Introduction to Experimental Environment

PING XU

xuping908@gmail.com

Connecting to the cluster machines

Remote Controlling a Linux Server

- We use ssh protocal to remotely control a linux server.
- After we use SSH to login to a server, we are given a command line interface (terminal) to interact with the server:
  - The command you entered in a SSH session is executed remotely on the server
  - The output of running the command is sent back to your local machine and displayed in the command line interface, just as what you've expect from a local command line session.
- This is Linux's "remote desktop", though it is command line only.

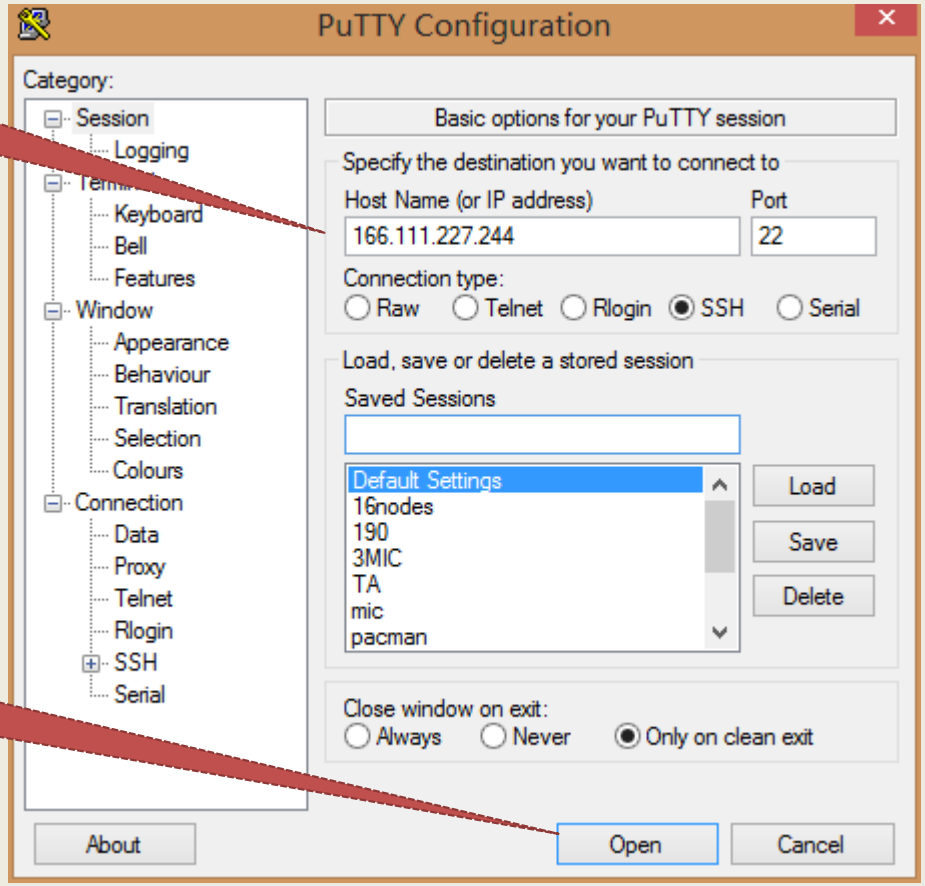
Log in to the server

- IP： 166.111.227.244
- User Id： Your Student ID Number (201xxxxxxx)
- Passwd： User Id@tsinghua (Don’t forget to change your password)
- Tool for log in： ssh (available in Linux and Mac)
  - On Windows, use putty (free): http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html or Tectia (commercial, https://www.ssh.com/products/tectia-ssh/)
- Tool for uploading & downloading files： scp
  - On Windows: WinSCP (http://winscp.net/eng/download.php) or Tectia

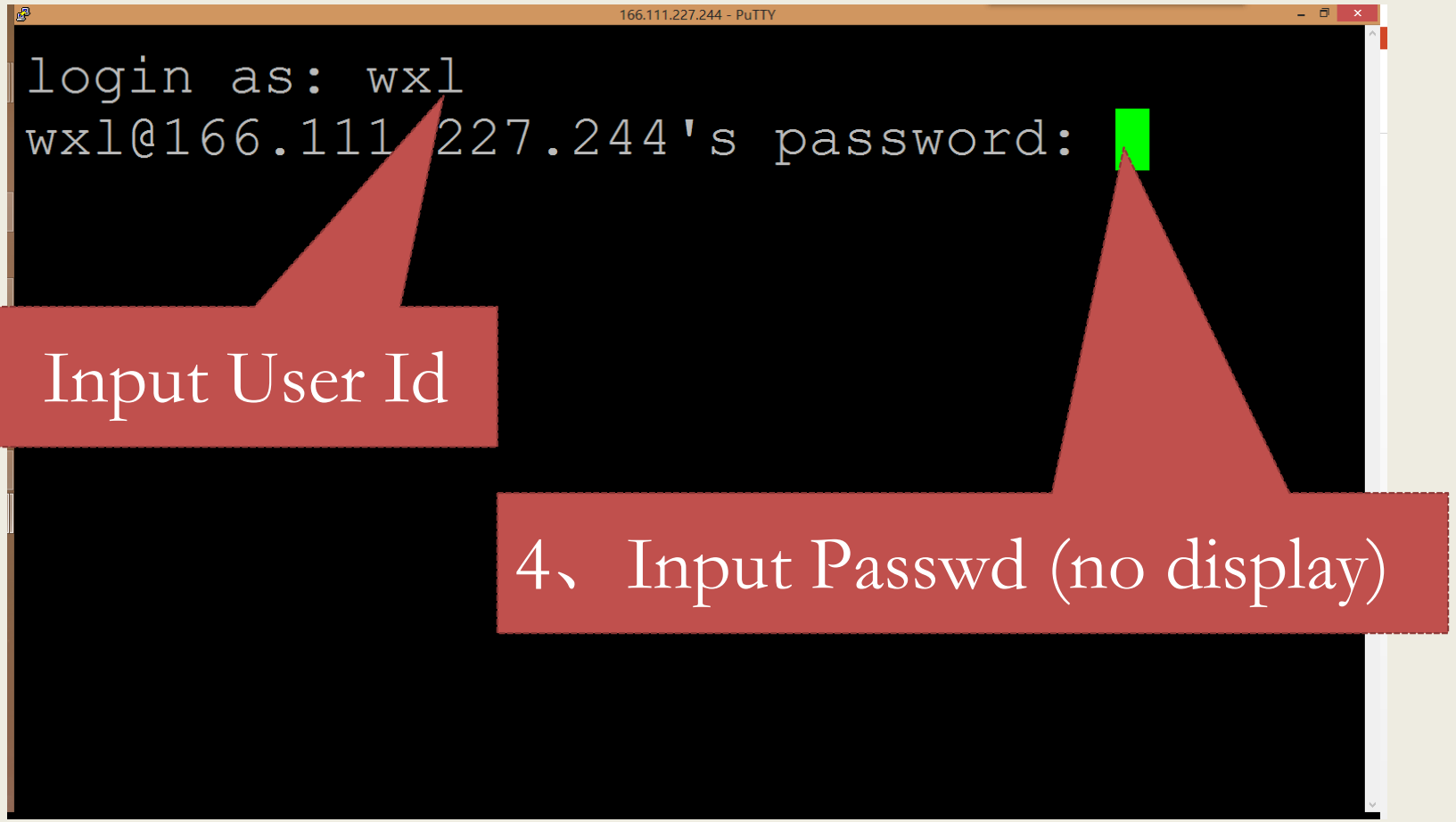
Example for Putty

1、Input IP

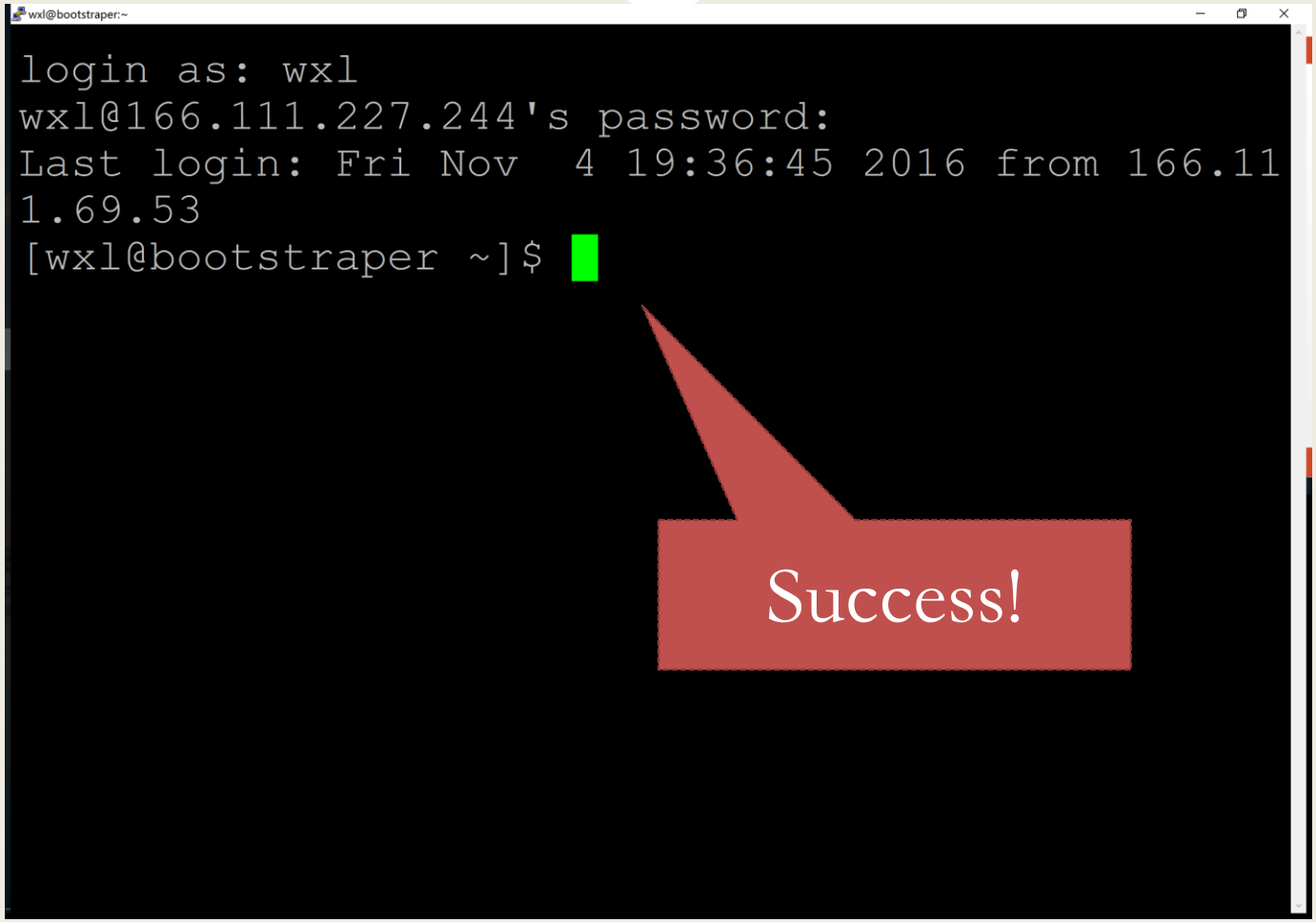
2、Click Open



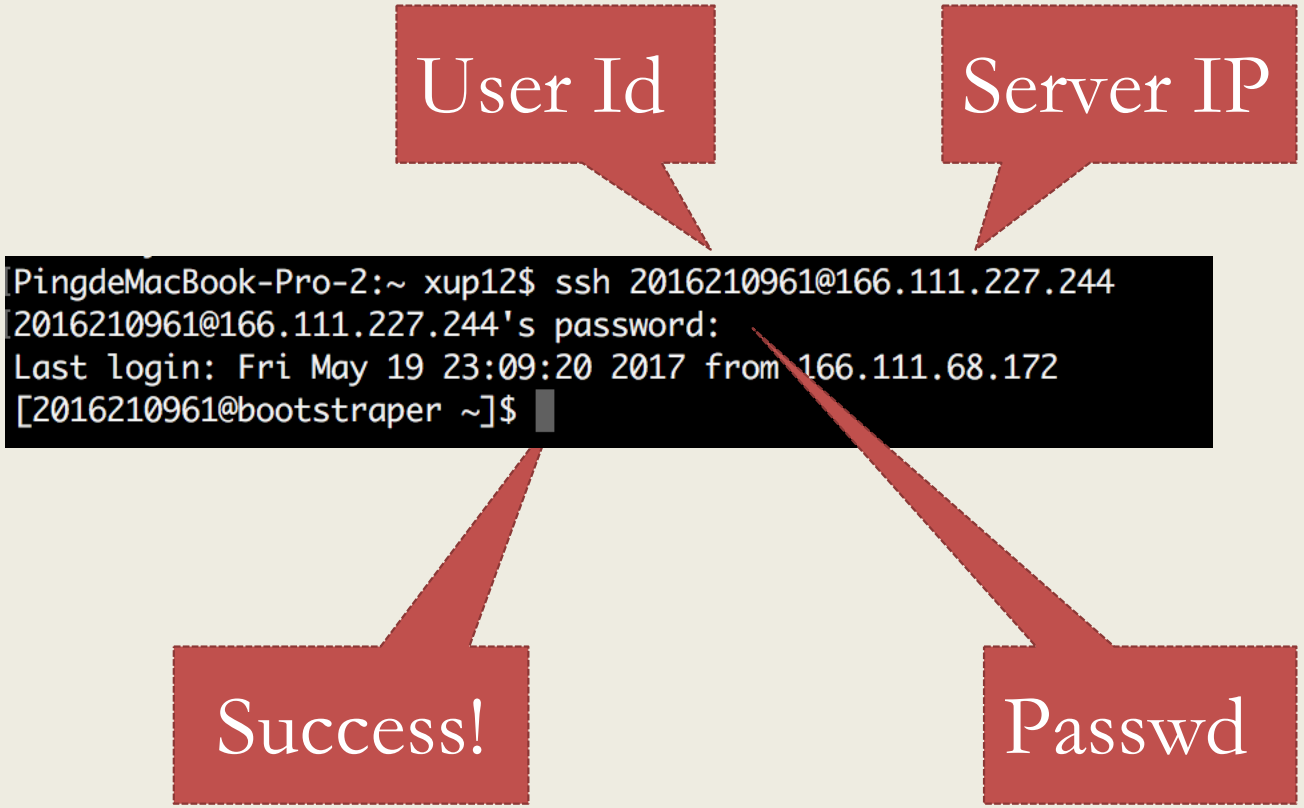
Example for Putty



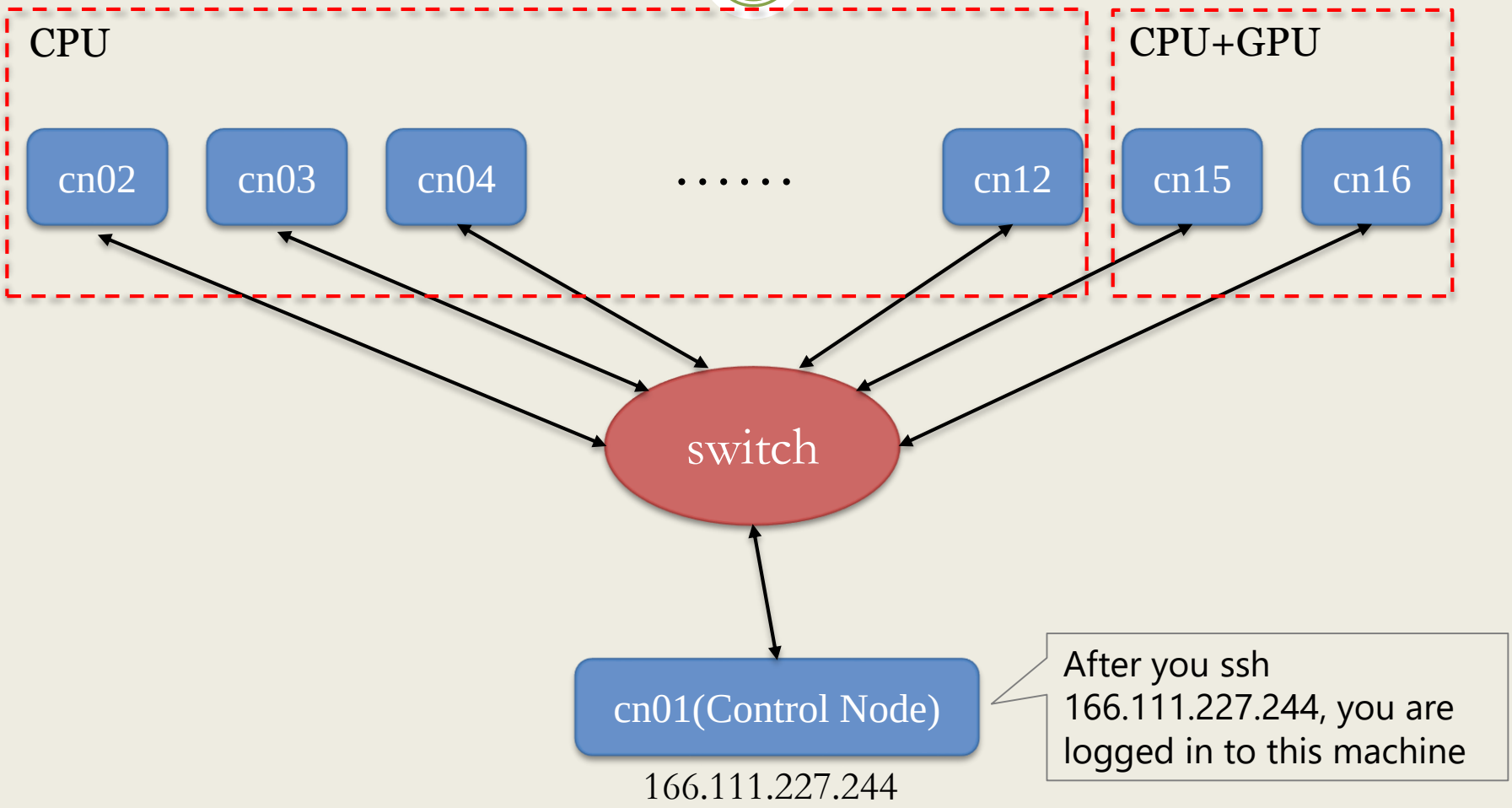
Example for Putty



Example for Mac



Architecture of the Cluster



Getting Assignment Files

How to Get the Assignments

- You can get Assignment 1 files to your user directory using git:

```
git clone ssh://166.111.227.244/data/home/HPC2017_Summer/class/asst1
```

```
[2016210961@bootstraper ~]$ git clone ssh://166.111.227.244/data/home/HPC2017_Summer/class/asst1
Initialized empty Git repository in /data/home/HPC2016/2016210961/asst1/.git/
remote: Counting objects: 52, done.
remote: Compressing objects: 100% (51/51), done.
remote: Total 52 (delta 12), reused 0 (delta 0)
Receiving objects: 100% (52/52), 9.73 MiB | 3 KiB/s, done.
Resolving deltas: 100% (12/12), done.
```

- Other assignments will soon be put up.

Use “make” to Build a Binary Executable File

Example: build prog3

```
cd prog3_Mandelbrot_ispc
ls
```

```
[2016210961@bootstraper prog3_mandelbrot_ispc]$ ls
main.cpp  Makefile  mandelbrot.ispc  mandelbrotSerial.cpp  proj3.vcxproj  proj3.vcxproj.filters
```

```
make
```

```
[2016210961@bootstraper prog3_mandelbrot_ispc]$ make
/bin/mkdir -p objs/
ispc -O2 --target=sse4-x2 --arch=x86-64 mandelbrot.ispc -o objs/mandelbrot_ispc.o -h objs/mandelbrot_ispc.h
g++ -std=c++11 -m64 main.cpp -I../common -Iobjs/ -O3 -Wall -c -o objs/main.o
g++ -std=c++11 -m64 mandelbrotSerial.cpp -I../common -Iobjs/ -O3 -Wall -c -o objs/mandelbrotSerial.o
g++ -std=c++11 -m64 ../common/ppm.cpp -I../common -Iobjs/ -O3 -Wall -c -o objs/ppm.o
g++ -std=c++11 -m64 ../common/tasksys.cpp -I../common -Iobjs/ -O3 -Wall -c -o objs/tasksys.o
g++ -std=c++11 -m64 -I../common -Iobjs/ -O3 -Wall -o mandelbrot_ispc objs/main.o objs/mandelbrotSerial.o objs/mandelbrot_ispc.o objs/ppm.o objs/tasksys.o -lm -lpthr
[2016210961@bootstraper prog3_mandelbrot_ispc]$ ls
main.cpp  Makefile  mandelbrot_ispc  mandelbrot.ispc  mandelbrotSerial.cpp  objs  proj3.vcxproj  proj3.vcxproj.filters  submit.sh
```

Binary file

End

xuping908@gmail.com