Carve your NetBSD

AsiaBSDCon 2014

March 16th 2014, Tokyo, Japan

Guillaume Lasmayous <gls@netbsd.org>
Pierre Pronchery <khorben@netbsd.org>

Introduction

- We grew "frustrated" about some aspects of NetBSD development

- Trying to address them independently from the project:

  - Birth of the EdgeBSD Project mid-2013

  - Place to experiment freely

  - Detailed paper in the proceedings

A look at the system: base

A look at the system: base

- Many releases of the base system
  - NetBSD-6.1.3
  - NetBSD-6.0.4
  - NetBSD-5.2.2
  - NetBSD-5.1.4
  - NetBSD-current

- Highly confusing for newcomers
  - Regular question on IRC or ML: which one is right for me ?

Possible alternatives

- We understand the reasons behind all these releases.

- Could NetBSD-6 be used as a "rolling" release, with regular snapshots, instead of formal releases ?

  - One "release" only to direct new users to

  - One "release" to generate packages against

- A snapshot every 3 months would allow to stay in sync with pkgsrc releases.

A look at the system: packages

A look at the system: packages

Package management should be a no-brainer

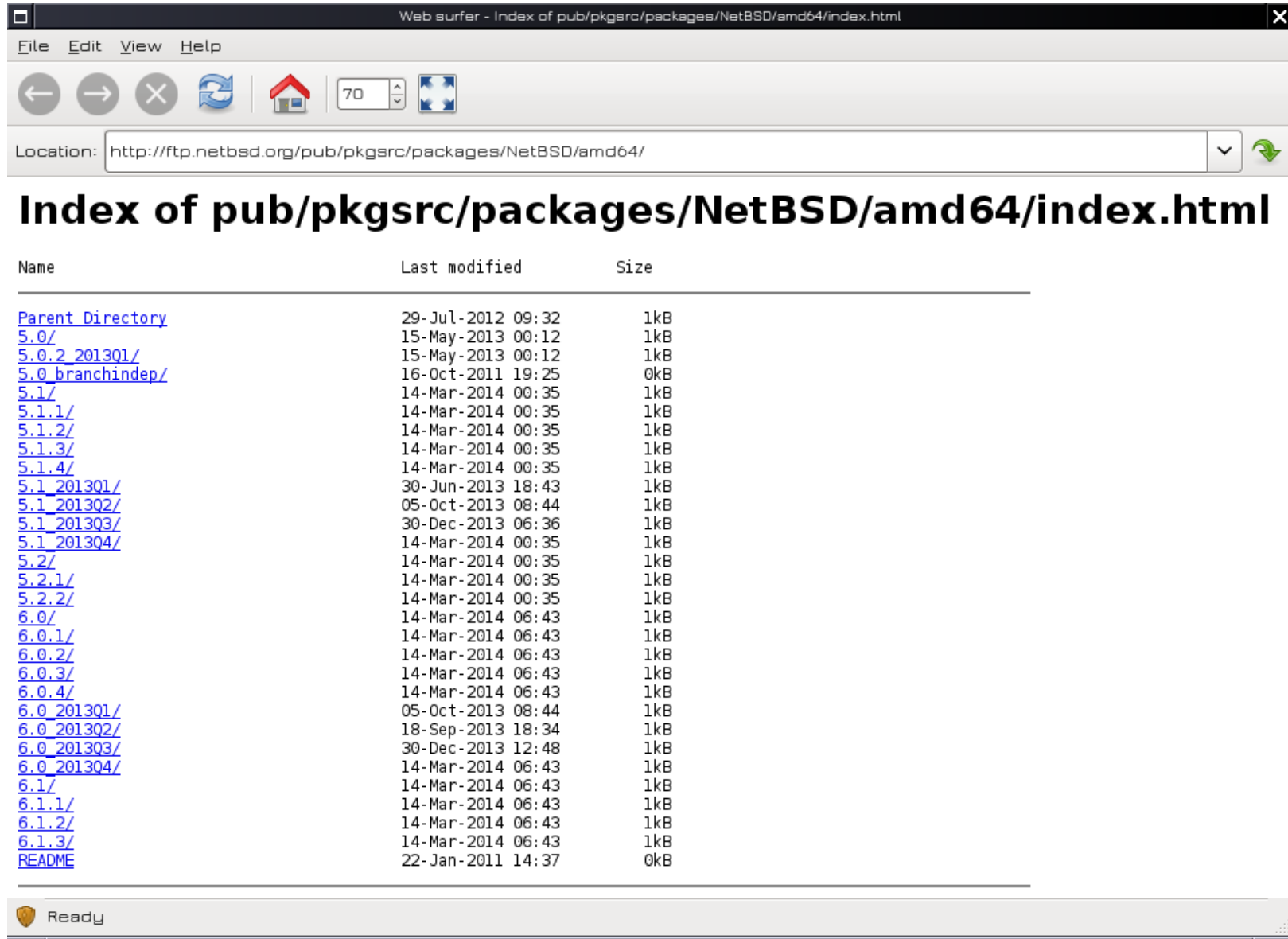- Choose

- Install

- Run

A look at the system: packages

- Several tools to deal with packages
  - pkg_add from base or pkgtools/pkg_install?
  - pkgin
  - nih
- Binary packages need not match the latest release:
  - NetBSD-6.1.3: packages compiled on NetBSD-6.0
  - NetBSD-5.2.2: packages compiled on NetBSD-5.1
- Confusing for newcomers again...

More confusion

- Compare these:
    - http://ftp.netbsd.org/pub/*NetBSD*/NetBSD-6.1.3/**amd64**/binary
    - http://ftp.netbsd.org/pub/pkgsrc/packages/*NetBSD*/**amd64**/6.1.3/All
    - ...why?
- Still does not mention the pkgsrc release

Even more confusion

A look at the system: packages

- Plethora of choice for upgrading:
    - pkgin
    - nih
    - pkg_comp
    - pkg_chk
    - pkg_rolling_replace

What should a newcomer choose?

A look at the system: packages

- And when it comes to building:
  - mk/bulk
    - old bulk framework, deprecated in favor of pbulk, but still used in some places
  - pkgtools/pbulk
    - "official" recommended tool, used in production
  - pkgtools/distbb

A look at the system: packages

- Binary package distribution:
  - Official packages are not signed
  - Trust based on their origin:
    - Built on the server farm from the NetBSD Foundation
    - No formal way to confirm their integrity
  - Difficult to integrate third-party repositories

A step in the right direction

- pkgin installation and configuration from sysinst
- Still, we dramatically need up to date binary packages:
    - Recent versions
    - Seamless upgrades
    - Security updates

Towards industry standards

Attracting vendors

- Vendors need decentralized development
- CVS is the obvious culprit there:
  - Commits and branches cannot be public
  - Would consume tons of network and disk activity!
- The industry has chosen Git:
  - There is a Git mirror maintained by joerg@
  - Several developers have switched to Git already

Long-Term Support (LTS)

- Already the case for the base system

- ...but not for packages

- Inconsistent and confusing policy

- Not sustainable for product development and maintenance

- We should consider LTS branches for pkgsrc

Binary packages

- OpenBSD does it:
  - More convenient for users
  - Traceability of issues
- Binary signatures fixed and working:
  - Not the most elegant design,
  - But it allows for traceability
  - Everyone can choose whose packages to trust
- Easy security updates with pkgin(1)

Challenges for binaries in pkgsrc

- Package options are not binary-friendly

- Need binary compatibility packages:

  - Supporting NetBSD 6.0, 6.1, -current...

- Embedded use has special requirements:

  - Stripping binaries by default already

  - Cross-compilation support improving!

  - Separate headers and development files?

  - Ship documentation separately?

Low-hanging security fruits

- Non exhaustive list:
    - Unprivileged package builds
    - Enabling SSP by default (base and packages)
    - Enabling ASLR by default
    - Building with relro by default: "-Wl,-z,relro,-z,now"
    - Non-executable /tmp
    - Filesystem encryption
- Such features are now industry standard

Introducing EdgeBSD

About EdgeBSD

- Started in mid-2013
- Freedom to experiment
- Testing different approaches to development
- Lowering the entrance barrier
- Attracting new developers to the community

EdgeBSD main objectives

- Stick to NetBSD's codebase

- Implement a rolling release:

  - Make easy upgrades a requirement

- One recommended way for common tasks

- Providing more tools to developers:

  - Development based on Git

  - Support a default development environment

  - Soon: calendar, VoIP...

More EdgeBSD objectives

- More user-friendly:

  - Graphical installer

  - Default desktop environment (DeforaOS?)

  - Ready-to-flash images (see etch.sh)

Current achievements

- Collaborative decentralized development (Git)
- Signed binary packages (GPG)
- Unprivileged binary package builds (fakeroot)
- LTS stable binary packages (pkgsrc-2013Q1)
- Collaborative port to a new architecture (lm32)
- Cross-platform Virtual Machine image generation (etch.sh)
- Attracted a new generation of developers

Who we are

- NetBSD committers:
  - gls@, khorben@, rkujawa@
- Newcomers:
  - drscream, fallen, kooda, yrmt
- More on IRC!
  - #EdgeBSD on Freenode

Ongoing challenges

- Tracking CVS commits with Git

- Therefore: tracking security issues

- Dedicated installer

- First release and documentation

- Userland support for NetBSD/lm32

- System identification (uname!)

Bringing improvements back to NetBSD

Thank you!

- The NetBSD Project, http://www.netbsd.org/
- The EdgeBSD Project, http://www.edgebsd.org/