

Tinderbox and Poudriere - Automatic Ports Testing and Package Building on FreeBSD

Guido Falsi

`<madpilot@FreeBSD.org>`



Institute of Biostructures and Bioimaging,
Naples, Italy
April 6, 2013

The problems at hand:

1. Test ports on all (supported) OS versions
 - ▶ Check if port fetches, builds and installs
 - ▶ Check if port packages correctly
 - ▶ Check if port disinstalls correctly
 - ▶ Check for leftovers



The problems at hand:

1. Test ports on all (supported) OS versions
 - ▶ Check if port fetches, builds and installs
 - ▶ Check if port packages correctly
 - ▶ Check if port disinstalls correctly
 - ▶ Check for leftovers
2. Create package sets/repositories for use with all (supported) OS versions
 - ▶ Build packages for both old pkg tools and pkgng
 - ▶ Build customized packages (make.conf, OPTIONS)
 - ▶ Build packages from modified ports trees



The problems at hand:

1. Test ports on all (supported) OS versions
 - ▶ Check if port fetches, builds and installs
 - ▶ Check if port packages correctly
 - ▶ Check if port disinstalls correctly
 - ▶ Check for leftovers
2. Create package sets/repositories for use with all (supported) OS versions
 - ▶ Build packages for both old pkg tools and pkgng
 - ▶ Build customized packages (make.conf, OPTIONS)
 - ▶ Build packages from modified ports trees

A solution:

Use Tinderbox and/or Poudriere.



Some concepts

Jail A base system version used to build/test ports



Some concepts

Jail A base system version used to build/test ports

Portstree A ports tree, with or without local modifications



Some concepts

- Jail** A base system version used to build/test ports
- Portstree** A ports tree, with or without local modifications
- Build** An environment made of a Jail and a Portstree (Tinderbox)



Some concepts

- Jail** A base system version used to build/test ports
- Portstree** A ports tree, with or without local modifications
- Build** An environment made of a Jail and a Portstree (Tinderbox)
- Set** Text string used to differentiate various customized environments (Poudriere)



Some concepts

- Jail** A base system version used to build/test ports
- Portstree** A ports tree, with or without local modifications
- Build** An environment made of a Jail and a Portstree (Tinderbox)
- Set** Text string used to differentiate various customized environments (Poudriere)
- DUD** Port which defines FORBIDDEN or BROKEN



About the projects

Tinderbox:

First released on: 2 February 2005

Project started by: Joe Marcus Clarke

Written in: Perl and PHP

Home page: <http://tinderbox.marcuscom.com/>

Poudriere:

First released on: 31 January 2012

Project started by: Baptiste Daroussin

Written in: sh

Home page: <https://fossil.etoilebsd.net/poudriere/>



Software requirements

Tinderbox:

- ▶ Any (supported) FreeBSD version
- ▶ Perl
- ▶ MySQL or PostgreSQL or SQLite (4.0 only)
- ▶ Webserver+PHP and PEAR::DB for web UI frontend
- ▶ NFS or nullfs will be used to mount things around

Poudriere:

- ▶ FreeBSD ≥ 8.3
- ▶ A ZFS pool with 7GB available space



Software requirements

Tinderbox:

- ▶ Any (supported) FreeBSD version
- ▶ Perl
- ▶ MySQL or PostgreSQL or SQLite (4.0 only)
- ▶ Webserver+PHP and PEAR::DB for web UI frontend
- ▶ NFS or nullfs will be used to mount things around

Poudriere:

- ▶ FreeBSD ≥ 8.3
- ▶ A ZFS pool with 7GB available space

Important:

Base system version must be equal or newer than all system versions being run in the jails.



Hardware Requirements

CPU The more the better!



Hardware Requirements

CPU The more the better!

RAM 1-2 GiB x core, more if using RAM disks



Hardware Requirements

CPU The more the better!

RAM 1-2 GiB x core, more if using RAM disks

Disk 1.5 GiB x Jail, 4 GiB x ports tree, variable space for distfiles and packages, plus space used by port build itself



Hardware Requirements

- CPU** The more the better!
- RAM** 1-2 GiB x core, more if using RAM disks
- Disk** 1.5 GiB x Jail, 4 GiB x ports tree, variable space for distfiles and packages, plus space used by port build itself
- Network** Connectivity to download sources and distfiles throughout the build



Tinderbox: preparing environment

- ▶ Create a jail from official release binaries:

```
./tc createJail -j 9.1-amd64 -d "9.1-RELEASE on amd64"  
-t 9.1-RELEASE -u LFTP -H ftp.freebsd.org -a amd64
```



Tinderbox: preparing environment

- ▶ Create a jail from official release binaries:

```
./tc createJail -j 9.1-amd64 -d "9.1-RELEASE on amd64"  
-t 9.1-RELEASE -u LFTP -H ftp.freebsd.org -a amd64
```

- ▶ Create a jail from sources via subversion:

```
./tc createJail -j 10-i386 -u SVN -d "10-CURRENT on i386"  
-D base/head -H svn.FreeBSD.org -P svn -a i386
```



Tinderbox: preparing environment

- ▶ Create a jail from official release binaries:

```
./tc createJail -j 9.1-amd64 -d "9.1-RELEASE on amd64"  
-t 9.1-RELEASE -u LFTP -H ftp.freebsd.org -a amd64
```

- ▶ Create a jail from sources via subversion:

```
./tc createJail -j 10-i386 -u SVN -d "10-CURRENT on i386"  
-D base/head -H svn.FreeBSD.org -P svn -a i386
```

- ▶ Create at least one "portstree":

```
./tc createPortsTree -p FreeBSD -d "ports tree" -u SVN  
-P http -H svn.FreeBSD.org -D ports/head  
-w http://svnweb.FreeBSD.org/ports/
```



Tinderbox: preparing environment

- ▶ Create a jail from official release binaries:

```
./tc createJail -j 9.1-amd64 -d "9.1-RELEASE on amd64"  
-t 9.1-RELEASE -u LFTP -H ftp.freebsd.org -a amd64
```

- ▶ Create a jail from sources via subversion:

```
./tc createJail -j 10-i386 -u SVN -d "10-CURRENT on i386"  
-D base/head -H svn.FreeBSD.org -P svn -a i386
```

- ▶ Create at least one “portstree”:

```
./tc createPortsTree -p FreeBSD -d "ports tree" -u SVN  
-P http -H svn.FreeBSD.org -D ports/head  
-w http://svnweb.FreeBSD.org/ports/
```

- ▶ Create a build by combining a jail and a “portstree”:

```
./tc createBuild -b 9.1-FreeBSD-amd64 -j 9.1-amd64  
-p FreeBSD -d "9.1-RELEASE with ports tree"
```



Tinderbox: preparing environment

- ▶ Create a jail from official release binaries:

```
./tc createJail -j 9.1-amd64 -d "9.1-RELEASE on amd64"  
-t 9.1-RELEASE -u LFTP -H ftp.freebsd.org -a amd64
```

- ▶ Create a jail from sources via subversion:

```
./tc createJail -j 10-i386 -u SVN -d "10-CURRENT on i386"  
-D base/head -H svn.FreeBSD.org -P svn -a i386
```

- ▶ Create at least one “portstree”:

```
./tc createPortsTree -p FreeBSD -d "ports tree" -u SVN  
-P http -H svn.FreeBSD.org -D ports/head  
-w http://svnweb.FreeBSD.org/ports/
```

- ▶ Create a build by combining a jail and a “portstree”:

```
./tc createBuild -b 9.1-FreeBSD-amd64 -j 9.1-amd64  
-p FreeBSD -d "9.1-RELEASE with ports tree"
```

- ▶ Logging needs to be configured:

```
./tc configLog -d tinderbox/logs
```



Tinderbox: Supported methods of getting sources and ports trees

- SVN** Use subversion
- LFTP** Depends on port ftp/lftp, downloads binary distributions from official mirrors if available (jails)
- USER** Will use user provided script in
tinderbox/jails|portstrees/<jail>|<portstree>/
/update.sh
- NONE** Will do nothing. User needs to manually setup
everything



Poudriere: preparing environment

- ▶ Create a jail from official release binaries:

```
poudriere jail -c -j 91amd64 -v 9.1-RELEASE -a amd64
```



Poudriere: preparing environment

- ▶ Create a jail from official release binaries:

```
poudriere jail -c -j 91amd64 -v 9.1-RELEASE -a amd64
```

- ▶ Create a jail from sources via subversion:

```
poudriere jail -c -j 10i386 -v head -m svn -a i386
```



Poudriere: preparing environment

- ▶ Create a jail from official release binaries:

```
poudriere jail -c -j 91amd64 -v 9.1-RELEASE -a amd64
```

- ▶ Create a jail from sources via subversion:

```
poudriere jail -c -j 10i386 -v head -m svn -a i386
```

- ▶ Create at least one “portstree”:

```
poudriere ports -c
```



Poudriere: preparing environment

- ▶ Create a jail from official release binaries:

```
poudriere jail -c -j 91amd64 -v 9.1-RELEASE -a amd64
```

- ▶ Create a jail from sources via subversion:

```
poudriere jail -c -j 10i386 -v head -m svn -a i386
```

- ▶ Create at least one “portstree”:

```
poudriere ports -c
```

Attention

No . (dots) in poudriere jail names!



Poudriere: Supported methods of getting sources and ports trees

svn Use subversion

ftp Downloads binary distributions from official mirrors if available

allbsd, gjb Will download binary snapshots from allbsd or gjb respectively (No automatic updating, jails only)

portsnap Will download and update a ports tree using portsnap

git Will download portstree using git



Customizing tinderbox builds

- ▶ Performed through sh scripts in `tinderbox/scripts/etc/env` named with following logic:

GLOBAL

`jail.<JAIL>`

`portstree.<PORTSTREE>`

`build.<BUILD>`



Customizing tinderbox builds

- ▶ Performed through sh scripts in `tinderbox/scripts/etc/env` named with following logic:
GLOBAL
jail.<JAIL>
portstree.<PORTSTREE>
build.<BUILD>
- ▶ These files contain env vars with the same values which can go in `make.conf` or `src.conf`



Customizing tinderbox builds

- ▶ Performed through sh scripts in `tinderbox/scripts/etc/env` named with following logic:
GLOBAL
jail.<JAIL>
portstree.<PORTSTREE>
build.<BUILD>
- ▶ These files contain env vars with the same values which can go in `make.conf` or `src.conf`
- ▶ To use options run `./tc configOptions -e && ./tc configOptions -o /options` and populate the `<build>` subdirectory with options



Customizing tinderbox builds

- ▶ Performed through sh scripts in `tinderbox/scripts/etc/env` named with following logic:
GLOBAL
jail.<JAIL>
portstree.<PORTSTREE>
build.<BUILD>
- ▶ These files contain env vars with the same values which can go in `make.conf` or `src.conf`
- ▶ To use options run `./tc configOptions -e && ./tc configOptions -o /options` and populate the <build> subdirectory with options

Tip:

```
cp -R /var/db/ports/* tinderbox/options/<BUILD>
```



Customizing poudriere builds

- ▶ Performed by concatenating files in /usr/local/etc/poudriere.d in the following order:

```
make.conf | src.conf
```

```
<setname>-make.conf | src.conf
```

```
<tree>-make.conf | src.conf
```

```
<jailname>-make.conf | src.conf
```

```
<jailname>--<tree>-make.conf | src.conf
```

```
<jailname>--<setname>-make.conf | src.conf
```



Customizing poudriere builds

- ▶ Performed by concatenating files in /usr/local/etc/poudriere.d in the following order:

```
make.conf | src.conf  
<setname>-make.conf | src.conf  
<tree>-make.conf | src.conf  
<jailname>-make.conf | src.conf  
<jailname>--<tree>-make.conf | src.conf  
<jailname>--<setname>-make.conf | src.conf
```

- ▶ Tip: use the += make assignment operator!



Customizing poudriere builds

- ▶ Performed by concatenating files in /usr/local/etc/poudriere.d in the following order:

```
make.conf | src.conf  
<setname>-make.conf | src.conf  
<tree>-make.conf | src.conf  
<jailname>-make.conf | src.conf  
<jailname>-<tree>-make.conf | src.conf  
<jailname>-<setname>-make.conf | src.conf
```

- ▶ Tip: use the += make assignment operator!
- ▶ Options will be searched for in these directories:

```
<jailname>-<setname>-options  
<setname>-options  
<jailname>-options  
options
```



Customizing poudriere builds

- ▶ Performed by concatenating files in /usr/local/etc/poudriere.d in the following order:

```
make.conf | src.conf  
<setname>-make.conf | src.conf  
<tree>-make.conf | src.conf  
<jailname>-make.conf | src.conf  
<jailname>--<tree>-make.conf | src.conf  
<jailname>--<setname>-make.conf | src.conf
```

- ▶ Tip: use the += make assignment operator!
- ▶ Options will be searched for in these directories:

```
<jailname>--<setname>-options  
<setname>-options  
<jailname>-options  
options
```

- ▶ To populate options directories:

```
poudriere options -j <jailname> -p <tree>  
-f <list> [ -z <set> ]
```



Using tinderbox

- ▶ Can use a distfile cache:

```
./tc configDistfile -c localhost:/distfiles
```



Using tinderbox

- ▶ Can use a distfile cache:

```
./tc configDistfile -c localhost:/distfiles
```

- ▶ To build some ports from command line:

```
./tc addPort -b 9.1-FreeBSD-amd64 -d x11/gnome2
```

```
./tc tinderbuild -b 9.1-FreeBSD-amd64
```



Using tinderbox

- ▶ Can use a distfile cache:
`./tc configDistfile -c localhost:/distfiles`
- ▶ To build some ports from command line:
`./tc addPort -b 9.1-FreeBSD-amd64 -d x11/gnome2`
`./tc tinderbuild -b 9.1-FreeBSD-amd64`
- ▶ Create a file named `.keep` in port's direcotry to save its workdir in `tinderbox/wrkdirs/<build>`



Using tinderbox

- ▶ Can use a distfile cache:
`./tc configDistfile -c localhost:/distfiles`
- ▶ To build some ports from command line:
`./tc addPort -b 9.1-FreeBSD-amd64 -d x11/gnome2`
`./tc tinderbuild -b 9.1-FreeBSD-amd64`
- ▶ Create a file named `.keep` in port's directory to save its workdir in `tinderbox/wrkdirs/<build>`
- ▶ Create a file named `.sleepme` in port's directory to suspend it and enter in it with:
`./tc enterBuild -b <build> -d <port>`



Using tinderbox

- ▶ Can use a distfile cache:

```
./tc configDistfile -c localhost:/distfiles
```

- ▶ To build some ports from command line:

```
./tc addPort -b 9.1-FreeBSD-amd64 -d x11/gnome2
```

```
./tc tinderbuild -b 9.1-FreeBSD-amd64
```

- ▶ Create a file named `.keep` in port's directory to save its workdir in `tinderbox/wrkdirs/<build>`
- ▶ Create a file named `.sleepme` in port's directory to suspend it and enter in it with:

```
./tc enterBuild -b <build> -d <port>
```

- ▶ Packages will be in `tinderbox/packages/<build>`



Using poudriere

- ▶ Test a single port:

```
poudriere testport -o x11/gnome2 -j 91amd64
```



Using poudriere

- ▶ Test a single port:

```
poudriere testport -o x11/gnome2 -j 91amd64
```

- ▶ Test a single port from a standalone directory:

```
poudriere testport -d ~/foo/myport -j 10amd64
```



Using poudriere

- ▶ Test a single port:

```
poudriere testport -o x11/gnome2 -j 91amd64
```

- ▶ Test a single port from a standalone directory:

```
poudriere testport -d ~/foo/myport -j 10amd64
```

- ▶ Bulk build all ports listed in ~/portlist on all the jails:

```
poudriere bulk -f ~/portlist
```



Using poudriere

- ▶ Test a single port:

```
poudriere testport -o x11/gnome2 -j 91amd64
```

- ▶ Test a single port from a standalone directory:

```
poudriere testport -d ~/foo/myport -j 10amd64
```

- ▶ Bulk build all ports listed in ~/portlist on all the jails:

```
poudriere bulk -f ~/portlist
```

- ▶ Bulk build all ports in myports tree defining a custom set in a jail saving the workdir in case of failure:

```
poudriere bulk -a -j 10amd64 -p myports -w -z myset
```



Using poudriere

- ▶ Test a single port:

```
poudriere testport -o x11/gnome2 -j 91amd64
```

- ▶ Test a single port from a standalone directory:

```
poudriere testport -d ~/foo/myport -j 10amd64
```

- ▶ Bulk build all ports listed in ~/portlist on all the jails:

```
poudriere bulk -f ~/portlist
```

- ▶ Bulk build all ports in myports tree defining a custom set in a jail saving the workdir in case of failure:

```
poudriere bulk -a -j 10amd64 -p myports -w -z myset
```

- ▶ Packages will be in

```
poudriere/data/packages/<jail>-<tree>[-<set>]
```



Using poudriere

- ▶ Test a single port:

```
poudriere testport -o x11/gnome2 -j 91amd64
```

- ▶ Test a single port from a standalone directory:

```
poudriere testport -d ~/foo/myport -j 10amd64
```

- ▶ Bulk build all ports listed in ~/portlist on all the jails:

```
poudriere bulk -f ~/portlist
```

- ▶ Bulk build all ports in myports tree defining a custom set in a jail saving the workdir in case of failure:

```
poudriere bulk -a -j 10amd64 -p myports -w -z myset
```

- ▶ Packages will be in

```
poudriere/data/packages/<jail>-<tree>[-<set>]
```

- ▶ Logs will be in poudriere/data/logs/(bulk|testport)/
/<jail>/<repo>[-<set>]



Using the bulk built repositories

- ▶ To make them generate pkgng files for 9.x this needs to be defined in the environment:

```
WITH_PKGNG=yes
```

```
PKGSUFFIX=.txz          (Tinderbox)
```



Using the bulk built repositories

- ▶ To make them generate pkgng files for 9.x this needs to be defined in the environment:

```
WITH_PKGNG=yes
```

```
PKGSUFFIX=.txz          (Tinderbox)
```

- ▶ Tinderbox does not build a repository catalogue for pkgng:

```
cd tinderbox/packages/<build> && \  
pkg repo tinderbox/packages/<build>
```



Using the bulk built repositories

- ▶ To make them generate pkgng files for 9.x this needs to be defined in the environment:

```
WITH_PKGNG=yes
```

```
PKGSUFFIX=.txz          (Tinderbox)
```

- ▶ Tinderbox does not build a repository catalogue for pkgng:

```
cd tinderbox/packages/<build> && \  
pkg repo tinderbox/packages/<build>
```

- ▶ Poudriere always builds a repository catalogue after a successful bulk build



Using the bulk built repositories

- ▶ To make them generate pkgng files for 9.x this needs to be defined in the environment:

```
WITH_PKGNG=yes
```

```
PKGSUFFIX=.txz          (Tinderbox)
```

- ▶ Tinderbox does not build a repository catalogue for pkgng:

```
cd tinderbox/packages/<build> && \  
pkg repo tinderbox/packages/<build>
```

- ▶ Poudriere always builds a repository catalogue after a successful bulk build

- ▶ To use with old pkg tools:

```
setenv PACKAGESITE http://example.org/91amd64/
```



Using the bulk built repositories

- ▶ To make them generate pkgng files for 9.x this needs to be defined in the environment:

```
WITH_PKGNG=yes
```

```
PKGSUFFIX=.txz          (Tinderbox)
```

- ▶ Tinderbox does not build a repository catalogue for pkgng:

```
cd tinderbox/packages/<build> && \  
pkg repo tinderbox/packages/<build>
```

- ▶ Poudriere always builds a repository catalogue after a successful bulk build

- ▶ To use with old pkg tools:

```
setenv PACKAGESITE http://example.org/91amd64/
```

- ▶ To use with pkgng:

```
packagesite: http://example.org/91amd64/  
in /usr/local/etc/pkg.conf
```



Demonstration time!

madpilot Tinderbox

Current And Latest Builds | Failed builds in this build | All Failures | Queue | Config | Add User | Modify Me | Logout

S = Number of ports built successfully
U = Number of ports with unknown status
F = Number of ports that failed to build
D = Number of ports that were not built due to a dependency failure
L = Number of ports with liveness
DD = Number of builds
R = Number of ports to be requeued
T = Total

Build Name	Build Description	S	U	F	D	L	DD	R	T	Build Packages
10-CURRENT-amd64-motest	madpilot net porttree on 10-CURRENT-amd64	260	-	-	-	-	-	-	260	Backaux,Discosoft
10-CURRENT-i386-motest	madpilot net porttree on 10-CURRENT-i386	261	-	-	-	-	-	-	261	Backaux,Discosoft
7-STABLE-amd64-motest	madpilot net porttree on 7-STABLE-amd64	262	-	-	-	-	-	-	262	Backaux,Discosoft
7-STABLE-i386-motest	madpilot net porttree on 7-STABLE-i386	262	-	-	-	-	-	-	262	Backaux,Discosoft
8-STABLE-amd64-motest	madpilot net porttree on 8-STABLE-amd64	262	-	-	-	-	-	-	262	Backaux,Discosoft
8-STABLE-i386-motest	madpilot net porttree on 8-STABLE-i386	262	-	-	-	-	-	-	262	Backaux,Discosoft
9-STABLE-amd64-motest	madpilot net porttree on 9-STABLE-amd64	262	-	-	-	-	-	-	262	Backaux,Discosoft
9-STABLE-i386-motest	madpilot net porttree on 9-STABLE-i386	262	-	-	-	-	-	-	262	Backaux,Discosoft

Find ports by name

10-CURRENT-amd64-motest

Maintainer

All Build Failures for the maintainer: RSS feed for the maintainer:

Modify User

madpilot Modify User

home | back | top | LocalTime: 2013-02-17 19:36:57 building Prepare

```
====> [07] Finished build of x11-themes/icons-tango-extras: Success
====> [07] Starting build of sysutils/conky
====> [03] Finished build of emulators/uae: Success
====> [03] Starting build of x11-wm/fvwm2
====> [06] Finished build of graphics/viewnior: Success
====> [06] Starting build of x11/xscreensaver
====> [07] Finished build of sysutils/conky: Success
====> [07] Starting build of audio/gmtp
====> [07] Finished build of audio/gmtp: Success
====> [07] Starting build of x11/xlogo
====> [07] Finished build of x11/xlogo: Success
====> [07] Starting build of multimedia/vlc

load: 9.90  cmd: clang 96294 [running] 154.98r 137.63u 0.16s  100% 114280k
[10amd64] [parallel_build:] [231/243] Built: 231 Failed: 0 Ignored: 0 Skip
ped: 0
[01]: build:emulators/virtualbox-ose
[02]: build:emulators/mame
[03]: build:x11-wm/fvwm2
[04]: build:www/firefox
[05]: build:graphics/gimp-app
[06]: build:x11/xscreensaver
[07]: depends:multimedia/vlc
[08]: build:mail/thunderbird
```

Also a good time to ask questions



Thanks for your attention!

