Guide pratique d'utilisation de BIND 8 en environnement restreint

Version française du Chroot-BIND8 HOWTO

Scott Wunsch

<scott CHEZ wunsch POINT org>

Adaptation française : Vincent Loupien

Relecture de la version française : Isabelle Hurbain

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

Version : 1.4.fr.1.1

21 août 2005

Historique des versions		
Version 1.4.fr.1.1	2005-08-21	VL,IH,JPG
Correction de l'url de la version française		
Version 1.4.fr.1.0	2004-03-05	VL,IH,JPG
Première traduction française		
Version 1.4	2001-07-01	SW

Résumé

Ce document décrit l'installation du serveur de noms BIND 8 dans un environnement restreint en tant qu'utilisateur non privilégié. Ce qui permet de disposer d'une sécurité améliorée et de réduire au minimum les effets potentiels d'une compromission. Cette version du document couvre l'ancienne version 8 de BIND, toujours populaire ; il existe un autre document qui fournit le même type d'informations, mais pour BIND 9.

Table des matières

1. Introduction

Ceci est le guide pratique de BIND 8 en environnement restreint ; allez voir Section 1.3, « Où ? » [p 3] pour le site principal, qui contient la dernière version de ce document. Nous supposons 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 lire d'abord le Guide pratique du DNS (*DNS HOWTO*). Nous supposons également que vous avez une connaissance suffisante de compilation et d'installation d'un logiciel sur votre système de type Unix.

1.1. Objet de ce document

Ce document décrit quelques précautions de sécurité supplémentaires que vous pouvez prendre quand vous installez BIND. Il explique comment configurer BIND de sorte qu'il réside dans un environnement restreint, ceci signifiant qu'il ne peut pas voir ou avoir accès aux fichiers à l'extérieur de sa propre arborescence. Nous le configurerons également pour l'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 avec une racine différente du système de fichier — 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 en dehors de son environnement. Par exemple, dans ce document, nous placerons BIND pour être exécuté en environnement restreint dans le répertoire `/chroot/named`. Cependant, pour BIND, le contenu de ce répertoire apparaîtra comme étant `/`, la racine. Il ne pourra accéder à rien d'autre en dehors de ce répertoire. Vous avez probablement déjà rencontré un environnement restreint auparavant, si vous avez déjà fait un **ftp** vers un serveur de fichier public.

1.2. Pourquoi ?

Le principe lors de l'exécution de BIND dans un environnement restreint est de limiter la quantité d'accès que n'importe quel individu malveillant pourrait gagner 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 pas comme une solution de remplacement de 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 Berstein. 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 Logiciel Libre de Regina, Saskatchewan, Canada (LOSURS) à l'adresse http://www.losurs.org/docs/howto/Chroot-BIND8.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-BIND8-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 8 est 8.2.4. BIND 9.x est maintenant sorti, et il fonctionne depuis un petit moment. Vous pouvez envisager de mettre à jour vers cette version, la procédure d'environnement restreint y est vraiment beaucoup plus simple et propre. Si vous exploitez BIND 9, alors utilisez le « guide pratique d'utilisation de BIND en environnement restreint » qui doit être disponible au même emplacement que ce document.

Gardez à l'esprit que des trous de sécurité sont *connus* dans toutes les versions de BIND 8 inférieure à *8.2.3*, assurez-vous que vous exécutez bien la dernière version !

1.4. Comment ?

J'ai écrit ce document à partir de 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 paquetage provenant de ma distribution Linux. Je vais supposer 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 le paquetage avant d'installer le nouveau. Ne désinstallez pas le paquetage 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 copie un fichier existant, vous devrez d'abord le créer vous-même. Le guide pratique du DNS peut être utile pour cela.

1.5. Mise en garde

Cette procédure a fonctionné pour moi, sur mon système. Vous pouvez 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 dans ce document doivent être facilement applicables à d'autres saveurs d'UNIX, et j'essaierai d'indiquer les éventuelles différences dont j'ai la connaissance.

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 en 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`) ; vous pouvez simplement adapter cet utilisateur à nos desseins, si vous le souhaitez.

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 une comme ceci dans `/etc/group` :

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

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

2.2. Arborescence de répertoire

Nous devons maintenant mettre en place l'arborescence de répertoire que nous allons utiliser pour l'environnement restreint dans lequel BIND s'exécutera. Cela peut être n'importe ou dans votre système de fichiers ; celui qui est vraiment paranoïaque pourra même la mettre dans un volume séparé. Je supposerai que vous emploierez `/chroot/named`. Commençons en créant l'arborescence de répertoire suivante :

```
/chroot
  +-- named
       +-- bin
       +-- dev
       +-- etc
       |    +-- namedb
       +-- lib
       +-- 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 zone 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 devra sûrement écrire dans le répertoire `namedb`, et probablement dans certains des fichiers contenus dans ce répertoire. Par exemple, si votre serveur DNS est esclave pour une zone, il devra y mettre à jour les fichiers de cette zone. De plus, BIND peut générer des statistiques, ce qu'il fait dans ce répertoire. Pour cette raison, vous devrez probablement faire de l'utilisateur `named` le propriétaire de ce répertoire et de son contenu :

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

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

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

2.4. Fichiers pour le support du système

Une fois que BIND s'exécute dans l'environnement restreint, il n'est pas capable *du tout* d'avoir accès aux fichiers en dehors de celui-ci. Cependant, il doit avoir accès à quelques fichiers clefs, comme la bibliothèque C du système. Les bibliothèques exigées dépendront de votre saveur d'Unix. Pour la plupart des systèmes Linux modernes, les commandes suivantes seront suffisantes pour mettre en place les bibliothèques nécessaires :

```
# cd /chroot/named/lib
# cp -p /lib/libc-2.*.so .
# ln -s libc-2.*.so libc.so.6
# cp -p /lib/ld-2.*.so .
# ln -s ld-2.*.so ld-linux.so.2
```

Vous pouvez aussi simplement compiler des versions statiquement liées des binaires de BIND pour les placer dans votre environnement restreint. Vous devez aussi copier **ldconfig** dans l'environnement restreint et l'exécuter pour créer un `etc/ld.so.cache` pour l'environnement restreint. Les commandes suivantes devraient le permettre :

```
# cp /sbin/ldconfig /chroot/named/bin/
# chroot /chroot/named /bin/ldconfig -v
```

BIND a encore besoin d'un fichier système dans son environnement restreint : le bon vieux `/dev/null`. De nouveau, la commande exacte nécessaire pour créer ce fichier spécial peut varier de système en système ; vérifiez votre script `/dev/MAKEDEV` pour être sûr. Quelques systèmes peuvent également exiger `/dev/zero`. Pour la plupart des systèmes Linux, nous pouvons employer la commande suivante :

```
# mknod /chroot/named/dev/null c 1 3
```

Pour terminer, vous avez besoin de quelques fichiers supplémentaires dans le répertoire `/etc` à l'intérieur de l'environnement restreint. En particulier, vous devez copier `/etc/localtime` (parfois nommé `/usr/lib/zoneinfo/localtime` sur certains systèmes) de façon à ce que BIND enregistre les évènements avec un horodatage correct. Vous devez également créer un petit fichier `group` contenant uniquement le groupe `named`. Les deux commandes suivantes se chargeront de ceci :

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

# echo 'named:x:200:' > /chroot/named/etc/group
```

Gardez à l'esprit que le GID, 200 dans cet exemple, doit correspondre à celui que vous avez défini dans le vrai `/etc/group` défini au dessus.

2.5. Journalisation des évènements

BIND a beau être prisonnier de son environnement restreint, il ne peut pas graver son journal sur les murs de sa cellule. :-). Normalement, BIND écrit les journaux grâce à **syslogd**, le démon de journalisation des évènements du système. Cependant, ce type de journalisation est effectué en envoyant les entrées d'évènements vers le socket spécial `/dev/log`. Puisqu'il est à l'extérieur de l'environnement restreint, BIND ne peut plus l'employer désormais. Heureusement, il existe deux solutions pour contourner cela.

2.5.1. La solution idéale

La solution idéale de ce dilemme exige une version raisonnablement récente de **syslogd** qui prend en charge le paramètre `-a` introduit par OpenBSD. Reportez-vous aux pages de manuel de votre syslogd(8) pour voir si vous avez une telle version.

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 lorsque vous lancez **syslogd**. Sur les systèmes qui utilisent un init SysV complet (ce qui inclut la plupart des distributions Linux), vous pouvez faire cela dans 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
```

Les systèmes Caldera OpenLinux utilisent un démon de lancement appelé **ssd**, qui lit la configuration dans `/etc/sysconfig/daemons/syslog`. Il vous suffit de modifier la ligne d'options pour que cela ressemble à ceci :

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

De la même façon sur les systèmes SuSE, je me suis dit que le meilleur endroit pour ajouter ce paramètre est le fichier `/etc/rc.config`. Changer la ligne

```
SYSLOGD_PARAMS=""
```

en

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

devrait faire l'affaire. Une fois que vous avez compris comment faire cette modification sur votre système, il vous suffit de redémarrer **syslogd**, que cela soit en l'arrêtant 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 devez voir dans `/chroot/named/dev` un « fichier » appelé `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**, alors vous devez trouver une autre façon de faire vos journalisations. Il existe quelques programmes pour faire ça, comme `holelogd`, qui est conçu pour agir comme un « proxy » en acceptant les entrées d'évènements du BIND en environnement restreint pour les passer au véritable socket `/dev/log`.

Vous pouvez aussi tout simplement configurer BIND pour journaliser les évènements dans un fichier au lieu de les passer à syslog. Voyez la documentation de BIND pour plus de détails si vous choisissez d'utiliser cette méthode.

3. Compilation de BIND

Vous devriez pouvoir trouver les sources de BIND en visitant http://www.isc.org/bind.html. Vous avez besoins du paquet `bind-src.tar.gz`. Assurez-vous de bien récupérer la dernière version !

3.1. Modifier les chemins

Les choses peuvent s'embrouiller un peu à partir de maintenant, parce que les différentes parties du paquetage BIND se réfèrent aux mêmes répertoires par des noms différents (dépendant du fait qu'ils s'exécutent ou non dans l'environnement restreint). Je vais essayer de ne pas *trop* vous embrouiller.

Le répertoire dont nous devons nous occuper en priorité est `/var/run` car son contenu est nécessaire à la fois pour le démon **named** (à l'intérieur de l'environnement restreint) et pour l'utilitaire **ndc** (à l'extérieur). Nous allons commencer par paramétrer ce qu'il faut pour trouver ce répertoire depuis le monde extérieur. Pour cela, nous devons modifier `src/port/linux/Makefile.set` (substituez

par le répertoire de votre architecture si vous ne fonctionnez pas sur Linux), et changez la ligne

```
DESTRUN=/var/run
```

en

```
DESTRUN=/chroot/named/var/run
```

Tant que vous êtes là, vous pouvez changer l'autre chemin de destination `/usr` en `/usr/local`. Maintenant, tout devrait être capable de trouver ce répertoire… excepté le démon **named** lui-même, pour qui c'est toujours le vrai `/var/run` dans l'environnement restreint. Nous pouvons contourner ceci en faisant un petit changement dans les sources de **named**. Dans le fichier `src/bin/named/named.h`, trouvez la ligne

```
#include "pathnames.h"
```

et ajouter la ligne suivante immédiatement après

```
#define _PATH_NDCSOCK    "/var/run/ndc"
```

De cette façon, **named** ignorera notre définition de `DESTRUN` dans `Makefile.set` et emploiera l'emplacement correct (par rapport à sa perspective dans l'environnement restreint). Vous remarquerez quelques avertissements au sujet des redéfinitions de _PATH_NDCSOCK quand vous faites la compilation ; vous pouvez les ignorer.

3.2. Compiler

Vous devriez maintenant être capable de compiler normalement BIND, en suivante les instructions du fichier `INSTALL`. À cette étape, nous voulons seulement compiler BIND, sans l'installer. N'allez pas trop loin en suivant le fichier `INSTALL`. Globalement, il faut juste faire **make clean**, **make depend** et **make**.

4. Installer votre beau BIND tout neuf

Je dois signaler que si vous avez une installation existante de BIND, par exemple en provenance d'un RPM, vous devrez probablement la désinstaller avant d'installer la nouvelle. Sur un système Red Hat, cela implique probablement de désinstaller les paquetages *bind* et *bind-utils*, et peut-être *bind-devel* et *caching-nameserver*, si vous les avez.

Vous voudrez sans doute sauvegarder une copie du script d'init (par exemple `/etc/rc.d/init.d/named`), s'il y a en un, avant de faire ceci ; ce sera utile plus tard.

4.1. Installer les outils en dehors de l'environnement restreint

C'est la partie facile :-). Il vous suffit d'exécuter **make install** et laissez faire le tout pour vous. Vous pouvez vouloir faire un **chmod 000 /usr/local/sbin/named** par la suite, pour être sûr que vous n'exécutez pas accidentellement une copie de BIND hors environnement restreint (il s'agit de `/usr/sbin/named` si vous ne lui avez pas dit d'aller dans `/usr/local/sbin` comme je l'ai suggéré).

4.2. Installer les binaires dans l'environnement restreint

Seuls deux parties du paquetage doivent s'exécuter à l'intérieur de l'environnement restreint : le démon principal **named** lui-même, et **named-xfer**, qui est utilisé pour les transferts de zone. Vous pouvez simplement les copier depuis l'arborescence source :

```
# cp src/bin/named/named /chroot/named/bin

# cp src/bin/named-xfer/named-xfer /chroot/named/bin
```

4.3. 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 **/chroot/named/bin/named**, avec les paramètres appropriés. Les paramètres sont… (roulement de tambour s'il vous plaît…)

- `-u named`, qui demande à BIND de s'exécuter en tant qu'utilisateur `named`, plutôt que `root`.
- `-g named`, pour exécuter BIND avec le groupe `named` également, plutôt que `root` ou `wheel`.
- `-t /chroot/named`, qui demande à BIND de s'exécuter dans l'environnement restreint que nous avons construit.

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. J'ai aussi modifié la commande **`ndc restart`** de façon à ce qu'elle redémarre le serveur correctement, et le garde à l'intérieur de l'environnement restreint. Vous pouvez probablement faire la même chose dans votre script d'init, sans avoir à copier celui-ci.

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

# Bibliothèque basique de fonctions.
. /etc/rc.d/init.d/functions

# Configuration basique du réseau.
. /etc/sysconfig/network

# Vérifie que la gestion du réseau est assurée
[ ${NETWORKING} = "no" ] && exit 0

[ -f /chroot/named/bin/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 /chroot/named/bin/named -u named -g named -t /chroot/named
        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)
        /usr/local/sbin/ndc status
        exit $?
        ;;
  restart)
        /usr/local/sbin/ndc -n /chroot/named/bin/named "restart -u named -g named -t /chroot/named"
        exit $?
        ;;
  reload)
        /usr/local/sbin/ndc 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/ndc reload >/dev/null 2>&1 || echo start
        exit 0
        ;;

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

exit 0
```

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

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

4.4. Changement de configuration

Vous devez aussi ajouter ou modifier quelques options dans votre `named.conf` pour avoir vos différents répertoires en ordre. En particulier, vous devez ajouter (ou changer, si vous les avez déjà) les directives suivantes dans la section `options` :

```
directory "/etc/namedb";
pid-file "/var/run/named.pid";
named-xfer "/bin/named-xfer";
```

Puisque ce fichier est lu par le démon **named**, tous les chemins sont naturellement relatifs à l'environnement restreint.

Quelques personnes ont aussi rapporté devoir ajouter quelques lignes supplémentaires à leur `named.conf` pour obtenir un fonctionnement correct de **ndc** :

```
controls {
    unix "/var/run/ndc" perm 0600 owner 0 group 0;
};
```

5. La fin

5.1. Lancement de BIND

Tout devrait être configuré, et vous devriez être prêt à lancer votre nouveau BIND plus sécurisé. Si vous utilisez un script d'init du style SysV, vous pouvez simplement le lancer par :

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

Assurez-vous d'avoir arrêté toutes les anciennes versions de BIND qui pourraient encore fonctionner avant de faire cela.

Si vous jetez un coup d'œil à votre journal système, vous devez trouver les messages d'initialisations que BIND crache quand il démarre. (si ce n'est pas le cas ; il y a un problème avec votre configuration de journalisation [p 6] que vous devez résoudre.) Parmi ces messages, BIND devrait vous indiquer qu'il est passé avec succès dans l'environnement restreint (*chroot* et qu'il fonctionne en tant qu'utilisateur et groupe `named`. Si ce n'est pas le cas, vous avez un problème.

5.2. Voilà !

Vous pouvez aller faire un petit somme maintenant ;-).

A. Annexes

1. Mises à jour ultérieures de BIND

Ainsi, vous avez un BIND 8.2.2_P7 tout joliment placé dans son environnement restreint et assez peaufiné à votre goût… et vous entendez parler de cette rumeur désagréable qu'il y a une faille root exploitable à distante dans cette version également, et vous avez besoin de mettre à jour en 8.2.3 tout de suite. Devez-vous repasser entièrement par ce long processus pour installer cette nouvelle version ?

Non. En fait, vous avez juste besoin de la section Section 3, « Compilation de BIND » [p 7] et les deux premières parties de la section Section 4, « Installer votre beau BIND tout neuf » [p 8] (installation des binaires en dehors et à l'intérieur de l'environnement restreint, respectivement).

Le reste de ce guide pratique traite de la mise en place de l'environnement restreint et d'autres choses de ce genre-là, qui ne devrait pas devoir être changé entre les versions du BIND. Vous devez juste déposer les nouveaux binaires par-dessus les anciens, et vous pouvez continuer. Mais n'oubliez pas d'arrêter et de redémarrer BIND par la suite ou c'est l'ancienne version, vulnérable, qui continuera à tourner !

2. Remerciements

Je voudrais remercier les gens suivants 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 POINT b CHEZ cableinet POINT co POINT uk>, Robert Cole <rcole CHEZ metrum-datatape POINT com>, Colin MacDonald <colinm CHEZ telus POINT net>, et d'autres qui ont mis le doigt sur des erreurs, des omissions, et prodigué d'autres conseils utiles pour faire que ce guide pratique soit meilleur encore.

- Erik Wallin <erikw CHEZ sec POINT se> et Brian Cervenka <brian CHEZ zerobe-low POINT org> pour avoir fourni de bonnes suggestions pour mieux sécuriser encore l'environnement restreint.

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 8 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