rsync: Versatile Data Copy

Table of Contents

Overview

When it comes to copying, updating, or in some other way moving your data around, there really is little that can compare to **rsync**.

As was discussed during the Aeolus batch/training seminars, we suggest using **rsync** to upload and manage your data within Aeolus. **cp** and **mv** don't always maintain permissions, attributes, file stability, et cetera while moving data across filesystem types or servers (especially across the network). If you are not satisfied with "*cp -au*", we suggest again **rsync**. In this tutorial, we will also be covering the use of **screen** a lite terminal emulator and **tail** a terminal printer. When in doubt, use/check the man page(s).

Learn about the Commands

Consider the following **rsync** options:

- h : output numbers in human readable format
- a : archive or "-rlptgoD" for recursion, permissions, etc
- i : output a change summary of all changes
- v : increase verbosity
- z : compress file data during the transfer
- --dry-run : perform a trial run with no changes made
- --progress : print information showing progress of transfer

Consider the following **screen** options:

- S : when creating a new session, specify a name
- R : attempt to resume first detached session it sees
- ls : prints a list of screen sessions

Consider the following **tail** options:

- f : output appended data as the file grows

Run the Commands

The basic rsync command holds the following syntax:

```
rsync  [from] [to]
rsync  [source] [target]
```

Putting that all together, you should be able to do something along the lines of:

```
[aeolus #]: screen -S syncdata
[aeolus #]: mkdir ~/logs
[aeolus #]: rsync -haiv --progress /fastscratch/myDIR/mysubdir /data/myLAB/users/myDIR/ > ~/logs/syncdata.log
```

> **NOTE:** trailing forward slash "/" will grab data in a directory or move data into a directory
>
> **NOTE:** no trailing forward slash "/" will grab the directory or write on top of a directory

By typing **CTRL + a**, then separately **"d"** to detach the screen session, you will be returned to your previous shell.

To watch the progress of the data copy:

```
[aeolus #]: tail -f ~/logs/syncdata.log
```

Finish with the Commands

By typing **CTRL + "c"**, you can kill the appending tail stream/view.

To list out active screen sessions, you can type the following:

```
[aeolus #]: screen -ls
```

Then to re-attach your previous screen session:

```
[aeolus #]: screen -R syncdata
```

Once in, by typing **CTRL + "c"**, you will kill the current re-attached screen session.

> **NOTE:** It will be up to you to check your log file for errors. Configured as above, rsync will not be able to copy files with read-only permissions (as it has to be written somewhere).

Bonus rsync Commands

> **Warning**
>
> **WARNING:** These are to only be exercised with extreme caution. If you get the paths wrong, rsync will do what you said and likely purge unintended data (that you have access to).

If the data that you were copying has been updated (more or less written to it) since you started the copy or your last copy, that's not an issue! You can just run an rsync as described above and then manually handle duplicate data, or you can use -*-delete-delay* in your rsync command.

- --delete-delay : find data that no longer exists in source and delete it from the target after the transfer

```
[aeolus #]: rsync -haiv --delete-delay --progress /fastscratch/myDIR/mysubdir /data/myLAB/users/myDIR/ > ~/logs
/syncdata.log
```

Additionally, you can remove the data from the source, similarly to "mv" but better.

- --remove-source-files : sender removes synchronized files that are part of the transfer

```
[aeolus #]: rsync -haiv --remove-source-files --progress /fastscratch/myDIR/mysubdir /data/myLAB/users/myDIR/ >
~/logs/syncdata.log
```

Using rsync over SSH (to replace scp)

Syntax is similar to scp, you add **`user@aeolus.wsu.edu:`** to either the source or destination you want to be remote. As an example, here it is as the destination for the first example from the beginning of the document. Don't forget to replace **`user`** with the user on the remote system that you want to connect as.

```
[aeolus #]: rsync -haiv --progress /fastscratch/myDIR/mysubdir user@aeolus.wsu.edu:/data/myLAB/users/myDIR/ > ~/logs/syncdata.log
```