

10 Practical Examples of Rsync Command in Linux

Rsync (Remote Sync) is the most commonly used command for [copying and synchronizing files and directories remotely](#) as well as **locally** in **Linux/Unix** systems.

With the help of the **rsync** command, you can copy and synchronize your data remotely and locally across directories, disks, and networks, perform data backups, and [mirror between two Linux machines](#).

This article explains **10** basic and advanced usage of the **rsync** command to transfer your files remotely and locally in **Linux-based** machines. You don't need to be a **root** user to run the **rsync** command.

Some Advantages and Features of Rsync Command

- It efficiently copies and sync files to or from a remote system.
- Supports copying links, devices, owners, groups, and permissions.
- It's faster than [scp \(Secure Copy\)](#) because **rsync** uses a remote-update protocol which allows transferring just the differences between two sets of files. The first time, it copies the whole content of a file or a directory from source to destination but from next time, it copies only the changed blocks and bytes to the destination.
- Rsync consumes less [bandwidth utilization](#) as it uses compression and decompression method while sending and receiving data on both ends.

The basic syntax of the rsync command

```
# rsync options source destination
```

Some common options used with rsync commands

- **-v** : verbose
- **-r** : copies data recursively (but don't preserve timestamps and permission while transferring data).
- **-a** : archive mode, which allows copying files recursively and it also preserves symbolic links, file permissions, user & group ownerships, and timestamps.
- **-z** : compress file data.
- **-h** : human-readable, output numbers in a human-readable format.

[You might also like: [How to Sync Files/Directories Using Rsync with Non-standard SSH Port](#)]

Install Rsync in Linux System

We can install the **rsync** package with the help of the following command in your Linux distribution.

```
$ sudo apt-get install rsync      [On Debian/Ubuntu & Mint]
$ pacman -S rsync                 [On Arch Linux]
$ emerge sys-apps/rsync           [On Gentoo]
$ sudo dnf install rsync          [On Fedora/CentOS/RHEL and Rocky Linux/AlmaLinux]
$ sudo zypper install rsync       [On openSUSE]
```

1. Copy/Sync Files and Directory Locally

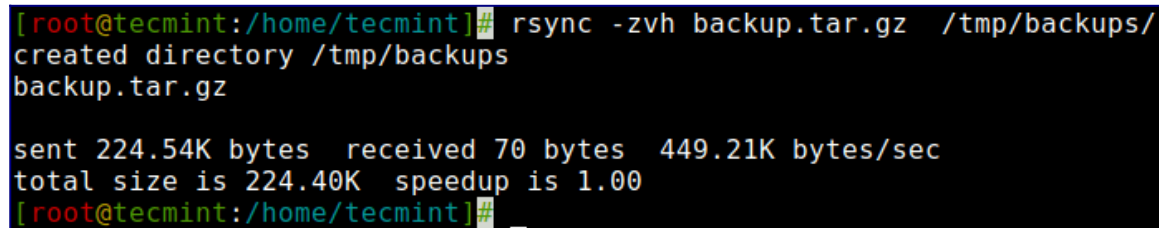
The following command will sync a single file on a local machine from one location to another location. Here in this example, a file name **backup.tar** needs to be copied or synced to **/tmp/backups/** folder.

```
[root@tecmint]# rsync -zvh backup.tar.gz /tmp/backups/

created directory /tmp/backups
backup.tar.gz

sent 224.54K bytes  received 70 bytes  449.21K bytes/sec
total size is 224.40K  speedup is 1.00
```

In the above example, you can see that if the destination is not already existed rsync will create a directory automatically for the destination.

A terminal window with a dark background and light-colored text. It shows the execution of the command 'rsync -zvh backup.tar.gz /tmp/backups/'. The output indicates that a directory was created at /tmp/backups and the file backup.tar.gz was transferred. The transfer statistics show 224.54K bytes sent, 70 bytes received, and a speedup of 1.00. The prompt at the end is '[root@tecmint:/home/tecmint]# _'.

```
[root@tecmint:/home/tecmint]# rsync -zvh backup.tar.gz /tmp/backups/
created directory /tmp/backups
backup.tar.gz

sent 224.54K bytes  received 70 bytes  449.21K bytes/sec
total size is 224.40K  speedup is 1.00
[root@tecmint:/home/tecmint]# _
```

Copy/Sync a Directory on Local Computer

The following command will transfer or sync all the files from one directory to a different directory in the same machine. Here in this example, **/root/rpmpkgs** contains some rpm package files and you want that directory to be copied inside **/tmp/backups/** folder.

```
[root@tecmint]# rsync -avzh /root/rpmpkgs /tmp/backups/

sending incremental file list
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aa6c63b.x86_64.rpm
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aa6c63b.x86_64.rpm
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm

sent 3.47M bytes  received 96 bytes  2.32M bytes/sec
total size is 3.74M  speedup is 1.08
```

```
[root@tecmint:~]# rsync -avzh /root/rpmpkgs /tmp/backups/
sending incremental file list
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm

sent 3.47M bytes  received 96 bytes  2.32M bytes/sec
total size is 3.74M  speedup is 1.08
[root@tecmint:~]# _
```

2. Copy/Sync Files and Directory to or From a Server

This command will sync a directory from a local machine to a remote machine. **For example**, there is a folder in your local computer “**rpmpkgs**” that contains some **RPM** packages and you want that local directory’s content sends to a remote server, you can use the following command.

```
[root@tecmint:~]# rsync -avzh /root/rpmpkgs root@192.168.0.141:/root/
```

```
The authenticity of host '192.168.0.141 (192.168.0.141)' can't be established.
ED25519 key fingerprint is SHA256:bH2tiWQn4S5o6qmZhmtXcBROV5TU5H4t2C42QDEMx1c.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.141' (ED25519) to the list of known hosts.
root@192.168.0.141's password:
sending incremental file list
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm

sent 3.74M bytes  received 96 bytes  439.88K bytes/sec
total size is 3.74M  speedup is 1.0
```

```
[root@tecmint:~]# rsync -avzh /root/rpmpkgs root@192.168.0.141:/root/
The authenticity of host '192.168.0.141 (192.168.0.141)' can't be established.
ED25519 key fingerprint is SHA256:bH2tiWQn4S5o6qmZhmtXcBROV5TU5H4t2C42QDEMx1c.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.141' (ED25519) to the list of known hosts.
root@192.168.0.141's password:
sending incremental file list
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm

sent 3.74M bytes  received 96 bytes  439.88K bytes/sec
total size is 3.74M  speedup is 1.00
[root@tecmint:~]# _
```

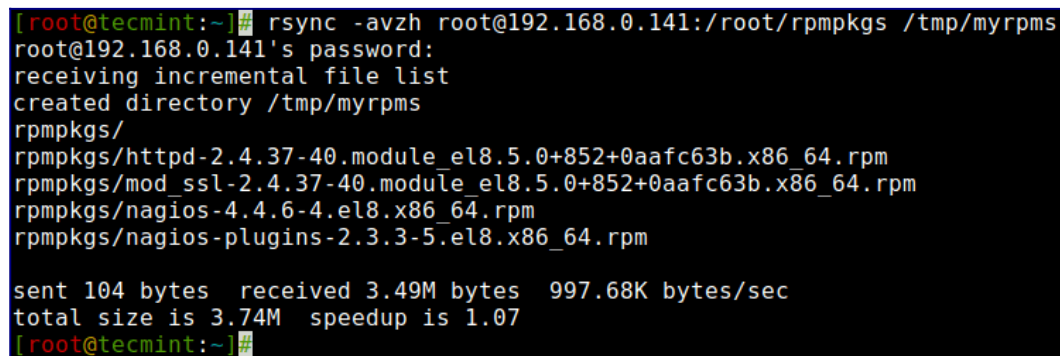
Copy/Sync a Remote Directory to a Local Machine

This command will help you sync a remote directory to a local directory. Here in this example, a directory **/root/rpmpkgs** which is on a remote server is being copied in your local computer in **/tmp/myrpms**.

```
[root@tecmin:~]# rsync -avzh root@192.168.0.141:/root/rpmpkgs /tmp/myrpms
```

```
root@192.168.0.141's password:
receiving incremental file list
created directory /tmp/myrpms
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aafc63b.x86_64.rpm
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aafc63b.x86_64.rpm
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm
```

```
sent 104 bytes  received 3.49M bytes  997.68K bytes/sec
total size is 3.74M  speedup is 1.07
```



```
[root@tecmin:~]# rsync -avzh root@192.168.0.141:/root/rpmpkgs /tmp/myrpms
root@192.168.0.141's password:
receiving incremental file list
created directory /tmp/myrpms
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aafc63b.x86_64.rpm
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aafc63b.x86_64.rpm
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm

sent 104 bytes  received 3.49M bytes  997.68K bytes/sec
total size is 3.74M  speedup is 1.07
[root@tecmin:~]# _
```

3. Rsync Over SSH

With **rsync**, we can use **SSH (Secure Shell)** for data transfer, using **SSH** protocol while transferring our data you can be ensured that your data is being transferred in a secured connection with encryption so that nobody can read your data while it is being transferred over the wire on the internet.

[You might also like: [How to Secure and Harden OpenSSH Server](#)]

Also when we use **rsync** we need to provide the **user/root** password to accomplish that particular task, so using the **SSH** option will send your logins in an encrypted manner so that your **password** will be safe.

Copy a File from a Remote Server to a Local Server with SSH

To specify a protocol with **rsync** you need to give the “-e” option with the protocol name you want to use. Here in this example, We will be using the “**ssh**” with the “-e” option and perform data transfer.

```
[root@tecmin:~]# rsync -avzhe ssh root@192.168.0.141:/root/anaconda-ks.cfg /tmp
```

```
root@192.168.0.141's password:
receiving incremental file list
anaconda-ks.cfg
```

```
sent 43 bytes  received 1.10K bytes  325.43 bytes/sec
total size is 1.90K  speedup is 1.67
```

```
[root@tecmint:~]# rsync -avzhe ssh root@192.168.0.141:/root/anaconda-ks.cfg /tmp
root@192.168.0.141's password:
receiving incremental file list
anaconda-ks.cfg

sent 43 bytes  received 1.10K bytes  325.43 bytes/sec
total size is 1.90K  speedup is 1.67
[root@tecmint:~]# _
```

Copy a File from a Local Server to a Remote Server with SSH

```
[root@tecmint:~]# rsync -avzhe ssh backup.tar.gz root@192.168.0.141:/backups/

root@192.168.0.141's password:
sending incremental file list
created directory /backups
backup.tar.gz

sent 224.59K bytes  received 66 bytes  64.19K bytes/sec
total size is 224.40K  speedup is 1.00
```

```
[root@tecmint:~]# rsync -avzhe ssh backup.tar.gz root@192.168.0.141:/backups/
root@192.168.0.141's password:
sending incremental file list
created directory /backups
backup.tar.gz

sent 224.59K bytes  received 66 bytes  64.19K bytes/sec
total size is 224.40K  speedup is 1.00
[root@tecmint:~]# _
```

[You might also like: [How to Use Rsync to Sync New or Changed/Modified Files in Linux](#)]

4. Show Progress While Transferring Data with rsync

To show the progress while transferring the data from one machine to a different machine, we can use the **‘-progress’** option. It displays the files and the time remaining to complete the transfer.

```
[root@tecmint:/]# rsync -avzhe ssh --progress /root/rpmpkgs
root@192.168.0.141:/root/rpmpkgs

root@192.168.0.141's password:
sending incremental file list
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aafc63b.x86_64.rpm
      1.47M 100%   31.80MB/s   0:00:00 (xfr#1, to-chk=3/5)
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aafc63b.x86_64.rpm
      138.01K 100%    2.69MB/s   0:00:00 (xfr#2, to-chk=2/5)
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
      2.01M 100%   18.45MB/s   0:00:00 (xfr#3, to-chk=1/5)
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm
      120.48K 100%    1.04MB/s   0:00:00 (xfr#4, to-chk=0/5)

sent 3.74M bytes  received 96 bytes  1.50M bytes/sec
total size is 3.74M  speedup is 1.00
```

```
[root@tecmint:/]# rsync -avzhe ssh --progress /root/rpmpkgs root@192.168.0.141:/root/rpmpkgs
root@192.168.0.141's password:
sending incremental file list
rpmpkgs/
rpmpkgs/httpd-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
  1.47M 100% 31.80MB/s 0:00:00 (xfr#1, to-chk=3/5)
rpmpkgs/mod_ssl-2.4.37-40.module_el8.5.0+852+0aaafc63b.x86_64.rpm
  138.01K 100% 2.69MB/s 0:00:00 (xfr#2, to-chk=2/5)
rpmpkgs/nagios-4.4.6-4.el8.x86_64.rpm
  2.01M 100% 18.45MB/s 0:00:00 (xfr#3, to-chk=1/5)
rpmpkgs/nagios-plugins-2.3.3-5.el8.x86_64.rpm
  120.48K 100% 1.04MB/s 0:00:00 (xfr#4, to-chk=0/5)

sent 3.74M bytes received 96 bytes 1.50M bytes/sec
total size is 3.74M speedup is 1.00
[root@tecmint:/]# _
```

5. Use of `--include` and `--exclude` Options

These two options allow us to **include** and **exclude** files by specifying parameters with these option helps us to specify those files or directories which you want to include in your sync and exclude files and folders with you don't want to be transferred.

Here in this example, the rsync command will include those files and directory only which starts with '**R**' and exclude all other files and directory.

```
[root@tecmint:/]# rsync -avze ssh --include 'R*' --exclude '*' root@192.168.0.141:/var/lib/rpm/ /root/rpm
```

```
root@192.168.0.141's password:
receiving incremental file list
created directory /root/rpm
./
```

Requirerename

```
sent 61 bytes received 273,074 bytes 60,696.67 bytes/sec
total size is 761,856 speedup is 2.79
```

```
[root@tecmint:/]# rsync -avze ssh --include 'R*' --exclude '*' root@192.168.0.141:/var/lib/rpm/
root@192.168.0.141's password:
receiving incremental file list
created directory /root/rpm
./
Requirerename

sent 61 bytes received 273,074 bytes 60,696.67 bytes/sec
total size is 761,856 speedup is 2.79
[root@tecmint:/]# _
```

6. Use of **--delete** Option

If a file or directory does not exist at the source, but already exists at the destination, you might want to delete that existing file/directory at the target while syncing.

We can use the '**--delete**' option to delete files that are not there in the source directory.

Source and target are in sync. Now create a new file **test.txt** at the target.

```
[root@tecmint:~]# cd /root/rpm/
[root@tecmint:~/rpm]# touch test.txt
[root@tecmint:~/rpm]# rsync -avz --delete root@192.168.0.151:/var/lib/rpm/
/root/rpm/
```

```
root@192.168.0.151's password:
receiving incremental file list
deleting test.txt
```

```
./
.dbenv.lock
.rpm.lock
Basenames
Conflictname
Dirnames
Enhancename
Filetriggername
Group
Installtid
Name
Obsoletename
Packages
Providename
Shalheader
Sigmd5
Suggestname
Supplementname
Transfiletriggername
Triggername
__db.001
__db.002
__db.003
```

```
sent 445 bytes  received 18,543,954 bytes  2,472,586.53 bytes/sec
total size is 71,151,616  speedup is 3.84
```

Target has the new file called **test.txt**, when synchronizing with the source with the '**--delete**' option, it removed the file **test.txt**.

```

[root@tecmin:/home/tecmin]# cd /root/rpm/
[root@tecmin:~/rpm]# touch test.txt
[root@tecmin:~/rpm]# rsync -avz --delete root@192.168.0.151:/var/lib/rpm/ /root/rpm/
root@192.168.0.151's password:
receiving incremental file list
deleting test.txt
./
.dboenv.lock
.rpm.lock
Basenames
Conflictname
Dirnames
Enhancename
Filetriggername
Group
Installtid
Name
Obsoletename
Packages
Providename
Shalheader
Sigmd5
Suggestname
Supplementname
Transfiletriggername
Triggername
__db.001
__db.002
__db.003

sent 445 bytes  received 18,543,954 bytes  2,472,586.53 bytes/sec
total size is 71,151,616  speedup is 3.84

```

7. Set the Max Size of Files to be Transferred

You can specify the **Max** file size to be transferred or sync. You can do it with the “**--max-size**” option. Here in this example, the Max file size is **200k**, so this command will transfer only those files which are equal to or smaller than **200k**.

```

[root@tecmin:~]# rsync -avzhe ssh --max-size='200k' /var/lib/rpm/
root@192.168.0.151:/root/tmprpm

```

```

root@192.168.0.151's password:
sending incremental file list
created directory /root/tmprpm
./
.dboenv.lock
.rpm.lock
Conflictname
Enhancename
Filetriggername
Group
Installtid
Name
Obsoletename
Recommendname
Requirename
Shalheader
Sigmd5

```

```
Suggestname  
Supplementname  
Transfiletriggername  
Triggername  
__db.002
```

```
sent 129.52K bytes  received 396 bytes  28.87K bytes/sec  
total size is 71.15M  speedup is 547.66
```

```
[root@tecmint:~]# rsync -avzhe ssh --max-size='200k' /var/lib/rpm/ root@192.168.0.151:
root@192.168.0.151's password:
sending incremental file list
created directory /root/tmprpm
./
.dbenv.lock
.rpm.lock
Conflictname
Enhancename
Filetriggername
Group
Installtid
Name
Obsoletename
Recommendname
Requirename
Shalheader
Sigmd5
Suggestname
Supplementname
Transfiletriggername
Triggername
__db.002

sent 129.52K bytes  received 396 bytes  28.87K bytes/sec
total size is 71.15M  speedup is 547.66
[root@tecmint:~]# _
```

8. Automatically Delete source Files After Successful Transfer

Now, suppose you have the main web server and a data backup server, you created a daily backup and synced it with your backup server, now you don't want to keep that local copy of backup in your web server.

So, will you wait for the transfer to complete and then delete that local backup file manually? Of Course NO. This automatic deletion can be done using the **'--remove-source-files'** option.

```
[root@tecmint:~]# rsync --remove-source-files -zvh backup.tar.gz  
root@192.168.0.151:/tmp/backups/
```

```
root@192.168.0.151's password:  
backup.tar.gz
```

```
sent 795 bytes  received 2.33K bytes  894.29 bytes/sec  
total size is 267.30K  speedup is 85.40
```

```
[root@tecmint:~]# ls -l backup.tar.gz
```

```
ls: cannot access 'backup.tar.gz': No such file or directory
```

```
[root@tecmint:~]# rsync --remove-source-files -zvh backup.tar.gz root@192.168.0.151:/tmp/backups/
root@192.168.0.151's password:
backup.tar.gz

sent 795 bytes  received 2.33K bytes  894.29 bytes/sec
total size is 267.30K  speedup is 85.40
[root@tecmint:~]# ls -l backup.tar.gz
ls: cannot access 'backup.tar.gz': No such file or directory
[root@tecmint:~]# _
```

9. Do a Dry Run with rsync

If you are a newbie using rsync and don't know what exactly your command going to do. Rsync could really mess up the things in your destination folder and then doing an undo can be a tedious job.

[You might also like: [How to Sync Two Apache Web Servers/Websites Using Rsync](#)]

Use of this option will not make any changes to the files and shows the output of the command, if the output shows exactly the same you want to do then you can remove the '**--dry-run**' option from your command and run on the terminal.

```
[root@tecmint:~]# rsync --dry-run --remove-source-files -zvh backup.tar.gz
root@192.168.0.151:/tmp/backups/
```

```
root@192.168.0.151's password:
backup.tar.gz
```

```
sent 50 bytes  received 19 bytes  19.71 bytes/sec
total size is 267.30K  speedup is 3,873.97 (DRY RUN)
```

```
[root@tecmint:~]# rsync --dry-run --remove-source-files -zvh backup.tar.gz root
root@192.168.0.151's password:
backup.tar.gz

sent 50 bytes  received 19 bytes  19.71 bytes/sec
total size is 267.30K  speedup is 3,873.97 (DRY RUN)
[root@tecmint:~]# _
```

10. Rsync Set Bandwidth Limit and Transfer File

You can set the bandwidth limit while transferring data from one machine to another machine with the help of '**--bwlimit**' option. This option helps us to limit **I/O** bandwidth.

```
[root@tecmint]# rsync --bwlimit=100 -avzhe ssh /var/lib/rpm/
root@192.168.0.151:/root/tmprpm/
root@192.168.0.151's password:
sending incremental file list
sent 324 bytes  received 12 bytes  61.09 bytes/sec
total size is 38.08M  speedup is 113347.05
```

Also, by default rsync syncs changed blocks and bytes only, if you want explicitly want to sync the whole file then you use the ‘**-W**’ option with it.

```
[root@tecmint]# rsync -zvhW backup.tar /tmp/backups/backup.tar
backup.tar
sent 14.71M bytes  received 31 bytes  3.27M bytes/sec
total size is 16.18M  speedup is 1.10
```

That’s all with rsync now, you can see **man pages** for more options. Stay connected with **Tecmint** for more exciting and interesting tutorials in the future. Do leave your **comments** and **suggestions**.