

# Fifteen Years and Fifteen Minutes: Applying Occam's Razor to FreeBSD with OccamBSD

@MichaelDexter

NYC\*BUG 2021-02-03

“Will think critically for food” – Occam’s Razor

*“Pluralitas non est ponenda sine necessitate”*

“Plurality should not be posited without necessity”

“The law of parsimony”

“The simplest explanation is usually the right one”

# “Will think critically for food” – Occam’s Razor

“Simple theories are easier to verify,  
simple solutions are easier to execute”

“No more assumptions should be made  
than are necessary”

“Keep It Simple”

“Less is more, `more(1)` or `less(1)`”

# Berkeley Software Distribution (BSD)

A largely-permissively licensed unified Unix operating system originating at the University of Berkeley and expanded by many, many people around the world

# Background, briefly

“4.3BSD in college was clearly a behind-the-camera view of the computer. *That* is a very familiar concept.”

“Coherent Unix was interesting but limited.”

“UnixWare looks great on paper.”

“The RedHat 5.1 documentation does not match the product.”

# Background, briefly

“RedHat 5.2 is a great all-around Unix clone with a packaged base. Rejoice!”

“My search is over!”

## Background, briefly

“RedHat 6.0 turned a good Unix clone into a stunningly-bad Windows clone with GNOME.”

“Binary updates and upgrades will come someday.”

“RPM HELL” and `linuxconf`

# Solution? Move to Latvia

The pitch for \*BSD in magazines and the Walnut Creek CD-ROM catalog was compelling but I didn't know anyone.

The "assman@" chroot jail did not sound workable.

jail(8) however sounds like a solution

# BSD.lv

`jail(8)` on 4.7/4.8 is very promising!

`jail(8)` on 5.0 with `jls(1)` and  
`jexec(8)` are ALL THE THINGS

“My search is *finally* over!”

# The BSD/FreeBSD Value Proposition

Before ZFS, DTrace, CTL, `gpart(8)`, PF, and...

Largely Permissively-Licensed

Unified OS/Unified Build

`jail(8)`

# An Excellent Narrative

*“A permissively-licensed OS that is perfect for embedded products and the things we will call ‘containers’ and ‘IoT’ many years from now.”*

*“Simply set a few build and `KERNCONF` options to reduce the system to its bare minimum and add what you need.”*

*“Just like `block all` on your packet filter.”*

*AWESOME!!! Just add a hypervisor!*

About that...

FreeBSD 5.0 with the j-tools delivered  
uptimes measured in days

Those *build options* didn't quite work

That was early 2003

# *Interlude*

OpenBSD SysJail with Kristaps

NetBSD/Xen for mult and general use

Patience

More patience...

BHyVe

# Updated Narrative

The highly-flexible, permissively-licensed  
OS with a “container” mechanism  
now has a hypervisor

**TIME TO REVISIT THOSE BUILD OPTIONS**

That was 2012 and narrative remains valid

# Build Options

`man src.conf`

`WITHOUT_ACCT`

Set to not build process accounting tools such as `accton(8)` and `sa(8)`.

`WITHOUT ACPI`

Set to not build `acpiconf(8)`, `acpidump(8)` and related programs.

`WITH_...`

230+ Options in total for 13.0

# Build Options

*“Are you absolutely sure I need to build CLANG/LLVM to test this one thing?”*

*“This thing we are shooting into space (er, digital sign) doesn’t speak TCP/IP...”*

*(Insert clever yet discouraged-for-years use case)*

# Build Options

*“The resulting system will be totally  
unrecognizable!”*

*Actually...*

# The Minimum OS

Documenting it? – Start here

Auditing it? – Start here

Learning it? – Start here

Fast forward to today, early 2021

Years of testing, reporting, nudging, and repairing has resulted in massive progress

FreeBSD 13.0-RELEASE will... (did we make it?)

~~Fifteen~~ Eighteen\* Years in the Making

OccamBSD: An application of  
Occam's Razor to FreeBSD

a.k.a. "super svelte stripped down FreeBSD"

`github.com/michaeldexter/occambsd`

\*Insert vote/drive/drink/buy a gun jokes here

# OccamBSD

`occambsd.sh`

`/usr/src/tools/tools/  
build_option_survey/listallopts.sh`

# The Build Option Survey (BOS) August, 2005

```
/usr/src/tools/tools/  
build_option_survey/
```

```
listallopts.sh
```

```
option_survey.sh
```

```
reduce.sh
```

```
mkhtml.sh
```

# The Build Option Survey (BOS) August, 2005

WITHOUT_CXGBETOOL=foo	MK_CXGBETOOL = no	+0	-0	<a href="#">*1577</a>	3426728	-100	+0	<a href="#">-3</a>	<a href="#">*1</a>	3423676	-3152	+0	<a href="#">-3</a>	<a href="#">*1578</a>	3423576	-3252
WITHOUT_CXX=foo	MK_CLANG = no MK_CLANG_FULL = no MK_CXX = no MK_GOOGLETEST = no MK_LLVM_COV = no MK_TESTS = no	failed					+0	<a href="#">-8501</a>	<a href="#">*3</a>	1889620	-1537208	+0	<a href="#">-8501</a>	<a href="#">*1186</a>	1889580	-1537248
WITHOUT_DEBUG_FILES=foo	MK_DEBUG_FILES = no	failed					+0	<a href="#">-1902</a>	<a href="#">*21</a>	2054824	-1372004	+0	<a href="#">-1932</a>	<a href="#">*2923</a>	1519148	-1907680
WITHOUT_DIALOG=foo	MK_BSDINSTALL = no MK_DIALOG = no	failed					+0	<a href="#">-222</a>	<a href="#">*1</a>	3416836	-9992	+0	<a href="#">-222</a>	<a href="#">*1566</a>	3416704	-10124

[callfortesting.org/results/](http://callfortesting.org/results/)

# The BOS-ng December, 2019

`github.com/michaeldexter/bos-ng`

`calctimes.sh`

`findfaults.sh`

## *With the hard parts out of the way...*

```
sh /usr/src/tools/tools/build_option_survey/listallopts.sh | \  
grep -v WITH_ | sed 's/$/=YES/' | \  
grep -v WITHOUT_AUTO_OBJ | grep -v WITHOUT_UNIFIED_OBJDIR | \  
grep -v < INSERT BROKEN OPTIONS HERE > | \  
grep -v WITHOUT_BOOT | grep -v WITHOUT_LOADER_LUA | \  
grep -v WITHOUT_LOCALES | \  
grep -v WITHOUT_ZONEINFO | \  
grep -v WITHOUT_VI \  
> /etc/src.conf
```

# *With the hard parts out of the way...*

<code>/tmp/occambsd</code>	
<code>src.conf</code>	OccamBSD <code>src.conf</code>
<code>OCCAMBSD</code>	OccamBSD kernel configuration file
<code>bhyve-kernel</code>	bhyve kernel directory
<code>bhyve-mnt</code>	bhyve disk image mount point
<code>bhyve.raw</code>	bhyve disk image with kernel
<code>load-bhyve-vmm-module.sh</code>	Script to load <code>vmm.ko</code>
<code>load-bhyve-disk-image.sh</code>	Script to load bhyve kernel from disk image
<code>load-bhyve-directory.sh</code>	Optional script to load bhyve kernel from
<code>directory</code>	
<code>boot-occambsd-bhyve.sh</code>	Script to boot bhyve
<code>destroy-occambsd-bhyve.sh</code>	Script to clean up the bhyve virtual machine

# *With the hard parts out of the way...*

/tmp/occambsd

xen-kernel	Xen kernel directory
xen-mnt	Xen disk image mount point
xen.raw	Xen disk image with kernel
xen-occambsd.cfg	Xen disk image boot configuration file
xen-occambsd-kernel.cfg	Xen directory boot configuration file
boot-occambsd-xen.sh	Script to boot Xen kernel from disk image
boot-occambsd-xen-kernel.sh	Script to boot Xen kernel from directory
destroy-occambsd-xen.sh	Script to clean up the Xen virtual machine

/tmp/occambsd/jail	Jail root directory
/tmp/occambsd/jail.conf	Jail configuration file
/tmp/occambsd/boot-occambsd-jail.sh	Script to boot the jail(8)

Demon

# Initial Feedback



# Initial Feedback



# Initial Feedback



# Choose Your Own Adventure

Should this...

Offer a choice of bhyve, Xen, or Jail?

Make rebuilding world or kernel easier?

< Your amazing idea here >

# Observations

`/usr/share/locale/C.UTF-8 (37K)` is all you need

`/usr/share/zoneinfo/UTC (512B)` is...

Separate those from their options?

# Related: up.bsd.lv

## FreeBSD CURRENT and STABLE Binary Upgrades

---

*"Because tracking CURRENT shouldn't be a rite of passage"*

### Disclaimer

This proof-of-concept is not a publication of FreeBSD.

### Description

up.bsd.lv is a proof-of-concept of binary updates using [freebsd-update\(8\)](#) for FreeBSD 13.0-CURRENT and 12-STABLE to facilitate the exhaustive testing of FreeBSD and the bhyve hypervisor and OpenZFS 2.0, and to help elevate ARM64 to Tier 1 status. Updates are based on the SVN revisions of official FreeBSD Release Engineering weekly snapshots.

### Requirements

Upgrades use SSL to the extent possible and thus require the [ca\\_root\\_nss](#) package, [certctl\(8\)](#), or another certificate bundle to support the retrieval of two `.ssl` files. Note that an accurate system time and date may be required for proper SSL handling.

Upgrades can be performed on 12.1 RELEASE or updated systems, or with a provided `disc1.iso` or `memstick` image.

Related: headbanger.sh

OccamBSD's 1650+ line older sibling

```
cd /usr/src ; make buildworld ; make buildkernel  
cd release ; make release
```

*With many, many options.*

# Related: headbanger.sh

```
sh headbanger.sh -s "/build/7ae27c2d6c4/src" \  
-o "/build/7ae27c2d6c4/obj" \  
-l "/build/7ae27c2d6c4/log" -r 7ae27c2d6c4 \  
-g "/pub/FreeBSD/git/src" \  
-t arm64 -T aarch64 -R "10.0.0.40" -k "GENERIC-NODEBUG" -u
```

```
sh headbanger.sh -O "WITHOUT_AUTO_OBJ WITHOUT_UNIFIED_OBJDIR  
WITHOUT_INSTALLLIB WITHOUT_BOOT WITHOUT_LOADER_LUA WITHOUT_LOCALES  
WITHOUT_ZONEINFO WITHOUT_VI" -K /tmp/OCCAM"
```

# What about nanobsd.sh?

```
# root@xen:/usr/src/tools/tools/nanobsd # more nanobsd.sh
```

```
#!/bin/sh
```

```
# Copyright (c) 2005 Poul-Henning Kamp.
```

```
...
```

```
00:00:00 ## Construct build make.conf  
(/usr/obj/nanobsd.full/make.conf.build)
```

```
00:00:00 ## run buildworld
```

```
00:00:01 ### log: /usr/obj/nanobsd.full/_.bw
```

```
NANO_PMAKE="make -j 3"
```

# What about nanobsd.sh?

```
--- ASTMatchers/Dynamic/Registry.o ---
```

```
Killed
```

```
*** [ASTMatchers/Dynamic/Registry.o] Error code 137
```

```
make[6]: stopped in /usr/src/lib/clang/libclang
```

```
00:00:05 ## Construct build make.conf
```

```
(/usr/obj/nanobsd.full/make.conf.build)
```

```
00:00:05 ## run buildworld
```

```
00:00:05 ### log: /usr/obj/nanobsd.full/_.bw
```

```
1h43m5.37s real
```

```
6h23m19.11s user
```

```
11m45.98s sys
```

# EPYC THANKS

Kyle Evans

Mitchell Horne

Ed Maste

NYC\*BUG