Guide pratique d’utilisation de BIND en environnement restreint

Adaptation française du guide pratique Chroot-BIND HOWTO

Scott Wunsch

<scott CHEZ wunsch POINT org>

Adaptation française : Vincent Loupien

Relecture de la version française : Benoît Rouits, Jean-Philippe Guérard

Préparation de la publication de la v.f. : Jean-Philippe Guérard

Version : 1.5.fr.1.0

20 juillet 2005

Historique des versions		
Version 1.5.fr.1.0	2005-07-30	VL, BR, JPG
Version française relue par Benoît Rouits.		
Version 1.5.fr.0.9	2004-06-28	VL, JPG
Première version française (non relue).		
Version 1.5	2001-12-01	SW

Résumé

Ce document décrit l’installation du serveur de noms BIND 9 dans un environnement d’exécution restreint en tant qu’utilisateur non root. Cette configuration offre une meilleure sécurité et permet de limiter les effets potentiels d’une compromission. Ce document a été mis à jour pour la version 9 de BIND ; si vous utilisez toujours la version 8 de BIND, référez-vous plutôt au « Guide pratique d’utilisation de BIND 8 en environnement restreint ».

Table des matières

1. Introduction

Ce document est le guide pratique de BIND en environnement restreint ; reportez-vous à la Section 1.3, « Où ? » [p 3] pour trouver la version la plus récente de ce document. Nous supposerons que vous savez déjà configurer et utiliser BIND (le serveur de Noms de Domaines Internet de Berkeley). Si ce n'est pas le cas, je vous recommande de commencer par lire le guide pratique du DNS (DNS HOWTO). Nous supposerons également que vous avez une connaissance suffisante de la compilation et de l'installation d'un logiciel sur un système de type Unix.

1.1. Objet de ce document

Ce document présente quelques précautions de sécurité supplémentaires applicables lors de l'installation de BIND. Il explique comment configurer BIND de sorte qu'il réside dans un environnement restreint, ce qui signifie qu'il ne peut voir ou accéder aux fichiers à l'extérieur de sa propre arborescence. Nous le configurerons également pour s'exécuter en tant qu'utilisateur non root.

Le principe d'un environnement restreint est assez simple. Lorsque vous exécutez BIND (ou tout autre processus) dans un environnement restreint (c'est-à-dire en utilisant pour le système de fichier une racine différente — d'où le nom de la commande utilisée « *chroot* », c'est-à-dire, en anglais, « changer la racine »), le processus ne peut tout simplement pas voir les autres parties du système de fichiers (situées hors de son environnement). Vous avez probablement déjà rencontré un environnement restreint auparavant, si vous avez déjà utilisé un client **ftp** pour vous connecter à un serveur de fichier public.

Étant donné que le processus d'exécution en environnement restreint est beaucoup plus simple avec BIND 9, j'ai commencé à développer légèrement ce document, pour y inclure des astuces plus générales sur la manière de sécuriser une installation BIND. Néanmoins, ce document n'est pas (et ne souhaite pas devenir) une référence complète pour la sécurisation de BIND. Faire uniquement ce qui est décrit dans ce document ne suffit pas à sécuriser un serveur de nom !

1.2. Pourquoi ?

Le principe de l'exécution de BIND en environnement restreint est de limiter le degré d'accès dont pourrait bénéficier un individu malveillant en exploitant une des vulnérabilités de BIND. C'est pour la même raison que nous exécutons BIND en tant qu'utilisateur non root.

Ceci devrait être considéré comme un supplément aux précautions normales de sécurité (exécution de la dernière version, utilisation des listes de contrôle d'accès, et cætera), et non comme une solution de remplacement à ces dernières.

Si la sécurité du DNS vous intéresse, quelques autres produits pourraient également vous intéresser. Compiler BIND avec StackGuard peut être une bonne idée pour assurer une plus grande protection. Son utilisation est simple ; elle équivaut à utiliser un gcc standard. Il existe aussi une alternative sécurisée à BIND, DNScache, écrit par Dan Bernstein. Dan est l'auteur de qmail et DNScache semble en suivre la même philosophie.

1.3. Où ?

La dernière version française de ce document est toujours disponible sur le site du projet Traduc.org : http://www.traduc.org/docs/howto/lecture/Chroot-BIND8-HOWTO.html.

La dernière version originale de ce document est toujours disponible à partir du site Web des Utilisateurs de Linux et de logiciels libres de Regina, Sask. :
http://www.losurs.org/docs/howto/Chroot-BIND.html.

Il existe maintenant une traduction japonaise de ce document, maintenue par <`nakano CHEZ apm POINT seikei POINT ac POINT jp`>. Elle est disponible à l'adresse http://www.linux.or.jp/JF/JFdocs/Chroot-BIND-HOWTO.html. BIND est disponible à l'adresse de l'Internet Software Consortium à l'adresse http://www.isc.org/bind.html. Au moment de la publication de ce document, la version courante de BIND 9 est la version 9.2.0. La version 9 de BIND est sortie depuis longtemps et est déjà largement employée en production. Néanmoins, nombre de traditionalistes préfèrent encore utiliser BIND 8. Si c'est votre cas, reportez-vous à mon « Guide pratique de l'utilisation de BIND 8 en environnement restreint » (disponible au même endroit) qui vous expliquera comment l'exécuter en environnement restreint. Cependant, soyez conscient que BIND 8 est plus difficilement exécutable en environnement restreint. Gardez à l'esprit qu'il existe des trous de sécurité *connus* sur toutes les versions de BIND. Assurez-vous que vous exécutez bien la dernière version !

1.4. Comment ?

J'ai écrit ce document en me basant sur mon expérience du paramétrage de BIND dans un environnement restreint. Dans mon cas, j'avais déjà un BIND en exploitation sous la forme d'un paquet fourni par de ma distribution Linux. Je suppose que beaucoup d'entre-vous êtes dans la même situation, que vous allez juste récupérer et modifier les fichiers de configuration provenant de votre installation actuelle de BIND, puis désinstaller l'ancien paquet avant d'installer le nouveau. Ne désinstallez pas le paquet tout de suite ; nous pourrions avoir besoin d'y récupérer quelques fichiers.

Si vous n'êtes pas dans ce cas, vous devriez néanmoins être capable de comprendre ce document. La seule différence est que, lorsque je parle de copier un fichier existant, vous devrez d'abord le créer vous-même. Le guide pratique du DNS pourra vous être utile pour cela.

1.5. Mise en garde

Cette procédure a fonctionné pour moi, sur mon système. Vous pourriez avoir à la modifier. Ce n'est qu'une façon d'aborder la question ; il y a d'autres moyens d'arriver à la même solution (cependant l'approche restera la même). Il s'est juste trouvé que ma première tentative a fonctionné, et j'ai donc tout noté.

À ce jour, mon expérience de BIND se limite à l'installation sur des serveurs Linux. Cependant, la plupart des instructions de ce document devraient être facilement applicables à d'autres saveurs d'UNIX et j'essaierai d'indiquer les éventuelles différences dont j'aurais connaissance. J'ai également reçu des suggestions de personnes utilisant d'autres distributions et d'autres plates-formes, et j'ai essayé d'incorporer leurs commentaires lorsque cela était possible.

Si vous utilisez Linux, vous devez être sûr d'utiliser un noyau 2.4 avant d'essayer ceci. Le paramètre `-u` (exécution par un utilisateur non root) requiert cette version du noyau.

2. Préparation de l'environnement restreint

2.1. Création d'un utilisateur

Comme cela est mentionné dans l'introduction, il n'est pas conseillé de faire fonctionner BIND sous le compte root. Ainsi, avant de commencer, créons un utilisateur spécifique pour BIND. Notez que vous ne devez jamais employer un utilisateur générique comme `nobody` pour cela. Ainsi, quelques distributions, comme SuSE et Mandrake Linux ont commencé à fournir un utilisateur spécifique (généralement appelé `named`) ; si vous le souhaitez, vous pouvez tout simplement adapter cet utilisateur à nos desseins. Ceci exige l'ajout d'une ligne comme celle qui suit dans `/etc/passwd` :

```
named:x:200:200:Serveur de noms:/chroot/named:/bin/false
```

Et d'une ligne de ce type dans `/etc/group` :

```
named:x:200:
```

Ceci crée pour BIND un utilisateur et un groupe appelés `named`. Assurez-vous que l'UID et le GID (tous deux valant 200 dans cet exemple) sont uniques sur votre système. L'interpréteur de commande est mis à `/bin/false` car cet utilisateur n'aura jamais besoin de se connecter.

2.2. Arborescence de répertoires

Nous devons maintenant mettre en place l'arborescence de répertoires que nous allons utiliser pour l'environnement restreint d'exécution de BIND. Elle peut se situer n'importe où dans votre système de fichiers ; si vous êtes vraiment paranoïaque, vous pourrez même la placer dans un volume séparé. Je supposerai que vous allez employer `/chroot/named`. Commençons en créant l'arborescence de répertoires suivante :

```
/chroot
  +-- named
       +-- dev
       +-- etc
       |    +-- namedb
       |         +-- slave
       +-- var
            +-- run
```

Si vous utilisez la commande GNU **mkdir** (tel que présente sur les systèmes Linux), vous pourrez créer l'arborescence de répertoires ainsi :

```
# mkdir -p /chroot/named
# cd /chroot/named
# mkdir -p dev etc/namedb/slave var/run
```

2.3. Mise en place des données de BIND

Si vous avez déjà fait une installation conventionnelle de BIND et si vous l'utilisez, votre fichier `named.conf` et vos fichiers de zones existent déjà. Ces fichiers doivent être déplacés (ou copiés pour plus de sûreté) dans l'environnement restreint, de sorte que BIND puisse les atteindre. `named.conf` ira dans `/chroot/named/etc`, et les fichiers de zone pourront aller dans `/chroot/named/etc/namedb`. Par exemple :

```
# cp -p /etc/named.conf /chroot/named/etc/
# cp -a /var/named/* /chroot/named/etc/namedb/
```

BIND a normalement besoin d'écrire dans le répertoire `namedb`, mais pour renforcer la sécurité, nous ne l'autoriserons pas à le faire. Si votre serveur de nom est esclave pour une zone quelconque, il aura besoin de mettre à jour ces fichiers de zones, ce qui veut dire nous devrons les enregistrer dans un répertoire séparé, auquel BIND aura accès.

```
# chown -R named:named /chroot/named/etc/namedb/slave
```

Gardez à l'esprit que vous devrez déplacer toutes vos zones esclaves dans ce répertoire et que vous devez mettre à jour votre `named.conf` en conséquence.

BIND aura aussi besoin d'écrire dans le répertoire `/var/run`, pour y mettre ses fichiers pid et ses fichiers de statistiques, donc permettons-lui de le faire :

```
# chown named:named /chroot/named/var/run
```

2.4. Fichiers de support système

Lorsque BIND s'exécute dans l'environnement restreint, il ne peut plus *du tout* accéder aux fichiers situés hors de celui-ci. Cependant, il a besoin d'accéder à quelques fichiers clefs, bien que leur nombre soit bien moindre que ce dont BIND 8 avait besoin.

Un fichier dont BIND aura besoin à l'intérieur de sa prison est le bon vieux `/dev/null`. Notez que la commande exacte nécessaire pour créer ce fichier spécial peut varier de système à système ; vérifiez le script `/dev/MAKEDEV` pour vous en assurer. Quelques systèmes peuvent également exiger `/dev/zero`, que nous pourrons créer de la même façon. Il a été mentionné que les version préliminaires de BIND 9.2.0 ont maintenant également besoin de `/dev/random`. Pour la plupart des systèmes Linux, nous pourrons employer les commandes suivantes :

```
# mknod /chroot/named/dev/null c 1 3
# mknod /chroot/named/dev/random c 1 8
# chmod 666 /chroot/named/dev/{null,random}
```

Pour FreeBSD 4.3, ce sera :

```
# mknod /chroot/named/dev/null c 2 2
# mknod /chroot/named/dev/random c 2 3
# chmod 666 /chroot/named/dev/{null,random}
```

Vous aurez besoin de disposer d'un autre fichier dans le répertoire `/etc` de l'environnement restreint. Vous devrez copier `/etc/localtime` (nommé `/usr/lib/zoneinfo/localtime` sur certains systèmes), afin que BIND puisse enregistrer les évènements avec un horodatage correct. La commande suivante s'en chargera :

```
# cp /etc/localtime /chroot/named/etc/
```

2.5. Journalisation des évènements

BIND a beau être prisonnier de son environnement restreint, contrairement à un prisonnier ordinaire, il ne peut écrire son journal sur les murs de sa cellule `:-)` Normalement, BIND enregistre un journal des évènements grâce à syslogd, le démon de journalisation système. Cependant, ce type de journalisation est effectué en envoyant les enregistrements d'évènements vers le connecteur (*socket*) spécial `/dev/log`. Puisqu'elle se trouve désormais à l'extérieur de l'environnement restreint, BIND ne peut plus l'employer. Heureusement, il existe quelques solutions pour contourner le problème.

2.5.1. La solution idéale

La solution idéale de ce dilemme exige une version raisonnablement récente de syslogd qui prenne en charge le paramètre `-a` introduit par OpenBSD. Reportez-vous à la page de manuel de votre syslogd(8) pour vérifier si elle offre cette option. Si c'est la cas, la seule chose que vous ayez à faire est d'ajouter le paramètre « `-a /chroot/named/dev/log` » à la ligne de commande utilisée pour lancer syslogd. Sur les systèmes qui utilisent un init SysV complet (ce qui inclut la plupart des distributions Linux), vous pouvez faire cela en modifiant le fichier `/etc/rc.d/init.d/syslog`. Par exemple, sur mon système Linux Red Hat, j'ai changé la ligne

```
daemon syslogd -m 0
```

en

```
daemon syslogd -m 0 -a /chroot/named/dev/log
```

Il est intéressant de noter qu'à partir de la Red Hat 7.2, Red Hat a apparemment rendu ce processus plus facile. Il y a maintenant un fichier appelé `/etc/sysconfig/syslog` dans lequel on peut définit des paramètres supplémentaires pour syslogd. Les systèmes OpenLinux de Caldera utilisent un démon de lancement appelé ssd, qui lit la configuration depuis `/etc/sysconfig/daemons/syslog`. Il vous suffit de modifier la ligne d'options pour qu'elle ressemble à ceci :

```
OPTIONS_SYSLOGD="-m 0 -a /chroot/named/dev/log"
```

De la même façon sur les systèmes SuSE, il m'a été indiqué que le meilleur endroit pour ajouter ce paramètre est le fichier `/etc/rc.config`. Changez la ligne

```
SYSLOGD_paraMS=""
```

en

```
SYSLOGD_paraMS="-a /chroot/named/dev/log"
```

devrait faire l'affaire. Enfin, le dernier mais non le moindre, pour FreeBSD 4.3, il suffit apparemment de modifier le fichier `rc.conf` et d'y ajouter :

```
syslogd_flags="-s -l /chroot/named/dev/log"
```

Le `-s` est là pour des raisons de sécurité, et fait partie des paramètres par défaut. Le `-l` doit être suivi d'un chemin local, dans lequel on souhaite placer une autre interface de journalisation. Une fois que vous avez compris comment faire cette modification sur votre système, il vous suffira de redémarrer syslogd, que cela soit en le tuant (avec un **kill**) et en le relançant (avec les paramètres supplémentaires) ou en employant le script d'init SysV qui le fera pour vous :

```
# /etc/rc.d/init.d/syslog stop
# /etc/rc.d/init.d/syslog start
```

Une fois redémarré, vous devriez voir dans `/chroot/named/dev` une entrée appelée `log` qui ressemble à ceci :

```
srw-rw-rw-   1 root     root            0 Mar 13 20:58 log
```

2.5.2. L'autre solution

Si vous avez un ancien syslogd, vous devrez trouver une autre façon d'écrire dans le journal des évènements. Il existe quelques programmes pour cela, comme holelogd, qui est conçu pour agir comme un serveur mandataire en acceptant les entrées d'évènements du BIND en environnement restreint pour les passer au véritable connecteur `/dev/log`. Vous pouvez aussi tout simplement configurer BIND pour journaliser les évènements dans un fichier au lieu de les passer à syslog. Reportez-vous à la documentation de BIND pour plus d'informations si vous choisissez d'utiliser cette méthode.

2.6. Resserrer les permissions

Tout d'abord, n'hésitez à restreindre à l'utilisateur `root` l'accès à la totalité du répertoire `/chroot`. Bien sur, tout le monde ne voudra pas faire cela, particulièrement si cela indisposait d'autres logiciels installés dans la même arborescence.

```
# chown root /chroot
# chmod 700 /chroot
```

Vous pouvez aussi sans danger restreindre l'accès de `/chroot/named` à l'utilisateur `named`.

```
# chown named:named /chroot/named
# chmod 700 /chroot/named
```

Pour renforcer un peu plus la sécurité, sur les systèmes Linux utilisant le système de fichier ext2, nous pouvons rendre certains fichiers et répertoires immuables, en utilisant l'utilitaire `chattr`.

```
# cd /chroot/named
# chattr +i etc etc/localtime var
```

De même, sur FreeBSD 4.3, **chflags** est l'outil à examiner pour rendre certains fichiers immuables. À titre d'exemple, ce qui suit devrait rendre immuable le contenu du répertoire `/chroot/named/etc`:

```
# chflags schg /chroot/named/etc/*(*).
```

Il serait pratique de pouvoir faire la même chose sur le répertoire `dev`, mais malheureusement cela empêcherait syslogd de créer son connecteur `dev/log`. Vous pourrez aussi choisir de positionner le bit immuable sur d'autres fichiers de votre prison, comme par exemple vos fichiers de zone primaire, à condition qu'ils ne soient pas supposés changer.

3. Compiler et installer votre beau BIND tout neuf

3.1. Compiler

Compiler BIND 9 pour l'utiliser dans un environnement restreint est une expérience plus plaisante que cela ne l'était avec BIND 8. En fait, vous n'avez rien de spécial à faire ; le classique **`./configure && make`** suffira. Gardez à l'esprit que, sur des systèmes Linux, si vous voulez activer la compatibilité IPv6 de BIND (`--enable-ipv6`), vous devrez disposer de versions correspondantes du noyau et de la bibliothèque glibc. Si vous avez un noyau 2.2, vous aurez besoin de la glibc 2.1. Si vous avez un noyau 2.4, vous avez besoin de la glibc 2.2. BIND est plutôt pointilleux à ce propos.

4. Installer votre beau BIND tout neuf

Vous devez aussi savoir que, si vous avez déjà installé BIND, par exemple en utilisant un paquet RPM, vous devrez probablement le désinstaller avant d'installer votre nouvelle version. Sur un système Red Hat, cela implique probablement de désinstaller les paquets `bind` et `bind-utils`, et peut-être aussi `bind-devel` et `caching-nameserver`, si vous les avez.

Vous devriez sauvegarder une copie du script d'init (en général `/etc/rc.d/init.d/named`), s'il y a en un, avant la désinstallation ; ce sera utile plus tard.

Si vous réalisez une mise à jour depuis une ancienne version de BIND, tel que BIND 8, vous devriez lire le document de migration contenu dans le fichier `doc/misc/migration` du paquet source de BIND. Ce document ne traite pas du tout de la migration ; il part simplement de l'hypothèse que vous remplacez une installation existante et fonctionnelle de BIND 9.

4.1. Installer les binaires

C'est la partie facile :-) Lancez juste **`make install`** et laissez-le tout faire pour vous. Et voilà, c'est aussi simple que cela.

4.2. Mise en place du script d'init

Si vous avez un script d'init provenant de votre distribution, le mieux serait probablement de simplement le modifier pour exécuter le nouveau binaire, avec les paramètres appropriés. Les paramètres sont… *(roulement de tambour s'il vous plaît…)*

- `-u named`, pour exécuter BIND avec l'utilisateur `named` , plutôt que `root`.
- `-t /chroot/named`, pour que BIND s'exécute dans l'environnement restreint que nous avons mis en place.
- `-c /etc/named.conf`, pour que BIND trouve sa configuration à l'intérieur de la prison.

Ce qui suit est le script d'init que j'utilise avec mon système Red Hat 6.0. Comme vous pouvez voir, il est presque identique à celui livré par Red Hat. Je n'ai pas encore essayé la commande **rndc**, mais je ne vois pas pour quelle raison elle ne fonctionnerait pas.

```
#!/bin/sh
#
# named   Le rôle de ce script est de démarrer et d'arrêter
#         named (serveur DNS BIND).
#
# chkconfig: 345 55 45
# description: named (BIND) est le serveur de nom (DNS) \
# utilisé pour résoudre les noms de domaines en adresses IP.
# probe: true

# Lecture de la bibliothèque de fonctions.
. /etc/rc.d/init.d/functions

# Lecture des paramètres réseau.
. /etc/sysconfig/network

# Vérifie que le réseau fonctionne.
[ ${NETWORKING} = "no" ] && exit 0

[ -f /usr/local/sbin/named ] || exit 0

[ -f /chroot/named/etc/named.conf ] || exit 0

# En fonction de ce qui est appelé.
case "$1" in
  start)
        # Démarrer le démon.
        echo -n "Démarrage de named : "
        daemon /usr/local/sbin/named -u named -t /chroot/named \
                                     -c /etc/named.conf
        echo
        touch /var/lock/subsys/named
        ;;
  stop)
        # Arrêter le démon.
        echo -n "Arrêt de named : "
        killproc named
        rm -f /var/lock/subsys/named
        echo
        ;;
  status)
        status named
        exit $?
        ;;
  restart)
        $0 stop
        $0 start
        exit $?
        ;;
```

```
  reload)
        /usr/local/sbin/rndc reload
        exit $?
        ;;
  probe)
        # named sait comment redémarrer intelligemment ;
        # nous ne voulons pas que linuxconf nous propose
        # de le redémarrer à chaque fois
        /usr/local/sbin/rndc reload >/dev/null 2>&1 || echo start
        exit 0
        ;;

  *)
        echo "Utilisation: named {start|stop|status|restart|reload}"
        exit 1
esac

exit 0
```

Comme pour syslogd, à partir de la version 7.2 de Red Hat, ce processus est devenu encore plus simple. Il existe maintenant un fichier nommé `/etc/sysconfig/named` dans lequel il est possible d'ajouter des paramètres pour syslogd. Cependant, dans la distribution Red Hat 7.2, la version par défaut de `/etc/rc.d/init.d/named`, vérifie l'existence de `/etc/named.conf` avant de lancer BIND. Vous devrez corriger ce chemin.

Sur les systèmes OpenLinux de Caldera, vous avez juste besoin de modifier les variables définies au début et le script s'occupera apparemment du reste pour vous :

```
NAME=named
DAEMON=/chroot/named/bin/$NAME
OPTIONS="-t /chroot/named -u named -g named"
```

Et sous FreeBSD 4.3, vous pouvez éditer le fichier `rc.conf` et y ajouter les lignes suivantes :

```
named_enable="YES"
named_program="chroot/named/bin/named"
named_flags="-u named -t /chroot/named -c /etc/namedb/named.conf"
```

4.3. Changement de configuration

Vous devrez aussi ajouter ou modifier quelques options dans votre `named.conf` pour que vos divers répertoires soient correctement définis. En particulier, vous devrez ajouter (ou changer, si vous les avez déjà) les directives suivantes dans la section `options` :

```
directory "/etc/namedb";
pid-file "/var/run/named.pid";
statistics-file "/var/run/named.stats";
```

Ce fichier étant lu par le démon named, tous les chemins sont relatifs à l'environnement restreint. Au jour de la rédaction de ce document, BIND 9 ne permettait pas d'utiliser nombre des fichiers de statistiques et de vidage qu'il était possible d'utiliser avec la version précédente. Présumons que les prochaines le pourront ; si vous exécutez de telles configurations, vous devrez ajouter des entrées additionnelles pour forcer BIND à également écrire ces fichiers dans le répertoire `/var/run`.

5. Fin

5.1. Lancement de BIND

Tout devrait être configuré et vous devriez être prêt à lancer BIND, dans cette nouvelle version plus sûre. Si vous utilisez un script d'init de type Système V, vous pourrez le lancer tout simplement en utilisant la commande :

```
# /etc/rc.d/init.d/named start
```

Avant de faire cela, assurez-vous d'avoir arrêté toutes les anciennes versions de BIND qui pourraient encore fonctionner.

5.2. Voilà !

Vous pouvez aller faire un petit somme maintenant ;-)

A. Annexes

1. Mises à jour ultérieures de BIND

Vous avez maintenant un BIND 9.1.2 tout joliment casé dans son environnement restreint et assez peaufiné à votre goût… et vous entendez parler de cette désagréable rumeur disant que BIND 9.1.3 est finalement sorti. Vous vous sentez donc obligé de l'essayer sans attendre. Devrez-vous repasser entièrement par ce long processus pour installer cette nouvelle version ?

Pas du tout. En fait, vous aurez juste besoin de compiler le nouveau BIND et l'installer par dessus l'ancien. N'oubliez pas d'arrêter l'ancienne version et de redémarrer BIND, ou c'est l'ancienne version qui continuera à tourner !

2. Remerciements

Je voudrais remercier les personnes suivantes pour leur aide dans la création de ce guide pratique :

- Lonny Selinger <lonny CHEZ abyss POINT za POINT org> pour l'évaluation» de la première version de ce guide pratique et pour s'être assuré que je n'avais rien oublié.
- Chirik <chirik CHEZ CastleFur POINT COM>, Dwayne Litzenberger <dlitz CHEZ dlitz POINT net>, Phil Bambridge <phil.b CHEZ cableinet POINT co POINT uk>, Robert Cole <rcole CHEZ metrum TIRET datatape POINT com>, Colin MacDonald <colinm CHEZ telus POINT net>, et tous ceux qui ont mis le doigt sur des erreurs, des omissions et prodigué d'autres conseils utiles pour rendre encore meilleur ce guide pratique.
- Erik Wallin <erikw CHEZ sec POINT se> et Brian Cervenka <brian CHEZ zerobelow POINT org> pour avoir fourni de bonnes suggestions pour rendre encore plus sûr l'environnement restreint.

* Robert Dalton <support CHEZ accesswest POINT com> pour avoir suggéré une paire d'exemples supplémentaires et m'avoir indiqué que les BIND 9.2.0 avaient besoin de `/dev/random`.

* Eric McCormick <hostmaster CHEZ cybertime POINT net> pour les informations sur FreeBSD 4.3.

* Tan Zheng Da <tzd CHEZ pobox POINT com> pour les informations sur les changements survenus dans la version 7.2 de la distribution Red Hat, qui ont rendu le processus un peu plus facile.

Et le dernier mais certainement pas le moindre, je voudrais remercier Nakano Takeo <nakano CHEZ apm POINT seikei POINT ac POINT jp> pour avoir traduit en japonais ce guide pratique de BIND en environnement restreint. Vous pouvez trouver sa traduction à l'adresse http://www.linux.or.jp/JF/JFdocs/Chroot-BIND-HOWTO.html.

3. Politique de distribution de ce document