不加-D 可能拷贝过去就是一个非常普通的文件，不能当设备来用
- -t 保持文件的时间属性；
- --delete 删除DEST中SRC没有的文件；
- --exclude 过滤指定文件，如--exclude "logs"会把文件名包含logs的文件或者目录过滤掉，不同步；
- -P 显示同步过程，比如速率，比-v更加详细；
- --progress 同-P功能一样；
- -u 加上该选项后，如果DEST中的文件比SRC新，则不同步；
- -z 传输时压缩；

上述选项较多，常用的有：a,-v,-z,--delete和--exclude

3. rsync通过ssh方式同步

把文件同步到远程机器

```
[root@damozhiying ~]# rsync -av /etc/passwd 192.168.112.138:/tmp/ying.txt
root@192.168.112.138's password:
sending incremental file list
passwd

sent 1,747 bytes received 35 bytes 209.65 bytes/sec
total size is 1,655 speedup is 0.93
```

在ying02机器 (192.168.112.138) 上, 查看同步情况

```
[root@ying02 ~]# cat /tmp/ying.txt|head
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
[root@ying02 ~]# ip addr show ens33: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
qdisc pfifo_fast state UP qlen 1000
link/ether 00:0c:29:c6:2c:24 brd ff:ff:ff:ff:ff:ff
inet 192.168.112.138/24 brd 192.168.112.255 scope global ens33
valid_lft forever preferred_lft forever
inet6 fe80::964f:be22:ddf2:54b7/64 scope link
valid_lft forever preferred_lft forever
```

把远程文件拉回来

```
[root@damozhiying ~]# rsync -avP 192.168.112.138:/tmp/ying.txt /tmp/123.txt //把远程的文件搬回到本机
root@192.168.112.138's password:
receiving incremental file list
ying.txt
1,655 100% 1.58MB/s 0:00:00 (xfr#1, to-chk=0/1)

sent 43 bytes received 1,749 bytes 210.82 bytes/sec
total size is 1,655 speedup is 0.92
```

假如对方的机器端口不是22;

```
[root@damozhiying ~]# rsync -avP -e "ssh -p 22" /etc/passwd 192.168.112.138:/tmp/ying.txt //指定端口, -e后面的命令
root@192.168.112.138's password:
sending incremental file list

sent 45 bytes received 12 bytes 12.67 bytes/sec
```

```
total size is 1,655 speedup is 29.04
```

ssh -p 22 是一个命令，可以直接连接对方机器；

```
[root@damozhiying ~]# ssh -p 22 192.168.112.138root@192.168.112.138's password:
Last login: Mon Jun 18 10:49:34 2018 from 192.168.112.1
```

"--rsh=ssh -p 22" 也可以指定端口；注意与 -e "ssh -p 22" 的区别

```
[root@damozhiying ~]# rsync -P ./123.txt 192.168.112.138:/root/
root@192.168.112.138's password:
123.txt
152 100% 0.00kB/s 0:00:00 (xfr#1, to-chk=0/1)
[root@damozhiying ~]# rsync -P -e "ssh -p 22" ./520.txt 192.168.112.138:/root/ // -e "ssh -p
22" 指定端口
root@192.168.112.138's password:
520.txt
68 100% 0.00kB/s 0:00:00 (xfr#1, to-chk=0/1)
[root@damozhiying ~]# rsync -P "--rsh=ssh -p 22" ./999.txt 192.168.112.138:/root/ //"--
rsh=ssh -p 22" 指定端口
root@192.168.112.138's password:
999.txt
0 100% 0.00kB/s 0:00:00 (xfr#1, to-chk=0/1)
```

4. rsync通过服务同步

这种方式可以理解为：在远程主机上建立一个rsync服务器，在服务器上配置好rsync的各种应用，然后将本机作为rsync的一个客户端连接远程rsync服务器，下面是步骤说明：

1) 通过设置etc/rsyncd.conf演示

把下面文件写入配置文件/etc/rsyncd.conf

```
[root@damozhiying ~]# vim /etc/rsyncd.conf //编辑文件

port=873
log file=/var/log/rsync.log
pid file=/var/run/rsyncd.pid
address=192.168.112.136
[test]
path=/tmp/rsync
use chroot=true
max connections=4
read only=no
list=true
uid=root
gid=root
auth users=test
secrets file=/etc/rsyncd.passwd
hosts allow=192.168.112.138
```

启动rsync服务

```
[root@damozhiying ~]# rsync --daemon //启动rsync服务
[root@damozhiying ~]# ps aux |grep rsync
root 1317 0.0 0.0 114740 568 ? Ss 09:22 0:00 rsync --daemon
root 1343 0.0 0.0 112720 980 pts/0 S+ 09:24 0:00 grep --color=auto rsync
[root@damozhiying ~]# netstat -lntp //监听IP, 端口
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name
tcp 0 0 0.0.0.0:22 0.0.0.0:* LISTEN 688/sshd
tcp 0 0 127.0.0.1:25 0.0.0.0:* LISTEN 930/master
tcp 0 0 192.168.112.136:873 0.0.0.0:* LISTEN 1317/rsync
tcp6 0 0 :::22 :::* LISTEN 688/sshd
tcp6 0 0 ::1:25 :::* LISTEN 930/master
```

建立试验目录, 并给予777权限

```
[root@damozhiying ~]# mkdir /tmp/rsync
[root@damozhiying ~]# chmod 777 /tmp/rsync
[root@damozhiying ~]# ls -ld !$
ls -ld /tmp/rsync
drwxrwxrwx 2 root root 6 6月 19 10:14 /tmp/rsync
```

telnet 192.168.112.136 873 检测端口是否通畅

```
[root@ying02 ~]# telnet 192.168.112.136 873
Trying 192.168.112.136...
telnet: connect to address 192.168.112.136: No route to host
```

关闭firewalld

```
[root@damozhiying ~]# systemctl stop firewalld
[root@damozhiying ~]# iptables -nvLChain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
```

在ying02客户端上给服务器写文件;

```
[root@ying02 ~]# rsync -avP /tmp/ying.txt 192.168.112.136::test/ying02.txt
Password:
@ERROR: auth failed on module test //配置文件中auth错误
```

```
rsync error: error starting client-server protocol (code 5) at main.c(1648) [sender=3.1.2]
```

修改配置文件/etc/rsyncd.conf

```
port=873
log file=/var/log/rsync.log
pid file=/var/run/rsyncd.pid
address=192.168.112.136
[test]
path=/tmp/rsync
use chroot=true
max connections=4
read only=no
list=true
uid=root
gid=root
#auth users=test //定义用户名; 加上#, 意味着不生效#secrets file=/etc/rsyncd.passwd //定义密码;
加上#, 意味着不生效
hosts allow=192.168.112.138
```

再次执行传输, 此时不需要, 密码

```
[root@ying02 ~]# rsync -avP /tmp/ying.txt 192.168.112.136::test/ying02.txt
sending incremental file list
ying.txt
1,655 100% 0.00kB/s 0:00:00 (xfr#1, to-chk=0/1)

sent 1,749 bytes received 35 bytes 3,568.00 bytes/sec
total size is 1,655 speedup is 0.93
```

查看传输结果

```
[root@damozhiying ~]# ls /tmp/rsync/
ying02.txt
```

2) 配置文件详解

etc/rsyncd.conf文件内容释义:

- port : 指定在哪个端口启动rsyncd服务, 默认是873端口。
- log file : 指定日志文件。
- pid file : 指定pid文件, 这个文件的作用涉及服务的启动、停止等进程管理操作。
- address : 指定启动rsyncd服务的IP。假如你的机器有多个IP, 就可以指定由其中一个启动rsyncd服务, 如果不指定该参数, 默认是在全部IP上启动。
- [] : 指定模块名, 里面内容自定义。
- path : 指定数据存放的路径。

- use chroot true|false : 表示在传输文件前首先chroot到PATH参数所指定的目录下。这样做的原因是实现额外的安全防护, 但缺点是需要以roots权限, 并且不能备份指向外部的符号连接所指向的目录文件。默认情况下chroot值为true, 如果你的数据当中有软连接文件, 建议设置成false。
- max connections : 指定最大的连接数, 默认是0, 即没有限制。
- read only ture|false : 如果为true, 则不能上传到该模块指定的路径下。
- list : 表示当用户查询该服务器上的可用模块时, 该模块是否被列出, 设定为true则列出, false则隐藏。
- uid/gid : 指定传输文件时以哪个用户/组身份传输。
- auth users 指定传输时要使用的用户名。
- secrets file : 指定密码文件, 该参数连同上面的参数如果不指定, 则不使用密码验证。注意该密码文件的权限一定要是600。用户名:密码
- hosts allow : 表示被允许连接该模块的主机, 可以是IP或者网段, 如果是多个, 中间用空格隔开。刚才提到了选项use chroot,默认为ture,首先在主机的130的/root/rsync/test1/目录下创建软链接文件:

- use chroot=true|false 当有软连接, 需要给其定义为false;

第一步: 在服务端, 建立一个软连接

```
[root@damozhiying ~]# cd /tmp/rsync/
[root@damozhiying rsync]# ls
ying02.txt
[root@damozhiying rsync]# ln -s /etc/passwd ./12.txt //建立软连接文件
[root@damozhiying rsync]# ls -l
总用量 4
lrwxrwxrwx 1 root root 11 6月 19 13:53 12.txt -> /etc/passwd //软连接文件创建成功
-rw-r--r-- 1 root root 1655 6月 10 07:49 ying02.txt
[root@damozhiying rsync]# vim /etc/rsyncd.conf //修改配置文件
```

第二步: 按下图操作, 修改配置文件/etc/rsyncd.conf

```
port=873
log file=/var/log/rsync.log
pid file=/var/run/rsyncd.pid
address=192.168.112.136
[test]
path=/tmp/rsync
use chroot=true → 改为false
max connections=4
```

第三步: 在客户端ying02上, 开始同步test模块

```
[root@ying02 ~]# rsync -avLP 192.168.112.136::test/ /tmp/test/ //从服务端传输到test目录
receiving incremental file list
12.txt
1,655 100% 1.58MB/s 0:00:00 (xfr#1, to-chk=1/3)

sent 43 bytes received 1,787 bytes 3,660.00 bytes/sec
total size is 3,310 speedup is 1.81
[root@ying02 ~]# ls -l /tmp/test/
总用量 8
```

```
-rw-r--r--. 1 root root 1655 6月 10 07:49 12.txt //成功接收到刚创建的软件接文件12.txt
-rw-r--r--. 1 root root 1655 6月 10 07:49 ying02.txt
```

```
[root@ying02 ~]# cat /tmp/test/12.txt |head //查看12.txt文件
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
```

- max connections: 指定最大的连接数; 默认为0

```
max connections=4 //我们配置文件为设置为4
```

- read only ture/false : 如果为true, 则不能上传到该模块指定的路径下。

```
read only=no //如果为ture, 则不能给服务端写数据
```

- list: 该模块是否被列出, 设定为true则列出, false则隐藏。 (--port 8730 指定端口)

配置文件为: list=true

```
[root@ying02 ~]# rsync --port=873 192.168.112.136::test
```

更改配置文件, list=false

```
[root@ying02 ~]# rsync --port=873 192.168.112.136::[root@ying02 ~]#
```

- uid/gid : 指定传输文件时以哪个用户/组身份传输。

```
[root@damozhiying ~]# ls -l /tmp/rsync/
总用量 4
lrwxrwxrwx 1 root root 11 6月 19 13:53 12.txt -> /etc/passwd //配置文件为root, 则传输文件也为root
-rw-r--r-- 1 root root 1655 6月 10 07:49 ying02.txt
```

-auth users 指定传输时要使用的用户名;


```

port=873
log file=/var/log/rsync.log
pid file=/var/run/rsyncd.pid
address=192.168.112.136
[test]
path=/tmp/rsync
use chroot=false
max connections=4
read only=no
list=false
uid=root
gid=root
auth users=test
secrets file=/etc/rsyncd.passwd
hosts allow=192.168.112.138

```

指定用户名

指定密码文件

- secrets file : 指定密码文件;

第一步:新建rsyncd.passwd文件, 并且设置600权限;

```

[root@damozhiying ~]# vim /etc/rsyncd.passwd

test:ying //书写格式, 用户名: 密码

[root@damozhiying ~]# chmod 600 !$ //指定600权限

```

第二步: 在客户端ying02, 同步test目录;此时不需要输入密码

```

[root@ying02 ~]# rsync -avP /tmp/test/ --port=873 test@192.168.112.136::test/ //注意用户要使用用户名
sending incremental file list
12.txt //同步的时候不需要输入密码
1,655 100% 0.00kB/s 0:00:00 (xfr#1, to-chk=1/3)

sent 1,787 bytes received 35 bytes 3,644.00 bytes/sec
total size is 3,310 speedup is 1.82

```

第三步:在客户端ying02, 设置密码文件

```

[root@ying02 ~]# vi /etc/rsync_pass.txt //在文本里面只输入 密码
[root@ying02 ~]# chmod 600 !$
chmod 600 /etc/rsync_pass.txt
[root@ying02 ~]# rsync -avP /tmp/test/ --port=873 --password-file=/etc/rsync_pass.txt
test@192.168.112.136::test/
sending incremental file list

```

```
sent 85 bytes received 12 bytes 194.00 bytes/sec
total size is 3,310 speedup is 34.12
[root@ying02 ~]# touch /tmp/test/3.txt //新建1个文件
[root@ying02 ~]# rsync -avP /tmp/test/ --port=873 --password-file=/etc/rsync_pass.txt
test@192.168.112.136::test/
sending incremental file list
./
3.txt //同步它，也不需要输入密码
0 100% 0.00kB/s 0:00:00 (xfr#1, to-chk=1/4)

sent 150 bytes received 38 bytes 376.00 bytes/sec
total size is 3,310 speedup is 17.61
```

- hosts allow=IP 定义哪个IP连接它

```
hosts allow=192.168.112.136 1.1.1.1 192.168.112.138 2.2.2.2.2 //指定多个IP,用空格隔开
```

```
hosts allow=192.168.133.0/24 //定义IP段
```