JASMIN Workshop: Exercise 2: Copy data to JASMIN to share with collaborators

Scenario

I am working with international colleagues on the project "workshop". The project has a JASMIN Group Workspace that all collaborators have access to. I want to share a directory of data files, which I currently have on my local machine, with my colleagues

Objectives

- Copy a directory of data files from my local machine to the “workshop” Group Workspace on JASMIN

JASMIN resources

- Transfer server: jasmin-xfer1.ceda.ac.uk
- Group workspace: /gws/nopw/j04/workshop/$USER/ex2

Local resources

- A linux terminal
- Local directory: /disks/my-data/obs1

Instructions

1. Make a directory on your local machine and create 2 files in it
2. Start ssh-agent session and add JASMIN private key
3. SSH to a transfer server
4. Identify path and create target directory, then change to that directory
5. Exit back to local machine and recursively copy a directory of data
6. Log back in to transfer server to check permissions are correct

Review

This exercise demonstrates how to:

- Copy data to a transfer (“xfer”) server
- Check permissions on the data to make sure it’s visible by collaborators

This is a basic workflow suitable for small datasets or where speed is not critical. For larger data transfers or over longer distances (international / intercontinental), it is recommended to consider other available options which could be more efficient, depending on source & destination. See more information at: https://help.jasmin.ac.uk/article/219-data-transfer-overview

Alternative approaches and best practice

- Use other tools for more sophisticated copying/syncing
- Use a faster server for scp, rsync to achieve higher throughput (better)
- Use Globus, an efficient bulk data transfer service (best)
- Internal transfers from one JASMIN filesystem to another

Cheat sheet for Exercise 2: Copy data to JASMIN to share with collaborators

1. Make a directory on your local machine and create 2 files in it.

```
mkdir -p tmp/my-data
cd tmp/my-data
echo "this is a readme" > README.txt
echo "1 2 3 4 5" > numbers.txt
ls -l
```

2. Start ssh-agent session and add JASMIN private key

```
exec ssh-agent $SHELL
ssh-add ~/.ssh/id_rsa_jasmin
```

3. SSH to a transfer server

```
ssh <username>@jasmin-xfer1.ceda.ac.uk
```

4. Identify path and create target directory, then change to that directory

```
cd /group_workspaces/jasmin2/workshop
mkdir -p users/$USER/ex02
cd users/$USER/ex02
```

NOTE: you can use this command as-is: the `$USER` environment variable should resolve to your system username

5. Exit back to local machine and recursively copy a directory of data

```
(on jasmin-xfer1): exit
(on local machine): scp -r my-data
<username>@jasmin-xfer1.ceda.ac.uk:/group_workspaces/jasmin2/workshop/users/<us
ername>/ex02/
```

NOTE: this time you need to replace "`<username>`" with your JASMIN system account username, in both places

6. Log back in to transfer server to check permissions are correct

```
(on local machine): ssh <username>@jasmin-xfer1.ceda.ac.uk
(on jasmin-xfer1): cd /group_workspaces/jasmin2/workshop/users/$USER/ex02
(on jasmin-xfer1): ls -l
...
(on jasmin-xfer1): chmod -R g+rX my-data
(on jasmin-xfer1): ls -l
drwxr-x--- <username> gws_workshop 1234 27 Jun my-data
```

Alternative approaches and best practice - specific examples

- Use other tools for more sophisticated copying/syncing/moving
  - `rsync` can synchronise a local and remote directory
  - Use as a drop-in replacement for `scp` in step 4

```
(on local machine): rsync -rv my-data
<username>@jasmin-xfer1.ceda.ac.uk:/group_workspaces/jasmin2/workshop/users/
<username>/ex02/
```

  - Advantage: could re-run this and it would only copy those files which have changed ("changed" can be defined in different ways: time, size, checksum)
  - See man page for full set of options: `$ man rsync`

- Use a faster server for scp, rsync to achieve higher throughput
  - jasmin-xfer[23].ceda.ac.uk are similar machines but physical machine and located in "Data Transfer Zone"
  - Require additional "hpxfer" access role and IP address if connecting from outside
  - jasmin-xfer2 : tuned for shorter paths (UK/Europe/Eastern US)
  - jasmin-xfer3: tuned for longer paths (Western US/Australia/Far East)

- Use Globus, an efficient bulk data transfer service (recommended method)
  - Demo videos:
    - https://youtu.be/fQ-J8l0PLw8
      - Quick (<2min) quick demo of moving data from ARCHER/RDF to JASMIN)
    - https://youtu.be/f9tSdoFK0Jk
      - Longer (30min) overview of how to use Globus for data transfer on JASMIN. Note that this uses the old Globus web interface which has since been replaced with https://app.globus.org

  - A good investment of your time
    - It takes a little time to get started with Globus on JASMIN, but it's worth the effort.
    - Globus makes the task of moving data much easier and is usually the fastest and most reliable method, even (or in fact, particularly) for multi-Terabyte transfers between continents, but also works well on a local or smaller scale.
    - Features:
      - Manages transfers between dedicated transfer endpoints:
        - You don't need to be connected to either, once the transfer is underway
        - You can be notified of the status of a transfer
      - Web, command-line and Python APIs available

- Internal transfers from one JASMIN filesystem to another:
  - For LARGE internal transfers, you can use the "bcopy" transfer service, which initiates a set of LOTUS batch jobs to do your transfer for you.
  - Best followed up with an rsync to "mop up" any remaining files and/or permissions/ownership changes
  - See https://help.jasmin.ac.uk/article/3844-copy-service