Hardening servers for the modern internet

Philip Paeps

The FreeBSD Foundation

SANOG32 – 7 August 2018 – Dhaka, Bangladesh

Session 1 (09:00 – 11:00)

1. Presentation: Introduction to the FreeBSD project (30 minutes)
2. Presentation: FreeBSD tips for Linux refugees (30 minutes)
3. Lab: Installing a FreeBSD server in a virtual machine (60 minutes)

Goal: have FreeBSD installed on your laptop before the tea break.

Session 2 (11:30 – 13:00)

1. Presentation: Introduction to the pf packet filter (60 minutes)
2. Lab: pf exercises

Goal: have pf configured on your laptop before lunch.

Session 3 (14:30 – 16:00)

1. Presentation: Introduction to jails and ezjail (60 minutes)
2. Lab: Configuring jails and ezjail in your virtual machines (30 minutes)

Goal: have jails on your laptop before the tea break.

Session 4 (16:30 – 18:00)

1. Presentation: introduction to letsencrypt.org (30 minutes)
2. Demo: getting a letsencrypt.org certificate in a jail (30 minutes)
3. Lab: get a letsencrypt.org certificate for your jail (30 minutes)

Goal: Have an HTTPS secured webserver by the end of the day

Documentation everywhere

- Online documentation
  - Installed by default
  - Primarily Unix man pages
- Cookbook-style FreeBSD Handbook
  https://www.freebsd.org/doc/handbook/book.html
- FreeBSD Wiki
  https://wiki.freebsd.org/

Finding your way around the system (1)

`/boot`	Bootloader configuration, kernel and kernel modules
`/bin, /sbin`	Essential user commands
`/dev`	Device special files (managed by `devfs(5)`)
`/etc`	System configuration files and scripts
`/lib, /libexec`	Critical system libraries/utilities
`/tmp`	Temporary files not guaranteed to persist across reboots
`/usr`	Majority of user utilities and applications
`/var`	Multi-purpose log, temporary, transient, and spool files

Finding your way around the system (2)

- Third-party applications installed by pkg(8) live in `/usr/local`
- `/proc` is not mounted by default
- Runtime kernel parameters managed by `sysctl(8)`
- No systemd!

The `pkg(8)` package manager

- New since FreeBSD 10.0-RELEASE (2014)
- Basic commands
  - `pkg install`
  - `pkg remove`
  - `pkg upgrade`
  - `pkg audit`

- See also chapter 4 of the FreeBSD Handbook
  https://www.freebsd.org/doc/handbook/ports.html

Starting and stopping services

- System configuration lives in `/etc/rc.conf`
- The `sysrc(8)` utility can be used to configure services
- Services are started and stopped with the `service(8)` utility

```
sysrc sshd_enable="YES"
service sshd start
service sshd stop
```

Network configuration

- Boot-time configuration lives in `/etc/rc.conf`

```
sysrc ifconfig_em0="DHCP"
sysrc ifconfig_em0="inet 192.0.2.42/24"
sysrc ifconfig_em0_ipv6="inet6 2001:db8::42/64"
```

- Run-time configuration is managed by `ifconfig(8)`
- ISC DHCP client is available out of the box – `dhclient(8)`

Further documentation in the manual pages and in the FreeBSD handbook!

Tips for Linux refugees (1)

- vi is really vi

  `pkg install vim` if you prefer Vim
  or `pkg install emacs` if you enjoy pain

- sh is really sh

  `pkg install zsh` if you prefer zsh
  or `pkg install bash` if you enjoy pain

- root's shell is csh

  Don't mess with that!

FreeBSD FOUNDATION

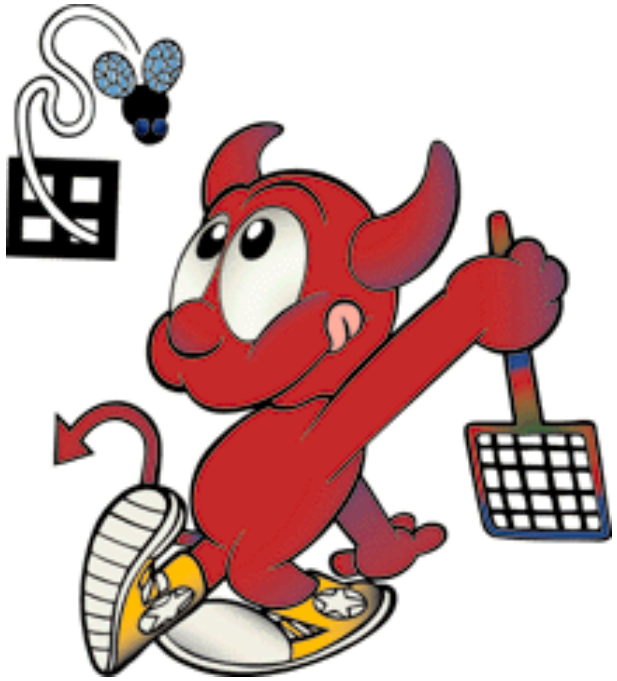
Tips for Linux refugees (2)

- FreeBSD aims to be consistent and self-documenting
- Warnings and errors generally provide clues to the solution
- Consulting man pages is often much faster than Google

The FreeBSD community is very helpful!

Subscribe to the questions@FreeBSD.org mailing list
https://lists.freebsd.org/mailman/listinfo/freebsd-questions

Keeping up to date (1)



The FreeBSD security team regularly issues security advisories and errata notices for supported FreeBSD releases.

- Stable branches (e.g. X-STABLE) are supported for five years after X.0-RELEASE
- Individual point releases (e.g. X.0-RELEASE) are supported for three months after the next point release (e.g. X.1-RELEASE)

Keeping up to date (2)

- Use `freebsd-update(8)` to patch a supported -RELEASE

```
freebsd-update fetch
freebsd-update install
```

- Check for updates nightly from `/etc/crontab`

```
0 3 * * * root /usr/sbin/freebsd-update cron
```

NOTE WELL!
This only checks for updates, it does not install them!

Keeping up to date (3)

- Upgrade to a newer supported -RELEASE

  `freebsd-update upgrade -r X.Y-RELEASE`

  Follow instructions for merging configuration files and rebooting

`freebsd-update(8)` usually will not destroy your system but…

There is no excuse for not having (tested) backups!

Installing your first FreeBSD server in a virtual machine

Downloading FreeBSD

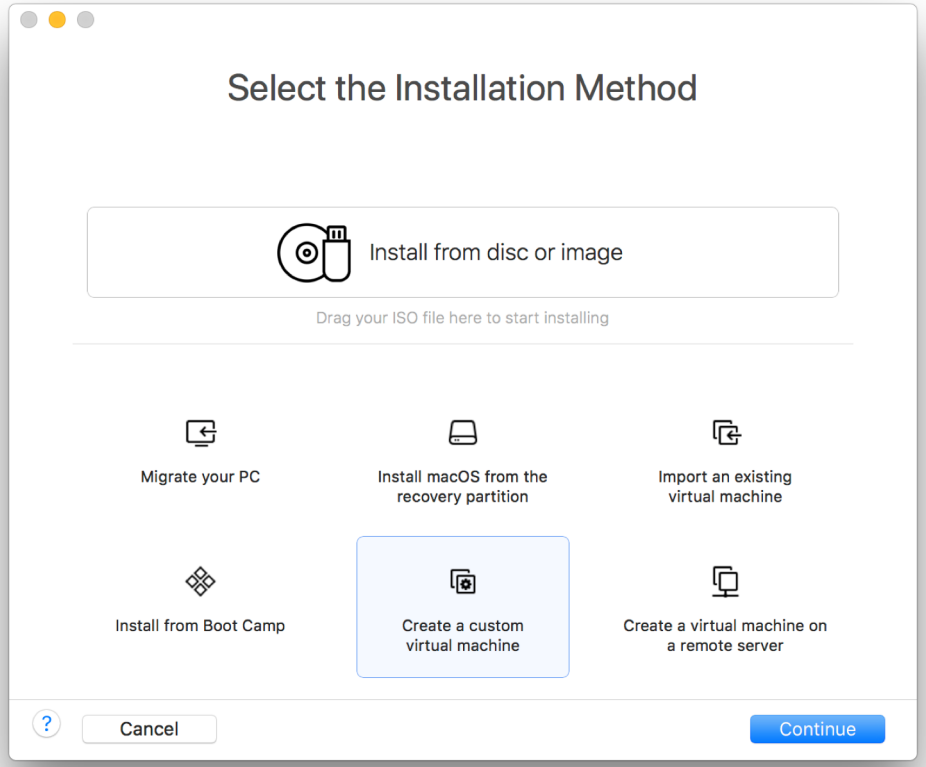
On the SANOG32 wireless network

http://172.16.0.246

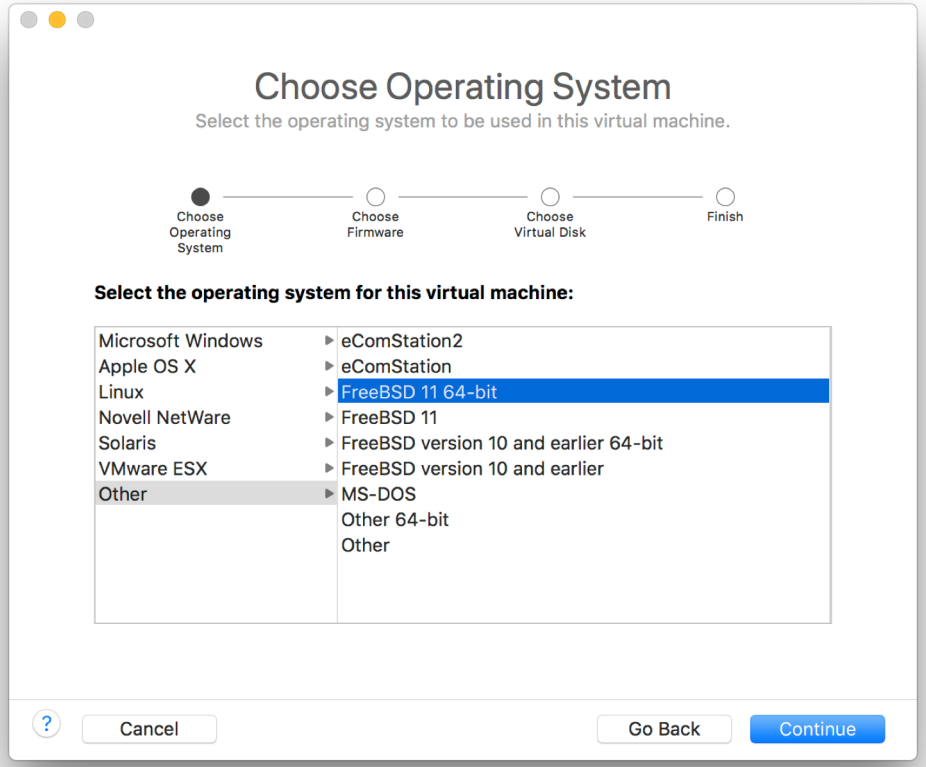
VMware	FreeBSD-11.2-RELEASE-amd64.vmdk.xz
VirtualBox	FreeBSD-11.2-RELEASE-amd64.vhd.xz
CD-ROM ISO image	FreeBSD-11.2-RELEASE-amd64-disc1.iso.xz
USB memstick image	FreeBSD-11.2-RELEASE-amd64-memstick.img

FreeBSD FOUNDATION

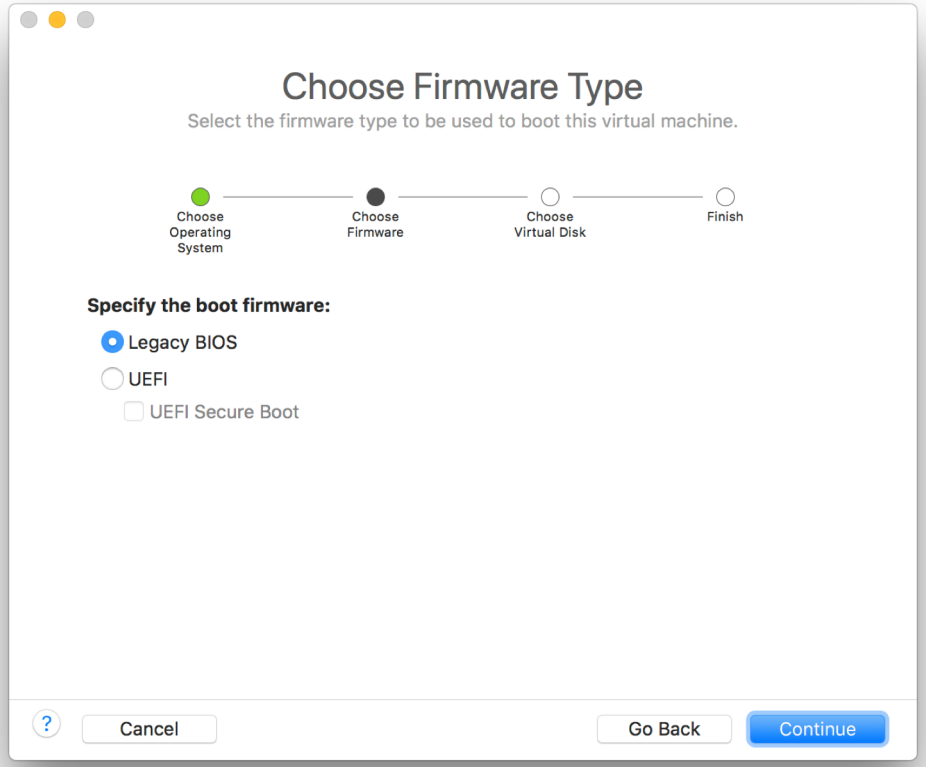
Installing FreeBSD in VMware (1)



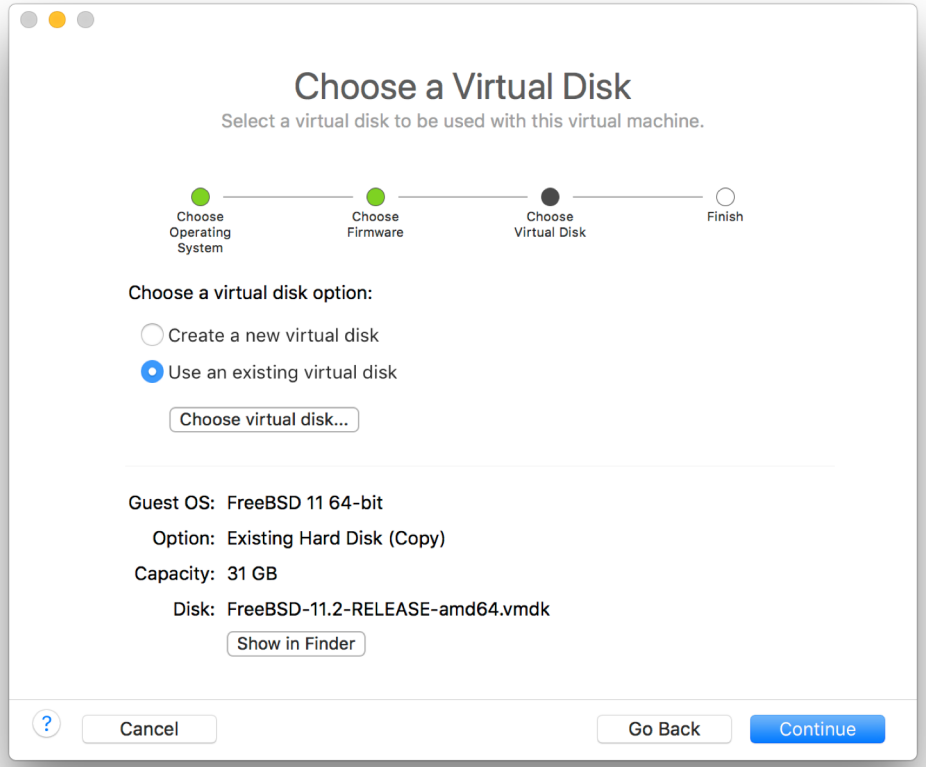
Installing FreeBSD in VMware (2)



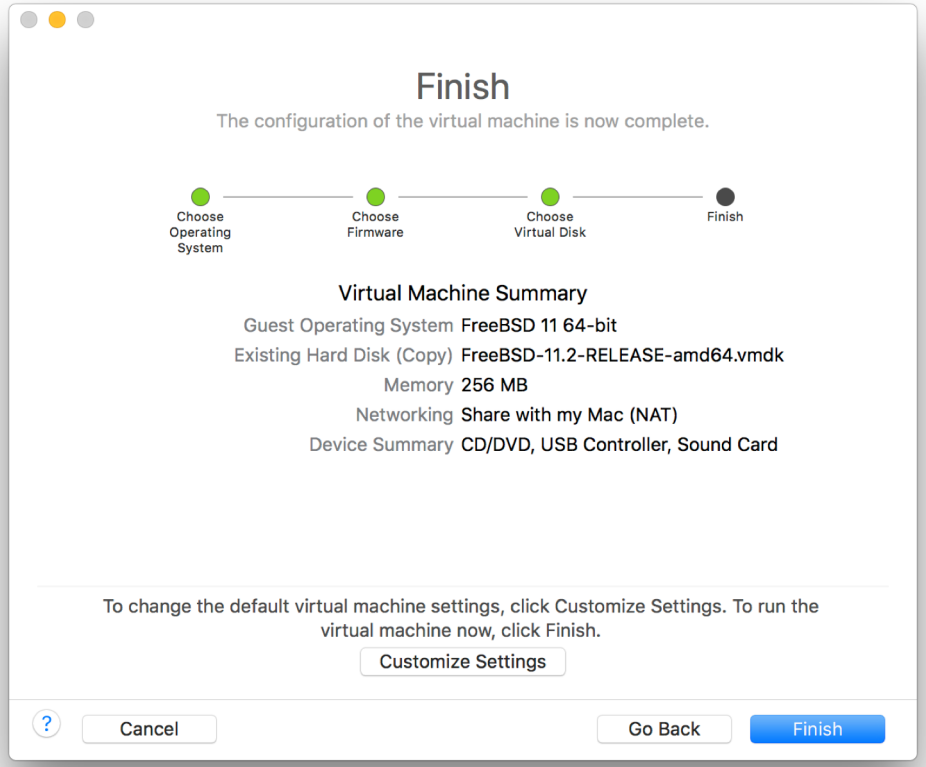
Installing FreeBSD in VMware (3)



Installing FreeBSD in VMware (4)



Installing FreeBSD in VMware (5)



Exercises (1)

1. Install FreeBSD in a virtual machine on your own laptop
2. Connect your virtual machine to the SANOG32 network
   - Bridge your virtual machine to the wireless network
   - Use **`dhclient em0`** to get an IP address
   - Ping a machine on the internet to verify it works
   - Make your configuration persistent in `/etc/rc.conf`
   - Reboot and verify again!

FreeBSD FOUNDATION

Exercises (2)

1. Install the nginx web server using `pkg(8)`
2. Enable nginx in `/etc/rc.conf`
3. Start nginx
4. Verify you and your neighbour can connect to your web server
5. Reboot your virtual machine to verify everything still works

The pf packet filter

High-level overview

- Operate on the IP and transport layers of the network stack
- Packets are recognised based on
  - Source and destination address
  - IP protocol (TCP, UDP, ICMP,…)
  - Source and destination port (TCP, UDP)
  - Type and code (ICMP)
- Matching packets are either allowed or refused

History of pf

- Originally developed on OpenBSD to replace ipfilter (2001)
- Primary goals
  - Security
  - Ease of configuration
- Imported in FreeBSD (2003)
- Optimised for FreeBSD (2003–present)
  - Performance
  - Multi-threading

FreeBSD FOUNDATION

Finding pf documentation

- pf on FreeBSD is not in sync with pf in OpenBSD!
- Blindly following OpenBSD documentation will not work
  - No “match” keyword on FreeBSD
- FreeBSD handbook and online man pages are accurate

pf features

- (Switch to mlaier's slides from SUCON)

pf exercises (1)

1. Make a clone of your virtual machine
2. Configure both virtual machines to NAT through your laptop
3. Write down the IP addresses of your two virtual machines
4. Write a pf filter to block all but outgoing traffic
    1. Test that this works!
5. Change your filter on one VM to allow ssh from the other VM
6. Change your filter on one VM to allow HTTP from the other VM

FreeBSD FOUNDATION

pf exercises (2)

- Configure NAT on one VM and route through the other VM
- Block all traffic to your web server from 192.168.42.0/24

Configuring pf

- All configuration is kept in /etc/pf.conf (one single file)
- Order matters – but not very much
  - Settings
  - Macros
  - NAT
  - Rules
- Syntax is simple and easy to understand

Jails: light-weight multi-tenancy

What are jails

- Essentially a chroot(8) with an IP address
- Actually: any number of IP addresses
  - IPv4 or IPv6
- Simply: confine a process and all its children to a sandbox

Configuring jails

- Manually using /etc/jail.conf and the jail command
- More usefully using ezjail

- https://www.freebsd.org/doc/handbook/jails-ezjail.html
- https://www.freebsd.org/doc/handbook/jails.html