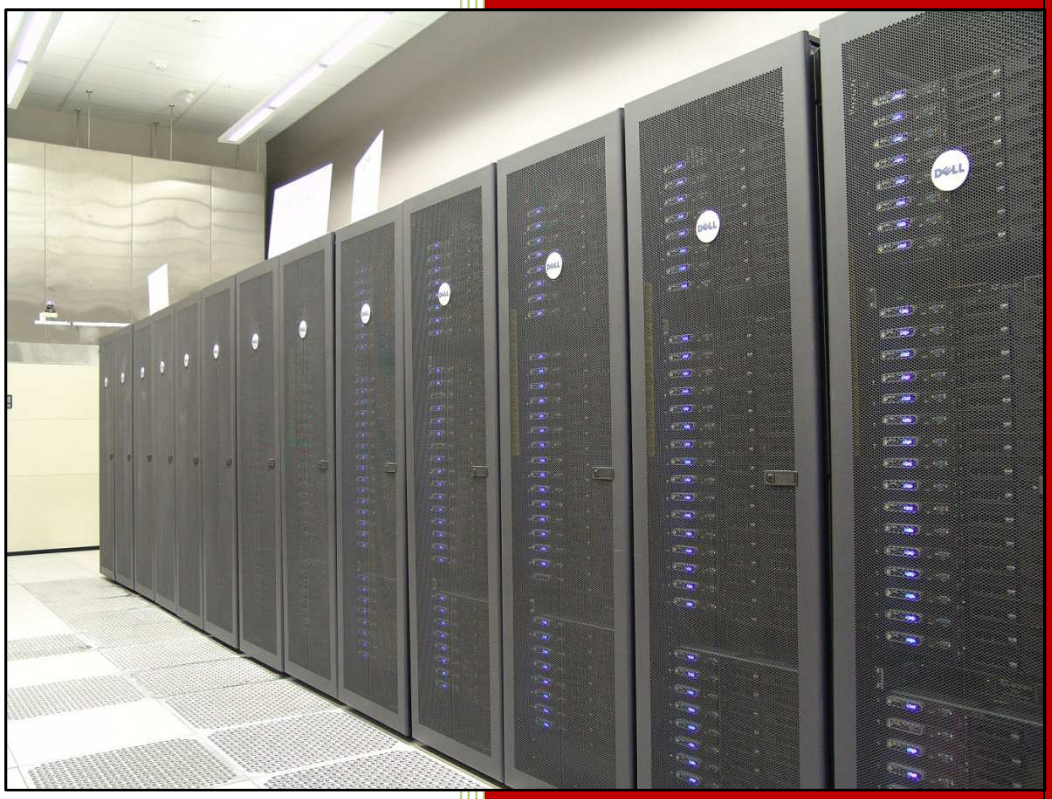
HPCC - Hrothgar

Getting Started User Guide – Cplex



High Performance Computing Center
Texas Tech University

Table of Contents

1. Introduction

ILOG CPLEX's mathematical optimization technology enables better decision-making for efficient resource utilization. With ILOG CPLEX, complex business problems can be represented as mathematical programming models. Advanced optimization algorithms allow you to rapidly find solutions to these models.

2. Setting up the environment

Hrothgar is equipped with SoftEnv to set up the environment with minimum work by users. The use of SoftEnv is not required but highly recommended by HPCC staff.

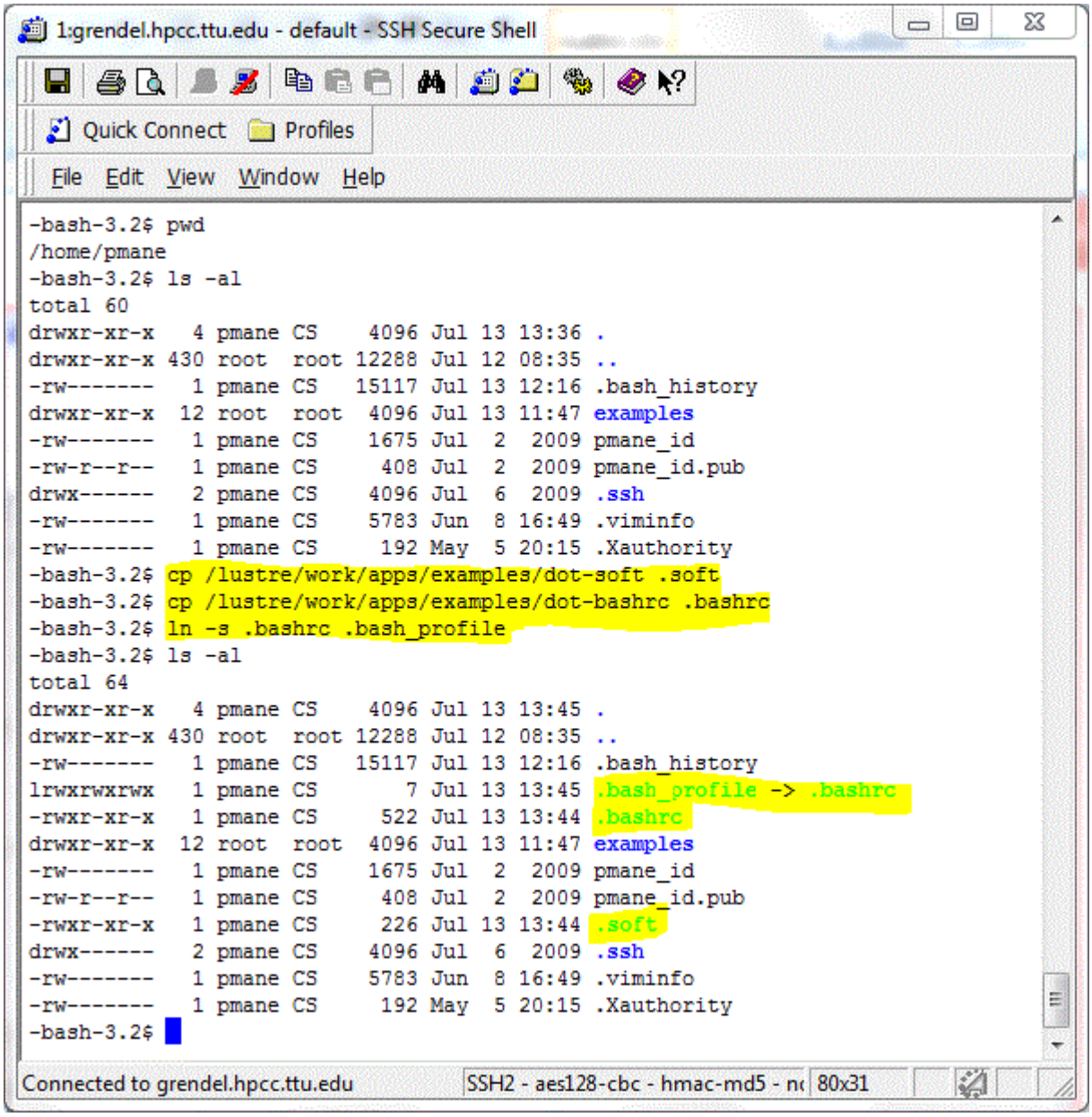
Step 1: setting up user environment

If the user environment is already set up, please skip this step.

At the first use, the user should copy two sample dot-files: dot-bashrc is the start up script which evokes SoftEnv; dot-soft contains a list of software whose specific environment variables will be set up for the user.

```
$ cp  /lustre/work/apps/examples/dot-bashrc .bashrc
$  cp /lustre/work/apps/examples/dot-soft .soft
$ ln –s .bashrc .bash_profile
```

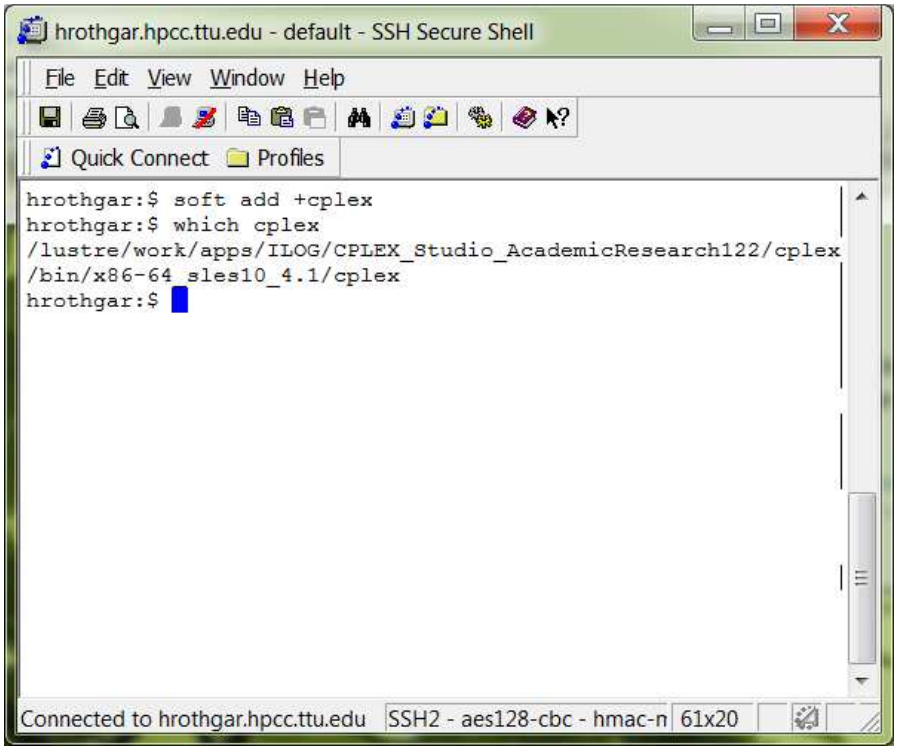

Log out and log in again.



Step 2: setting up CPLEX environment

To run CPLEX on Hrothgar,

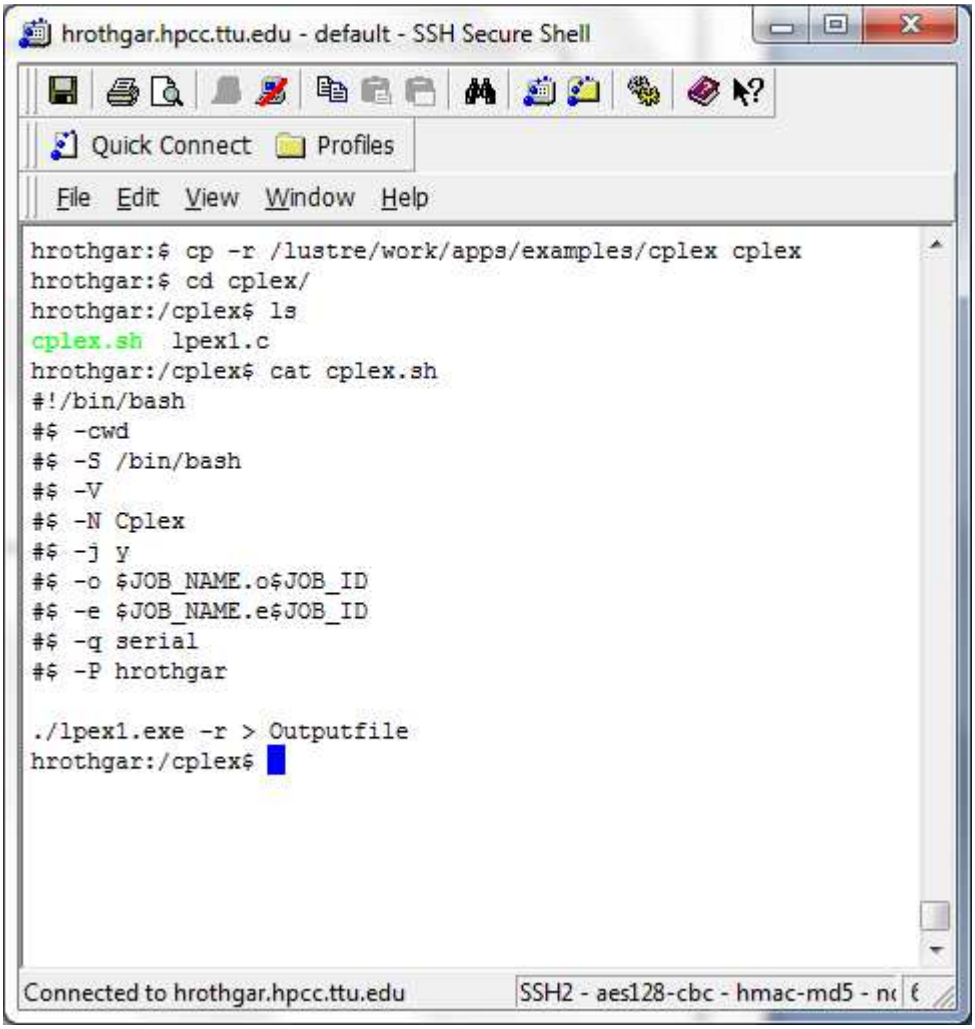
1. Add cplex to user environment by typing the command:
   $ soft add +cplex
2. To check the environment setting: which cplex
   $ which cplex



3. Job Submission

3.1. Script for Batchjob submission

The following is a script file (Cplex.sh) to submit a CPLEX job on Hrothgar. It assumes that input files are in the same directory with the script.



3.2. Job submission

qsub cplex.sh - To submit your Cplexjob to Hrothgar

qstat – To check the status of the job

3.3 Interactive Job submission

Interactive jobs are currently only allowed in . Command To submit a job interactively on Hrothgar

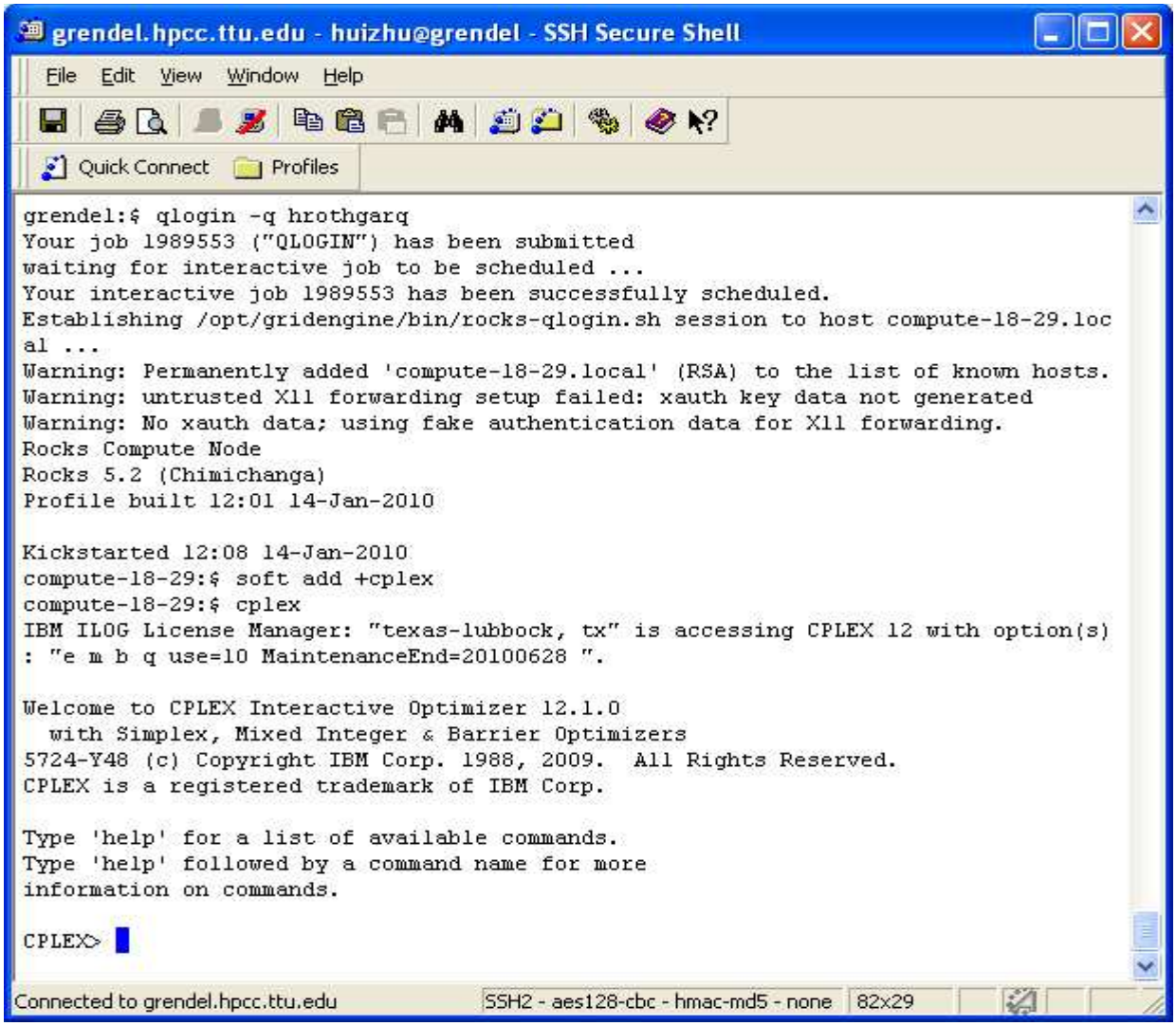
$ qlogin –q serial –P hrothgar –pe fill 1

A node will be assigned for your job and user will be logged in to the node. Users need to add cplex to SoftEnv to set up the environment on the compute node.

$ soft add +cplex

To use cplex on this node type cplex

$ cplex



4. Using CPLEX libraries to compile and link C code:

Example in /lustre/work/apps/examples/cplex directory

$ soft add +cplex

$ cp -r /lustre/work/apps/examples/cplex cplex

Step 1: Generate Object file

$ gcc -c -m64 -fPIC -I/$CPLEXINC lpex1.c -o lpex1.o

Step 2: Link object file with cplex libs.

$ gcc -m64 -fpic -I/$CPLEXINC lpex1.o -o lpex1.exe -L/$CPLEXLIB -lcplex -lm –pthread

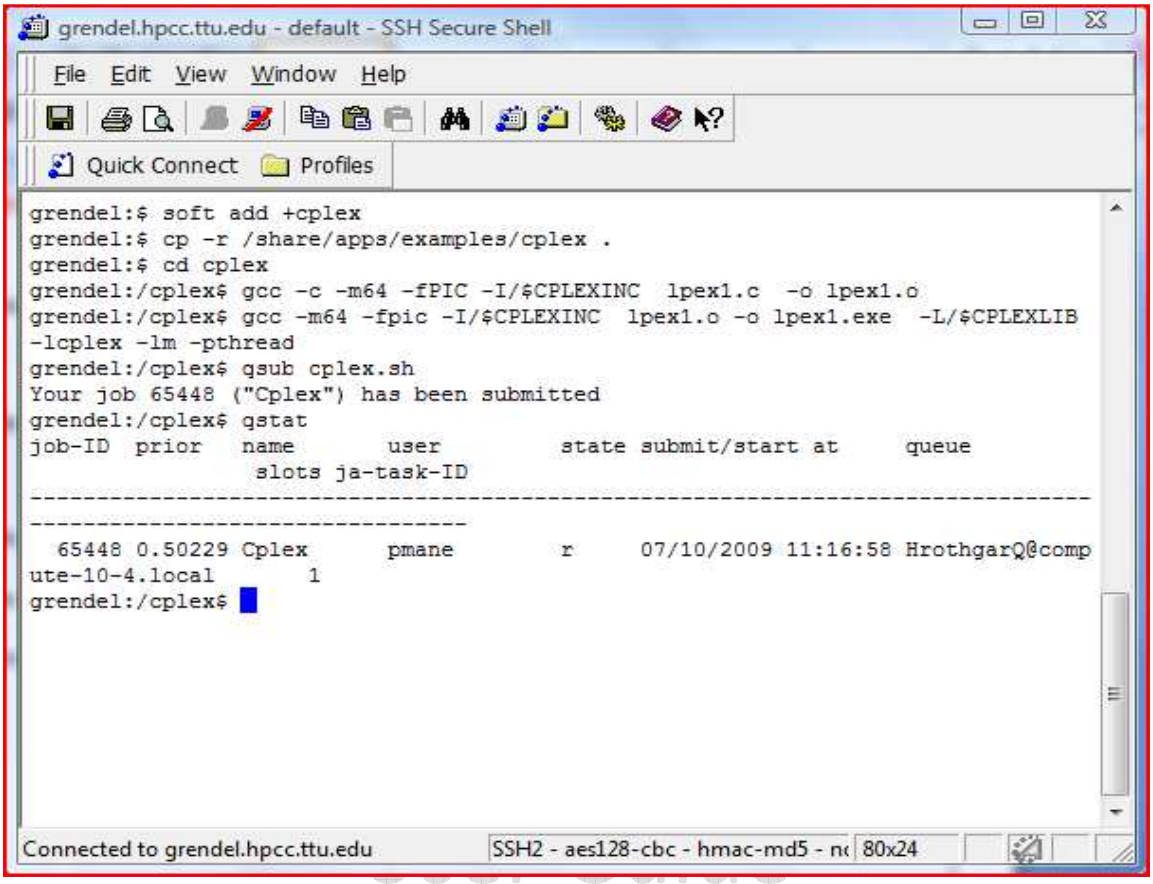
The script to submit this program is available in the same directory, called cplex.sh. It submit the job to run lpex1.exe, and write the output to the file "Outputfile".

```
#!/bin/bash
#$ -cwd
#$ -S /bin/bash
#$ -V
#$ -N Cplex
#$ -j y
#$ -o $JOB_NAME.o$JOB_ID
#$ -e $JOB_NAME.e$JOB_ID
#$ -q serial
#$ -P hrothgar

./lpex1.exe -r > Outputfile
```

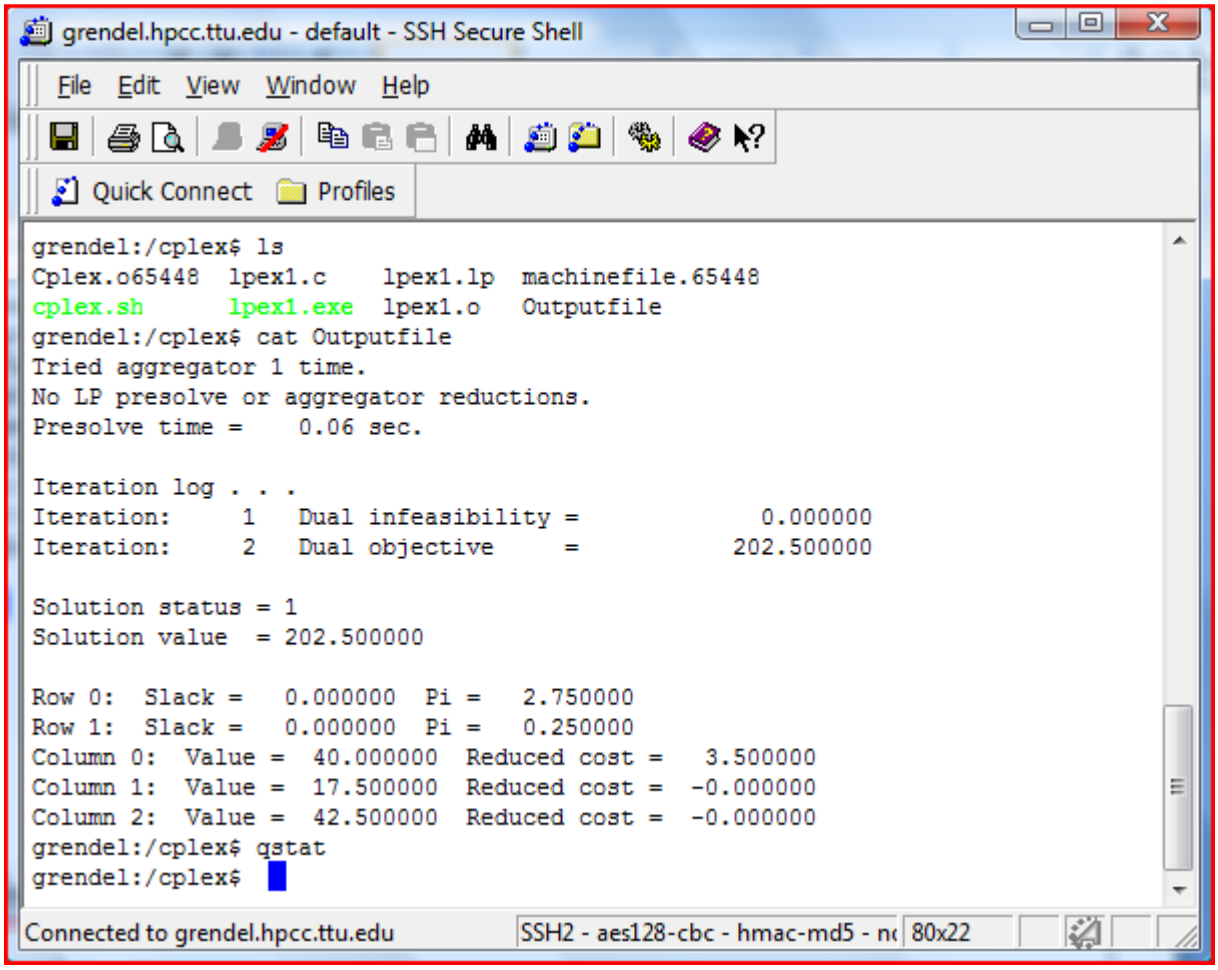
Submit Test Job

$ qsub cplex.sh



Use command “cat Outputfile” to check the results when the job finishes.

$ cat OutputFile


grendel.hpcc.ttu.edu - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
grendel:/cplex$ ls
Cplex.o65448 lpex1.c lpex1.lp machinefile.65448
cplex.sh lpex1.exe lpex1.o Outputfile
grendel:/cplex$ cat Outputfile
Tried aggregator 1 time.
No LP presolve or aggregator reductions.
Presolve time = 0.06 sec.
Iteration log . . .
Iteration: 1 Dual infeasibility = 0.000000
Iteration: 2 Dual objective = 202.500000
Solution status = 1
Solution value = 202.500000
Row 0: Slack = 0.000000 Pi = 2.750000
Row 1: Slack = 0.000000 Pi = 0.250000
Column 0: Value = 40.000000 Reduced cost = 3.500000
Column 1: Value = 17.500000 Reduced cost = -0.000000
Column 2: Value = 42.500000 Reduced cost = -0.000000
grendel:/cplex$ qstat
grendel:/cplex$
Connected to grendel.hpcc.ttu.edu
SSH2 - aes128-cbc - hmac-md5 - n
80x22

Last updated: 10/30/2013

For Additional Assistance Contact: hpccsupport@ttu.edu

For Comments/Suggestions on user guide hpcc@ttu.edu

User Guide