

Rsync

Simple exact copy

Including all permissions, time, etc.

```
rsync -az --verbose --itemize-changes --times --recursive --whole-file --  
delete --links root@192.168.178.80:/source_dir/ /target_dir
```

Notice the '/' behind the source_dir

Output flags

Each bit position and value in rsync's output:

```
YXcstpoguax  path/to/file  
|||||||  
|||||||- x: The extended attribute information changed  
|||||||-- a: The ACL information changed  
|||||||- u: The u slot is reserved for future use  
|||||||- g: Group is different  
|||||||- o: Owner is different  
|||||- p: Permission are different  
|||||- t: Modification time is different  
|||- s: Size is different  
|||- c: Different checksum (for regular files), or  
      changed value (for symlinks, devices, and special files)  
  
|- the file type:  
  f: for a file,  
  d: for a directory,  
  L: for a symlink,  
  D: for a device,  
  S: for a special file (e.g. named sockets and fifos)  
  
|- the type of update being done::  
  <: file is being transferred to the remote host (sent)  
  >: file is being transferred to the local host (received)  
  c: local change/creation for the item, such as:  
    - the creation of a directory  
    - the changing of a symlink,  
    - etc.  
  h: the item is a hard link to another item (requires  
    --hard-links).  
  .: the item is not being updated (though it might have  
    attributes that are being modified)
```

*: means that the rest of the itemized-output area contains a message (e.g. "deleting")

Some example output from rsync for various scenarios:

```
>f++++++ some/dir/new-file.txt
.f....og..x some/dir/existing-file-with-changed-owner-and-group.txt
.f.....x some/dir/existing-file-with-changed-unnamed-attribute.txt
>f...p....x some/dir/existing-file-with-changed-permissions.txt
>f..t...g..x some/dir/existing-file-with-changed-time-and-group.txt
>f.s.....x some/dir/existing-file-with-changed-size.txt
>f.st.....x some/dir/existing-file-with-changed-size-and-time-stamp.txt
cd++++++ some/dir/new-directory/
.d....og... some/dir/existing-directory-with-changed-owner-and-group/
.d..t..... some/dir/existing-directory-with-different-time-stamp/
```

Full backup

```
rsync -haxAX --stats --delete --info=progress2 --info=name0 /* "$TargetMnt"
--exclude={"/dev/*,/proc/*,/sys/*,/tmp/*,/run/*,/mnt/*,/media/*,/lost+found}
```

Some notable comments about parameters:

--stats gives information on files added, changed and deleted from the clone (called backup in your case).

--delete instructs rsync to delete files in the clone that no longer exist in the source directory.

info=progress2 gives a modern looking progress display whilst cloning directories.

--info=name0 prevents every single filename from being displayed as it is being copied. This gives less screen clutter but you may want to omit this parameter.

/* tells rsync where to start synchronizing files. In this example it's the root directory but you want to change it to /media/user1/DATA4/FolderA.

"\$TargetMnt" tells rsync where to clone to. In your case change it to "/media/ivan/Seagate Backup Plus Drive/FolderA". The double quotes are important because your directory names contain spaces in them.

The second line starting with --exclude={"/dev/* you don't need at all because these directories aren't in the list. Don't use this line and drop the line continuation character \ at the end of the first line.

As with all backup scenarios always test the backups to make sure all files are there and contain the appropriate information.

From:
<https://wiki.oscardegrout.nl/> - **HomeWiki**



Permanent link:
<https://wiki.oscardegrout.nl/doku.php?id=linux:system:backup-clone:rsync&rev=1694169476>

Last update: **2023/09/08 10:37**