GridFTP how-to (draft)

Note: this document is currently being extensively revised. There may be mismatches between older and newer versions until it is complete. Proceed at your own risk.

Overview:

GridFTP is a way of transferring large files quickly between computers which have it installed and enabled. To use gridFTP to transfer large files among ORNL, NERSC, and NCAR (or other sites), you must have access to systems at these sites which have the Globus toolkit installed. You will either need to create an Globus Online account to use the web-GUI transfer, or obtain a certificate from the Open Science Grid (OSG) certificate authority and install it on these systems to use the globus-url-copy command line interface, or both. (There are situations in which one or the other may be preferable.) If you plan to transfer data to/from the ORNL systems, you will need to obtain an OSG certificate in any event.

There is some useful documentation at ORNL at: https://www.olcf.ornl.gov/kb_articles/transferring-data-with-gridftp/ which explains the process of obtaining an OSG certificate and importing it to the ORNL computers.

Step 0: make sure you have accounts on the appropriate computers.

To use gridFTP, the gridFTP software must be installed on the machines you are using at both ends of your data transfer. At ORNL and NERSC this is easy; if you have a login on any system at these sites, you will have a login on the data transfer systems, dtn03.ccs.ornl.gov and dtn04.ccs.ornl.gov (NOTE: as of May 2013, only dtn03 and 04 may be used for data transfers), and dtn01.nersc.gov, dtn02.nersc.gov, dtn03.nersc.gov, dtn04.nersc.gov, and garchive.nersc.gov (the HPSS; NOTE: as of February 2013, garchive is down indefinitely). The system intended for data transfer at NCAR is gridftp.ucar.edu, which shares /glade with yellowstone.ucar.edu. You can not connect directly to gridftp, but if you have a yellowstone account you can use the public ncar endpoint of Globus Online. If you are feeling adventurous, you can install gridFTP on your own system – see http://globus.org/toolkit/docs/4.2/4.2.1/admin/install/

You will also probably want a Globus Online account. Go to the signup page on the Globus Online website, http://globusonline.org/signup. It's pretty self explanatory.

Step 1: obtain Open Science Grid (OSG) certificate.

Note: this procedure is NEW as of March 2013!

Go to http://oim.grid.iu.edu/oim/certificate. On the left-hand side of the page there is a menu box. Under "User Certificates" select "Request New". For Affiliation (registration authority) choose ESGF. For Sponsor Information, enter Gary Strand, strandwg@ucar.edu, 303-497-1336. When you submit the form, it will ask you to choose a GRID passphrase to protect the private key. You will receive email from "Open Science Grid FootPrints" <osg@tick.globalnoc.iu.edu> in a day or so with a link to your certificate.

Open the link and click on the button on the bottom to download your certificate and save it on your machine. It will be a .p12 file named something like user_certificate_and_key.p12.

Step 2: import your certificate.

You will need to import your certificate on a machine at each site you plan to use for command-line transfers. You will need to import your certificate at ORNL whether you plan to use command-line transfers or Globus Online.

Create a $HOME/.globus directory on the remote machine and copy your .p12 file to that machine (I use the example name file.p12 below, but it doesn't matter what you name it). I did this using pscp (a program by the makers of putty, which I use to open terminal windows to remote machines) – you can obtain it from http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html and information on how to use it is at http://tartarus.org/~simon/putty-snapshots/htmldoc/Chapter5.html#pscp. Then extract the certificate:

```
> openssl pkcs12 -in file.p12 -clcerts -nokeys -out $HOME/.globus/usercert.pem
```

To get the encrypted private key :

```
> openssl pkcs12 -in file.p12 -nocerts -out $HOME/.globus/userkey.pem
```

For both of these you will have to enter the passphrase you selected when you created your certificate. You will also have to create a passphrase to use for getting your credentials.

You must then set file permissions as follows:

```
> chmod 644 $HOME/.globus/usercert.pem

> chmod 600 $HOME/.globus/userkey.pem
```

Step 3: register your certificate.

Note: even if you have previously registered a certificate obtained from CA, you will need to register your new OSG certificate (as obtained in step 1)

ORNL:

Log on to dtn03.ccs.ornl.gov (or dtn04; I'll just refer to dtn03 from here on)

> module load globus

> register_globus_creds

You will receive an email once your certificate has been successfully registered, most likely after several hours or a day.

NERSC:

Log on to dtn01.nersc.gov (or dtn02, 03, 04; I'll just refer to dtn01 from here on)

> module load globus

> grid-cert-info -subject

Which will give you a line of text beginning /DC=org/DC=doegrids. Then enter

> grid-cert-info -issuer

Which will give you another similar line of text.

Log in to the NIM website at https://nim.nersc.gov. On your account summary page click on the Grid Certificates tab. Click on the "Add existing Grid Certificate to NIM" link on the bottom of the page, which will bring you to a page where you can enter the above two lines of text for subject and issuer – copy and paste them so they are correct! Again, you will have to wait for several hours to a day before the registration has been approved.

NCAR:

Contact cislhelp@ucar.edu and request that you be enabled to do gridFTP. You will need to send them some of the information from your certificate. The easiest way to do this is to generate your proxy at ORNL or NERSC (see below) and then enter: grid-proxy-info -s. Copy the output and paste it into your email to CISL. They will let you know when you've been authorized. (The person at CISL who usually handles GridFTP issues is Sidd Ghosh.)

Step 4: generate your proxy.

Again, this can be (slightly) different depending on which site you are using to initiate the transfer. I usually use the site that I am sending the data from, but you can also initiate the transfer from the receiving site, or even from a third party computer. One consideration: the ORNL servers reject certificates unless they were generated with the SecureID passcode, so all transfers to or from ORNL should be done from ORNL. (I have heard from one user that he has been able to transfer data from ORNL initiating the transfer from the receiving site, so this perhaps may no longer be true.)

You must have a valid proxy to copy data. Proxies are good for 12 hours.

ORNL:

Log on to dtn03.ccs.ornl.gov.

> module load globus

> grid-proxy-init

Enter the GRID passphrase you specified when generating your certificate. This will create your credentials good for 12 hours.

Alternatively - and you will need to do this to create a private endpoint in Globus Online:

> module load globus

> myproxy-init -n

Enter the GRID passphrase you specified when generating your certificate. This creates a proxy that is good for a week and you don't need to repeat this step for subsequent logins during that period. However, you still need to create your credentials:

> myproxy-logon

Enter your ORNL passcode (pin plus SecurID code), *not the GRID pass phrase* (the entry text is misleading). This creates credentials which are good for 12 hours.

NERSC:

Log on to dtn01.nersc.gov.

> module load globus

> grid-proxy-init

Enter the GRID passphrase you specified when generating your certificate. This will create your credentials good for 12 hours.

NCAR:

If you can, initiate the transfer from the other site, or use Globus Online. If you absolutely need to transfer from NCAR, at the moment you will need to do the following (this information from Davide Del Vento):

Step 5: transfer data.

Step 2: create endpoints if needed

An 'endpoint' is a server which is running gridFTP. Several sites have public endpoints, so you can use them, rather than creating your own. The public endpoint for NCAR is ncar#gridftp. NERSC also has several public endpoints; you will most likely use nersc#dtn for dtn01/02. NERSC does not have a public endpoint for garchive (direct transfer from the HPSS) so when garchive is operational (NOTE: as of February 2013 garchive is indefinitely down) it is generally more efficient to do command-line transfers from NERSC.

ORNL does not currently have public endpoints, so if you want to transfer data to or from ORNL you will need to create your own. Note that you will need to have an OSG certificate first; see steps 1-3 under the globus-url-copy procedure.

To create a private endpoint for ORNL, you will first need to create your proxy certificate on myproxy1.ccs.ornl.gov.

Create a private endpoint this by choosing the menu item Manage Data -> manage endpoints and then selecting 'add an endpoint'. Give it a name (your own username will be part of the name).

Okay, you have loaded the globus module and obtained your proxy. Now it's easy. Mostly.

If you're using gridFTP, you are probably transferring files which reside on the MSS or HPSS. For files originating at ORNL or NCAR, you'll have to retrieve the files to dtn01 or bluefire, respectively, which can take a lot of time if you are transferring many large files. For files originating at NERSC, you can transfer files directly from the HPSS by specifying garchive.nersc.gov as the originating computer. And of course the same applies for the destination – at ORNL and NCAR you will (probably) need to move the files to MSS/HPSS, while for NERSC you can move the files directly to the HPSS.

The command for moving data is globus-url-copy. There are various parameters which relate in some arcane way to efficiency. I have not noticed that changing them from the suggested values makes a huge difference. The basic syntax is:

> globus-url-copy *[parameters] source destination*

Wildcards and regex specification are allowed (enclose source in double quotes).

The parameters that seem to result in the fastest transfer are -tcp-bs 8M -bs 8M -p 8 -vb. The first two are buffer sizes on the remote and local system, the third specifies 8 parallel streams, and the fourth specifies verbose mode. I typically get transfer rates around 50-80 MB/sec with these parameters, except for garchive.nersc.gov transfers which are typically around 30-40 MB/sec.

(I got my best results with only flags -vb -p 16 - JPE)

There are two ways to specify the source and destination: gsiftp://, and file://. The latter may only be used when you are transferring from or to the system you're initiating the transfer from, but it appears to be slightly faster than the former. You can also specify a directory rather than a file (which is what I usually do for the destination). It must have a trailing slash. If the source is a directory, all the files in that directory will be moved. If you want a recursive move, you must use the flag --r. I have not tried moving an entire directory other than by use of wildcards.

Examples:

> globus-url-copy -tcp-bs 8M -bs 8M -p 8 -vb file://`pwd`/bigfile.tar gsiftp://gridftp.ucar.edu/ptmp/user/
This transfers a local file bigfile.tar (on local storage at NERSC or ORNL, and issued from that machine) in the current directory to the gridftp machine at NCAR. The user can then login on bluefire, cd to /ptmp/user, and use msrcp or hsi to move the file to the MSS/HPSS.

> globus-url-copy -tcp-bs 8M -bs 8M -p 8 -vb gsiftp://gridftp.ucar.edu/ptmp/user/bigfile2.nc file://`pwd`/
This is the reverse of the above, transferring a file bigfile2.nc that has been moved to the ptmp area on bluefire.ucar.edu to the machine which has issued the command, to the current directory.

> globus-url-copy -tcp-bs 8M -bs 8M -p 8 -vb gsiftp:// dtn01.ccs.ornl.gov/`pwd`/bigfile.tar gsiftp://gridftp.ucar.edu/ptmp/user/

This is the same as the first example, but using gsiftp:// rather than file://, and transferring from ORNL.
> globus-url-copy -tcp-bs 8M -bs 8M -p 8 -vb gsiftp://garchive.nersc.gov//home/u/user/bigfile3.tar gsiftp://gridftp.ucar.edu/ptmp/ilana/

This transfers a file directly from the NERSC HPSS to the gridftp machine at NCAR. Note that you can issue this command from dtn01.nersc.gov, bluefire.ucar.edu, *or* from dtn01.ccs.ornl.gov! In general you can do third-party transfers from one machine to another initiated from a third machine, *unless* either the 'from' or 'to' machine doesn't accept them or (like ORNL) has restrictions on proxy certificates.

I was able to use these commands directly from the jaguar and jaguarpf login nodes at ORNL :

> globus-url-copy -p 16 -vb -fast -stripe gsiftp://dtn01.ccs.ornl.gov/tmp/work/jedwards/mytest.tgz gsiftp://gridftp.ucar.edu/glade/scratch/jedwards/

Step 5a: optimization

For staging large amounts of data from an HPSS, or for transferring large amounts of data from NERSC's garchive, it is recommended to access the files in tape/position order. Informal testing suggests this improves speed by about a factor of 5.

A quick and dirty way to do this is to use the -P flag for hsi ls to generate a list including all the position information and redirect to a file. Then extract the tape number, position, and file name and write them to a second file in that order, and sort on the first two fields. This will not properly sort cases where there are files at positions with different numbers of digits (as it will sort e.g. 230, 24, 25, 251) but it's good enough for most cases, and you can quickly scan in an edit and fix this if needed. (Or write a perl script to sort properly - I just haven't done it yet.) You can then insert the rest of the URL codes and feed it to the globus-url-copy command with the -f option.

Example:

I'm frequently transferring a lot of related data, for example, the atm, ocn, ice, and lnd data for a single CCSM experiment. It's more efficient to process it all at once in four batches on the different dtn machines, or in groups of up to 1000 files or so as then files on the same tape can be grouped regardless of component. This is how I do it:

1. Get all the filename and position information

```
> hsi -q "ls -P /home/c/ccsm/csm/[experiment]/atm/hist/[experiment.cam2.h0.??-.nc" >& rawlist

> hsi -q "ls -P /home/c/ccsm/csm/[experiment]/atm/hist/[experiment.cam2.h0.??-.nc" >>& rawlist

> hsi -q "ls -P /home/c/ccsm/csm/[experiment]/atm/hist/[experiment.cam2.h0.??-.nc" >>& rawlist

> hsi -q "ls -P /home/c/ccsm/csm/[experiment]/atm/hist/[experiment.cam2.h0.??-.nc" >>& rawlist
```

The file rawlist will look like this:

```
FILE    /home/c/ccsm/csm/b.e11.BRCP26C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP26C5CN.f09_g16.2300.001.cice.h.2100-01.nc72792184
72792184        58      EP037000        2       0       1       03/03/2012      23:22:54        06/06/2012      16:19:34
FILE    /home/c/ccsm/csm/b.e11.BRCP26C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP26C5CN.f09_g16.2300.001.cice.h.2100-02.nc72792184
72792184        2344    EP036700        2       0       1       03/03/2012      23:22:55        06/06/2012      16:24:55
FILE    /home/c/ccsm/csm/b.e11.BRCP26C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP26C5CN.f09_g16.2300.001.cice.h.2100-03.nc72792184
72792184        2345    EP036700        2       0       1       03/03/2012      23:22:56        06/06/2012      16:24:58
FILE    /home/c/ccsm/csm/b.e11.BRCP26C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP26C5CN.f09_g16.2300.001.cice.h.2100-04.nc72792184
72792184        61      EP037000        2       0       1       03/03/2012      23:23:56        06/06/2012      16:29:08
```

2. Extract relevant fields and sort

```
> awk '
```

Unknown macro: {print $6,$5,$2}

```
' rawlist | sort --key=1,2 > sortlist
```

The file sortlist will look like this:

```
EP036000 10 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2103-06.nc
EP036000 11 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2103-08.nc
EP036000 12 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2103-11.nc
EP036000 13 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2104-01.nc
```

3. The sorted list might have several thousand files. Suppose it has 3000; I'll divide it into four batches of about 750 files each. I edit the file (I use vi) and chop it up by going to line 750, then to the next line which shows a new tape and saving everything ahead of it in a new file, then repeating through the whole thing. E.g. in the below segment I'd split after the first three files, and put the fourth in the next batch.

```
EP036100 10948 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2120-09.nc
EP036100 10949 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2120-11.nc
EP036100 10950 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2121-06.nc
EP036200 10000 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2120-05.nc
```

I usually name the files list1.raw, list2.raw etc.

4. I glance through the segments in vi and fix any sort failures, e.g.

```
EP036000 68 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2107-08.nc
EP036000 69 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2108-07.nc
EP036000 7 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2101-05.nc
EP036000 70 /home/c/ccsm/csm/b.e11.BRCP60C5CN.f09_g16.2300.001/ice/hist/b.e11.BRCP60C5CN.f09_g16.2300.001.cice.h.2108-08.nc
```

Yeah, I need to write a script to do this, or figure out how to sort on two keys at once in proper numeric sort. Or you can ignore these; it will still be more efficient than totally random access. Then I add the URL bits:

```
:1,$s=/home=gsiftp://garchive/home=g
:1,$s=nc=nc gsiftp://gridftp.ucar.edu/glade/data01/CMIP5/proc/ilana/xfer/=g
```

5. Strip the tape info and put into a new file

```
> cut --delim ' ' --fields 3 list1.raw > list1.do
```

6. transfer

```
> globus-url-copy -fast -f sortlist &
```

Note that the files are all going to a temporary directory. That's so they can be sorted out into their proper directories (atm/hist, ocn/hist etc) later. You can also check on the progress of the transfer by doing ls -l | grep -v [bytes in complete file] (with as many pipes and greps as needed for all the possibilities). When it's complete, of course, nothing will be returned (if all goes well).