Poudrière Efficient package building

> Baptiste Daroussin bapt@FreeBSD.org





EuroBSDCon 2015 Stockholm Octobre 4th, 2015

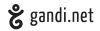
# poudri...GNI?: [pu.dui.je]



- Package building system
- Port tester
- Quality insurance on packages
- Package repository generator
- System stress tool



## History



- 2010-07: Initial work
- ▶ 2011: Start to be known and used in the french community
- ▶ 2012-01-31: 1.0 enter the ports tree
- > 2012-04-08: 1.2 limit network on fetch phase
- 2012-05-15: 1.3 pbi support, attract interest of bdrewery@
- 2012-08-28: 2.0 parallel build, ugly html UI (bapt as a designer)
- ▶ 2012-10-15: 2.2 Removal of pbi support, support for "sets"
- 2013-05-20: 3.0 ZFS optional, full tmpfs support, nice and reactive web UI (bdrewery designer)
- ▶ 2013-07: Used in the FreeBSD cluster
- 2013-09-22: 3.0.3 support staging, initial qemu support
- 2014-12-04: 3.1.0 Yet a better web UI



EuroBSDCon 2015

#### Design



- Simple
  - Easy to setup:
    - only depend on base (by default)
    - one simple configuration file
    - few command to prepare the resources
  - Easy to use
    - One single command
    - Simple subcommands
- Resource efficient
  - ▶ parallel build: by default 1 core == 1 package building
  - Iow overhead (resources should be dedicated to build sources not for poudriere itself)
- Safe and contained
  - all builds in clean jail(8)
  - only access network during fetch phase
  - build as regular user

EuroBSDCon 2015







Subcommands:

- bulk: Generate packages for given ports
- ▶ jail: Manage the jails used by poudriere
- ▶ ports: Create, update or delete the portstrees



#### Poudrière: jails





EuroBSDCon 2015





Fetch release/snapshot/old releases sets



EuroBSDCon 2015





- Fetch release/snapshot/old releases sets
- ▶ Build from sources: git, svn, file, support for branches







- Fetch release/snapshot/old releases sets
- Build from sources: git, svn, file, support for branches
- Full support for src.conf







- Fetch release/snapshot/old releases sets
- Build from sources: git, svn, file, support for branches
- Full support for src.conf
- Support for multiple arches (via gemu user emulation)







- Fetch release/snapshot/old releases sets
- Build from sources: git, svn, file, support for branches
- Full support for src.conf
- Support for multiple arches (via gemu user emulation)
- Can have kernel







- Fetch release/snapshot/old releases sets
- Build from sources: git, svn, file, support for branches
- Full support for src.conf
- Support for multiple arches (via gemu user emulation)
- Can have kernel
- Updateable (via sources or freebsd-update)







- Fetch release/snapshot/old releases sets
- Build from sources: git, svn, file, support for branches
- Full support for src.conf
- Support for multiple arches (via gemu user emulation)
- Can have kernel
- Updateable (via sources or freebsd-update)

Creating a jail poudriere jail -c -j 102 -v 10.2-RELEASE







- Fetch release/snapshot/old releases sets
- Build from sources: git, svn, file, support for branches
- Full support for src.conf
- Support for multiple arches (via gemu user emulation)
- Can have kernel
- Updateable (via sources or freebsd-update)

```
Creating a jail
| poudriere jail -c -j 102 -v 10.2-RELEASE
```

```
Updating a jail
| poudriere jail -u -j 102
```



EuroBSDCon 2015

# Poudrière: ports





EuroBSDCon 2015



▶ Fetch from portsnap, git, svn



EuroBSDCon 2015



- Fetch from portsnap, git, svn
- Notion of "default" ports tree





- ▶ Fetch from portsnap, git, svn
- Notion of "default" ports tree





- ▶ Fetch from portsnap, git, svn
- Notion of "default" ports tree

Creating a ports tree poudriere ports -c -p portstree





- Fetch from portsnap, git, svn
- Notion of "default" ports tree

Creating a ports tree poudriere ports -c -p portstree

Updating a ports tree | poudriere ports -u -p portstree



EuroBSDCon 2015





EuroBSDCon 2015



Associate a ports tree, a jail and a list of packages to build



EuroBSDCon 2015



- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support
- Repository generation support (including signature)





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support
- Repository generation support (including signature)
- Default ports tree support





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support
- Repository generation support (including signature)
- Default ports tree support
- Incremental support (aggressive)





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support
- Repository generation support (including signature)
- Default ports tree support
- Incremental support (aggressive)
- Restricted support





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support
- Repository generation support (including signature)
- Default ports tree support
- Incremental support (aggressive)
- Restricted support
- Saving workdir after failure





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support
- Repository generation support (including signature)
- Default ports tree support
- Incremental support (aggressive)
- Restricted support
- Saving workdir after failure

► Autodetection of rebuild





- Associate a ports tree, a jail and a list of packages to build
- Massively parallelized (1 port per core, can be find tuned)
- Support ccache
- Tuneable via: make.conf (fine grained [<jailname>-[<setname>-[<portstree>-]]]make.conf)
- Nice WebUI (static files made dynamic via js)
- Nice cli (with colors and siginfo support)
- Hooks support
- Repository generation support (including signature)
- Default ports tree support
- Incremental support (aggressive)
- Restricted support
- Saving workdir after failure

► Autodetection of rebuild





Building packages: | poudriere bulk -j 102 -f listofpackages.txt

Building packages with Q/A:

Building all ports

Building all ports with a special "set"



EuroBSDCon 2015

Poudrière: a stress tool



In FreeBSD:

- ZFS deadlocks
- tmpfs deadlocks
- nullfs deadlocks
- ▶ tons of fixes in sh(1) in particular regarding job control
- highlight contentions

In Dragonfly:

- Used as a benchmark tool in 2013
- Lots of performance improvement between December 26, 2012 and March 15, 2013 (released in 3.4)
- Lots of panics fixed



Poudrière: under the hood





EuroBSDCon 2015



Mostly coded in sh(1) (clean and maintainable shell is possible!)



EuroBSDCon 2015



- Mostly coded in sh(1) (clean and maintainable shell is possible!)
- Small bits in C for perfomances



EuroBSDCon 2015

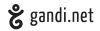


- Mostly coded in sh(1) (clean and maintainable shell is possible!)
- Small bits in C for perfomances
- Lots of care made on efficiency:
  - avoid subshells as much as possible
  - parallelize as many things as possible
  - reuse resources as much as possible
- Use filesystem as a Key/Value DB (on tmpfs for speed)



EuroBSDCon 2015

# Poudrière: image (soon)



- Associate jails, packages and overlays
- Able to generates usable images:
  - Isos: with or without mfsroot
  - Usb disk: with or without mfsroot
  - GPT base firmwares (NanoBSD-like)
  - plain mfsroot
  - rawdisk (VMs)







# Thanks



EuroBSDCon 2015