# UNIX Workshop 2010

http://uws.assembla.me

Angad Singh  Aldrian Obaja  Dang Cao Khoa  Chong Perry
Saiaswin Saikannan  Jialong Liu  Vu An Hoa  Gary Kwong  Melvin Zhang

August 2, 2010

"It's a UNIX system! I know this."

– Alexis "Lex" Murphy, Jurassic Park (1993)

## 1 Getting started

Every NUS student is provided with a NUSNET account which gives you access to university wide online services such as the Integrated Virtual Learning Environment (IVLE) and an email account with the nus.edu.sg domain.

In order to login to the PCs in the computer labs, you will need to use your NUSNET account.

**Activity: Login to NUSNET**

1. Press `Ctrl-Alt-Delete`.
2. Type in your NUSNET user name, password and select NUSSTU domain.
3. Click on the `Ok` button.

In addition, every SoC student can create a UNIX account that allows you to remotely access SoC's `sunfire` server. You are allowed to choose your own user name for your UNIX account.

In the following activity, you will create your UNIX account.

**Activity: Creating your UNIX account**

1. Goto `https://mysoc.nus.edu.sg/~newacct` using your NUSNET user name and password.
2. Read through the user-agreement and make sure you understand the obligations.
3. Decide your UNIX user name. Your user name should be between 5-8 characters and must be formed from your name. You may also use your NUSNET user name.
4. Type in your new password (twice).
5. Submit your application.

A number of IT facilities exclusive to SoC students can only be accessed via our `sunfire` server which runs Solaris, a UNIX-like operating system.


Figure 1: Modern UNIX-like operating systems (Linux, BSD, Solaris)

## 2 What is UNIX?

UNIX is the name of the operating system developed by a group of AT&T researchers at Bell Labs in 1969. Solaris and many other modern operating system, such as Linux, Mac OS X, and BSD, are descendants of UNIX. All of these operating systems provide a UNIX-like environment to the user.

Today, many of the operating system, such as the Mac OS X from Apple, Solaris from Sun and GNU/Linux, are descendants of the original UNIX system from Bell Labs (see Figure 1). The operating system you will be using on our UNIX server in SoC is Solaris.

## 3 Connecting to sunfire

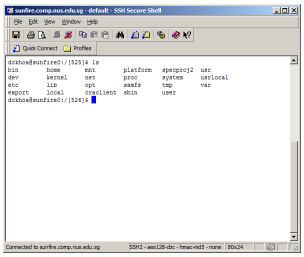
## 4 SoC's `sunfire` server

In the next activity, we will connect to the **`sunfire`** server (see Figure 2) and access the UNIX system via the Secure Shell protocol (ssh).

**Activity: Connecting to `sunfire`**

1. From the desktop, launch the `SSH Secure Shell Client` application.
2. Click on `Quick Connect` Host Name: `sunfire.comp.nus.edu.sg` User Name: your UNIX user name
3. Click on `Connect`.
4. Click on "Yes" at the next dialog.
5. Enter your UNIX password in the password dialog.

Figure 2: `sunfire` server located in the Machine Room with our Networks staff. Clockwise from top-left: Tan Chee Sin, Tan Kwang Pon, Budiman Tsjin (has since left SOC) and Lai Zit Seng.



Once you have logged in, you will see an interface like the following a copy of the shell is started and you are automatically placed in your home directory.

Launching program by typing name follow by parameters. Programs as functions, f(x)

**Command line interface**

- Silence is golden: there is usually no output when a program runs successfully
- Easy to automate repetitive tasks
- Default interface when accessing remote servers

# 5 Text editing

Types of files you will encounter on UNIX include the following:

- a document (report, essay etc.)
- the text of a program written in some high-level programming language
- instructions comprehensible directly to the machine and incomprehensible to a casual user, for example, a collection of binary digits (an executable or binary file)

- a directory, containing information about its contents, which may be a mixture of other directories (subdirectories) and ordinary files.

Text files are used extensively on a UNIX system for storing system data, program configuration files, scripts and source code. They are preferable to binary files because they can be easily read/modified. The most important program is a text editor, a program to interactively create/edit text files. Examples of editors on UNIX are Vim, Emacs, nano, and pico.

In the following activity, we will use nano, http://www.nano-editor.org/, to create a simple text file.

**Activity: Text editing with nano**

1. From the Secure Shell Client window start Vim and create a new file using the command

   ```
   nano hello.txt
   ```

2. Type the following paragraph as carefully as possible.

3. Save the file and exit nano by pressing

   ```
   Ctrl-x
   ```

# 6 Working with text files

**Activity: playing with `diff` and `grep` I**

1. What is `diff`? – compare differences between files

2. Text editing usually leaves a lot of backup files ending with `~`. One day you want to figure out the differences between a file `text` and its backup `text~`... Open in two editors and then eye-ball?

   ```
   diff firstFile secondFile
   ```

3. A quick how-to

**Activity: playing with `diff` and `grep` II**

Let's see how `text` and `text~` look like first

Output of `cat text`

```
same text
same text
Hello World!
still the same
still the same
```

Output of `cat text~`

```
same text
same text
Hello World~
still the same
still the same
```

**Activity: playing with `diff` and `grep` III**

Output of `diff text text~` :

```
3c3
< Hello World!
---
> Hello World~
```

**Activity: playing with `diff` and `grep` IV**

1. What is `grep`? *– look for a pattern in file(s)

   ```
   grep pattern file
   ```

2. Sometimes it is useful to find the occurrences of some word in a (list of) file. Say you suspect a typo in you source code, open a text editor and 'Find'?

3. But what if you made the same typo in a lot of files? 'grep' makes your life easier Let's find out how to 'grep'

**Activity: playing with `diff` and `grep` V**

Sample output of `grep h1 a.html`:

```
grep h1 a.html

  <h1>Hello World!</h1>
```

Contents of a.html

```
<html>
  <body>
    <h1>Hello World!</h1>
  </body>
</html>
```

A bit too easy, isn't it? Ready to get nasty?

1. Output of `grep h1 a.html*`

   ```
   a.html:<h1>Hello World!</h1>
   a.html~:<h1>Hello World~</h1>
   ```

2. Output of `grep -n h1 a.html*`

   ```
   a.html:3:<h1>Hello World!</h1>
   a.html~:3:<h1>Hello World~</h1>
   ```

3. Output of `grep -n -i 'heLlO wORlD' a.html*`

   ```
   a.html:3:<h1>Hello World!</h1>
   a.html~:3:<h1>Hello World~</h1>
   ```

4. Find out more in `'man grep'` !

5. grep on Linux is more fun! :p

# 7 Combing multiple tools

UNIX utilities are simple focused programs operating using a common communication protocol. This is summed up by Douglas McIlroy (inventor of UNIX pipes) as the UNIX philosophy.

**The UNIX Philosophy**

> Write programs that do one thing and do it well.
>
> Write programs to work together.
>
> Write programs to handle text streams, because that is a universal interface.
>
> – Douglas McIlroy

We will see the UNIX philosophy in action in our next activity, which makes use of three UNIX utility programs to analyse SMS messages.

**Activity: SMS Word Count**

Your friend from FASS is studying SMS language as part of a course project. She collected a number of SMS messages and would like to find out the frequency of each word.

For example, given the following text file:

```
U wan 2 haf lunch i'm in da
canteen now.
Haf u found him? I feel so
stupid da v cam was working.
Where r we meeting?
I went to ur hon lab but no
one is there.
```

The desired output is:

```
      .
      .
      .
      1 we
      1 went
      1 Where
      1 working.
      2 da
      2 I
```

**Activity: `sort` and `uniq`**

Two UNIX utility programs are related to our task.

`sort`

| Input: | | Output: |
|---|---|---|
| dog | | bat |
| bat | $\longrightarrow$ | cat |
| log | | dog |
| cat | | log |

`uniq`

| Input: | | Output: |
|---|---|---|
| dog | | dog |
| dog | $\longrightarrow$ | cat |
| cat | | dog |
| cat | | cat |
| dog | | |
| cat | | |
| cat | | |

**Activity: SMS Word Count**

1. Download the file containing sms messages from `http://uws.assembla.me/SMSwords.txt` using wget

   ```
   wget http://uws.assembla.me/SMSwords.txt
   ```

2. Sort the file.

   ```
   sort SMSwords.txt
   ```

3. Sort and remove duplicates.

   ```
   sort SMSwords.txt | uniq
   ```

4. We need to use a particular option of `uniq` which counts the number of duplicates, read the manual page for `uniq`. Press `q` to leave the manual page.

   ```
   man uniq
   ```

5. Sort and count words,

   ```
   sort SMSwords.txt | uniq -???
   ```

6. Sort by the frequency, so that more frequent words appear later,

   ```
   sort SMSwords.txt | uniq -???  | sort -n
   ```

# 8 Summary

Finally, after we are done with what we need to do in the UNIX environment, we have to logout of the system.

**Activity: Logging out of `sunfire`**

To log out of `sunfire`, use the `logout` command,

```
logout
```

# 9 Resources

**Useful programs/websites**

- KiTTY SSH client `http://www.9bis.net/kitty/`
- Computing facilities in SoC `http://docs.comp.nus.edu.sg/cf`
- mySoC (Web services) `https://mysoc.nus.edu.sg`