

SF2568: Instructions for Lab 1

Michael Hanke

January 20, 2020

Overview

The aim of this lab work is twofold,

- to become comfortable with the PDC environment and tegner and to run and modify a simple MPI program;
- to start with homework 1, problem 6.

Finding information

One of your most important skills is to be able to find information in order to learn and understand new things. For this lab your main sources of information are:

- `man`, `--help`, `-h`, `apropos`, ...
- Lecture notes:
<https://pdc-support.github.io/hpc-intro/>
- PDC support pages, to be found at
<https://www.pdc.kth.se/support/>
Note: The scheduling system on tegner is SLURM!
- <http://www.pdc.kth.se/>
- Internet

If you get stuck, have any problems with your account or in any other way want to check that you have grasped something in this lab correctly — please ask the course assistants for help! What you learn today is the basis for the following weeks of lab exercises.

Exercise

- Copy the files `hello.c` into your work directory.
- Compile and run the program `hello.c` using 4 processes. What are the results?
- Modify the program `hello.c` such that, instead of printing the acknowledgement, sends a message back to the master containing the worker's rank. Also modify the program so the master receives the messages and prints out the corresponding ranks to stdout.

How to work on tegner

1. If you are working from a KTH Ubuntu workstation, there are special commands available. They have usually the prefix `pdcc-`. Use them! A description can be found here:

https://www.pdc.kth.se/support/documents/login/kth_ubuntu_login.html

2. On a computer with a Kerberos software installed (your laptop, or for this lab a CSC workstation), make sure that you have forwardable Kerberos tickets:

```
klist -f
```

3. If you don't have forwardable Kerberos tickets, create them on your LOCAL COMPUTER using:

```
kinit --forwardable your-user-name-at-pdc@NADA.KTH.SE
```

(The capitalized domain is important!)

Note: Never type your password on any remote connection — only on your local computer!

4. Log in to PDC's tegner by issuing the command

```
ssh -Y your-user-name-at-pdc@tegn.pdc.kth.se
```

Check your Kerberos ticket and AFS tokens on the login node. Are they forwarded and forwardable?

5. Get the proper environment for doing the lab using the command

```
module add i-compilers intelmpi
```

6. Change to your work directory.

7. Compile your program by running `mpicc`.

8. Run your program using the command (one line!)

```
srun --nodes=1 -A edu20.sf2568 --reservation=2568.0127 -t 1
```

```
mpirun -np <#processes> 'pwd' /<your_program>
```

(' indicates an accent grave!)

Each node features (at least) 24 cores. Limiting the number of tasks per node to 24 switches off hyperthreading.

Note: srun will allocate a node for you and release it after your job has completed. It may take some time before your request will be served. Be patient! Another possibility is to preallocate nodes by `salloc`. You will release them via `exit`. Note that the reservation is only valid for this lab!

9. When your code is successfully running on interactive nodes, try submitting a run on dedicated nodes on tegner. On the login node do:

```
sbatch ./job.sh
```

It should be located in your working directory. Please have a look at it using an editor of your choice to see how to use it and how it works! `stdout` and `stderr` are redirected to a file `slurm-%jobid.out`.

10. Note: A hanging program can be interrupted using **Control-C**.
11. The program can be edited on your local Linux workstation using your favorite editor. The compilation (and execution, of course) must be done on tegner. For copying files to tegner use the `scp` command (or, `pdcp`). Technical details on how this should be done are provided at the web https://www.pdc.kth.se/support/documents/login/kth_ubuntu_login.html.