Gentoo Linux

by
Geert Bevin,
Uwyn
http://www.uwyn.com

Summary

- Installing Gentoo
- Benefits of Gentoo
- Using Portage
- Who is Uwyn?

Installing Gentoo

- Boot from CD

- Setup networking

- Create partitions and mount them

- Extract stage1, stage2 or stage3

- Setup compilation flags and bootstrap (stage2)

- Emerge system (stage3)

- Compile kernel and configure machine
  development code, no smp, buslogic scsi, amd pcnet, devfs, virtual ram, ext3

- Setup grub and reboot

Benefits of Gentoo

- Everything is compiled from source

- You're in control

- Powerful ports-based package manager

- Easy package format

- Centralized internet repository of packages

- Intelligent and dependency-based init scripts

Everything is compiled from source

- Optimized for your hardware

- Packages are built from the original sources

- Fast package updates when new version appear

- Can be turned into binary packages during or after package installation

You're in control

- Minimal installation

- No services are activated by default

- No configuration is done automatically

- Default configurations are available but have to be performed explicitly

- Your configurations are protected and will never be overwritten

Powerful package manager 1/2

- Packages are first installed in a staging directory, and are only then merged into your live system

- Performs installations in a protecting sandbox

- Safe library updates on a running system

- Resolves dependencies automatically

- Keeps you informed about new updates

- Updates your repository from the internet

- Allows you to safely remove outdated packages

Powerful package manager 2/2

- Can create and use binary packages

- Low-level (developer) tool : ebuild

- High-level (user) tool : emerge

- System-wide configuration of your usage preferences (use-flags)

- Virtual packages

Easy package format

- Closely mimics the bash commands you'd execute to install manually

- Clearly separated installation phases : unpack, compile, install, merge

- Pre and post actions for each installation phase

- Additional configuration is done through the definition of variables : description, homepage, download url, dependencies, binary compatibility

Centralized internet repository of packages

- Immediate access to new versions

- Intuitive synchronization of your local package tree

- Direct access by developers to the same repository

Intelligent and dependency-based init scripts

- Not numbered

- Scripts use or need each-other

- Stopping a service auto-stops all services that need it

- Services can be paused (stop function in other distributions)

- Services can't be started or stopped multiple times

- Easy tool to manage services

Using Portage

- emerge
- ebuild
- quickpkg
- rc-update
- etc-update
- /var/db/pkg

emerge

- High-level functions to handle packages

- Merge packages

- Update packages

- Query the portage package tree

- Safely remove outdated packages

- Forcibly erase packages

- Create a binary packages during installation

- Expressive package selection syntax

ebuild

- Direct access to ebuilds

- Execution of individual package stages : fetch, unpack, compile, install, merge (qmerge), unmerge, clean

quickpkg

- Creates a binary package of an already installed package

rc-update

- Manages the activated/deactivated init scripts for each run-level

- Add

- Remove

etc-update

- Manages upgrades of configuration files

- Shows differences with installed config files

- Allows you to merge differences

- Allows you to replace to current version with the new one

/var/db/pkg

- Tree of the installed packages

- Contains meta-data about each package installation

- Category, compilation flags and host, contents, license, virtual package it provides, dependencies, binary compatibility slot, use flags

Who is Uwyn?

- Computer software company, specialized in custom application development

- Development of high-performance portage in C++ with robust embedded database and expressive and versatile Portage Query Language.

- Developed the Portage protective sandbox.

- Developed distinct parts of the current portage.

- All projects are open-source and we maintain an independent framework with associated tools.