Lab 1

Introduction to Programming Laboratory

Goals

- Platform introduction
- Connect to server
- Compile & run a hello world program

Platform Introduction

Hardware

20 nodes

- Intel X5670 2x6 cores @ 2.93GHz
- 96GB Memory
- 5.5TB RAID5 disk
- QDR Infiniband

Software

- OS: Arch Linux lastupdate=1561870580
- Compilers: GCC 9.1.0, Clang 8.0.0
- MPI: MVAPICH-2.3.1
- Workload scheduler: 18.08.3

Available resources

- 1 login node (200% CPU max)
- 16 compute nodes (submit jobs via SLURM)
- 48GB disk space per user

Connect to server

Credentials

Address

ipl.cs.nthu.edu.tw

Username

(check iLMS)

Password

(check iLMS)

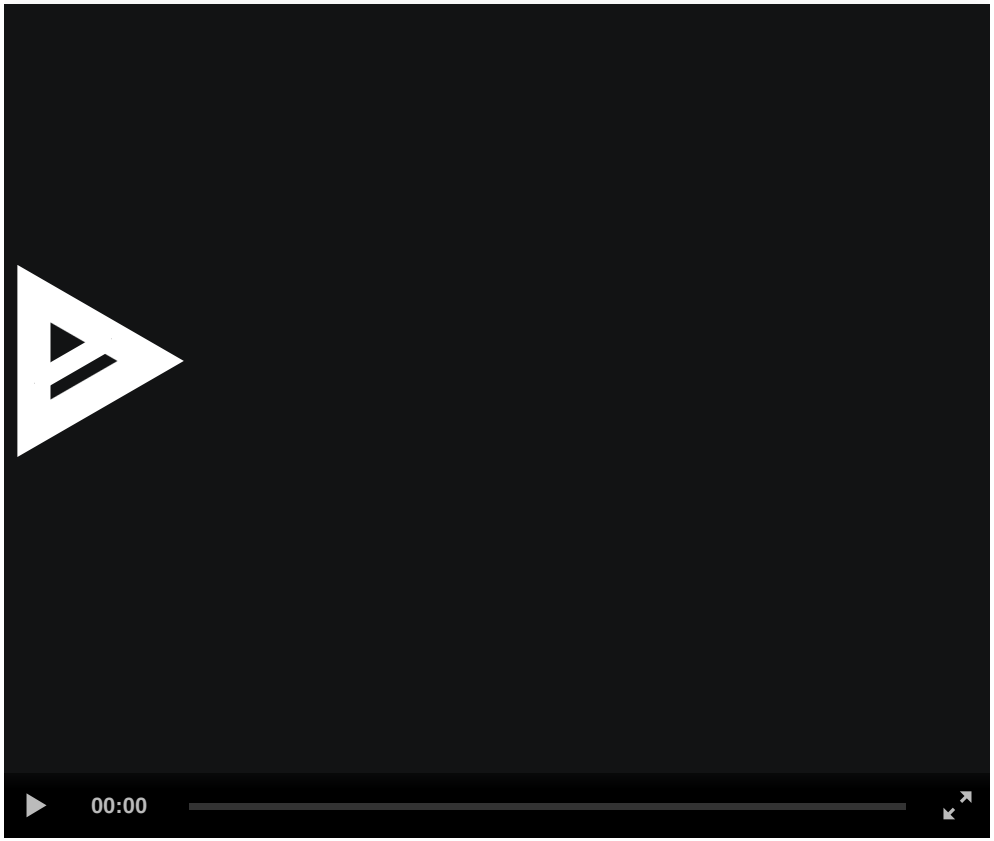
Login via SSH (Windows)

- **MobaXterm**
- **Putty**
- Or anything that works for you

Login via SSH (Unix)

1. Open Terminal
2. Type `ssh ipl19-XX@ipl.cs.nthu.edu.tw`
3. Enter password
4. You'll be ask to change your password on first login

SSH walkthrough



Compile & run a hello-world program

*Please demo with TA in class after you completed this task

Edit code

- vim
- emacs
- nano

We have these editors on the server.
If your preferred editor is not available, you can ask TA to install it for you.

Alternatively, you can edit code on your own computer and use `rsync` to transfer the code

Command examples for transfering files

Windows users

You can just drag and drop with MobaXterm

Transfer file to server

`rsync -ahPv *filename* ipl19-*XX*@ipl.cs.nthu.edu.tw:*remote-filename*`

Transfer file from server

`rsync -ahPv ipl19-*XX*@ipl.cs.nthu.edu.tw:*remote-filename* *filename*`

Compiling code

After editing your code, you can compile it with `g++`:

```
g++ -std=c++11 -O3 hello.cc -o hello
```

- `-std=c++11` we are using c++11
- `-O3` optimize level 3
- `hello.cc` the name of the source code
- `-o hello` the name of the output executable

Running code

Type `./hello` in your terminal.

Task: Hello world

1. Write a hello world program outputing `"Hello, world\n"`
2. Compile your code
3. Run your code
4. Show TA your results

Task: Hello world walkthrough