



链滴

# rsync 使用介绍

作者: [qq1072023998](#)

原文链接: <https://ld246.com/article/1606375411602>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

# 1.rsync简介

rsync是一种备份工具。支持增量备份和差异备份。rsync有三种备份方式，第一种是本地备份类似与c命令，第二种是远程备份，类似与scp，使用的是ssh传输通道（需要进行ssh身份验证）。第三种是守护进程模式，需要有一台linux主机作为服务器监听端口，其他主要通过服务器的ip和端口进行访问（份）操作

## 2.rsync用法

### 2.1选项

rsync有许多选项：

- n: 在不确定命令是否能按意愿执行时，务必要事先测试；-n可以完成此功能；
- v: --verbose, 详细输出模式
- q: --quiet, 静默模式
- c: --checksum, 开启校验功能，强制对文件传输进行校验
- r: --recursive, 递归复制；
- a: --archives, 归档，保留文件的原有属性
- p: --perms 保留文件的权限
- t: --times 保留文件的时间戳
- l: --links 保留文件的符号链接
- g: --group 保留文件的属组
- o: --owner 保留文件的属主
- D: --devices 保留设备文件
- H, --hard-links 保留硬链接；
- S, --sparse 对稀疏文件进行特殊处理以节省DST的空间；
- e ssh: 表示使用ssh协议作承载
- z: 对文件压缩后传输
- progress: 显示进度条
- stats: 显示如何执行压缩和传输
- delete: 删除那些DST中有而SRC没有的文件，即删除DST中多余的文件

### 2.2命令格式

```
<pre>Local: rsync [OPTION...] SRC... [DEST]
```

Access via remote shell:

```
Pull: rsync [OPTION...] [USER@]HOST:SRC... [DEST]
```

```
Push: rsync [OPTION...] SRC... [USER@]HOST:DEST
```

Access via rsync daemon:

```
Pull: rsync [OPTION...] [USER@]HOST::SRC... [DEST]
      rsync [OPTION...] rsync://[USER@]HOST[:PORT]/SRC... [DEST]
Push: rsync [OPTION...] SRC... [USER@]HOST::DEST
      rsync [OPTION...] SRC... rsync://[USER@]HOST[:PORT]/DEST</pre>
```

由此语法可知，rsync有三种工作方式：

- (1).本地文件系统上实现同步。命令行语法格式为上述"Local"段的格式。
- (2).本地主机使用远程shell和远程主机通信。命令行语法格式为上述"Access via remote shell"段的格式。
- (3).本地主机通过网络套接字连接远程主机上的rsync daemon。命令行语法格式为上述"Access via rsync daemon"段的格式。

前两者的本质是通过管道通信，即使是远程shell。而方式(3)则是让远程主机上运行rsync服务，使其听在一个端口上，等待客户端的连接。

## 2.3 配置文件

rsync配置文件一般为/etc/rsyncd.conf:

```
fake super = yes
uid = rsync
gid = rsync
use chroot = no
max connections = 10
timeout = 600
motd file = /var/rsyncd/rsyncd.moth
pid file = /var/rsyncd/rsync.pid
lock file = /var/run/rsync.lock
log file = /var/rsyncd/rsyncd.log
dont compress = *.gz *.tgz *.zip *.z *.Z *.rpm *.deb *.bz2
[backup]
path = /backup
ignore errors
read only = false
write only = false
list = false
auth users = rsyncd_backup
secrets file = /etc/rsyncd.passwd
```

### rsyncd.conf配置文件常用参数说明

## rsyncd.conf参数

uid=rsync

gid=rsync

use chroot=no

传输文件前“chroot to the path”。这是一种安全配置，因为我们大多数都在内网，所以不配也没系

max connections=200

意思无限制，负值为关闭这个模块

timeout=400

00-600 (5-10分钟)

pid file

文件。如果这个文件存在，rsync不会覆盖该文件，而是会终止

lock file

s”参数，使得总连接数不会超过限制

log file

出相关日志信息

ignore errors

read only=false

默认对所有模块为true

list=false

默认为可以

hosts allow

址或地址段，默认情况没有此参数，即都可以连接

hosts deny

址或地址段，默认情况没有此参数，即都可以连接

auth users

哪些模块，用户不需要在本地系统中存在。默认为所有用户无密码访问

secrets file

用户名；密码，密码不超过8位

[backup]

起名称没有特殊要求，但最好是有意义的名称，便于以后维护

path

，目录的权限要注意和配置文件中的权限一致，否则会遇到读写的问题

## 参数说明

rsync使用的用户。

rsync使用的用户组（用户所在的组）

如果为true，daemon会在客户

设置最大连接数，默认0

默认为0，表示no timeout，建议

rsync daemon启动后将其进程pid写入

指定lock文件用来支持“max connectio

不设或者设置错误，rsync会使用rsyslog

忽略I/O错误

指定客户端是否可以上传文件

是否允许客户端可以查看可用模块列表

指定可以联系的客户端主机名或和ip

指定不可以联系的客户端主机名或ip

指定以空格或逗号分隔的用户可以使

指定用户名和密码存放的文件，格式

这里就是模块名称，需用中括号扩起来

这个模块中，daemon使用的文件系统或目

## 3.实验

### 3.1 本地传输

文件同步

```
rsync -avz /etc/fstab /data/
```

目录同步

```
rsync -avz /data /opt
```

data 后面加/表示将data下的内容同步过去 不加/表示将data目录同步过去

--delete 让目标目录与原目录数据同步

```
[root@145 ~]# rsync -avz /tmp/tmp.tar.gz /root
sending incremental file list
tmp.tar.gz

sent 378 bytes  received 35 bytes  826.00 bytes/sec
total size is 10,240  speedup is 24.79
```

## 3.2 远程传输

```
Last login: Sat Nov 21 18:43:02 2020 from 192.168.65.1
[root@192 ~]# rsync -avz root@192.168.65.132:/tmp /root
receiving incremental file list
tmp/
tmp/1.txt
tmp/2.txt
tmp/3.txt
tmp/4.txt
tmp/5.txt
tmp/6.txt
tmp/.ICE-unix/
tmp/.Test-unix/
tmp/.X11-unix/
tmp/.XIM-unix/
tmp/.font-unix/
tmp/systemd-private-df3df9299ea64e9d9e5288b79f2df7ed-chronyd.service-UGWnn1/
tmp/systemd-private-df3df9299ea64e9d9e5288b79f2df7ed-chronyd.service-UGWnn1/tmp/
tmp/systemd-private-df3df9299ea64e9d9e5288b79f2df7ed-vgauthd.service-EnIFWB/
tmp/systemd-private-df3df9299ea64e9d9e5288b79f2df7ed-vgauthd.service-EnIFWB/tmp/
tmp/systemd-private-df3df9299ea64e9d9e5288b79f2df7ed-vmtoolsd.service-q3svEj/
tmp/systemd-private-df3df9299ea64e9d9e5288b79f2df7ed-vmtoolsd.service-q3svEj/tmp/
tmp/systemd-private-df3df9299ea64e9d9e5288b79f2df7ed-vmtoolsd.service-q3svEj/tmp/vmware-ro
ot/

sent 289 bytes  received 825 bytes  2228.00 bytes/sec
total size is 25  speedup is 0.02
```

拉数据

```
[root@192 ~]# rsync -avz /tmp root@192.168.65.132:/root
sending incremental file list
tmp/
tmp/yum_save_tx.2020-11-21.19-08.VHI0fi.yumtx
tmp/yum_save_tx.2020-11-21.21-21.8eurSR.yumtx
tmp/yum_save_tx.2020-11-21.21-21.R0vRks.yumtx
tmp/.ICE-unix/
tmp/.Test-unix/
tmp/.X11-unix/
tmp/.XIM-unix/
tmp/.font-unix/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-chronyd.service-VBYaFu/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-chronyd.service-VBYaFu/tmp/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-httpd.service-J3en8g/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-httpd.service-J3en8g/tmp/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-vgauthd.service-dhcfrd/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-vgauthd.service-dhcfrd/tmp/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-vmtoolsd.service-sNdNDQ/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-vmtoolsd.service-sNdNDQ/tmp/
tmp/systemd-private-1ef3c56d98ef4900aa5ef48ea36190ef-vmtoolsd.service-sNdNDQ/tmp/vmware-ro
ot/
tmp/systemd-private-62e936eba1d8481b9dd6136a3e356dcf-chronyd.service-HABMeh/
tmp/systemd-private-62e936eba1d8481b9dd6136a3e356dcf-chronyd.service-HABMeh/tmp/
tmp/systemd-private-62e936eba1d8481b9dd6136a3e356dcf-vgauthd.service-1Ck6be/
tmp/systemd-private-62e936eba1d8481b9dd6136a3e356dcf-vgauthd.service-1Ck6be/tmp/
tmp/systemd-private-62e936eba1d8481b9dd6136a3e356dcf-vmtoolsd.service-pgkjab/
```

推数据

备注：需要注意的是，远程传输方式使用的是ssh传输通道，ssh传输通道本身传输数据加密的，是安全的，但也有几个缺点：

- 1.用户权限问题
- 2.用户身份暴露（安全隐患）
- 3.文件路径暴露（安全隐患）

tcp传输过程，每传输一个数据都会有连接的建立，维护，断开过程。所以传输多个文件会比较慢且两边都需要安装rsync程序。

第一次传输需要进行密钥确定，且每次登陆需要输入密码，如不想输入密码，可生成密钥并发送到远主机。

```
[root@192 ~]# ssh 180.76.146.54
The authenticity of host '180.76.146.54 (180.76.146.54)' can't be established.
ECDSA key fingerprint is SHA256:11Daqs22bj8hhXFCbAdH20L8p0aV7Cd5FAGZMidMNVY.
ECDSA key fingerprint is MD5:04:86:44:7d:ab:43:9c:c9:0d:41:d6:0f:ed:34:6a:ae.
Are you sure you want to continue connecting (yes/no)?
```

```
[root@192 ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)?
[root@192 ~]# ssh-copy-id root@180.76.146.54
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '180.76.146.54 (180.76.146.54)' can't be established.
ECDSA key fingerprint is SHA256:11Daqs22bj8hhXFCbAdH20L8p0aV7Cd5FAGZMidMNVY.
ECDSA key fingerprint is MD5:04:86:44:7d:ab:43:9c:c9:0d:41:d6:0f:ed:34:6a:ae.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any th
at are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it
is to install the new keys
root@180.76.146.54's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@180.76.146.54'"
and check to make sure that only the key(s) you wanted were added.

[root@192 ~]#
```

使用ssh-keygen生成密钥，并使用ssh-copy-id 发送至远程主机

此时传输数据便不会提示输入密码（ssh登陆亦是如此）

### 3.3 守护进程方式

守护进程方式需要一台机器作为server端，启动rsync daemon程序，监听端口，通过ip+socket来进行交互。

- 1.rpm -ql rsync查看rsync是否有安装（有的rsync把rsync与rsync-daemon分开，需要注意下）

```
[root@192 ~]#  
[root@192 ~]# rpm -ql rsync  
/etc/rsyncd.conf  
/etc/sysconfig/rsyncd  
/usr/bin/rsync  
/usr/lib/systemd/system/rsyncd.service  
/usr/lib/systemd/system/rsyncd.socket  
/usr/lib/systemd/system/rsyncd@.service  
/usr/share/doc/rsync-3.0.9
```

如果没有的话则需要安装

```
yum -y install rsync-daemon rsync
```

2.编辑配置文件

```
vim /etc/rsyncd.conf
```

3.创建程序用户

```
useradd -M -s /sbin/nologin rsync
```

```
chown -R rsync.rsync /backup
```

4.创建目录

```
mkdir /backup && chown -R rsync.rsync /backup
```

5.创建虚拟用户认证文件

```
echo "rsync_backup:1" > /etc/rsync.password
```

```
chmod 600 rsync.password
```

6.启动守护进程

```
systemctl start rsyncd
```