Using the Hábrók HPC Cluster

An Overview

Lennart August

March 7, 2024

Logging In

- Access Hábrók via SSH: `ssh sNumber@login1.hb.hpc.rug.nl`
- Use your s-number password.
- Enter your authentication code.

Logging In from Windows

- Use PuTTY or another SSH client:
  - Download and install PuTTY from `https://www.putty.org/`.
  - Open PuTTY, enter `login1.hb.hpc.rug.nl` as the Host Name.
  - Enter your username and password when prompted.
- For key-based authentication, use PuTTYgen to generate and manage SSH keys.

Copying Data to Hábrók

- Use `scp` for secure file transfer:
  - From your local machine: `scp /path/to/local/file sNumber@login1.hb.hpc.rug.nl:/path/to/destination`
  - From Hábrók to local: `scp sNumber@login1.hb.hpc.rug.nl:/path/to/file /path/to/local/destination`
- Large transfers might benefit from using `rsync`.

Copying Data to and from Hábrók in Windows

- Use WinSCP for a graphical interface:
  - Download and install WinSCP from `https://winscp.net/eng/download.php`.
  - Open WinSCP and enter `login1.hb.hpc.rug.nl` as the Host Name, and your credentials.
  - Drag and drop files between the local and remote file systems.
- Alternatively, use PuTTY's pscp command in the command line:

```
# To copy from your local machine to Hábrók:
pscp C:\local\path\file
    sNumber@login1.hb.hpc.rug.nl:/remote/path
# To copy from Hábrók to your local machine:
pscp sNumber@login1.hb.hpc.rug.nl:/remote/path/file
    C:\local\path
```

Setting Up an Environment

- Load necessary modules for software and libraries.
- Example: `module load Python/3.10.4-GCCcore-11.3.0`
- Check available modules with `module avail [string]`. This lists all available modules that contain "string".
- Useful modules: `Python/3.10.4-GCCcore-11.3.0`, `TensorFlow/2.11.0-foss-2022a-CUDA-11.7.0`, `PyTorch/1.12.1-foss-2022a-CUDA-11.7.0`

Creating CPU Scripts

- Use SLURM for job submission: `sbatch my_script.sh`
- When you have a larger dataset: zip and copy your data to the local drive. (`cp /path/to/scratch/archrive.tar $TMPDIR/archive.tar`)
- Example script (`my_script.sh`):

```
#!/bin/bash
#SBATCH --job-name=test_cpu
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=4
#SBATCH --time=01:00:00
#SBATCH --mem=4G

module Python/3.10.4-GCCcore-11.3.0
python my_script.py
```

Creating GPU Scripts

- Specify GPU resources in your SLURM script.
- Example:

```
#!/bin/bash
#SBATCH --job-name=test_gpu
#SBATCH --gpus-per-node=v100:1
#SBATCH --time=03-00:00:00
#SBATCH --mem=124G
#SBATCH --cpus-per-task=8

module load Python/3.10.4-GCCcore-11.3.0
module load PyTorch/1.12.1-foss-2022a-CUDA-11.7.0

python my_gpu_script.py
```

- Use the `--gpus-per-node=gpu_type:⟨count⟩` option to request GPU resources. Possible GPU options are a100, a100.20gb, v100.