

1 Cluster Machines

The base machine that you can use for development and/or testing your code is going to be `pcc16.njit.edu` whose IP address is `128.235.32.116`. This machine is part of a tiny cluster whose other nodes will become available in the context of PA4.

The other three nodes of the cluster are `pcc13`, `pcc14`, `pcc15` with similar IP addresses (i.e. add to 100 the two-digit suffix number of the name of the machine to get the fourth IP address byte). Among these 3, you might also have access to `pcc15` before PA4.

You can connect from one machine to the other without supplying a password; for example you can type `rsync pcc15.njit.edu -l u10` to connect to `pcc15` from `pcc16` as user `u10`, if you are also user `u10` on `pcc16`.

2 What login name to use

On Monday January 30, 2006 during the break, the instructor will ask you to pick at random a piece of paper that includes information such as a login name in the form `u2X` or `u3X` where `X` is a digit from 1 to 8. On the other side of the paper a password is written that contains four random lower-case English characters and four-digits. So if a password is `a2222bc1`, then the last character must be character `el` rather than one (the digit)!

3 How can you login to the cluster

You can connect to the cluster only from an NJIT IP address. This can be an NJIT machine to which you have already logged on (eg. an AFS machine). Alternatively you might enable an NJIT VPN client (freely available by NJIT) if you plan to connect from home or from outside NJIT and you want to avoid using the former option.

You can only connect to `pcc16.njit.edu` from outside the cluster. This is the base machine. From that you can connect to the other machines by the method described in section 1.

You can connect to the base machine through a secure shell program such as `OPENSSH` available for free under Red Hat/Fedora Linux.

For Windows, NJIT offers a secure shell client at <http://csd.njit.edu/software/> Make sure that if you don't have such a program/client on your machine, you download, install and run it to connect to the cluster. A VPN client is also available through this site or if you also want installation instructions through <http://telecom.njit.edu/vpn/>.

A connection say from AFS to `pcc16.njit.edu` can be initiated through `ssh` by the command
`ssh pcc16.njit.edu -l u10`
if your login name is `u10` at `pcc16`.

4 Account Information

We are going to use the `u10` user as an example for the remainder. Substitute your actual account for `u10` in the remainder.

4.1 Account Policies: Expiration and backup

How long will the accounts be active? The accounts will be deactivated shortly after May 15, 2006.

Backup policies. None. You are responsible for backing up your own files. One simple way is to create a `.tar` file with all your relevant files and upload them to an afs machine or a machine of your preference.

4.2 SSH operations

How to use ssh from within pcc16. Use the command `ssh` to create a telnet equivalent connection, or `sftp` / `scp` to create an ftp-like connection for file transfer or copy files remotely and securely.

An ssh shell is initiated For example `ssh afs10.njit.edu -l jolly-roger` allows you to connect to afs10 as user jolly-roger.

Remote secure copy. If you issue a `scp jolly.tar jolly-roger@afs10.njit.edu:~/` then this command allows you to copy file `jolly.tar` to the remote machine `afs10` in the space of user `jolly-roger`. The file will be copied at the top-level of the directory hierarchy as indicated by the string after the colon `:`. If you want to copy things elsewhere in that hierarchy you can define another directory or file name.

If you are a Windows users you can initiate the copy from your Windows machine. With a ssh Window open, choose `Window` tab and option `New File Transfer`. A new window pops-up and choose `Operation Upload` to move file to the remote machine or `Download` to download files from there. You can navigate around you local machine hierarchy through the newly opened window. You can navigate around the remote machine when you choose an operation.

4.3 Sample Code.

Sample code. User account `u20` will be open (i.e. readable) by all of you (you can read but not write). Comments, and also a copy of the Web crawler's activity on my Web-page will become available there.

4.4 Tar and Untar

How to untar a tar file.

```
% tar xvf testing.tar
```

if the file name is `testing.tar`.

How to create a tar file

```
% tar cvf testing.tar file1 dir1 file2
```

to tar `file1` , `file2` and the contents of directory `dir1`. Sometimes you might need to write `dir1/` instead of just `dir1`.

5 Online material

Account `/home/u20` contains a copy of the "test" Web-data that will be used for this assignment. There is a file `README.1st` that you can read that explains the files/directories available there.

6 Programming Guidelines

A number of programming guidelines that need to be followed for the completion of the programming requirements of this course are listed below.

6.1 How do you submit program files?

Create a directory with name `homework` at the top-level of your home directory say `/home/u10`. Create within it 4 subdirectories labeled `pa1` through `pa4`. Make sure that permissions of `homework` and all four subdirectories are `700` (i.e. readable/writeable/executable only by you). If you don't know how to turn on and off permissions do a `man chmod` and follow the instructions.

A successful completion of this task will earn you 10 points out of the points of PA1.

In the future a homework submission will mean copying files into the corresponding `paZ` directory. The work in `paZ` should be frozen before midnight of the day the homework is due; otherwise a late submission will be recorded.

For every source code file (or result file that you may generate) clearly identify yourself in the first line of that file. The first few lines could look like the following ones.

```
/* Alex. Gerbessiotis: u10 PA1 Programming */
// Use the common convention of of the programming language you use for
// your implementation to signify a comment line
int main()
{
    return(0);
}
```

If for some reason you lose access to the base machine and you have the relevant files stored elsewhere, you can send us an email to `alg750@cs.njit.edu` but the Subject: line of your email must clearly state PA number and your user name, for example

Subject: `u10pa2`

If you decide not to follow the standard guidelines (per assignment) for file-names you must provide a `README` file that tells us what is going on.

It is your responsibility that your code is runnable on the text platform and corresponding pa directory.

In order to compile and run the files, each one of you will have to create a Makefile and use program `make`. Information about `make` is also available at the site `www.gnu.org`

A sample Makefile is also provided at `/home/u20/Sample.make`.

6.2 C/C++ Compilers and Perl

The default one for Red Hat Linux 9.0 i.e. `gcc/g++ 3.2.2-5`. Perl is version `v5.8.0`.

6.3 Java interpreter

For Java, `j2sdk-1.4.2` and `j2re-1.4.2` have been installed on `pcc16`. However to ease compilation problems copy the `/home/u10/cshrc` file into your own directory as file `.cshrc` (the file-name has a period `.` before `cshrc`) by typing for example

```
% cp /home/u10/cshrc /home/YOUR_USER_NAME/.cshrc
```

and then run

```
% cd /home/YOUR_USER_NAME/
% source .cshrc
```

The latter source command will point the `path` environment variable to the location of the `javac` compiler and the `java` bytecode interpreter. More on `/home/u20/Readme.1st`.