服务器知识手册（十九）

狄登峰 主编

需要全本请访问 www.ertongbook.com

目　　录

域名服务器.....1
BIND8+域名服务器安全增强.....1
设置过多 DNS 服务器地址的问题.....14
如何在 Linux 服务器上设置 DNS.....15
谈谈 DNS 在活动目录中的应用.....20
动态改变/添加网络设置中 TCP/IP 的 DNS.....22
数据库应用.....23
MySQL 数据库和备份与恢复.....23
MySQL 导入导出方法与工具.....33
MySQL 如何对抗解密高手.....45
Oracle 数据库的灾难恢复.....47
保持多台服务器数据的一致性.....54
Oracle 数据库的启动和停止.....58
复杂应用环境 Oracle 性能监控.....60
备份与恢复 Win2003 的 AD 数据库.....75
SQL 导出到 MYSQL 的简单方法.....79
有效提高 WebSphere 数据库性能.....81
Oracle 数据库安全策略分析.....83
Oracle 中如何直接运行 OS 命令.....90
Oracle 密码文件的使用和维护.....98
细说 Sybase 数据库日志.....103
MySQL 数据导入与导出.....106
Sybase 数据库的碎片整理.....113
InformixOnline 数据库维护技巧.....118

域名服务器

BIND8+域名服务器安全增强

为什么要写这篇文章？第一个原因当然就是前段时间出现的 BIND8.2.xTSIG 安全漏洞（还有去年公布的 BIND8.1.x/8.2.xNXT 安全漏洞），直到目前为止，国内也还没有关于 DNS 服务安全配置方面的较为完整的文章（即使是国外也不多见）。

另一个原因是经过调查发现，几乎任何一种 UNIX 家族的操作系统，都使用 BIND 软件作为其 DNS 的唯一实现，比起其它诸如 ftp/http/pop3 等网络服务有各种各样的发行版本，所以一旦被发现有安全问题，则受影响的主机之多也是其它漏洞很难比拟的。所以觉得应该写一份针对 BINDDNS 服务软件的安全配置资料，充分利用 BIND 自身已经实现的保护功能，加强 BIND 安全性，从而能抵御目前已知的 BIND 安全漏洞，并使潜在的安全漏洞所可能对服务器造成的影响尽可能地减少。

一.配置环境：

FreeBSD4.1-RELEASE

BIND8.2.3

二.启动安全选项

named 进程启动选项：

-r：关闭域名服务器的递归查询功能（缺省为打开）。该选项可在配置文件的

options 中使用"recursion"选项覆盖。

-u 和-g：定义域名服务器运行时所使用的 UID 和 GID。

这用于丢弃启动时所需要的 root 特权。

-t：指定当服务器进程处理完命令行参数后所要 chroot()的目录。

三.配置文件中的安全选项

1、假如希望记录安全事件到文件中，但同时还希望保持原有的日志模式，可以添加以下内容：

```
logging{
channelmy_security_channel{
file"my_security_file.log"versions3size20m;
severityinfo;
};
categorysecurity{
my_security_channel;
default_syslog;default_debug;};
}
```

其中 my_security_channel 是用户自定义的 channel 名字，my_security_file.log 是安全事件日志文件，可包含全路径（否则是以 named 进程工作目录为当前目录）。

安全事件日志文件名为 my_security_file.log，保存三个最近的备份（my_security_file.log0、my_security_file.log1、my_security_file.log2），日志文件的最大容量为 20MB（如果达到或超这一数值，直到该文件被再次打开前，将不再记录任何日志消息。缺省（省略）时是没有大小限制。）

2、在 options 节中增加自定义的 BIND 版本信息，

可隐藏 BIND 服务器的真正版本号。

version"Whoknows?";

//version9.9.9;

此时如果通过 DNS 服务查询 BIND 版本号时，返回的信息就是"Whoknows?"。

3、要禁止 DNS 域名递归查询，在 options（或特定的 zone 区域）节中增加：

recursionno;

fetch-glueno;

4、要增加出站查询请求的 ID 值的随机性，在 options 节中增加：

use-id-poolyes;

则服务器将跟踪其出站查询 ID 值以避免出现重复，并增加随机性。注意这将会使服务器多占用超过 128KB 内存。（缺省值为 no）

5、要限制对 DNS 服务器进行域名查询的主机，在 options（或特定的 zone 区域）节中增加：

allow-query{};

address_match_list 是允许进行域名查询的主机 IP 列表，如"1.2.3.4;5.6.7/24;"。

6、要限制对 DNS 服务器进行域名递归查询的主机，在 options（或特定的 zone 区域）节中增加：

allow-recursion{};

address_match_list 是允许进行域名递归查询的主机 IP 列表，如

"1.2.3.4;5.6.7/24;"。

需要全本请访问 www.ertongbook.com

7、要指定允许哪些主机向本 DNS 服务器提交动态 DNS 更新，在 options（或特定的 zone 区域）节中增加：

allow-update{};

address_match_list 是允许向本 DNS 服务器提交动态 DNS 更新的主机 IP 列表，如"1.2.3.4;5.6.7/24;"。

缺省时为拒绝所有主机的提交。

8、要限制对 DNS 服务器进行区域记录传输的主机，在 options（或特定的 zone 区域）节中增加：

allow-transfer{};

address_match_list 是允许进行区域记录传输的主机 IP 列表，如"1.2.3.4;

5.6.7/24;"。

9、要指定不接受哪些服务器的区域记录传输请求，在 options（或特定的 zone 区域）节中增加：

blackhole{};

address_match_list 是不接受区域记录传输请求的主机 IP 列表，如"1.2.3.4;

5.6.7/24;"。

10、在 options 节中还有一些资源限制选项，不同用户可根据实际情况灵活设置，但一定要注意不当的设置会损失 DNS 服务的性能。

coresize;//coredump 的最大值。缺省为 default。

datasize;//服务器所使用的最大数据段内存。缺省为 default。

files;//服务器能同时打开的最大文件数。缺省为 //unlimited（不限制）。

//（注意，并非所有操作系统都支持这一选项。）

max-ixfr-log-size;//（目前版本暂不使用。）限制增量区域

记

录传输时会话日志的大小。

stacksize;//服务器所使用的最大堆栈段内存。缺省为

default。

11、定义 ACL 地址名（即用于上面的）。注意，如果要使用这里定义的列表名，必须先定义，后使用！

例如：

aclintranet{

192.168/16;

};

aclpartner{

!172.16.0.1;

172.16/12;//除 172.168.0.1 外 172.16.0.0/12 网络中其它主机

};

BIND 已内置以下四个 ACL：

all//允许所有主机

none//禁止所有主机

localhost//本机的所有网络接口

localnets//本机所在网络

12、BIND 域名服务器的一个有用功能（慎用！！！）：

控制管理接口 controls 节语法格式：

```
controls{
[inetip_addr
portip_port
allow{;};]
[unixpath_name
permnumber
ownernumber
groupnumber;]
};
```

controls 节提供管理接口。如果使用第一种(inet)，则在指定 IP（接口）和端口上监听，但只允许在 allow 中限定允许与其连接的 IP 地址列表。如果使用第二种(unix)，则产生一个 FIFO 的控制管道，权限、属主和用户组都由其参数限定。

四.通过 TSIG 对区域记录传输进行认证和校验

首先请确保你的 BIND 域名服务器软件已更新到最新版本！

在 BIND8.2+中，能够使用事务签名（TransactionSignatures，即 TSIG！）来对区域记录数据进行验证和校验。它要求在主域名服务器和辅助域名服务器上配置好加密密钥，并通知服务器使用该密钥与其它域名服务器通讯。（注意，TSIG 的使用要求域名服务器必须进行时钟同步！）

A、如果需要用 TSIG 签名来进行安全的 DNS 数据库手工更新，具体操作步骤很简单：

1、使用 BIND 自带的 dnskeygen 工具生成 TSIG 密钥。

#dnskeygen-H128-h-ntsig-key.

则会生成两个文件。’Ktsig-key.+157+00000.key内容如下：

tsig-key.INKEY5133157awwLOtRfpGE+rRKF2+DEiw==

’Kvip-key.+157+00000.private内容如下：

Private-key-format:v1.2Algorithm:157(HMAC)Key:awwLOtRfpGE+rRKF2+DEiw==

注意这些密钥都已经过 BASE64 编码了。将它们放到本地域名服务器的配置文件中。例如：

keytsig-key.{algorithmhmac-md5;secret"awwLOtRfpGE+rRKF2+DEiw==";};

zone"dns.nsfocus.com"{

...

...

allow-update{keytsig-key.;};

}

记住要重启 named 守护进程。

然后将这两个密钥文件复制到客户端系统（或辅助域名服务器），例如为/var/named/tsig 目录。最后运行如下命令即可：

nsupdate-k/var/named/tsig:tsig-key.

B、如果需要对区域记录传输（自动或手工）进行 TSIG 签名，则：

1、用 dnskeygen 生成 TSIG 密钥，方法同上。

2、主域名服务器配置文件的内容（节选）如下：

//定义认证的方法和共享密钥

```
keymaster-slave{
algorithmhmac-md5;
secret"mZiMNOUYQPMNwsDzrX2ENw==";
};
```

//定义辅助域名服务器的一些特性

```
server192.168.8.18{
transfer-formatmany-answers;
keys{master-slave;};
};
```

//区域记录定义

```
zone"nsfocus.com"{
typemaster;
filedb.nsfocus.com;
allow-transfer{192.168.8.18;};
};
```

3、辅助域名服务器配置文件的内容（节选）如下：

//定义认证的方法和共享密钥

```
keymaster-slave{
algorithmhmac-md5;
secret"mZiMNOUYQPMNwsDzrX2ENw==";
};
```

//定义与主域名服务器通讯时的一些特性

```
server192.168.8.19{
transfer-formatmany-answers;
keys{master-slave;};
};
```

//区域记录定义

```
zone"nsfocus.com"{
```

```
typeslave;
file"bak.db.nsfocus.com";
masters{192.168.8.19;};
allow-transfer{none;};
};
```

五.实施 DNSSec 功能

说实在的，我一直在考虑需不需要在目前的版本中实施 DNSSec **功能。因为虽然** ISC **早已在** BIND8.1.x **版本后增加了** DNSSec **的实现，但实际的应用却不常见，而且去年公布的** NXT **远程安全漏洞和** DNSSec **有关（实际上** NXT **就属于** DNSSec **实现的功能之一）。**

最后我决定在本文不讨论如何实施 DNSSec **安全功能。但不可否认，**DNSSec **确实是一项很好的安全技术，它通过加密** DNS **数据来提高了** DNS **服务的安全性。主加密密钥用于对第一级域名的加密密钥进行加密签字，第一级域名(**.com,.cn **等)密钥用于对自身数据及其子域名密钥进行加密签名，以此类推。例如，**nsfocus.com **的域名服务器由**.com **域密钥签名，**nsfocus.com **域密钥则用于对** www.nsfocus.com **域名进行加密签名。**

六.实现 BIND 的 chroot

（以 FreeBSD **系统平台为例）**

步骤一：BIND-8 **的最新源代码版本获取和安装**

请到 ISCFTP **站点下载** BIND **的最新版本。**

BIND8：http://www.isc.org/products/BIND/bind8.html

BIND9：http://www.isc.org/products/BIND/bind9.html

需要全本请访问 www.ertongbook.com

步骤二：构造静态(static)**的** named **和** named-xfer **二进制文件**

在编译和安装后，你需要构造可执行文件的静态链接版本。只要对%BIND%/src/port/freebsd **目录下的** Makefile.set **文件稍加修改后即可。**

修改文件内容：

’CDEBUG=-O2-g’

替换为：

’CDEBUG=-O2-static’

切换到 BIND **的源代码路径，执行**"makeclean"**和**"make"**命令。在下面的步骤中将会把这些文件复制到** chroot()**目录下。**

#cd/tmp/bind/src

#makeclean;make

本步骤构造的静态链接执行文件在运行时无需装载动态链接库。在 chroot()**环境中，这种“独立”可执行文件可避免出现缺少链接库文件问题。它在** chroot()**环境中无需任何静态链接库，可使服务配置简单化。其它所有的网络守护进程也可以编译和使用这种静态链接版本。**

步骤三：构造 BIND **目录**

为 chroot()**环境构造** BIND **目录。这个目录将在** chroot()**环境中被** BIND **当作系统根目录。在这里我使用**/chroot/bind **作为** chroot **后的根目录。**

#cd/chroot/bind

#mkdir/chroot

#mkdir/chroot/dev

#mkdir/chroot/etc

#mkdir/chroot/etc/namedb

#mkdir/chroot/usr

#mkdir/chroot/usr/sbin

#mkdir/chroot/var

#mkdir/chroot/var/run

需要复制以下文件到其下的相应子目录中，和进行一些必要的处理：

#cp/etc/namedb/named.conf/chroot/bind/etc/

#cp/etc/localtime/chroot/bind/etc/

#grepbind/etc/group>/chroot/bind/etc/group

#cp-R/etc/namedb//chroot/bind/etc/namedb/

#mknod/chroot/bind/dev/nullc22

#chmod666/chroot/bin/dev/null

#cp/tmp/bind/src/bin/named/named/chroot/bind/usr/sbin/

#cp/tmp/bind/src/bin/named-xfer/named-xfer/chroot/bind/

步骤四：添加 bind 用户和组（如果没有的话。如果已经有 bind 或 named 之类的用户和组，请跳过本步骤。）

在/etc/passwd 和/etc/group 文件中添加 bind 用户和组。它们是 DNS 服务器运行时的 UID/GID。

此时，你可以到 chroot 环境中执行"chown-Rbind.bind/chroot/bind/etc/namedb"命令。这样当你向系统发送中断信号(kill-INT)时，named 进程能够保存服务器缓存和统计信息。如果该目录为 root 所有则 named 进程无法将输出写到目录中，但不会影响 named 服务器功能。另一个选择是仅改变目录权限（使 named 用户具有写权限），而属主仍然是 root。这种方法也是可行的，但必须小心

设置，确保其它用户不会修改 named 记录！

重要警告

不要用一个已存在的 UID/GID（如"nobody"）运行 named。记住，以 chroot 环境中使用任何已存在的 UID/GID 都可能会影响到服务的安全性。必须养成在 chroot 环境中为每一个守护进程提供独立的 UID/GID 的习惯。

步骤五：其它必要调整

如果在 named.conf 中还指定了另外的目录和文件，也要相应地在 chroot()环境中（在本例中即/chroot/bind/目录）进行设置。

步骤六：调试

1、终止系统中原有的 syslogd 和 named 守护进程。

#killallsyslogdnamed

2、用适当的参数重新启动 syslogd 守护进程。

#syslogd-s-p/chroot/bind/var/run/log

3、用适当参数重新启动 named 守护进程。

#/chroot/bind/named-ubind-gbind-t/chroot/bind

4、检查 syslogd/named 守护进程、监听端口是否正常，和/var/log/messages 文件中 named 进程启动时是否正常。

#psauwx|grepsyslogd

root58960.01.7896508??Ss9:44PM0:00.10syslogd-s-p

/chroot/bind/var/run/log

#psauwx|grepnamed

bind59410.04.916521444??Is9:52PM0:00.01

/chroot/bind/usr/sbin/named-ubind-gbind-t/chroot/bind

#netstat-an|grep53

tcp400127.0.0.1.53*.*LISTEN

tcp400192.168.8.19.53*.*LISTEN

udp400127.0.0.1.53*.*

udp400192.168.8.19.53*.*

步骤七：修改系统启动脚本

对于 FreeBSD **系统，在**/etc/rc.conf **文件中增加如下内容即可：**

syslogd_enable="YES"

#**如果希望禁止向外发送日志，将**-s **换成**-ss。

syslogd_flags="-s-p/chroot/bind/var/run/log"

named_enable="YES"

named_program="/chroot/bind/usr/sbin/named"

named_flags="-ubind-gbind-t/chroot/bind"

注：如果在其它系统平台，如 OpenBSD、Linux、Solaris，**则可能会稍有不同。主要是不同平台上的** syslog **实现不尽相同。例如对于** OpenBSD **和** Linux **系统，打开日志别名** socket **的命令是**"syslogd-a/chroot/bind/var/run/log"，**而** Solaris **的** syslogd **守护进程则不支持别名。因此** Solaris **系统平台上的** chroot **需要通过另外的方法实现，关于具体的实现过程我会在另外的文章中说明。**

七.结束语

安全增强配置的文章写完了，但并不是说只要你按本文提到的方法和技术实施就能万无一失，高枕无忧了。其实以上设置对 NXT **和** TSIG **远程漏洞攻击并不没太多的防御作用，充其量只不过是要编写更多的** shellcode **代码来突破** chroot **环境的限制。即使用** allow-query **等极其严格地限制查询客户端（实际上在互联网上并不现实），基于** UDP **协议的** TSIG **攻击技术也只需构造伪造** IP **地**

址的数据包即可绕过其限制。

所以，在对 BIND（还有其它应用服务）进行安全增强配置的基础上，安全管理员仍然需要密切关注最新的安全公告、安全补丁和安全技术，经常与专业的计算机安全专家交流知识和经验，再辅以必要的安全产品和安全服务，才能更充分地保护好自己的网络和计算机用户，抵御各种恶意攻击。

设置过多 DNS 服务器地址的问题

设置过多 DNS 服务器地址将会导致服务器不支持动态 DNS 错误

很多人可能都遇到过这个问题:将域内的 DNS 安装成了与 ActiveDirectory 集成模式并启动了动态更新,但每一次 DC 启动后总在事件查看器的系统日志中提示如下错误:

事件类型:警告

事件来源:NETLOGON

事件种类:无

事件 ID:5773

日期:2001-6-18

事件:11:48:42

用户:N/A

计算机:MYSERVER

描述:

该 DC 的 DNS 服务器不支持动态 DNS。将文件'%SystemRoot%\System32\Config\netlogon.dns'中的 DNS 记录添加到伺服那个文件中引用的域的 DNS 服务器。

数据:

0000:2c230000,#..

明明 DNS 服务器已经启动了动态更新了的,为什么系统还提示"该 DC 的 DNS 服务器不支持动态 DNS"呢?

出现这个错误的原因是我们在 DC 上设置了多个 DNS 地址,而 DC 启动时会依次向每一个 DNS 服务器进行注册.如果这些 DNS 中某一台或几台不支持动态更新,那么 DC 就会提示上述错误.

假设我们设置的 DNS 如下:

10.10.10.3(支持动态更新)

10.10.9.3(不支持动态更新)

10.10.8.3(不支持动态更新)

使用诊断命令行工具 netdiag 进行诊断,会发现:

DNStest.....:Passed

PASS-AlltheDNSentriesforDCareregisteredonDNSserver’10.10.10.3’.

[WARNING]TheDNSentriesforthisDCarenotregisteredcorrectlyonDNSserver’10.10.9.3’.Pleasewaitfor30minutesforDNSserverreplication.

[WARNING]TheDNSentriesforthisDCarenotregisteredcorrectlyonDNSserver’10.10.8.3’.Pleasewaitfor30minutesforDNSserverreplication.

纠正这个错误的办法比较简单,将多余的 DNS 条目删除即可。

如何在 Linux 服务器上设置 DNS

背景:

需要全本请访问 www.ertongbook.com