Unix Environment
An Advanced Introduction to Unix/C Programming

Dennis Ritchie   Ken Thompson   Linus Torvalds   Richard Stallman   Brian Kernighan

John Dempsey
COMP-232 Programming Languages
California State University, Channel Islands

Before You Can Login

Your Unix administrator will create an entry in the /etc/passwd file assigning you a unique user id and will set your primary group, your home directory, and your default shell.

% **sudo adduser julie**

% **sudo adduser --home /home/amy --shell /bin/bash --uid 1001 --gid 50 amy**

Adding user `amy' ...

Adding new user `amy' (1001) with group `staff' ...

Creating home directory `/home/amy' ...

Copying files from `/etc/skel' ...

New password: **xxxxxxxx**

Retype new password: **xxxxxxxx**

passwd: password updated successfully

Changing the user information for amy

Enter the new value, or press ENTER for the default

    Full Name []: **Amy Smith**

    Room Number []:

    Work Phone []: **805-111-2222**

    Home Phone []: **805-333-4444**

    Other []: **Remote**

Is the information correct? [Y/n] **Y**

john@oho:~$ **grep amy /etc/passwd**

amy:x:1001:50:Amy Smith,,805-111-2222,805-333-4444,Remote:/home/amy:/bin/bash

john@oho:~$ **ls -ld /home/amy**

drwxr-xr-x 1 amy staff 4096 Dec 27 14:46 /home/amy

Logging In

- After the Unix administrator provides you with your user id and password, you can login.
- You should change your password using the passwd command, like so:

login: **amy**

Password: **NiceTry123**

Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 4.4.0-19041-Microsoft x86_64)

amy@oho:~$ **id**

uid=1001(amy) gid=50(staff) groups=50(staff)

amy@oho:~$ **grep amy /etc/passwd**

amy:x:1001:50:Amy Smith,,805-111-2222,805-333-4444,Remote:/home/amy:/bin/bash

amy@oho:~$ **grep staff /etc/group**

staff:x:50:

amy@oho:~$ **passwd**   ← **passwd is used to change your password.**

Changing password for amy.

Current password: **NiceTry123**

New password: **NewPassword123**

Retype new password: **NewPassword123**

PATH

PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin

When you type in a command, above directories are searched in order to locate the command to run.

If two directories have the same command name, the first command found in the PATH is run.

What's missing in the above PATH?

PATH
What's missing?

If you type a command found only in your directory, you'll see "command not found" message.

amy@oho:~$ **cat c.c**

#include <stdio.h>

int main()

{

    printf("Hello World!\n");

}

amy@oho:~$ **gcc c.c**

amy@oho:~$ **ls**

a.out c.c

amy@oho:~$ **a.out**   **← See "a.out: command not found"**

a.out: command not found

amy@oho:~$ **./a.out**   **← To run, use ./a.out.   The ./ says to look in my current directory only.**

Hello World!

How To Correct?

To correct, you can update your PATH by typing:

% **PATH=.:$PATH**

or

% **PATH=$PATH:.**

But wait! One is correct. The other is wrong. Which is best?

Correct Way

% **PATH=$PATH:.**

% **echo $PATH**

/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:.

A hacker can create an executable file, named say "ls", and add it to one of your directories. When you run "ls", you could run their "ls" program using all of your permissions instead of the actual /bin/ls command.

By adding the dot (.) to the end of your PATH, you prevent this attack from happening.

But How Are Environment Variables Initialized?

When you login, environment variables, like $PATH, are set by running these scripts in the following order:

1.	**/etc/profile**	← System profile for all users.
2.	**/etc/environment**	← Sets the initial PATH for all users.
3.	**~/.profile**	← Your .profile file found in your home directory.
4.	**/etc/bashrc**	← System bashrc file.
5.	**~/.bashrc**	← Your .bashrc file in your home directory.

As a user, you can edit and update your .profile and .bashrc files, e.g., to add . to your PATH. (.bashrc is used if logging in using /sbin/bash.)

% **more /etc/environment**

PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin"

~/.profile

You can set your Unix prompt (PS1), directories, and aliases in your home directory.
----

export PS1='% '   ← Just % as the prompt.
export PS1="${USER}:${PWD}: "   ← This prompt displays user name, directory

export ORACLE_HOME=/oracle/product/19.20   ← Need for PATH
export ORACLE_SID=MYDB

PATH=$PATH:$ORACLE_HOME/bin:.

alias src='cd /home/julie/project_snowcrab/src'
alias n='cat notes.txt'

Type env to display environment. Type aliases to view aliases defined.

Your Unix Environment

Typing "env" will display your environment variables.

amy@oho:~$ **env**

SHELL=/bin/bash   ← You're using the bash shell.

PWD=/home/amy   ← Print Working Directory (displays the current directory you are in).

LOGNAME=amy   ← Name of the user logged in.

MOTD_SHOWN=update-motd   ← Runs message of the day (MOTD) scripts found in /etc/update-motd.d.

HOME=/home/amy   ← Your home directory.

LANG=C.UTF-8   ← Computer. Unified Transformation Format (UTF) – 8-bit variable width
character set.

LS_COLORS=rs=0:di=01;34:ln=01;36: … ← Colors used for the ls command output.

TERM=xterm-256color   ← Terminal type set to xterm-256color.

LESSOPEN=| /usr/bin/lesspipe %s

USER=amy   ← Your user id amy

PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin

MAIL=/var/mail/amy   ← Filename where your email is stored on the system.

~/.vimrc

Configure your vim editor using .vimrc file.

---

set noswapfile

set background=dark

set tabstop=4

syntax on

colorscheme evening