Gentoo, un système pour ceux et celles qui aiment tout ajuster

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 29 novembre 2011

https://www.bortzmeyer.org/gentoo.html

Il existe des tas de systèmes d'exploitation libres. Certains sont très différents de la moyenne et, comme j'ai géré pendant des années un serveur sous Gentoo, voici quelques notes sur Gentoo et deux ou trois choses que j'ai apprises à son sujet.

Le système Gentoo utilise le noyau Linux, la GNU libc et les outils GNU habituels. (C'est donc ce que certains appellent, par erreur, une distribution Linux <https://www.bortzmeyer.org/distribution-linux.html>.) Mais Gentoo se distingue de systèmes comme Debian ou CentOS par le fait que tout est compilé à partir des sources et que l'administrateur système a bien plus de liberté pour ajuster le système à ses goûts. Gentoo n'est pourtant pas forcément plus difficile en utilisation quotidienne.

Un des problèmes du terme « distribution Linux » est qu'il suppose que tous les systèmes ainsi nommés ont des points en commun. Par exemple, j'ai lu lors d'une discussion un partisan de FreeBSD affirmer que « les distributions Linux sont en binaire », ce qui est certainement vrai de Debian ou CentOS, mais pas de Gentoo, qui est au contraire le royaume de la compilation.

Donc, le principe est que tout logiciel installé sur la machine a été compilé, avec des options choisies par l'administrateur. Contrairement à FreeBSD, il n'y a qu'une seule méthode (la bonne) pour maintenir l'arbre à jour (et c'est la même pour tout le système, le noyau, la libc, MySQL).

Lorsqu'on tape :

```
% emerge postfix
```

On ne se contente pas d'installer un paquetage binaire de Postfix, on télécharge les sources et on recompile. Les options de compilation (dites "*USE flags*") sont centralisés dans `/etc/portage/package.use` et, par exemple, on peut trouver :

```
mail-mta/postfix -ldap -mailwrapper
```

qui indiquent de compiler Postfix sans LDAP. Autre exemple :

```
net-dns/opendnssec -mysql sqlite
```

indique qu'OpenDNSSEC doit être compilé avec SQLite mais sans MySQL. Les valeurs possibles sont indiquées dans `/usr/portage/profiles/use.desc` pour les variables globales, et dans `/usr/portage/pro` pour celles spécifiques à une application donnée.

Ces compilations systématiques sont évidemment mauvaises du point de vue écologique (on fait tourner la machine et consommer davantage d'électricité). Il existe un moyen de fabriquer des paquetages binaires mais c'est juste en interne, il n'y a pas d'archive publique de tels paquetages.

Gentoo permet également une grande souplesse dans le choix des versions installées. On peut masquer ou démasquer des programmes par leur numéro de version, sélectionnant ainsi le choix de telle ou telle version. Par exemple, si on n'est pas prêt pour MySQL 5.1, on interdit `mysql>=5.1`).

Quelques commandes Gentoo utiles (avec leur équivalent Debian entre parenthèses pour ceux qui sont plus familiarisés avec ce système) :

```
[Compile et installe le paquetage XXX ("aptitude install XXX")]
% emerge XXX

[Mets à jour la liste des paquetages connus ("aptitude update")]
% emerge --sync
[Ou emerge-webrsync si on est coincé derrière un pare-feu pénible.]

[Cherche le nom d'un paquetage ("apt-cache search XXX")]
% emerge --search XXX

[Supprime le paquetage XXX]
% emerge --unmerge XXX
```

La compilation peut rater. Par exemple, en essayant `emerge bind` :

```
!!! The ebuild selected to satisfy "net-dns/bind" has unmet requirements.
- net-dns/bind-9.8.1_p1::gentoo USE="berkdb doc ipv6 (multilib) ssl xml -caps -dlz -geoip -gost -gssapi -idn

  The following REQUIRED_USE flag constraints are unsatisfied:
    berkdb? ( dlz )

  The above constraints are a subset of the following complete expression:
    postgres? ( dlz ) berkdb? ( dlz ) mysql? ( dlz !threads ) odbc? ( dlz ) ldap? ( dlz ) sdb-ldap? ( dlz )
```

C'est beau comme message, non ? Pour le supprimer, il a fallu ajouter dans `package.use` un `net-dns/bind -mysql -ldap -berkdb -dlz`.

Autre problème, certains programmes nécessitent une énorme quantité de mémoire lors de la compilation. C'est ainsi que Ruby ou Haskell (le compilateur ghc) ne peuvent tout simplement pas être compilés sur un VPS de petite taille.

emerge est l'outil en ligne de commande du système Portage de gestion des paquetages :

```
%  emerge --version
Portage 2.1.7.17 (default/linux/amd64/10.0/server, gcc-4.3.4, glibc-2.10.1-r1, 2.6.24-24-xen x86_64)
```

Mais Gentoo offre d'autres possibilités, par exemple certains préfèrent utiliser Paludis <http://paludis.pioto.org/>. D'une manière générale, Gentoo dispose de plein d'outils rigolos <http://serverfault.com/questions/24323/gentoo-linux-useful-utilities>.

Par exemple :

```
[Liste des paquetages ("dpkg -l")]
% equery list
```

equery est livré dans le paquetage gentoolkit. Il sert à bien d'autres choses comme à trouver le paquetage auquel appartient un fichier.

emerge permet de mettre à jour tout le système d'un coup :

```
% emerge --update --deep --tree --verbose --ask world
```

Inutile de dire que cela prend du temps. Pire, on découvre souvent à cette occasion des problèmes subtils de dépendance, de bloquage d'un paquetage par un autre, qui sont longs et difficiles à régler. Je ne peux que conseiller d'appliquer la commande précédente fréquemment : plus on attend, plus elle devient cauchemardesque.

Et tout ne se passe pas toujours bien, ce qui ramène au bon (?) vieux temps où on était obligé de tout compiler. Il y a des fois des messages assez mystérieux, notamment concernant le masquage. Mais quand ça marche, on est tout heureux.

```
% sudo emerge -uDavt world
These are the packages that would be merged, in reverse order:
Calculating dependencies  .... .. ..... done!
[ebuild     U  ]  x11-libs/fltk-2.0_pre6970-r1 [2.0_pre6970] USE="doc jpeg png xft zlib
-cairo -debug -opengl -xinerama" 2,470 kB [0]
[ebuild     U  ]   dev-php/PEAR-PhpDocumentor-1.4.3-r1 [1.4.1] USE="-minimal" 2,367 kB [
0]
[ebuild  NS   #]   dev-lang/php-5.2.17 [5.3.6] USE="apache2 berkdb bzip2 cgi cli crypt c
type curl doc filter ftp gd gdbm hash iconv imap ipv6 json mysql ncurses nls pcre pdo pi
c posix postgres readline reflection session simplexml snmp sockets spl sqlite ssl token
izer truetype unicode xml xmlrpc xsl zlib (-adabas) -bcmath (-birdstep) -calendar -cdb -
cjk -curlwrappers -db2 -dbase (-dbmaker) -debug -discard-path -embed (-empress) (-empres
s-bcs) (-esoob) -exif (-fdftk) -firebird -flatfile -force-cgi-redirect (-frontbase) -gd-
external -gmp -inifile -interbase -iodbc -kerberos -kolab -ldap -ldap-sasl -libedit -mha
sh -msql -mssql -mysqli -oci8 -oci8-instant-client -odbc -pcntl -qdbm -recode -sapdb -sh
aredext -sharedmem -soap (-solid) -spell -suhosin (-sybase-ct) -sysvipc -threads -tidy -
wddx -xmlreader -xmlwriter -xpm -zip" 8,888 kB [0]
[ebuild     U  ] net-analyzer/rrdtool-1.4.5-r1 [1.3.8] USE="doc perl python -lua% -rrdcg
i -ruby -tcl (-nls%*)" 1,318 kB [0]
...
```

(Rappelez-vous que "*merge*", en jargon Gentoo, signifie « installer ».)

Si vous êtes sur une machine AMD64 et qu'emerge refuse de compiler avec des messages comme "*(masked by : missing keyword)*", il faut typiquement ajouter `amd64` après le nom du paquet dans `/etc/portage/pac` La FAQ explique pourquoi `<http://www.gentoo.org/doc/en/gentoo-amd64-faq.xml#keyword>`.

Notez que Gentoo dispose d'un Bugzilla qui est très actif `<https://bugs.gentoo.org/>`.

Et pour ceux qui ne veulent pas tout mettre à jour (même en version stable, c'est-à-dire testée pendant suffisamment longtemps par suffisamment de gens), il y a un outil, `glsa-check`, qui ne met à jour que les paquetages qui ont des alertes de sécurité :

```
% glsa-check --test all
This system is affected by the following GLSAs:
200809-05
200903-25
201006-18
201009-03
```

Très utile lorsque vous reprenez en main la gestion d'une Gentoo après une longue période de non-maintenance.

Il y a aussi des particularités de Gentoo qui peuvent être déroutantes, voire agaçantes. Par exemple, les démons ne sont pas redémarrés automatiquement en cas de mise à jour. C'est embêtant lorsque c'est une mise à jour de sécurité.

D'autre part, lorsqu'on met à jour une bibliothèque, la mise à jour des paquetages qui en dépendent n'est pas automatique. Si on a changé MYLIB, il faut penser à `revdep-rebuild --library=MYLIB` (sinon, on se récupère des erreurs du genre « "*error while loading shared libraries : libjpeg.so.62 : cannot open shared object file : No such file or directory*" ». Voici un exemple, après avoir recompilé OpenSSL, on recompile tout ce qui dépendait de la vieille version :

```
# revdep-rebuild --library=/usr/lib/libssl.so.0.9.8
 * Configuring search environment for revdep-rebuild

 * Checking reverse dependencies
 * Packages containing binaries and libraries using /usr/lib/libssl.so
 * will be emerged.

 * Collecting system binaries and libraries
 ...
 * found /usr/bin/htdig
 * found /usr/bin/ncat
 * found /usr/bin/ssh-keygen
 ...
 * Assigning files to packages
 * Assigning packages to ebuilds
 ...
emerge --oneshot
www-misc/htdig:0
net-analyzer/nmap:0
net-misc/openssh:0
...
```

Même chose lorsqu'on a mis à jour des programmes d'infrastructure comme les interpréteurs Perl ou Python. Heureusement, il existe des programme perl-cleaner et python-updater, qui servent justement à cela, et qu'il ne faut pas oublier de lancer.

Enfin, les dépendances ne sont gérées qu'à l'installation, pas à la désinstallation. Rien, même pas un avertissement, ne prévient qu'on est en train de retirer une bibliothèque vitale.

Et pour les fichiers de configuration ? Une fois un emerge réalisé, on fait un `etc-update` :

```
# etc-update
Scanning Configuration files...
The following is the list of files which need updating, each
configuration file is followed by a list of possible replacement files.
1) /etc/bind/named.conf (1)
2) /etc/init.d/named (1)
Please select a file to edit by entering the corresponding number.
              (don't use -3, -5, -7 or -9 if you're unsure what to do)
              (-1 to exit) (-3 to auto merge all remaining files)
                           (-5 to auto-merge AND not use 'mv -i')
                           (-7 to discard all updates)
                           (-9 to discard all updates AND not use 'rm -i'): -3
Replacing /etc/bind/named.conf with /etc/bind/._cfg0000_named.conf
mv: overwrite `/etc/bind/named.conf'? y
Replacing /etc/init.d/named with /etc/init.d/._cfg0000_named
mv: overwrite `/etc/init.d/named'? y
Exiting: Nothing left to do; exiting. :)
```

Il existe une alternative à `etc-update`, `dispatch-conf`.

Je l'avais signalé, mais Gentoo permet également des paquetages binaires qu'on peut installer avec :

```
emerge --usepkg --getbinpkgonly XXX
```

mais il semble qu'il faille un arbre portage complet. De toute façon, en l'absence d'un dépôt public de tels paquetages binaires, ils ne sont utilisables qu'à l'intérieur d'une organisation (si on a des dizaines de machines et qu'on ne veut pas que chacune recompile).

Comment passe t-on d'une version de Gentoo à une autre ? Le processus est documenté <http://www.gentoo.org/doc/en/gentoo-upgrading.xml> mais, en gros :

```
# cd /etc
# ln -sf /usr/portage/profiles/default/linux/amd64/10.0/server make.profile
```

et cela fait passer en version 10.0 (au fur et à mesure des compilations). Autre solution : `eselect profile set default/linux/amd64/10.0/server` (`eselect list` permet de voir tous les choix possibles).

Bien sûr, Gentoo permet aussi de créer son propre paquetage si son programme favori n'est pas encore disponible. On trouve de bonnes documentations en ligne <http://linuxreviews.org/gentoo/ebuilds/> mais le principe est simple :

— On écrit un fichier `ebuild` en s'inspirant de ceux qui existent dans `/usr/portage`. Voici par exemple celui que j'avais fait (en ligne sur `https://www.bortzmeyer.org/files/nsd-2.3.5.ebuild`) pour NSD (depuis, le paquetage a évolué <https://bugs.gentoo.org/show_bug.cgi?id=128246>).
— On met le fichier dans `/usr/local/portage/$DIR/$PROGRAM`.
— On crée les fichiers auxiliaires nécessaires avec `ebuild $PROGRAM-$VERSION.ebuild digest`.
— On peut ensuite le traiter comme un paquetage normal (`emerge -u $PROGRAM`).

Pour l'anecdote, voici quels avaient été les temps de fonctionnement de ma machine Gentoo chez Slicehost <https://www.bortzmeyer.org/slicehost-debut.html> :

```
% uprecords
     #               Uptime | System                                     Boot up
----------------------------+-------------------------------------------------
     1   370 days, 18:57:23 | Linux 2.6.32.9-rscloud    Mon Mar 22 20:20:22 2010
     2   335 days, 22:40:29 | Linux 2.6.16.29-xen       Mon Sep 22 16:28:39 2008
     3   261 days, 23:45:51 | Linux 2.6.16.29-xen       Fri Sep 14 10:46:04 2007
->   4   245 days, 07:01:34 | Linux 2.6.32.9-rscloud    Mon Mar 28 16:19:07 2011
     5   154 days, 16:14:35 | Linux 2.6.16.29-xen       Thu Apr 12 16:47:20 2007
```

Si vous voulez en savoir plus sur Gentoo :

— Le saviez-vous ? Le Gentoo est un pingouin (oui, je sais, en fait, c'est un manchot). J'ai découvert cela avec ces jolies photos <http://wiinterrr.blogspot.com/2011/11/images-of-day-yesterday24.html>.
— La documentation officielle <http://www.gentoo.org/doc/en/handbook/handbook-x86.xml>.
— Le Wiki de Gentoo <http://gentoo-wiki.com/>.
— Un bon HOWTO en français <http://gordon.so/sysadmin/howto-pratique-installation-compl html>.
— Un projet original, le système de paquetage Gentoo avec le noyau et une partie de FreeBSD <http://www.gentoo.org/doc/fr/gentoo-freebsd.xml>.

Merci à Samuel Tardieu pour son aide et ses remarques sur Gentoo.