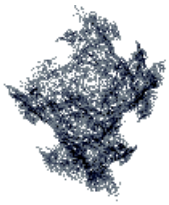

Sécuriser un serveur de nom

Home

Linux Mag
- automount
- Bastille Linux
- IPv6
- NFS
- NIS
- Secure Programming
- xinetd
- Apache

HS Sécurité (MISC)
- Cryptographie
- Sécuriser un réseau
- Virus !
- Root-kit et intégrité
- SSH
- Les attaques externes
- Filtrage sous Linux
- IDS
- Les tests d'intrusion
- Apache
- Bind
- ProFTP
- Postfix

Misc.
- fmtbuild-howto
- Nustra

Research
- Ant sort
- Evolutionary Algorithms
- IFS Museum
- L-Systems
- Primality Test
- RSA
- Watermarking

Download
- Download

Quelle version de BIND pour votre serveur ?

Ces derniers temps, de nombreuses erreurs d'implémentation (tsig, infoleak,...) ont été découvertes dans BIND. Le détail de ces problème http://www.isc.org/products/BIND/bind-security.html. Ainsi à l'heure actuelle, il vous faut installer la version 8.2.3 ou la version 9.1. Bind 9 i TSIG (disponible à partir de la version 8.2) pour signer les requêtes DNS, ce qui devrait résoudre à terme les différents problèmes liés au l le support d'IPv6 et de nouveaux protocoles (IXFR, DDNS, Notify, EDNSO).

Si vous vous demandez si BIND est capable de supporter de lourdes charges, le serveur F.root-servers.net utilisant bind 8.2.3 répond à pl

Installation

Vous pouvez aussi bien utiliser des packages pré-compilés que compiler bind vous-même à partir des sources. Celles-ci sont disponibles packages RedHat, il vous faut installer bind, bind-utils ainsi que bind-devel si vous programmez.

Configuration

Le fichier de configuration de bind est généralement le fichier /etc/named.conf. Quelques mots rapides sur la configuration :

- les commentaires sont notés /* */ ou // comme en C ou commencent par un ";"
- SOA: Source Of Autority
- NS: Name Server
- MX: Mail eXchange
- CNAME: alias de nom
- Les noms de machine doivent se terminer par un point.

Le fichier se découpe en sections: options, logging et zone. Je ne m'étendrais pas sur le sujet, j'espère que vous maîtrisez suffisamment le Bind.

Parano par défaut

Prenons le parti d'être parano, interdisons tout par défaut.

```
options {
  also-notify { none; };
  allow-transfer { none; };
  allow-query { none; };
};
```

also-notify doit contenir les DNS secondaires non officiels, cela permet de les prévenir immédiatement si une mise à jour d'une zone est et

allow-transfer indique quelles sont les machines autorisées à effectuer un transfert de zone, le transfert de zone étant comme son nom l'in totalité des paramètres d'une zone. Seuls DNS secondaires devraient y être autorisés.

allow-query indique qui peut interroger le serveur pour une zone donnée.

Les ACL (*access control lists*) permettent de définir des ensembles de machines et/ou de réseaux. Quatre ACL sont pré-définies : any, nor respectivement à tout le monde, personne, le serveur seulement et l'ensemble des réseaux définis par les adresses IP et netmasks de la r

Une ACL se définit de la façon suivante :

```
acl name {
  address_match_list
};
```

Par exemple, je peux définir l'acl dns_sec_non_officiel comportant deux adresses IP de la manière suivante :

```
acl dns_sec_non_officiel {
  192.168.1.2;
  192.168.3.4;
}
```

Définitions des zones et utilisation des ACL

Pour obtenir la liste des DNS primaires, il suffit d'exécuter la commande dig @ns.internic.net . ns > named.ca. Ils résolvent les adresses ip primaire, ni secondaire.

La zone pour les requêtes par défaut est la zone "." définie ci-dessous :

```
zone "." {
  type hint;
  file "named.ca";
  allow-query { mon_parc_info; };
};
```

mon_parc_info est une acl contenant les IPs des machines de mon réseau.

Dans certains cas d'architecture réseau ou pour des raisons de sécurité, on peut vouloir forcer un DNS interne à utiliser un serveur DNS b
Dans ce cas-là, on impose cette contrainte dans les options :

```
options {
        ....
        forward only;
        forwarders { mes_dns_en_sortie; };
};
```

Autre remarque, si votre machine utilise du "dial on demand" ou connexion à la demande (la connexion par modem s'active s'il y a des req
dialup yes; pour éviter que votre connexion ne soit démarrer de manière intempestive.

Une zone de résolution inverse, adresse IP vers nom de machine, pour l'adresse de loopback se définit ainsi

```
zone "0.0.127.in-addr.arpa" {
  type master;
  allow-query { mon_parc_info; };
  file "named.local";
  };
```

Un serveur primaire doit être interroger par tous mais n'autoriser que les dns secondaires à effectuer des transferts de zone.

```
zone "domaine1.org" {
  type master;
  file "domain1/domaine1.org";
  also_notify { dns_sec_non_officiel; };
  allow-transfer { dns_sec_officiel; dns_sec_non_officiel; };
  allow-query { any; };
}
```

On définie ces acls pour chaque zone.

Les DNS indiqués dans le paramètre also_notify sont avertis dès qu'une modification est intervenue dans une zone. C'est utilisé concrèter officiels, c'est à dire ceux qui ne sont pas explicitement déclaré comme NS dans le fichier de définissant la zone.

Certains services proposés par les ISP ne sont accessibles que par l'utilisation de leur DNS. Ainsi, pour accéder aux news ou lire son mail utiliser le DNS spécifié sur la feuille indiquant le login/password ou donné par le serveur DHCP.

```
zone "wanadoo.fr"
{
  type forward;
  forwarders  { 193.252.19.3; };
};
```

NB: Wanadoo filtre les IP accédant aux DNS, serveurs web,... réservés aux abonnés.

Cacher le numéro de version

Le numéro de version s'obtient facilement :

```
$ nslookup - 127.0.0.1
Default Server:  localhost
Address:  127.0.0.1

> set class=chaos
> set q=txt
> version.bind
Server:  localhost
Address:  127.0.0.1
```

```
VERSION.BIND    text = "8.2.3-REL"
```

Une possibilité est de définir le texte à renvoyer. Dans la rubrique options, il suffit de spécifier version "mon super texte". Mais cela a l'inco
peut ruser un peu plus en redéfinissant la zone "chaos" :

```
zone "bind" chaos {
  type master;
  file "bind";
  allow-query { localhost; };
};
```

Cette zone recouvre un ensemble d'information sur le serveur comme sa version`VERSIO`
`authors.bind` (Bind 9.1 et sup). Dans le fichier `bind`, je définis la classe `chaos`.

```
$TTL 1D
$ORIGIN bind.
@       1D  CHAOS SOA   localhost.      root.localhost. (
                                        1
                                        3H
                                        1H
                                        1W
                                        1D  )
CHAOS  NS       localhost.
```

Ainsi lorsqu'un petit malin cherche à obtenir la version de bind, sa tentative sera enregis

```
Mar 12 17:51:12 vectra named[17035]: unapproved query from [10.0.0.169]
```

Et on pourra continuer à récupérer le numéro de version en local sauf si on est encore pl
`allow-query { none; };`.

Sécurisation

Les ports inférieurs à 1024 sont des ports privilégiés, c'est-à-dire que seuls les programmes fonctionnant en tant que root peuvent les utilis
TCP/UDP 53 (domains).

Un mécanisme de sécurité est incorporé dans les versions modernes de bind qui lui permet, une fois qu'il s'est attribué le port domains (53
pouvoir, *named* généralement. Ainsi, en cas de faille de sécurité, le pirate prend l'identité named et non root.

Cependant, le pirate possède désormais un accès à la machine, où il est en mesure d'exploiter des failles locales. L'idée est de le confiner
exemple, le répertoire /var/chroot-named où se trouvent également les fichiers nécessaires à bind.

Création de la prison

Pour créer la prison, il faut commencer par être root.

```
  $ mkdir /var/chroot-named
  $ mkdir /var/chroot-named/dev
  $ mkdir /var/chroot-named/etc
  $ mkdir /var/chroot-named/var
  $ mkdir /var/chroot-named/var/run
  $ mkdir /var/chroot-named/usr
  $ mkdir /var/chroot-named/usr/sbin
  $ mknod /var/chroot-named/dev/null c 1 3
```

Création du compte

```
  $ adduser named -s /bin/false
  $ egrep "(^root:|^named:)" /etc/passwd > /var/chroot-named/etc/passwd
  $ egrep "(^root:|^named:)" /etc/group > /var/chroot-named/etc/group
```

Configuration du système de log

Si syslogd supporte l'option « -a », il faut qu'il soit démarrer avec dans /etc/rc.d/init.d/syslog :

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

Dans le cas contraire, il faut alors loguer directement :

```
  $ mkdir /var/chroot-named/var/log
  $ ln -s /var/chroot-named/var/log /var/log/dns
```

Dans named.conf :

```
logging {
  channel replace_syslog{
    file "/var/log/dns" versions 3 size 100k;
    severity info;
    print-category yes;
    print-severity yes;
    print-time yes;
  };
  category default { replace_syslog; default_debug; };
};
```

Les fichiers de log appartiennent à root.

Récupération de la configuration

```
  $ mv /etc/named.conf /var/chroot-named/etc
  $ mv /var/named /var/chroot-named/var
  $ ln -s /var/chroot-named/etc/named.conf /etc/named.conf
  $ ln -s /var/chroot-named/var/named /var/named
  $ chown -R named:named /var/chroot-named/var/named
```

Rappelz-vous également que bind doit pouvoir écrire dans les répertoires correspondant
donc les créer et les attribuer à l'utilisateur named, sans quoi il ne pourra pas récupérer l

Installation des binaires

Soit on compile named et named-xfer en statique, soit on utilise les binaires existants et on copie les librairies dynamiques nécessaires :

```
  $ cp /usr/sbin/named /var/chroot-named/usr/sbin
  $ cp /usr/sbin/named-xfer /var/chroot-named/usr/sbin
  $ mkdir /var/chroot-named/lib
  $ mkdir /var/chroot-named/usr/lib
  $ ldd /usr/sbin/named-xfer
         libc.so.6 => /lib/libc.so.6 (0x40022000)
         /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Cette dernière commande montre toutes les librairies dont bind a besoin pour fonctionne
également être présente dans la prison.

Utilisation

Dans /etc/rc.d/init.d/named (ici, supprimer le daemon si vous n'avez pas une RedHat), on signale que le démon tourne dans une prison :

```
daemon named -u named -t /var/chroot-named
```

L'option "-u" désigne l'utilisateur sous lequel fonctionnera bind et l'option "-t" le réperto
prison.

Pour la fine bouche

La sécurité TSIG permet de réduire les conséquences de l'IP spoofing entre les DNS primaires et secondaires. Ils partagent une clé secrè
grenier-supersecret dans l'exemple. La clé est encodée en base 64 head -c 16 /dev/urandom | uuencode -m - et se définit ainsi dans les fic

```
key grenier-supersecret. {
  algorithm hmac-md5;
  secret "mZiMNOUYQPMNwsDzrX2ENw==";
};
```

Dans la configuration du DNS maître, on ajoute :

```
server ip_dns_secondaire {
  transfer-format many-answers;
  keys { grenier-supersecret.; };
};
```

et pour les DNS secondaires, on met la même chose mais avec l'adresse IP du serveur D

Pour fonctionner, les horloges systèmes des deux serveurs doivent être synchronisées. Au besoin, on peut les synchroniser par xntp. La c serveur le plus vulnérable.

Conclusion

Avec l'apport de la cryptographie, les protocoles DNS sont plus fiables et les mécanismes de sécurité système comme le chrootage sont t d'implémentation =:-) (Bug TSIG).

Christophe GRENIER - grenier@nef.esiea.fr cgr@global-secure.fr

---

frederic.raynal@inria.fr

Last modified : Tuesday July 24 2001 -- 15:16