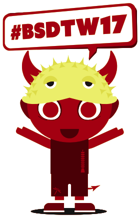
Building custom FreeBSD images

Baptiste Daroussin
bapt@FreeBSD.org
bapt@gandi.net



Taipei
2017-11-10

Gandi

gandi.net

- Domain names
- SSL Certificates
- Simple hosting
- IaaS

Gandi

- Domain names
- SSL Certificates
- Simple hosting
- IaaS

- An office here in Taiwan
- Website in ZH-HANT and ZH-HANS
- Support in Chinese

Poudrière

- Clean room package building
- Used in the FreeBSD cluster
- System stress tool
- Consistently non pronounceable

Poudrière: internal

- Always build inside jails
- No network access during build

Poudrière: internal

- Always build inside jails
- No network access during build
- Massive parallelisation (by default 1 build of package per core, tunable)

Poudrière: internal

- Always build inside jails
- No network access during build
- Massive parallelisation (by default 1 build of package per core, tunable)
- Multiple modes:
  - ZFS
  - UFS
  - TMPFS:
    - data
    - localbase
    - workdir
    - all

Poudrière: internal

- Always build inside jails
- No network access during build
- Massive parallelisation (by default 1 build of package per core, tunable)
- Multiple modes:
  - ZFS
  - UFS
  - TMPFS:
    - data
    - localbase
    - workdir
    - all
- port tester (in depth testing)

Poudrière: internal

- Always build inside jails
- No network access during build
- Massive parallelisation (by default 1 build of package per core, tunable)
- Multiple modes:
  - ZFS
  - UFS
  - TMPFS:
    - data
    - localbase
    - workdir
    - all
- port tester (in depth testing)
- extremely easy to use

Gandi's need

- Custom ramdisks (for FreeBSD based filers)
  - custom patches on the source tree
  - custom packages
  - minimalistic images
  - custom configuration files

Gandi's need

gandi.net

- Custom ramdisks (for FreeBSD based filers)
  - custom patches on the source tree
  - custom packages
  - minimalistic images
  - custom configuration files
- FreeBSD base VM images (for customers)
  - Vanilla FreeBSD
  - Few packages
  - Custom configuration files
  - Custom script to add to the base system
- Fast to regenerate (rebuilding is time consuming)

FreeBSD

State of art

- NanoBSD

State of art

- NanoBSD
- Crochet

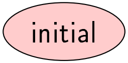
State of art

- NanoBSD
- Crochet
- FreeBSD wifi build

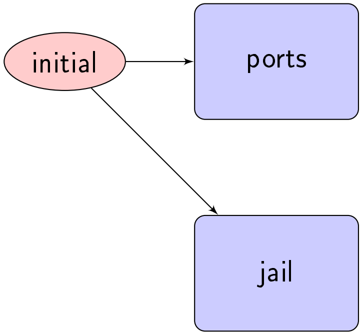
State of art

- NanoBSD
- Crochet
- FreeBSD wifi build
- release.sh

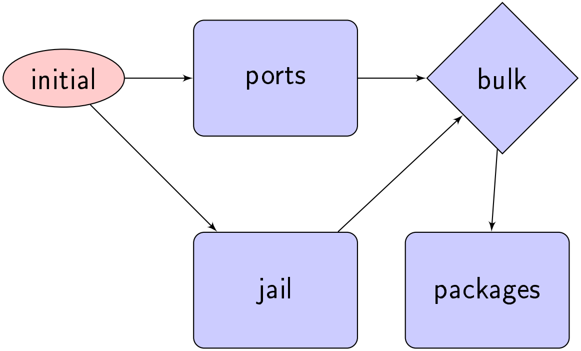
Poudrière image



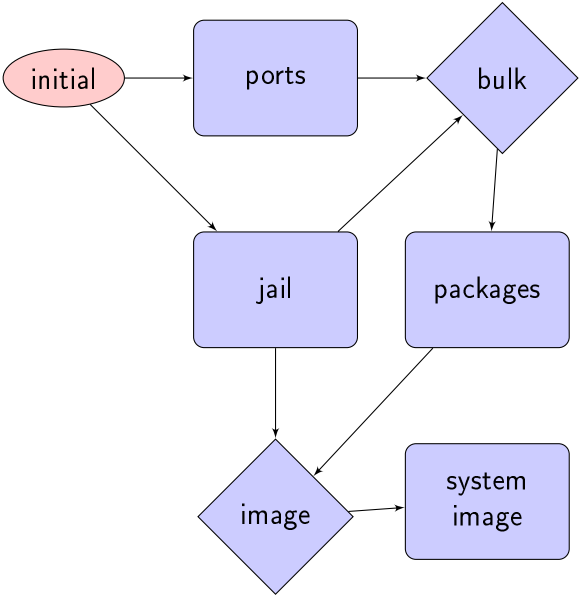
Poudrière image



Poudrière image



Poudrière image



Poudrière image: Features

- below 500 LoC (in shell)

Poudrière image: Features

- below 500 LoC (in shell)
- many output formats: iso, rawdisk, memdisk, firmware, tar files

Poudrière image: Features

- below 500 LoC (in shell)
- many output formats: iso, rawdisk, memdisk, firmware, tar files
- UEFI/bios ready images

Poudrière image: Features

- below 500 LoC (in shell)
- many output formats: iso, rawdisk, memdisk, firmware, tar files
- UEFI/bios ready images
- regenerate from binaries

Poudrière image: Features

- below 500 LoC (in shell)
- many output formats: iso, rawdisk, memdisk, firmware, tar files
- UEFI/bios ready images
- regenerate from binaries
- overlay

Poudrière image: Features

- below 500 LoC (in shell)
- many output formats: iso, rawdisk, memdisk, firmware, tar files
- UEFI/bios ready images
- regenerate from binaries
- overlay
- early support for cross targets: arm (thanks manu@)

Poudrière image: Features

- below 500 LoC (in shell)
- many output formats: iso, rawdisk, memdisk, firmware, tar files
- UEFI/bios ready images
- regenerate from binaries
- overlay
- early support for cross targets: arm (thanks manu@)

Poudrière image: how it works

Poudrière image: how it works

1. jail should be built with a -K so that it contains a kernel

Poudrière image: how it works

1. jail should be built with a -K so that it contains a kernel
2. bulk packages

Poudrière image: how it works

1. jail should be built with a -K so that it contains a kernel
2. bulk packages
3. grab jail files into a temporary directory

Poudrière image: how it works

1. jail should be built with a -K so that it contains a kernel
2. bulk packages
3. grab jail files into a temporary directory
4. run make delete-old base on an extra src.conf

Poudrière image: how it works

1. jail should be built with a -K so that it contains a kernel
2. bulk packages
3. grab jail files into a temporary directory
4. run make delete-old base on an extra src.conf
5. install packages

Poudrière image: how it works

1. jail should be built with a -K so that it contains a kernel
2. bulk packages
3. grab jail files into a temporary directory
4. run make delete-old base on an extra src.conf
5. install packages
6. apply an overlay or top of the system

Poudrière image: how it works

1. jail should be built with a -K so that it contains a kernel
2. bulk packages
3. grab jail files into a temporary directory
4. run make delete-old base on an extra src.conf
5. install packages
6. apply an overlay or top of the system
7. generate the final image
8. profit

Poudrière image: flexibility

- already there:
  - support custom source and ports tree
  - support extra for both ports and source tree
  - overlay (specific code/configuration) can be isolated
  - Use binaries (jail + packages) when assembling an image

Poudrière image: flexibility

- already there:
  - support custom source and ports tree
  - support extra for both ports and source tree
  - overlay (specific code/configuration) can be isolated
  - Use binaries (jail + packages) when assembling an image
- to come:
  - ports overlay to allow working with a vanilla ports tree

Poudrière image: TODO/which list

- support for vmdk/qcow/vhd

Poudrière image: TODO/which list

- support for vmdk/qcow/vhd
- improve makefs(1): inode exhaustion

Poudrière image: TODO/which list

- support for vmdk/qcow/vhd
- improve makefs(1): inode exhaustion
- extend makefs(1) to support zfs

Poudrière image: TODO/which list

- support for vmdk/qcow/vhd
- improve makefs(1): inode exhaustion
- extend makefs(1) to support zfs
- allow to add u-boot into embedded images

Poudrière image: TODO/which list

- support for vmdk/qcow/vhd
- improve makefs(1): inode exhaustion
- extend makefs(1) to support zfs
- allow to add u-boot into embedded images
- allow user to specify the disk layout

Poudrière image: TODO/which list

- support for vmdk/qcow/vhd
- improve makefs(1): inode exhaustion
- extend makefs(1) to support zfs
- allow to add u-boot into embedded images
- allow user to specify the disk layout
- add zfs snapshot output

Questions?

Thanks