

Installing Applications in FreeBSD

lctseng

Before we start

❑ Permission issue

- root: the super user
- Like administrator in M\$ Windows

❑ Don't execute commands as root directly

- It's **DANGEROUS**

❑ But sometimes you still need to be root to do something

- Install software
- Manage system file
- Create/modify users

Before we start

❑ Become root

- Console login with root
- By default, cannot login with root via SSH

❑ Change current user

- Don't need to login with console
- Use command 'su -', and then type root's password
- To see which credit you are using, use 'whoami'

```
1:55 lctsens@lctsens(10.0.2.15)[~]
[^_^] > whoami
lctsens
1:55 lctsens@lctsens(10.0.2.15)[~]
[^_^] > su -
Password:
1:55 root@lctsens(10.0.2.15)[~]
[^_^] # whoami
root
1:55 root@lctsens(10.0.2.15)[~]
[^_^] #
```

Before we start

- ❑ As mentioned before, don't run as root directly

- ❑ Can we execute with root's credential only for some specific commands?
 - Like 'Run as administrator' in Windows
 - Is there similar commands in FreeBSD?

Before we start

- ❑ Run commands with other user's permission

- ❑ 'sudo' command
 - Only simplest explanation here for basic usage
 - 'sudo' syntax and other details will explain in later chapter
 - Here only tell you how to simply enable 'sudo'

- ❑ How to enable sudo?
 - 'sudo' is not a built-in command, need to be installed manually

Before we start – Enable ‘sudo’

❑ Install the package

- Check Internet connection
 - `ping 8.8.8.8`
- Become root (`su -`)
- Execute `pkg install sudo`
 - This will install ‘sudo’ from Internet
 - Type ‘Y’(means yes) when it ask for conform

Before we start – Enable ‘sudo’

❑ Allowing your user to execute ‘sudo’

- Switch to root first
- Type ‘setenv EDITOR ee’ to change your editor for this time
 - Will explain this in later chapter
 - This will allow you to use a **notepad-like** editor
 - If you are familiar with default editor ‘vi’, just skip this step
- Type ‘visudo’ to edit the sudoer file
 - Specific who can use ‘sudo’
 - Around line 88, type ‘lctseng ALL=(root) ALL’
 - Please replace ‘lctseng’ with your own username
- Save the file and exit, back to normal user
 - Use ‘logout’ command or press Ctrl+D

Before we start – Using ‘sudo’

- ❑ Now, you can prepend ‘sudo’ before commands to run them as root
 - But please **think carefully before you type**

- ❑ Execute commands with ‘sudo’
 - `$ sudo whoami`
 - You have **root’s credential**
 - `$ sudo pkg install vim`
 - Install software without become root directly
 - You need to re-type your password
 - Don’t need to re-type within 5 minutes

Install software

❑ Package

- Pre-built
 - Like most of installer (.msi) in Windows
- Other unix-like system: rpm, yum, dpkg, ...
- FreeBSD: pkg
 - Don't use old pkg_* commands!

❑ Source

- Compile the source files first and then install
- Tar ball, a pack of source code
- `tar -xzf certain-source.tar.gz`
- `cd certain-source`
- `./configure --help`
- `./configure [options ...]`
- `make`
- `make install` (root permission)

Install software : Overview

- ❑ Three technologies
 - Packages
 - Ports
 - Tar ball
- ❑ Packages
 - pre-built ports, contain **pre-compiled** copies of all the commands for the application, as well as any configuration files or documentation.
- ❑ Ports
 - a collection of files designed to **automate** the process of **compiling** an application from source code and **additional patches**
 - a set of Makefile, patches, description files, ...
- ❑ Both packages and ports understand ***dependencies***
- ❑ Tar ball
 - fetch it, configure the installation options, and compile it by yourself. **NO DEPENDENCY CHECKING.**

Overview

❑ Package benefits

- Packages do not require any additional compilation
 - Benefit for slow machines

❑ Ports benefits

- You can tweak the compilation options to generate code that is **specific** to a different processor – speed
- Some applications have compile time options relating to what they can and cannot do – customization

❑ Why tar ball?

- Some software cannot be found in port collections
- Some latest version of software may have new configurations that do not exist in port

Package System (1)

- ❑ pkg
 - New generation of FreeBSD package system
- ❑ Install new software
 - Fetch packages from Internet
 - `pkg install <names of packages...>`
 - `pkg install vim screen tmux`
 - Run with root's permission (sudo)
 - Automatically update the database
 - Perform dependency check
 - Will install software that required by new software

Package System (2)

❑ Upgrade currently installed software

- `pkg upgrade <names of packages...>`
 - `pkg upgrade vim`
- `pkg upgrade`
 - Upgrade all installed software
- This will also update the database

❑ Update packages database only

- `pkg update`

❑ Delete a package

- `pkg delete <names of packages>`
 - `pkg delete php53`
- Will check dependency
- `pkg delete -f <names of packages>`
 - `-f`: force
 - Disable dependency check

Package System (3)

❑ Show information about installed packages

- `pkg info`
 - Show all installed packages
 - Use 'grep' to find specific packages
 - `pkg info | grep vim`
- `pkg info <name of package>`
 - Show detailed information
 - `pkg info php56`

❑ Show version of installed packages

- `pkg version`
 - `pkg version -v`

```

[...]
```

<code>apache24-2.4.16_1</code>	➤ Show detailed information
<code>apr-1.5.2.1.5.4</code>	➤ succeeds port (port has 2.4.10_2)
<code>aspell-0.60.6.1_5</code>	➤ <code>pkg info php56</code>
<code>aspell-ispell-0.60.6.1</code>	➤ succeeds port (port has 1.5.1.1.5.4)
<code>atk-2.16.0</code>	= up-to-date with port
<code>autocore-2.60</code>	= up-to-date with port

```

[...]
```

❑ Show version of installed packages

- `pkg version`
 - succeeds port (port has 2.14.0)
 - up-to-date with port

How to use ports

- ❑ Obtain the ports collection
 - List of ports available to be installed into system
- ❑ We should...
 - Find the application
 - Change to the directory for the port
- ❑ Ports will
 - Fetch the tar ball
 - Ask for configuration friendly
 - Compile the source code
 - Install your application

Obtaining the Ports Collection

□ Port directory

- /usr/ports/<category>/<name>

```

14:47 lctseng@lctseng(10.0.2.15)[/usr/ports]conf
[^^] > ls
CHANGES          Tools/           chinese/        ftp/            mail/           ports-mgmt/
CONTRIBUTING.md  UIDs            * http://www.freebsd.org/games/handbook/math/ap.html  portuguese/
COPYRIGHT         UPDATING        converters/     german/         misc/           print/
GIDs             accessibility/  databases/     graphics/       multimedia/     russian/
Keywords/        arabic/         deskutils/     hebrew/         net/            science/
LEGAL            archivers/      devel/          hungarian/      net-im/         security/
MOVED            astro/          dns/           irc/            net-mgmt/       shells/
Makefile         audio/          editors/        japanese/       net-p2p/        sysutils/
Mk/             benchmarks/    emulators/     java/           news/           textproc/
README          biology/       finance/        korean/         palm/           ukrainian/
Templates/      cad/           french/        lang/           polish/         vietnamese/
14:47 lctseng@lctseng(10.0.2.15)[/usr/ports]

```

```

14:49 lctseng@lctseng(10.0.2.15)[/usr/ports/editors/vim]
[^^] > ls
Makefile  distinfo  files/    pkg-descr  pkg-plist

```


Obtaining the Ports Collection

❑ portsnap(8)

- Fetch and update your port tree
- fetch, extract, update, cron
- `$ portsnap fetch extract update`
 - With root permission
- `/etc/portsnap.conf`
 - `$ sudo ee /etc/portsnap.conf`
 - `SERVERNAME=portsnap.tw.FreeBSD.org`

Ports system (1)

□ Find your application

- `cd /usr/ports`
- `make search name=program name`
- `make search key=string`

```

15:26 lctseng@lctseng(10.0.2.15)[/usr/ports]/ports]
[^^] [ >_make> search name=zh-mutt vim-lite
Port: P0zh-mutt-1.5.24-7.4.865_1
Path: Pa/usr/ports/chinese/mutt vim-lite
Info: InThe Mongrel of Mail User Agents with Chinese support (age)
Maint: Maports@FreeBSD.org FreeBSD.org
B-deps: B-autoconf-2.69 autoconf-wrapper-20131203 automake-1.15 auto
5 docbook-sgml-4.5_1 docbook-xml-5.0_3 docbook-xsl-1.76.1_2 expat-
-0.2.3 wiso8879-1986_3 libgcrypt-1.6.3 libgpg-error-1.20 libiconv-1
,1 m4-1.4.17_1,1 perl5-5.20.3_8 sdocbook-xml-1.1_1,2 xmlcatmgr-2.2
R-deps: cyrus-sasl-2.1.26_9 db5-5.3.28_2 gettext-runtime-0.19.5.1
autoconvert-0.3.16_5
WWW: http://www.mutt.org/

```

Ports system (2)

❑ psearch(1)

- Simple but useful tool to find ports
- ports-mgmt/psearch
 - Install it before you use
- pSearch <portname>
 - \$ psearch vim

```

3:02pm lctseng@nctucs [~]
[W1] > psearch vim
audio/vitunes
devel/clewn
devel/p5-Shell-EnvImporter
editors/cream
editors/neovim
editors/p5-Vimana
editors/vim
editors/vim-lite
Curses-based media player with vim-like keybinds
Clewn provides Gdb support within Vim
Inherit different shell environments and restore previous
Gvim extension with many features
Next generation Vim
make install clean
Improved version of the vi editor
Improved version of the vi editor (lite package)

```

Ports system (3)

❑ Type “make install clean” to install your application

- make config (/var/db/ports/)
- make fetch (/usr/ports/distfiles/)
- make checksum
- make extract
- make patch
- make configure
- make build
 - “make” means all of the above
- make install
- make clean
- make distclean
 - Clean files generated by configure process

Ports system (4)

❑ The ports system uses [fetch\(1\)](#) to download the files

- **MASTER_SITES** environment variable
- /etc/make.conf

```
MASTER_SITE_BACKUP?= \
    http://FreeBSD.cs.nctu.edu.tw/distfiles/${DIST_SUBDIR}/
MASTER_SITE_OVERRIDE?= ${MASTER_SITE_BACKUP}
```

❑ Options for ports

- make config
 - Won't build or install the port
 - Use this to re-configure ports (otherwise , it uses old one instead)
- hidden options (not shown in 'make config')
 - Edit the Makefiles under that port directory

Ports system (5)

- ❑ I installed the application but
 - Command not found...
 - Logout, and then login.
 - If you use (t)osh
 - rehash

Deinstall Applications

❑ Two methods

- pkg delete
 - Find the package name via pkg info
 - Dependency check
 - pkg delete -f : disable dependency check
- make deinstall
 - Change to the ports directory
 - make deinstall
 - Delete it anyway
 - Similar to ‘ pkg delete -f ’

Upgrading Ports using Portmaster

❑ ports-mgmt/portmaster

- A utility for easily upgrading and installing ports
- It was designed to make use of the tools found in the base system
 - ports-mgmt/portupgrade
- ```
$ cd /usr/ports/ports-mgmt/portmaster
$ make install clean
```

## ❑ Install or upgrade a port

- ```
$ portmaster sysutils/lsof
```
- /usr/ports/UPDATING
 - **Read before you update any port or software!!**

❑ Useful options

- -B, -D, -a, -r, -y, -H, -w
- ```
$ portmaster -dyBwH editors/vim
```



# Package/Port Security Issue

- ❑ Show security issues about installed packages
  - No matter from port or from package
  - `pkg audit`
  - Upgrade these packages to avoid security problems

```
2:36pm lctseng@nctucs [~]
[W1] > pkg audit
gstreamer1-libav-1.4.5 is vulnerable:
ffmpeg -- out-of-bounds array access
CVE: CVE-2015-3395
WWW: https://vuxml.FreeBSD.org/freebsd/80c66af0-d1c5-449e-bd31-63b12525ff88.html

gstreamer1-libav-1.4.5 is vulnerable:
ffmpeg -- use-after-free
CVE: CVE-2015-3417
WWW: https://vuxml.FreeBSD.org/freebsd/da434a78-e342-4d9a-87e2-7497e5f117ba.html

1 problem(s) in the installed packages found.
```

# Try to install from ports

---

- screen, tmux
- vim, emacs
- mutt
- wget, curl
- lftp
- lynx, w3m
- expect
- zh-telnet