



King Fahd University of Petroleum & Minerals
جامعة الملك فهد للبترول والمعادن

KING FAHD UNIVERSITY OF PETROLEUM AND MINIRALS

Collage of Computer Sciences and Engineering

Computer Engineering Department

Semester 082 (Spring), 6 October 2009

COE 501, Project Report -1

A Tutorial:

Remotely Accessing the KFUPM ITC-HPC Cluster & Installing Intel C++ Compiler

Course Instructor:

Dr. Mayez Abdullah Al-Mouhamed

Report Prepared By:

Ameer Ahmed Abbasi

ID: g200703670

Table of Contents

1- Introduction	3
2- Opening User Account on ITC-HPC Cluster	3
2.1 The Procedure	4
2.2 Allocation of HPC at KFUPM	4
2.3 ITC-HPC Technical Staff (Contact Information)	4
2.4 Acknowledging the use of HPC at KFUPM	4
3- Accessing ITC-HPC Cluster from KFUPM Desktop	4
3.1 Accessing the Linux Red Hat Cluster	5
3.1.1 PuTTY	5
a. Starting a session with PuTTY	6
b. Verifying the host key (SSH only)	7
c. Logging in	7
d. After Logging in	8
e. Logging out	8
3.1.2 WinSCP	8
a. Starting Session with WinSCP	9
b. After Logging in	9
4- Non-Commercial Version of Intel C++ OpenMP Compiler	13
4.1 Downloading the Compiler	14
5- Installing Intel C++ Compiler on ITC-HPC Linux Cluster	14
5.1 Installation	14
5.2 Removal/Uninstall	15
6- Setting up Environment & Testing your Compiler	15

1. Introduction

In an effort to advance research in High Performance Computing (HPC) and to provide premium computing services to researchers throughout the university, King Fahd University of Petroleum and Minerals (KFUPM) has recently created its highly specialized Center for HPC. The center has started to serve as a central gateway to the high performance computing resources of the university. The center with its recently acquired IBM cluster system enabling interdisciplinary research as well as research in individual disciplines and supporting users with the best possible computing infrastructure to carry out their research. The HPC project strengthens the position of KFUPM as a leader, not only in deploying high-tech infrastructure, but also in advancing it for scientific and engineering applications.

The Information technology Center (ITC) at KFUPM has a High-Performance Computing System (HPC) which consists of a 128-node Dual-boot Cluster with following resources (<http://www.kfupm.edu.sa/hpc/>):

- 128 compute-node e1350 IBM eServer cluster.
- The cluster is unique in its dual-boot capability with Microsoft Windows HPC Server 2008 and Red Hat Enterprise Linux 5 operating systems.
- The cluster has 3 master nodes, one for Red Hat Linux, one for Windows HPC Server 2008 and one for cluster management.
- The cluster has 128 compute nodes.
- Each compute node of the cluster is dual-processor having two 2.0 GHz x3550 Xeon Quad-core E5405 processors.
- The total number of cores in the cluster is 1024.
- Each master node has 1 TB of hard disk space.
- Each compute node has 500 GB of hard disk space.
- Each master node has 8 GB of RAM.
- Each compute node has 4 GB of RAM.
- The interconnect is 10 GBASE-SR

This document provides a quick tutorial on how to open user account with ITC-HPC and remotely access ITC-HPC cluster from your KFUPM desktop to perform high performance computing. In addition, the document provides information on how to download, install and test the non-commercial version of Intel C++ openMP compatible compiler for Linux.

2. Opening User Account on ITC-HPC Cluster

In order to use HPC computational resources at KFUPM and access ITC-HPC cluster (Linux or Windows), you must first request a user account. Any KFUPM affiliate or a collaborator of a KFUPM affiliate, including graduate or undergraduate students, may request an account.

2.1 The Procedure

Accounts for HPC Systems can be requested through a valid KFUPM email address by sending an email to **hpc@kfupm.edu.sa** . In a response from ITC, a link to the *account request form* will be sent to your KFUPM email address. To be directed to the *account request form*, you should click the link in the email message. In order to expedite the entire process, *account request form* needed to be read carefully and filled properly. If you are a student, postdoctoral researcher or research associate at KFUPM, you may input your advisor as a KFUPM Contact/Collaborator. Members of KFUPM faculty at or above Assistant Professor level are not required to input KFUPM Contact/Collaborator as they will be the primary users of the HPC resources themselves.

2.2 Allocation of HPC at KFUPM

Once you have a user account, you must request an allocation to use HPC computational resources at KFUPM. The allocations step is presently through email but once the user accounts and access procedures are finalized, you will be allowed to submit your jobs directly to the cluster from your desktops.

2.3 ITC-HPC Technical Staff (Contact Information)

This section provides information about the technical staff involve with the ITC-HPC project at KFUPM. The technical staff may be contacted on the following phone numbers:

- | | |
|---|-------------|
| 1. Mr. Nabil El-Halmoushi [Manager] | Phone: 3910 |
| 2. Mr. Syed Tariq Maghrabi [Project Head] | Phone: 3979 |
| 3. Mr. Farhan | Phone: 7325 |

2.4 Acknowledging the use of HPC at KFUPM

Users are asked to acknowledge their use of KFUPM HPC resources in resulting publications and reports with the following statement:

Portions of this research were conducted with high performance computing resources provided by King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia (<http://hpc.kfupm.edu.sa>).

3. Accessing ITC-HPC Cluster from KFUPM Desktop

SSH, Telnet and Rlogin are three ways of doing the same thing: logging in to a multi-user computer from another computer, over a network. Multi-user operating systems, such as UNIX, Linux and VMS, usually present a command-line interface to the user, much like the 'Command Prompt' or 'MS-DOS Prompt' in Windows. The system prints a prompt, and user type commands which the system will obey. Using this type of interface, there is no need for a user to be sitting at the same machine he is typing commands to. The commands,

and responses, can be sent over a network, so user can sit at one computer and give commands to another one, or even to more than one.

SSH, Telnet and Rlogin are *network protocols* that allow a user to do this. On the computer user sit at, he run a *client*, which makes a network connection to the other computer (the *server*). The network connection carries user's keystrokes and commands from the client to the server, and carries the server's responses back to the user.

In order to access the ITC-HPC cluster (Linux or Windows), you can connect and login remotely from anywhere within the KFUPM campus. ITC recommends that as a user you should make a remote connection from your desktop to utilize the high performance computing or submitting a job. Since the ITC-HPC cluster is dual boot and there exist Linux Red Hat and Windows operating systems, different kind of remote client software are needed to make a successful remote connection either with Linux or Windows cluster. The following sections provide information about the client software suitable to make remote connection with Linux cluster. In addition, information is provided on how to make a remote connection from your desktop to ITC-HPC Linux cluster. Please refer to (<http://www.microsoft.com/hpc/en/us/default.aspx>) to learn about Windows cluster HPC operating system and its remote connection client software.

3.1 Accessing the Linux Red Hat Cluster

To remotely access the command line on ITC-HPC cluster a terminal emulator application is required. Another application is needed in order to perform secure file transfer between your desktop and ITC-HPC cluster. Following software applications are highly recommended:

1. PuTTY [available at URL: <http://www.chiark.greenend.org.uk/~sgtatham/putty/>]
2. WinSCP [available at URL: <http://winscp.net/eng/index.php>]

Please note that these software applications are open source and can be downloaded from the above mentioned URLs free of cost. It is recommended to download .exe files of the above mentioned software which do not required to install them on a machine.

3.1.1 PuTTY

PuTTY is a terminal emulator application which can act as a client for the SSH, Telnet, rlogin, and raw TCP computing protocols. The name "PuTTY" has no definitive meaning, though 'tty' is the name for a terminal in the UNIX tradition, usually held to be short for teletype.

PuTTY was originally written for Microsoft Windows, but it has been ported to various other operating systems. Official ports are available for some UNIX-like platforms, with

work-in-progress ports to Classic Mac OS and Mac OS X, and unofficial ports have been contributed to platforms such as Symbian and Windows Mobile.

This section gives you a quick guide to the simplest types of interactive login session using PuTTY.

a. Starting a session with PuTTY

Once you start PuTTY, you will see a dialog box. This dialog box allows you to control everything PuTTY can do. You don't usually need to change most of the configuration options. To start the simplest kind of session, all you need to do is to enter a few basic parameters. In the 'Host Name' text box, enter the IP address of Linux head server to connect to. Currently, IP address of Linux head server is **10.146.1.130**. You should have been told this by the ITC-HPC technical staff that has provided you login account. Now you should select a login protocol to use, from the 'Connection type' buttons. For a login session, you should select Telnet, Rlogin or SSH. The fourth protocol, *Raw*, is not used for interactive login sessions; it is usually use for debugging other Internet services. The fifth option, *Serial*, is used for connecting to a local serial line, and works somewhat differently. By default, SSH protocol is selected and you should leave the protocol SSH.

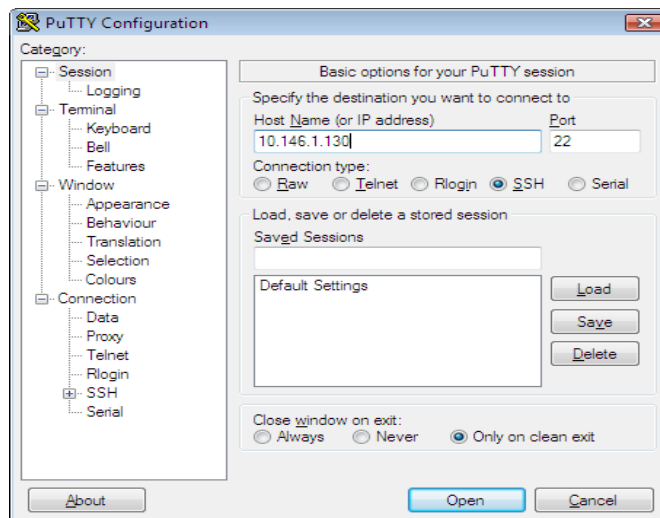


Figure 1: Session starting & PuTTY configuration screen.

If you change the selected protocol, the number in the 'Port' box will change. This is normal: it happens because the various login services are usually provided on different network ports by the server machine. Most servers will use the standard port numbers, so you normally will not need to change the port setting. If the server provides login services on a non-standard port, the system administrator from ITC-HPC should have told you which one (For example, many MUDs run Telnet service on a port other than 23.). By default, port 22 is available in the box and you should leave it as it is. Once you have filled in the 'Host Name', 'Protocol', and possibly 'Port' settings and your screen looks like the one

provided in Figure 1 above, you are ready to connect. By pressing the 'Open' button at the bottom of the dialog box, PuTTY will begin trying to connect you to the server.

b. Verifying the host key (SSH only)

If you are using SSH to connect to a server for the first time, you will probably see a message looking something like this:

```
The server's host key is not cached in the registry. You have no
guarantee that the server is the computer you think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 1024 7b:e5:6f:a7:f4:f9:81:62:5c:e3:1f:bf:8b:57:6c:5a
If you trust this host, hit Yes to add the key to PuTTY's cache and
carry on connecting.
If you want to carry on connecting just once, without adding the key
to the cache, hit No.
If you do not trust this host, hit Cancel to abandon the connection.
```

Since users from within the KFUPM network only can connect to the server, spoofing attacks are unlikely and you should choose to trust the key without checking it.

c. Logging in

After you have connected, and perhaps verified the server's host key, you will be asked to log in, probably using a username and a password. ITC-HPC system administrator should have provided you these username and password. Your screen should look like the one provided below in Figure 2.

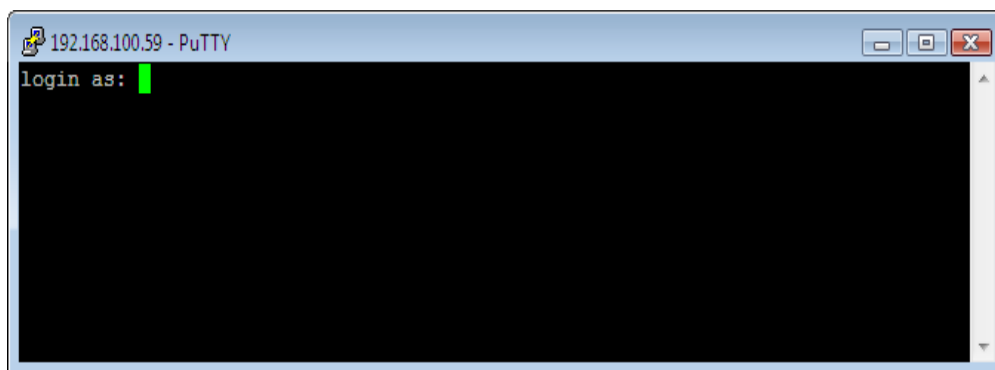


Figure 2: User login screen provided by the PuTTY client.

By entering the username and the password, the server should grant you access and begin a session for you. If user has mistyped the password, most servers will give him several chances to get it right. However, if user is using SSH (which is true in this case), he should be careful not to type his username wrongly, because he will not have a chance to correct it after he press Return; many SSH servers do not permit to make two login attempts using

different usernames. Thus, if you typed your username wrongly, you must close PuTTY and start again.

If the password is refused and you believe that you have typed it correctly, you should check that Caps Lock is not enabled. Many login servers, particularly UNIX/Linux computers, treat upper case and lower case as different when checking the password; so if Caps Lock is on, the password will probably be refused.

d. After Logging in

After you log in to the server, what happens next is up to the server! Most servers will print some sort of login message and then present a prompt, at which user can type commands which the server will carry out. Some servers will offer users on-line help; others might not. However, your screen should look similar to the one provided in Figure 3.

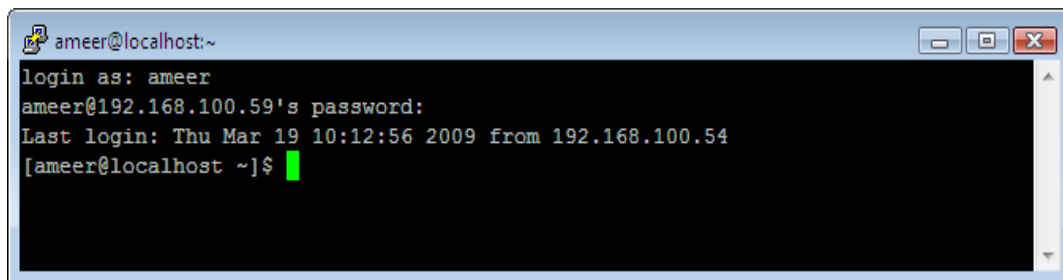


Figure 3: PuTTY command line screen after successful login.

e. Logging out

Once you want to finish your session, you should log out by typing the server's own logout command. This might vary between servers; if in doubt, try `logout` or `exit`, or consult a manual or system administrator. When the server processes user's logout command, the PuTTY window should close itself automatically.

You *can* close a PuTTY session using the Close button in the window border, but this might confuse the server - a bit like hanging up a telephone unexpectedly in the middle of a conversation. It is recommended not doing this unless the server has stopped responding and you cannot close the window any other way.

3.1.2 WinSCP

WinSCP (Windows Secure CoPy) is an open source SFTP and FTP client for Microsoft Windows. Its main function is secure file transfer between a local and a remote computer. Beyond this, WinSCP offers basic file manager and file synchronization functionality. For secure transfers, it uses Secure Shell (SSH) and supports the SCP protocol in addition to SFTP.

The following sub-sections give a quick guide on how to login to WinSCP. In addition, a pictorial example is provided on file transfer between a local and a remote computer using WinSCP.

a. Starting a session with WinSCP

Once you start WinSCP, you will see a dialog box. This dialog box allows you to control everything WinSCP can do. Like PuTTY, you don't need to change much of the default configuration settings. Simply, all you need to do is to enter a few basic parameters.

In the 'Host Name' text box, enter the IP address of Linux head server to connect to. Currently, IP address of Linux head server is **10.146.1.130**. You should have been told this by the ITC-HPC technical staff that has provided you login account. Since WinSCP uses SSH protocol, by default port 22 is selected in the port number box and you should leave it as it is. In the 'Username' and 'Password' text boxes, you should type the username and password that already provided by the ITC-HPC system administrator.

Once you have filled in the correct 'Host Name', 'Username' and 'Password' and your screen looks like the one provided below in Figure 4, you are ready to connect. Upon pressing the 'Login' button at the bottom of the dialog box, WinSCP will connect you to the server.

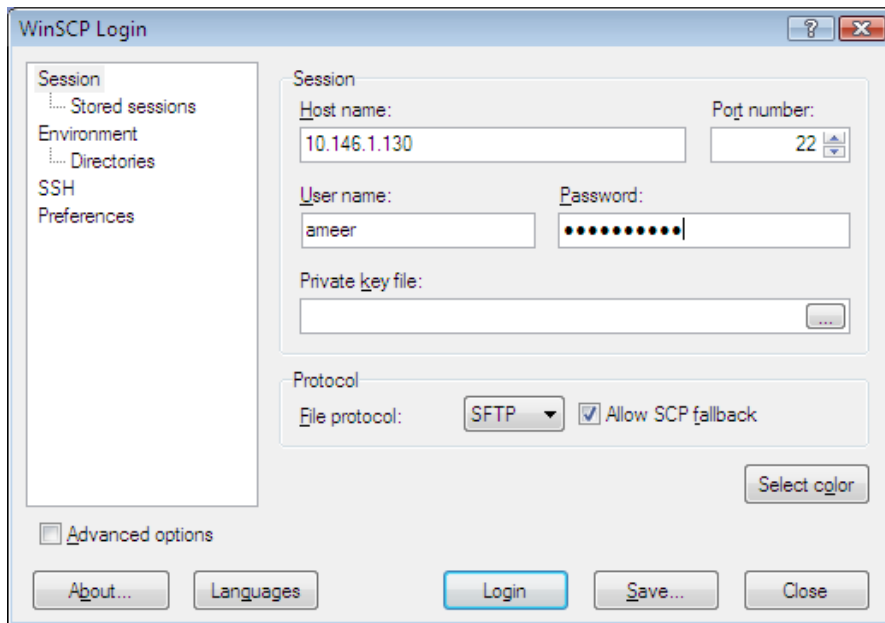


Figure 4: WinSCP Login screen.

If you are using a machine first time to login to ITC-HPC cluster by using WinSCP (remember WinSCP uses the SSH protocol), you will probably see a message to verify the host key. The details of it have already been explained above under the PuTTY sub-section.

b. After Logging in

Once connection has been established you will see window looks like the one provided below in Figure 5. Where on left side there are files of user local computer and on right side there are files that are on server where you have just connected. You can move up in folder tree by clicking the icon that has two dots (..). Moving down in directory is done by double clicking on the name of the folder.

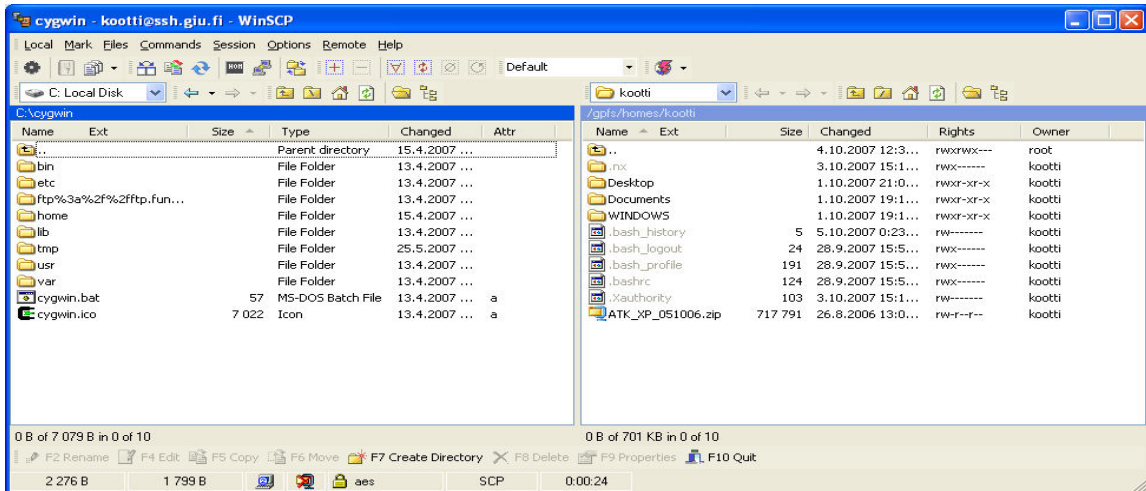


Figure 5: WinSCP graphical user interface.

Before you can do any copying, you must first select files that you want to operate. This is done by first clicking file/folder and then opening **Mark** menu and then clicking select. You can also use **Ins** keyboard shortcut to select/deselect files or folders. For example, in the Figure 6 below setup.ini is going to be selected.

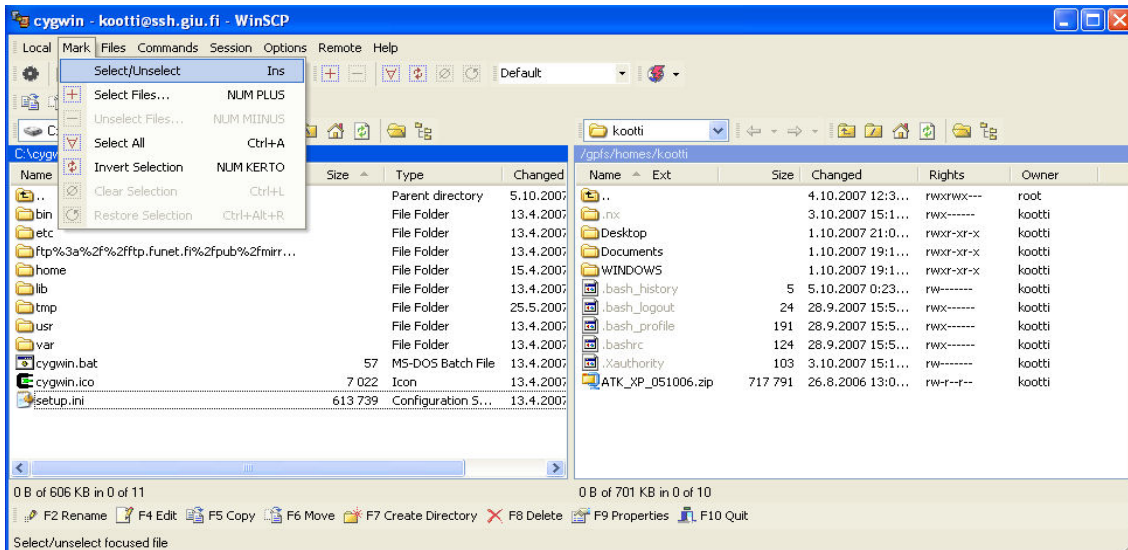


Figure 6: Selecting a file to transfer.

You can perform many different actions to selected files. For example, in Figure 7 below we are copying `setup.ini` file.

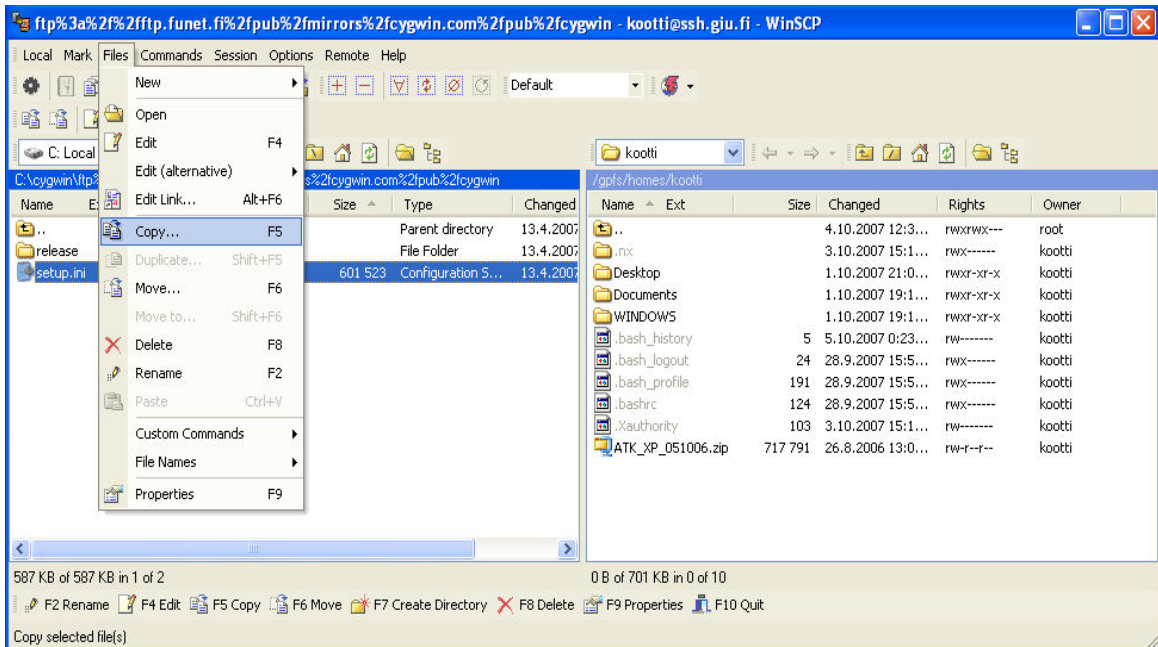


Figure 7: Copy a file to transfer

After choosing Copy (F5) a dialog box window will popup and lets you do many different changes to default settings (which in general are OK to use).

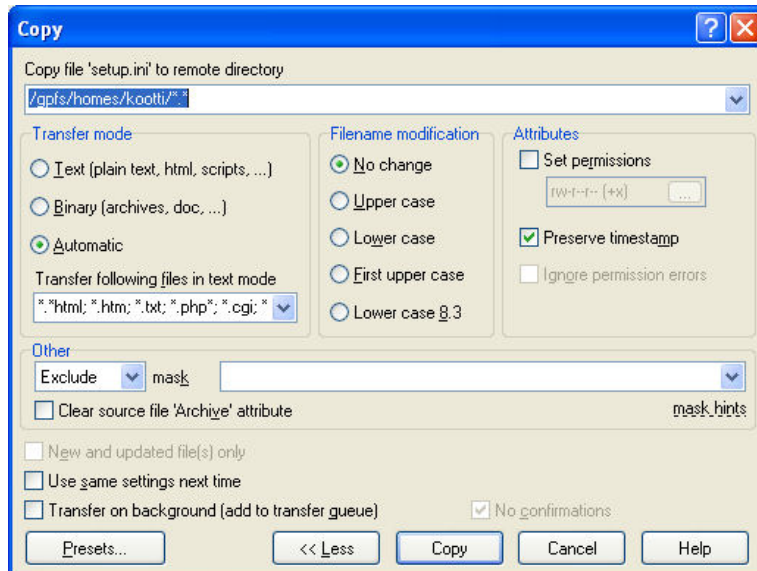


Figure 8: Copy confirmation dialog box.

The progress bar below in Figure 9 will lets you see current action, statistics and estimate of time left until action is finished.

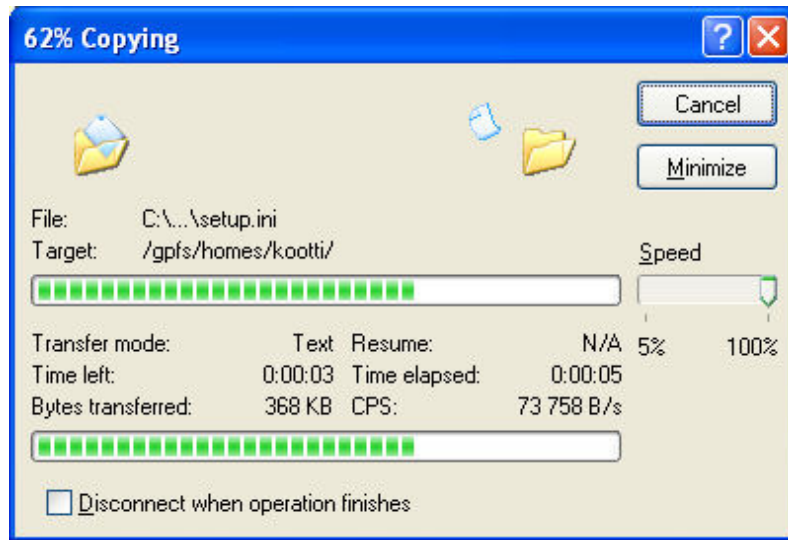


Figure 9: Progress of copying is reported by a progress bar.

Action is now finished and `setup.ini` is copied from your local drive to server home folder.

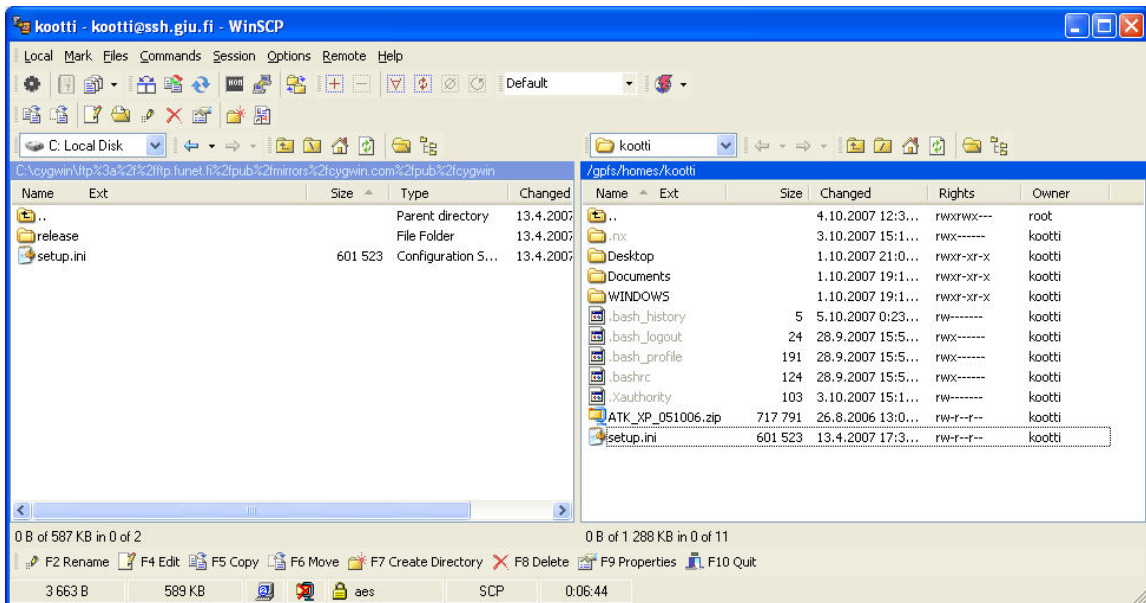


Figure 10: WinSCP window after copying is done.

As evident from the above discussions, transferring files with WinSCP is simple. As method presented above, you can drag and drop the file from a folder on your computer into the WinSCP window, or *vice versa*. Let's try it:

1. On the desktop, create a file called `winscp-test.txt`. Type something short into this file (e.g., "Hello!") and save it.

2. Now, drag the file from the desktop into the WinSCP window, and drop it (let go of the left mouse button). You will be prompted with a Copy dialog box. This is just to confirm that you really want to copy the file; hit OK.
3. The file now shows up in the remote file listing. File just has copied from your computer to GIU!
4. To transfer a file from GIU back to the user computer, simply reverse the process. First, delete the file `winscp-test.txt` from the Desktop.
5. Then, drag the file from the WinSCP window (GIU) back to the Desktop. Another confirmation window will be presented, just as before; confirm by clicking the Copy button.
6. The file will be transferred back to the Desktop. Open it to see it's the same as the one you just created before, only this one has been copied to GIU and back again.

In short, for any file that needs to be transferred to GIU, just connect with WinSCP and drag it into the WinSCP window. Very simple!

4. Non-Commercial Version of Intel C++ OpenMP Compiler

The Intel C++ Compiler Professional Edition delivers rapid development and winning performance for the full range of Intel processor-based platforms. The Professional Edition not only comes with the compiler's breadth of advanced optimization, multi-threading, and processor support, including automatic processor dispatch, vectorization, and loop unrolling, it also has highly optimized C++ templates for parallelism, math processing, and multimedia libraries.

The Intel C++ Compiler Professional Edition has the following special feature for fine-grain parallelization and multi-programming to get full advantage of recent multi-core technology.

Open MP 3.0

OpenMP raises the parallelism abstraction away from the API, simplifying threading and making code more portable. Previously limited to loop-based data-parallelism, the new 3.0 standard simplifies both data and task parallelism.

Eclipse IDE Integration

This integration provides GUI operation in addition to command-line execution for the Intel® Itanium processor.

Multi-Threaded Application Support

OpenMP and auto-parallelization allow you to take full advantage of multi-core technology.

In addition, software compiled using the Intel C++ Compiler for Linux benefits from advanced optimization features, including:

High Performance Parallel Optimizer (HPO) offers an improved ability to analyze, optimize, and parallelize more loop nests. This revolutionary capability combines vectorization, parallelization, and loop transformations into a single pass which is faster, more effective, and more reliable than prior discrete phases.

Automatic Vectorizer analyzes loops and determines when it is safe and effective to execute several iterations of the loop in parallel.

Profile-Guided Optimization (PGO) improves application performance by reducing instruction-cache thrashing, reorganizing code layout, shrinking code size, and reducing branch mispredictions.

Interprocedural Optimization (IPO) dramatically improves performance of small- or medium-sized functions that are used frequently, especially programs that contain calls within loops. The analysis capabilities of this optimizer can also give feedback on vulnerabilities and coding errors that cannot be as effectively detected by compilers that rely on front-end analysis.

Please refer to (<http://www.intel.com/software/products/compilers/clin>) for details on hardware and software requirements.

4.1 Downloading the Compiler

Intel has expanded its offerings of free tools for non-commercial software development. This offering is provided to developers who are developing software on their own time without compensation. Intel C++ Compiler Professional Edition 11.0 for Linux can be downloaded free of cost for non-commercial use from the Intel website.

To download, please refer to (<http://www.intel.com/cd/software/products/asm-na/eng/277618.htm>) and click on “Free Non-Commercial Download”. Once you accept the conditions and terms you will be redirected to another page. On this webpage, under compiler category you will see various Linux based Intel compilers which are available free for non-commercial use. Once you click on Intel C++ Compiler Professional Edition 11.0 for Linux you will be redirected to the registration page. This registration is free of charge. After completing the registration successfully you will be provided a product serial number and option to download the product. This serial number along with a link for download will also be emailed to your email address for future use. If you decided not to download now, you can download the product later just by clicking on the link that you have received in your email without registering again.

5. Installing Intel C++ Compiler on ITC-HPC Linux Cluster

This section describes how to install the Intel C++ Compiler Professional Edition 11.0 for Linux on ITC-HPC cluster remotely.

5.1 Installation

If you are installing the Intel C++ Compiler Professional Edition 11.0 for Linux for the first time, please be sure to have the product serial number available as you will be asked for it during installation. The product serial number for non-commercial use can be obtained free of charge from the Intel website as already been explained in section 4 of this document. A valid license is required for installation and use.

First of all, you need to login to ITC-HPC Linux cluster by using WinSCP application to copy the downloaded zipped file to the server. Once you copied the zipped file to the server, you need to access the server command line remotely. You should login to the server by using PuTTY. If java run time environment (jre) is not installed, you will get a warning

prompt about it during the installation. If you like to use GUI of Intel C++ debugger, make sure latest version of java run time environment (jre) is installed. Otherwise GUI of Intel C++ debugger will not work properly.

In addition, Intel C++ Compiler Professional Edition 11.0 for Linux required GNU gcc and gcc-c++ compiler collection to be installed and work properly. Make sure that latest versions of gcc and gcc-c++ are installed on the Linux server.

Since you received the product as a downloadable zipped file, first unpack it into a writeable directory of your choice using the command:

```
tar -xzvf name-of-downloaded-file
```

Then change the directory (cd) to the directory containing the unpacked files and begin the installation using the command:

```
./install.sh
```

If you received your product on DVD, mount the DVD, change the directory (cd) to the top-level directory of the mounted DVD and begin the installation using the command:

```
./install.sh
```

Follow the prompts to complete installation.

5.2 Removal/Uninstall

Removing (uninstalling) the product should be done by the same user who installed it (root or a non-root user). It is not possible to remove the compiler while leaving any of the performance library or Eclipse* integration components installed.

1. Open a terminal window and set default (cd) to any folder outside <install-dir>
2. Type the command: <install-dir>/bin/ia32/uninstall_cproc.sh (substitute intel64 or ia64 for ia32 as desired)
3. Follow the prompts
4. Repeat steps 2 and 3 to remove additional platforms or versions

If you have the same-numbered version of Intel® Fortran Compiler installed, it may also be removed. If you have added the Intel C++ Eclipse integration to an instance of Eclipse in your environment, you will need to update your Eclipse configuration by removing the Intel integration extension site from your Eclipse configuration.

6. Setting up Environment & Testing your Compiler

To start using the compiler you need to perform the following three steps:

1. Open a command prompt.
2. Set the environment variables for the compiler.
3. Invoke the compiler.

One way to set the environment variables prior to invoking the compiler is to "source" the compiler environment script, `iccvars.sh` (or `iccvars.csh`):

```
source <install-dir>/Compiler/11.0/xxx/bin/iccvars.sh argument
```

where `<install-dir>` is the installation directory (default for system-wide installation is `/opt/intel`) and `xxx` is the update number. The environment script takes an *argument* based on architecture; valid arguments are as follows:

- `ia32`: Compilers and libraries for IA-32 architectures only.
- `intel64`: Compilers and libraries for Intel(R) 64 architectures only.
- `ia64`: Compilers and libraries for IA-64 architectures only.

In the following a trivial test program has been provided that you can use to test that whether your compiler and environment support parallelization and multi-programming:

```
#include <omp.h>
#include <stdio.h>
int main() {
#pragma omp parallel
printf("Hello from thread %d, nthreads %d\n", omp_get_thread_num(),
omp_get_num_threads());
}
```

Type or copy the program into a C/C++/text editor and save file as `hello.c` (for C) or `hello.cpp` (for C++) on your local computer. Transfer the file to your home directory on the Linux server by using the WinSCP as explained under section 3.1.2 above.

To compile a C language source file (for example, `hello.c`), use the following command:

```
icc -openmp hello.c
```

To compile C++ source files, use this command:

```
icpc -openmp hello.cpp
```

Compilation should proceed with no errors or warnings.

Execute the output file, called `a.out`. You should see more than one "Hello" lines with different thread numbers. Note that the lines may be mixed together. If you see only one, try setting the environment variable `OMP_NUM_THREADS` to some number (say 4) and try again.