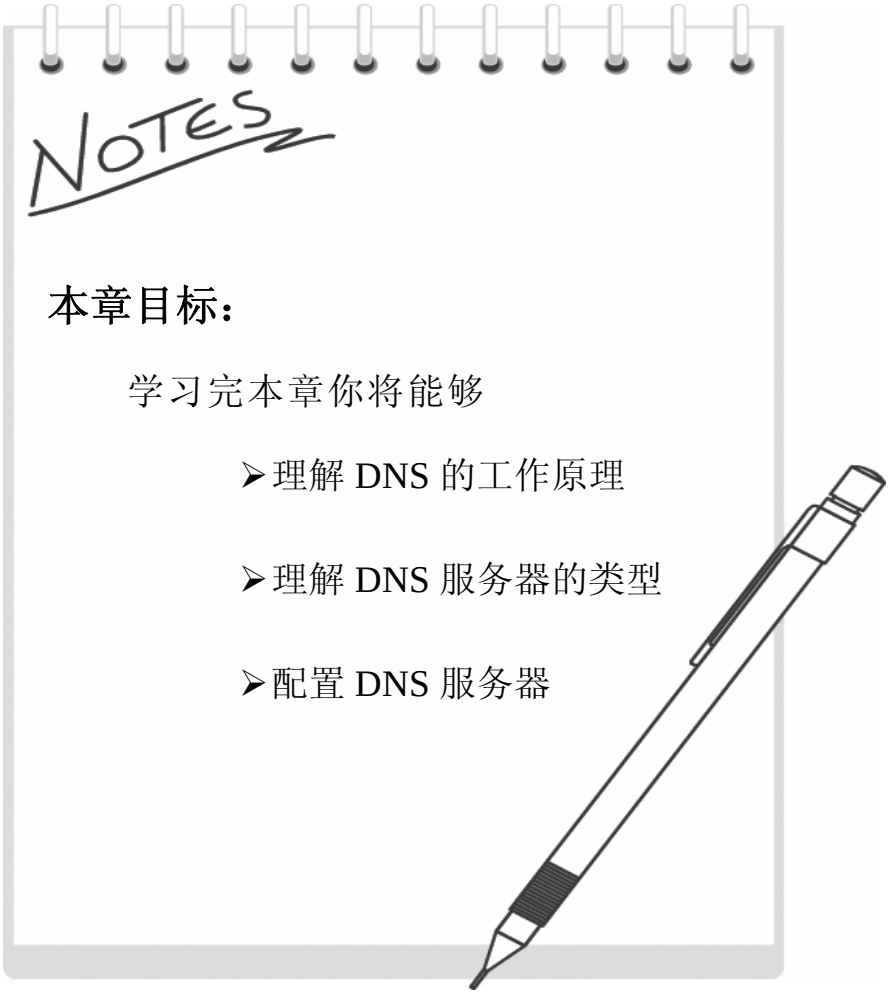
第三章　DNS 服务器的配置

网络发展到今天，仅仅用 IP 地址去标一台网络上的计算机是很难以让人接受的，人们不可能记住众多的 IP 地址，于是便有了域名系统，它可以利用形象的名称来标识一台网络上的计算机，这就是 DNS，本章将详细介绍有关 DNS 的一些知识。



本章目标：

学习完本章你将能够

- 理解 DNS 的工作原理
- 理解 DNS 服务器的类型
- 配置 DNS 服务器

1. DNS 简介

连接 TCP/IP 的每个网络接口用一个唯一的 32 位的 IP 地址来标识，但由于数字比较复杂、难以记忆而且没有形象性。因而，人们发明了用域名系统，在这种情况下，我们可以使用易于理解和较为形象的名称来标识一台计算机。在大多数情况下，数字地址和域名地址可以交替使用；但无论用数字地址还是域名进行网络应用时，网络总是以 IP 地址为基础来进行的。在网络进行连接前，系统必须将域名转换成 IP 地址。这就是 DNS 服务器的作用。

将域名转换成 IP 地址有两种常用的方法。一种较古老的方法是从一个称为“主机表”的文件中查找主机名；另一种是使用一个称为“域名服务（DNS）”的分布式数据库系统，将名字转换成 IP 地址。

主机表是一个简单的文本文件，可以使 IP 地址与主机名相关联。在 Linux 操作系统中，主机表文件为/etc/hosts，该文件的每一列表项包含一个 IP 地址和用空格隔开的与该地址关联的主机名。

- DNS 的优点：

DNS 的扩充性好。它并不单独依靠一个主机表，而是依靠一个分布式数据库系统，不会因为数据库的增长而陷入困境。

DNS 可确保在必要时将新主机的信息传播到网络中的其他部分。不仅能自动地传播信息，而且可以只传播所需的信息。

DNS 的工作原理为：

如果一台 DNS 服务器接收到一个要求获取有关主机信息的请求，它就将该请求发送给一台管理服务器。管理服务器是负责保持查询域的精确信息的服务器。当它响应该请求时，本地服务器就将回答信息保存在高速缓存中。当本地服务器再接收到要求获取关主机信息的请求时，它本身就回答这个请求。

2. 递归查询与迭代查询

DNS 进行查询时有两种查询方法：递归与迭代。

2.1 递归查询的工作方式

递归查询是最常见的查询方式，域名服务器将代替提出请求的客户机（下级 DNS 服务器）进行域名查询，若域名服务器不能直接回答，则域名服务器会在域各树中的各分支的上下进行递归查询，最终将返回查询结果给客户机，在域名服务器查询期间，客户机将完全处于等待状态。

图 3-1 是一个 DNS 查询的示例。

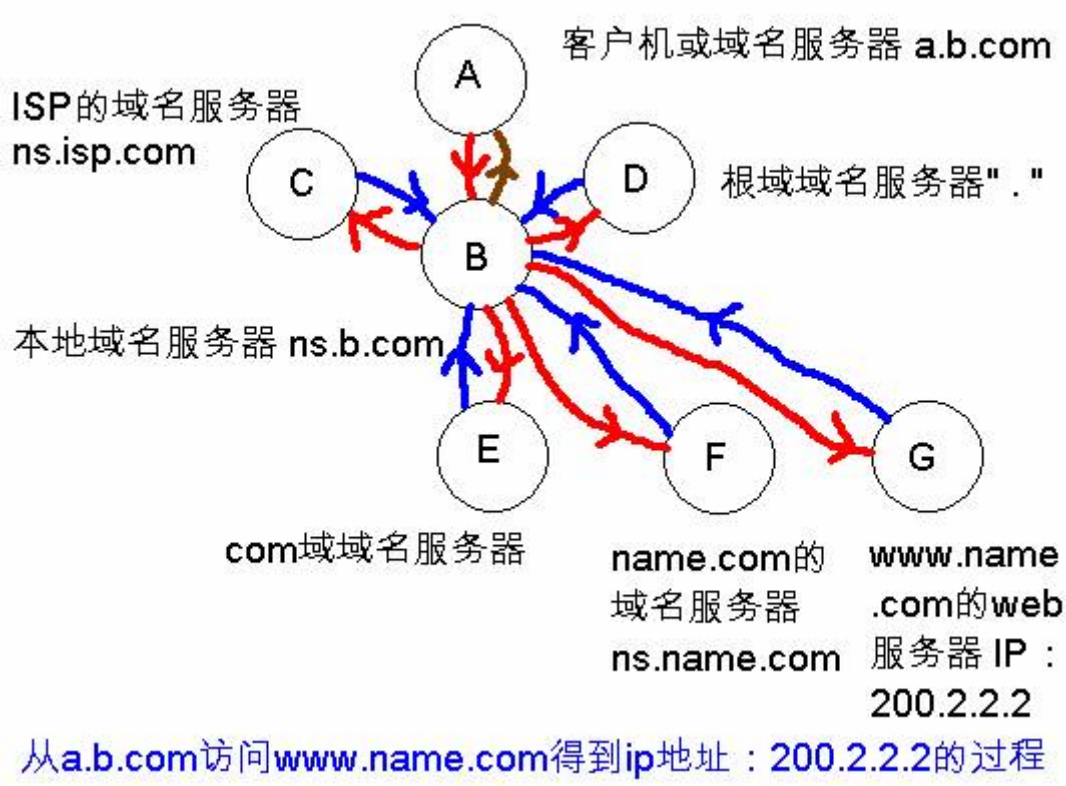


图 3-1

示例说明：A 向 B 发送递归查询请求，B 向 C 发送迭代查询请求（下一节将介绍迭代查询），得到 C 给出的提示后，B 向 D 发送迭代查询请求，得到 D 给出的提示后，B 向 E 发出迭代请求，得到 E 给出的提示后，B 向 F 发出迭代查询请求，得到 F 给出的提示后，B 得到了 F 返回 G 的 IP 地址，B 向 A 返回 G 的 IP 地址，整个查询结束。

从这个例子可以看出，A 向 B 发出查询（递归）后，等待 B 给出最终答案；而 B 向其它服务器查询（迭代）时，是在其它服务器的指引下有自己去寻找答案。

2.2 迭代查询的工作方式

迭代查询又称重指引，当服务器使用迭代查询时能够使其他服务器返回一个最佳的查询点提示或主机地址，若此最佳的查询点中包含需要查询的主机地址，则返

回主机地址信息，若此时服务器不能够直接查询到主机地址，则是按照提示的指引依次查询，直到服务器给出的提示中包含所需要查询的主机地址为止，一般的，每次指引都会更靠近根服务器（向上），查寻到根域名服务器后，则会再次根据提示向下查找。从上节的图中可以知道，B 访问 C、D、E、F、G，都是迭代查询，首先 B 访问 C，得到了提示访问 D 的提示信息后，开始访问 D，这时因为是迭代查询，D 又返回给 B 提示信息，告诉 B 应该访问 E，依次类推。

说明：假设你要寻找一家你从未去过的公司，你会有 2 种解决方案，1 是找一个人替你问路，那可能是你的助手，2 是自己问路，每走过一个路口，就问一个人，这就好比递归查询和迭代查询，递归查询在这里代表你的第 1 种解决方案，而迭代则是第 2 种解决方案。

一般的，你的企业内部如果有超过 300 台机器，你就应该在你的部署计划中建立多个 DNS 服务器了。根据活动目录或者物理位置将多个 DNS 平均分布。而根域名服务器总应该使用迭代查询，而不应该使用递归查询。同时，为了减轻客户机的负担，所有的下级域名服务器就都应该使用递归查询与迭代查询的混合模式。若你的企业整合了活动目录及有分公司分布在全球，通过使用多层的域名服务器，可以得到最佳的性价比！

3. DNS 服务器的类型

DNS 服务器分为以下几个类型：

1.“Cache-Only”DNS 服务器：

一个 DNS 服务器也不可能拥有国际网络上所有的主机信息，因此它提供一个转送方式，将自己的 DNS 服务器无法处理的查询要求转送至上一层的 DNS 服务器上查询，然后将所得到的查询结果传送给提出查询要求的主机。而“Cache-Only”DNS 就是该 DNS 服务器的主机里除了自己的信息就没有其它的了，它将所有的查询要求都转送至其他 DNS 服务器上。

2.“Primary”DNS 服务器：

一个功能完备的 DNS 服务器，管理一个或数个“Domain”（域）的计算机信息。而这些相关的资料都依照某种格式储存于服务器的档案目录中，然后会在服务器启动时将这些资料载入系统。

3.“Secondary”DNS 服务器：

“Secondary”DNS 服务器也是一个功能完备的 DNS 服务器，所不同的是，它的主机资料并非完全储存于服务器所在的档案目录中，而由某个“Primary DNS”来提供。

4. BIND 软件介绍

BIND (Berkeley Internet Name Domain)是 Domain Name System (DNS) 协议的一个实现，提供了 DNS 主要功能的开放实现，包括：

域名服务器 (named)

DNS 解析库函数

DNS 服务器运行调试所用的工具

是一款开放源码的 DNS 服务器软件，由美国加州大学 Berkeley 分校开发和维护的。按照 ISC 的调查报告，BIND 是世界上使用最多最广泛的域名服务系统。不论你的邮件服务器，WEB 服务器或者其他的 services 如何的安全可靠，DNS 的故障会给你带来用户根本无法访问这些服务。

BIND，也是我们常说的 named，由于多数网络应用程序使用其功能，所以在很多 BIND 的弱点及时被发现。主要分为三个版本：

v4

1998 年多数 UNIX 捆绑的是 BIND4，已经被多数厂商抛弃了，除了 OpenBSD 还在使用。OpenBSD 核心人为 BIND8 过于复杂和不安全，所以继续使用 BIND4。这样一来 BIND8/9 的很多优点都不包括在 v4 中。

v8

就是如今使用最多最广的版本。

v9

最新版本的 BIND，全部重新写过，免费（但是由商业公司资助），也添加了许多新的功能（但是安全上也可能有更多的问题）。BIND9 在 2000 年十月份推出，现在稳定版本是 9.3.2。RedHat Enterprise Linux AS 4 Update 2 中所带的是 9.2.4

BIND 软件使用文本数据库来存储数据，其数据库文件中有很多参数，在此先列出下面这些参数的含义：

- ttl

为了迫使解析器在一段时间后丢去信息，每条记录都有一个相应的“生存期”，简称 ttl。它的单位是秒，系统的缺省值为 86400。

- SOA

描述了一个授权区域，它表示了该区域的授权信息。

- Serial

表示该区域文件的版本号数。每当区域文件中的数据改变时，这个数值将要增加。通常用修改的时间来表示该版本号。例如 2002110501 表示 2002 年 11 月 5 日对该文件的第一次修改。从服务器在一定时间以后就请求主服务器的 SOA 记录，并将该序列号值与缓存中的 SOA 记录的序列号相比较，如果数值改变了，从服务器就从主服务器将整个区域的数据传输过来。

- Refresh

指定了从服务器将要检查主服务器的 SOA 记录的时间间隔，单位为秒。

- retry

它指定了从服务器的一个请求或一个区域刷新失败后，从服务器重新与主服务器联系的时间间隔，单位是秒。

- expire

在指定的时间内，如果从服务器还不能联系到主服务器（主服务器宕机），从服务器将丢去所有的区域数据。

- minimum

如果没有明确指定 ttl 的值，则 minimum 就是缺省的“生存期”。

- A

主机记录，用于将一个主机名与一个 IP 地址相关联（相对应）。

- NS

用来指定某个区域的主名字服务器和所有从名字服务器。一条 NS 记录指向一个给定区域的主名字服务器，以及包含该服务器主机名的资源记录。

- CNAME

用来关联一个主机名的别名和它的规范主机名，即该记录为规范主机名提供了一个别名。

- MX

指明了该区域中的邮件交换器（邮件服务器）和优先级。

5. DNS 服务器的配置

在 Linux 系统中，要配置 DNS 服务器首先安装软件。安装软件可以从源代码安装，也可以从 RPM 来安装。在 RedHat Linux 系统中，如果使用 RPM 包，除了需要安装 bind 软件包外，还需要安装示例配置文件软件包 caching-nameserver。

5.1 配置文件 named.conf

在 RedHat Linux 系统中，安装完 caching-nameserver 软件包后，DNS 服务器的缺省配置文件/etc/named.conf。文件中以“#”、“//”、“;”开头的行以及使用“/*　*/”扩起来的内容都是注释。下面分别对有效的部分进行简单说明：

```
options {
        directory "/var/named";
        dump-file "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
};
```

options：是全局服务器的配置选项，即在 options 中指定的参数，对配置中的任何域都有效，例如：在服务器上要配置两个域 koorka.com 和 clusting.com，那么在 option 中指定的选项对两个域都生效。

directory：指定 named 从 /var/named 目录下读取 DNS 数据文件，这个目录用户可自行指定并创建，指定后所有的 DNS 数据文件都存放在此目录下。

dump-file：当执行 rndc dumpdb，命令时，将 DNS 服务器的缓存数据转存到指定的文件中

statistics-file：指定 named 服务的统计文件。当执行 rndc stats 命令时，会将内存中的统计信息追加到该文件中。

```
controls {
        inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};
```

controls：定义使用 rndc（remote name daemon control）工具对 DNS 服务器进行管理时的通道。inet 指定要监听本机的哪个 IP 地址（如果是 127.0.0.1，表示只能通过本机对该 DNS 服务器进行管理）；allow 指定允许哪些主机使用 rndc 命令来管理；keys 表示在通讯时使用的加密字符串和算法。例如本机的对外 IP 为 202.127.124.110，允许从 202.127.124.109 来管理 DNS 服务器，则可以定义：

```
inet 202.127.124.110 allow{202.127.124.109;} keys{rndckey;};
```

```
zone "." IN {
        type hint;
        file "named.ca";
};
```

每一个 zone 就是定义一个区域。该段指定 named 从 named.ca 文件中获得 Internet 的顶层“根”服务器地址 。

```
zone "localhost" IN {
        type master;
        file "localhost.zone";
        allow-update { none; };
};
```

定义本机的解析。

```
zone "0.0.127.in-addr.arpa" IN {
        type master;
        file "named.local";
        allow-update { none; };
};
```

定义回环地址（loopback）的反向解析（通过 IP 解析出主机名称）。

```
zone "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa" IN {
        type master;
        file "named.ip6.local";
        allow-update { none; };
};
```

定义 ipv6 的回环地址的反向解析。

```
zone "255.in-addr.arpa" IN {
        type master;
        file "named.broadcast";
        allow-update { none; };
};
```

广播地址的反向解析。

```
include "/etc/rndc.key";
```

include 表示包含某个文件，named 会将该文件的内容放在 include 语句所在的位置。

以上是对 named.conf 文件的简要说明，下面通过几个配置实例来理解 DNS 服务器的配置。

5.2 纯缓存服务器的配置

北京千喜公司现用 ADSL 上网，目前已能上网，但是速度稍慢，为了提高访问互联网的速度，公司网络管理员决定在 Linux 系统下做一台 caching nameserver（域名缓存服务器）。试为该公司安装该服务器。

分析：该服务器的功能就是暂存上次访问过的域名，下次需要解析时，直接从缓存内读取结果，所以不用建立其他的区域。假设我们将服务器的 IP 设置为 192.168.0.197。

具体操作步骤如下：

1．获取并安装 DNS 服务器软件。

Linux/UNIX 系统内常用的 DNS 服务器软件为 Bind，RedHat Enterprise Linux 版本为 bind-9.2.4-2，也可以到 http://www.isc.org/products/BIND/ 或 ftp://ftp.isc.org/isc/bind9/获得新版本。

如果是下载的源代码（以 bind-9.3.0 为例），则执行以下操作：

（1）进入源代码所在的目录。例如：/tmp。

（2）tar –zxvf bind-9.3.0.tar.gz

（2）cd bind-9.3.0

（3）./configure

make

make install

如果是从源代码安装，默认的 named.conf 文件应该在安装目录的 etc 下，例如如果安装目录是/usr/local/named，则配置文件应该是/usr/local/named/etc/named.conf，也可以存在/etc/named.conf，在启动服务时指定：

```
/usr/local/named/sbin/named –c /etc/named.conf –u named
```

2．修改配置文件/etc/named.conf。

由于是域名缓存服务器，原则上来说，不用修改该配置文件的其它字段，但是必须确定该配置文件内有如下字段：

```
zone "." IN {
        type hint;
        file "named.ca";
};
```

但是，为了进一步减小网络流量，可以将设置一个转发，即在 options 段中，添加下面的内容：

```
forwarders {
        202.99.8.1;   //主域名服务器的 IP 地址
 };
```

所谓的 forwarder，就是当某一台 NS 主机遇到非本机负责的 zone (slave zone 也属于本机负责的范围) 查询请求的时候，将不直接向 "." zone 查询而把请求转交给指定的 forwarder (一台或多台) 主机代为查询。

我们知道，当 DNS 服务器接到客户端主机的查询请求时，首先会检查这个查询是否属于本机管辖，否则将转向 "." zone 再逐级的查询下去，最后再把查询结果告诉客户端。

在这个过程之中，DNS 服务器还会将查询到的结果存放到缓存中。只要缓存中的 TTL 没过期，在下次遇到同样查询的时候，就可以直接将结果响应给客户端，而无需再重复上次的查询流程。

如果 DNS 服务器上指定了 forwarder，那这个 DNS 发现缓存中没有记录时，将不向 "." 查询，而是向 forwarder 送出同样的请求（转发），然后等待查询结果，即把逐级往下查询这个耗费精力的动作，交给 forwarder 负责。但无论这个结果是自己直接查询得来的，还是 forwarder 送回来的，DNS 服务器都会保存一份数据在缓存中。

这样，以后的相同查询就快多了，这对于 DNS 所服务的 客户端而言查询效率会提高很多。

3．启动 dns 服务器。

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

为了每次开机都启动 DNS 服务，需要执行 ntsysv 命令，将 named 选中，如图 3-2 所示。

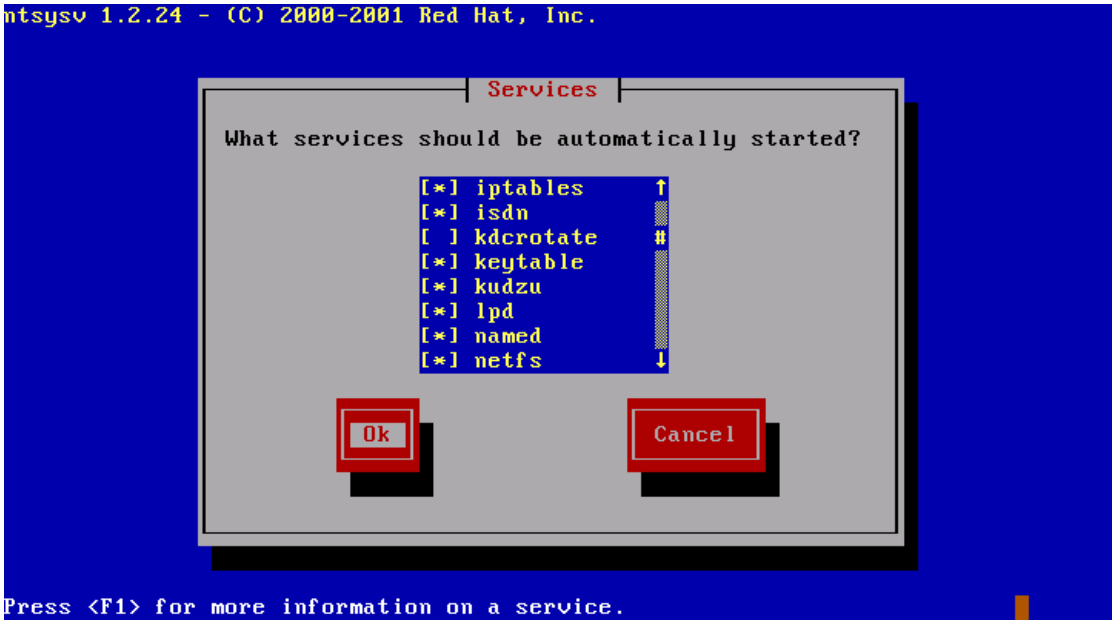


图 3-2

当执行完/etc/rc.d/init.d/named start 后，执行 tail –f /var/log/messages 命令，应出现类似如图 3-3 所示的画面，表示服务启动正常。

```
[root@www root]# /etc/rc.d/init.d/named start
Starting named:                                            [  OK  ]
[root@www root]# tail -f /var/log/messages
May 22 07:09:45 www named[1237]: starting BIND 9.1.3 -u named
May 22 07:09:45 www named[1237]: using 1 CPU
May 22 07:09:45 www named: named startup succeeded
May 22 07:09:45 www named[1240]: loading configuration from '/etc/named.conf'
May 22 07:09:45 www named[1240]: the default for the 'auth-nxdomain' option is n
ow 'no'
May 22 07:09:45 www named[1240]: no IPv6 interfaces found
May 22 07:09:45 www named[1240]: listening on IPv4 interface lo, 127.0.0.1#53
May 22 07:09:45 www named[1240]: listening on IPv4 interface eth0, 192.168.0.197
#53
May 22 07:09:45 www named[1240]: command channel listening on 127.0.0.1#953
May 22 07:09:45 www named[1240]: running
```

图 3-3

4．测试服务器。

（1）编辑/etc/resolve.conf，将其内容更改为（即指定 DNS 通过本机解析）：

nameserver 127.0.0.1

（2）执行 dig –x 127.0.0.1（验证本机解析是否正常）命令后，应出现如图 3-4 所示的画面。

```
[root@www root]# dig -x 127.0.0.1
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 63238
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;1.0.0.127.in-addr.arpa.                IN      PTR

;; ANSWER SECTION:
1.0.0.127.in-addr.arpa. 86400   IN      PTR     localhost.

;; AUTHORITY SECTION:
0.0.127.in-addr.arpa.   86400   IN      NS      localhost.

;; ADDITIONAL SECTION:
localhost.              86400   IN      A       127.0.0.1

;; Query time: 52 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Wed May 22 07:20:30 2002
;; MSG SIZE  rcvd: 93

[root@www root]# _
```

图 3-4

（3）执行 dig www.google.com（外部域名解析是否正常）命令，应出现如图 3-5 所示的画面。

```
; <<>> DiG 9.1.3 <<>> www.google.com
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21141
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 0

;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         300     IN      A       216.239.51.100

;; AUTHORITY SECTION:
google.com.             345600  IN      NS      ns4.google.com.
google.com.             345600  IN      NS      ns1.google.com.
google.com.             345600  IN      NS      ns2.google.com.
google.com.             345600  IN      NS      ns3.google.com.

;; Query time: 1944 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Wed May 22 09:50:20 2002
;; MSG SIZE  rcvd: 120

[root@www etc]# _
```

图 3-5

5.客户端设置。只需将客户端的首选 DNS 服务器设置为 192.168.0.197 即可。

5.3 主域名服务器

长城医药公司申请了域名 greatwall.com，现在公司的 DNS 服务器地址为：202.119.98.1，域名为 ns.greatwall.com，web 服务器地址为：202.119.98.10，域名为 www.greatwall.com，FTP 服务器地址为：202.119.98.100，域名为 ftp.greatwall.com

试为该公司安装一台 DNS 服务器。

分析：因为没有特殊要求，这是最简单的 DNS 服务器。只需要设置本地区域，并且能够起到缓存作用即可，而且内部通过此服务器也能解析外部的 DNS 地址。

具体操作步骤如下：

1．获得并安装 DNS 服务器软件（参看实验一的步骤 1）。

2．修改配置文件，即 vi /etc/named.conf。

（1）定义正解区域，在 named.conf 文件内插入以下内容：

```
zone "greatwall.com"{
        type master;
        file "dns.greatwall.com";
        allow-update { none; };
};
```

type 指名该区域为主服务器；

file 指定区域文件名称。

（2）定义反解区域，在 named.conf 文件内插入以下内容：

```
zone "98.119.202.in-addr.arpa"{
        type master;
        file "202.119.98";
        allow-update { none; };
};
```

反向区域可以不用配置。

3．用/etc/hosts 文件解析服务器域名，在/etc/hosts 文件内插入以下内容：

```
202.119.98.1      ns        ns.greatwall.com
```

4．创建 DNS 数据库文件。

（1）创建正解数据库文件/var/named/dns.greatwall.com，其内容如下：

```
$TTL86400
@  IN  SOA     ns.greatwall.com.      root.ns.greatwall.com（
                        199802151; serial
                        28800; refresh
                        14400; retry
                        3600000; expire
                        86400）; minimum，seconds;
        NS  ns.greatwall.com.
ns      A   202.119.98.1
www     A   202.119.98.10
```

```
ftp             A          202.119.98.100
……
```

（2）创建反解数据库文件/var/named/202.119.98，其内容如下：

```
$TTL86400
@       IN  SOA     ns.greatwall.com.   root.ns.greatwall.com（
                    199802151; serial
                    28800; refresh
                    14400; retry
                    3600000; expire
                    86400）; minimum
        IN  NS      ns.greatwall.com.
1       IN  PTR     ns.greatwall.com.
10      IN  PTR     www.greatwall.com.
100     IN  PTR     ftp.greatwall.com.
……
```

5．启动 DNS 服务，/etc/rc.d/init.d/named start（参看实验一）。

6．测试 DNS 服务器。

（1）设置/etc/resolv.conf，即将某台客户机的 DNS 设置为 202.119.98.1（或者将 DNS 服务器设置为 202.119.98.1，此时服务器也当客户机）：

```
nameserver 202.119.98.1
```

（2）执行 dig –x 202.119.98.1 命令，测试服务器是否正常，如图 3-6 所示。

```
[root@www named]# dig -x 202.119.98.1
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39602
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;1.98.119.202.in-addr.arpa.      IN      PTR

;; ANSWER SECTION:
1.98.119.202.in-addr.arpa. 86400 IN     PTR     ns.greatwall.com.

;; AUTHORITY SECTION:
98.119.202.in-addr.arpa. 86400  IN      NS      ns.greatwall.com.

;; ADDITIONAL SECTION:
ns.greatwall.com.       86400   IN      A       202.119.98.1

;; Query time: 19 msec
;; SERVER: 202.119.98.1#53(202.119.98.1)
;; WHEN: Thu May 23 07:54:51 2002
;; MSG SIZE  rcvd: 103

[root@www named]#
```

图 3-6

（3）执行 nslookup www.greatwall.com 命令，解析内部域名地址，如图 3-7 所示。

```
[root@www named]# nslookup www.greatwall.com
Note:  nslookup is deprecated and may be removed from future releases.
Consider using the `dig' or `host' programs instead.  Run nslookup with
the `-sil[ent]' option to prevent this message from appearing.
Server:         202.119.98.1
Address:        202.119.98.1#53

Name:   www.greatwall.com
Address: 202.119.98.10

[root@www named]# _
```

图 3-7

（4）执行 dig greatwall.com axfr 命令，查看 greatwall.com 域的全部记录，如图 3-8 所示。

```
[root@www named]# dig greatwall.com axfr

; <<>> DiG 9.1.3 <<>> greatwall.com axfr
;; global options:  printcmd
greatwall.com.          86400   IN      SOA     ns.greatwall.com. root.ns.greatw
all.com. 1997022700 28800 14400 3600000 86400
greatwall.com.          86400   IN      NS      ns.greatwall.com.
ftp.greatwall.com.      86400   IN      A       202.119.98.100
ns.greatwall.com.       86400   IN      A       202.119.98.1
www.greatwall.com.      86400   IN      A       202.119.98.10
greatwall.com.          86400   IN      SOA     ns.greatwall.com. root.ns.greatw
all.com. 1997022700 28800 14400 3600000 86400
;; Query time: 26 msec
;; SERVER: 202.119.98.1#53(202.119.98.1)
;; WHEN: Thu May 23 08:00:12 2002
;; XFR size: 7 records

[root@www named]# _
```

图 3-8

（5）执行 nslookup www.google.com 命令，解析外部域名：

```
C:\>nslookup www.google.com
Server:  ns.greatwall.com
Address:  202.119.98.1

Non-authoritative answer:
Name:    www.google.com
Address:  216.239.33.101
```

图 3-9

到此为止，服务器已经安装完成并且能够正常运行。由于在 named.conf 内定义的有“.”（根域），所以本服务器也能解析外部域名，它同时也是一台缓存服务器。

5.4 子域与备份服务器

霍普公司总部有一域名服务器，各地分公司也均有自己的域名服务器，现要为该公司建立域名服务系统，且总部和各地分公司的 web 服务器与 FTP 服务器用同一台主机，公司有一台邮件服务器和一台备份邮件服务器，由于公司总部的业务不能

中断，要求建立一台备份域名服务器。公司的部分资料如下：

域　名	对应的 IP	角　色	部　门
ns.hope.com	140.135.10.1	主域名服务器	总公司
nsback.hope.com	140.135.10.2	备份域名服务器	总公司
mail.hope.com	140.135.10.3	主邮件服务器	总公司
mailback.hope.com	140.135.10.4	备份邮件服务器	总公司
www.hope.com	140.135.10.5	Web 服务器	总公司
ftp.hope.com	140.135.10.5	FTP 服务器	总公司
download.hope.com	ftp.hope.com	FTP 服务器别名	总公司
ns.chengdu.hope.com	140.135.20.1	分公司域名服务器	成都分公司
www.chengdu.hope.com	140.135.20.2	分公司 Web 服务器	成都分公司
ftp.chengdu.hope.com	140.135.20.2	分公司 FTP 服务器	成都分公司
ns.shanghai.hope.com	140.135.30.1	分公司域名服务器	上海分公司
ns.changchun.hope.com	140.135.40.1	分公司域名服务器	长春分公司
ns.tianjin.hope.com	140.135.50.1	分公司域名服务器	天津分公司
……	……	……	……

分析：由于总公司与各分公司均有自己的域名服务器，要让总公司能够解析各分公司的域名，各分公司的 DNS 服务器应为总公司的子域。要让各分公司能够解析出总公司的域名，可以设置转发，也可以将分公司的域名服务器的根服务器设置为总公司的服务器。

具体操作步骤如下：

一、总公司主服务器的安装。

1．获得并安装 DNS 服务器软件（参看实验一的步骤 1）。

2．修改配置文件，即编辑/etc/named.conf：

（1）定义正解区域，在 named.conf 文件内插入如下内容：

```
zone "hope.com"{
        type master;
        file "dns.hope.com";
};
```

（2）定义反解区域，在 named.conf 文件内插入如下内容：

```
zone "10.135.140.in-addr.arpa"{
type master;
file "140.135.10";
};
```

3．用/etc/hosts 文件解析服务器域名，在/etc/hosts 内添加以下内容：

```
140.135.10.1        ns      ns.hope.com
```

4．创建 DNS 数据库文件：

（1）创建正解数据库文件/var/named/dns.hope.com，其内容如下：

```
$TTL86400
@       IN  SOA     ns.hope.com.    admin.hope.com（
                        199802151;serial
                        28800;refresh
                        14400;retry
                        3600000;expire
                        86400）;minimum
            IN      NS      ns.hope.com.
#以下为各地分公司的域名记录，即 NS 记录。
chengdu     IN      NS      ns.chengdu.hope.com.
shanghai    IN      NS      ns.chengdu.hope.com.
changchun   IN      NS      ns.changchun.hope.com.
tianjin     IN      NS      ns.tianjin.hope.com.
……
#以下为各地分公司的 DNS 服务器的主机记录，它是 A 记录。
ns.chengdu      IN      A           140.135.20.1
ns.shanghai     IN      A           140.135.30.1
ns.changchun        IN      A           140.135.40.1
ns.tianjin          IN      A           140.135.50.1
……
#以下两条为邮件服务器记录。
                MX 10   mail.hope.com.
                MX 20   mailback.hope.com.
#以下为总公司的主机记录和别名，即 A 记录和 CNAME 记录。
ns          IN      A       140.135.10.1
nsback      IN      A       140.135.10.2
mail            IN      A       140.135.10.3
mailback        IN      A       140.135.10.4
www         IN      A       140.135.10.5
ftp         IN      A       140.135.10.5
download        IN      CNAME ftp
……
```

（2）创建反解数据库文件/var/named/140.135.10，其内容如下：

```
$TTL86400
@           IN      SOA     ns.hope.com.    root.ns.hope.com
```

```
                        199802151; serial
                        28800; refresh
                        14400; retry
                        3600000; expire
                        86400）; minimum
                IN      NS      ns.hope.com.
1               IN      PTR     ns.hope.com.
```

5．启动 DNS 服务，/etc/rc.d/init.d/named start（参看实验一）。

6．测试 DNS 服务器：

（1）设置/etc/resolv.conf，即将某台客户机的 DNS 设置为 140.135.10.1（或者将 DNS 服务器设置为：140.135.10.1，此时服务器也当客户机）：

nameserver 140.135.10.1

（2）执行 dig –x 140.135.10.1 命令

二、建立各分公司的服务器（子域），下面以成都分公司为例介绍子域的建立。

1．获取并安装服务器软件。（参看实例一的步骤 1）

2．用/etc/hosts 文件解析 DNS 服务器。在/etc/hosts 文件内添加如下内容：

140.135.20.1　　ns　　ns.chengdu.hope.com

3．配置 named.conf 文件。

（1）在 named.conf 中添加转发字段：

```
    forward first;
    forwarders {
    140.135.10.1;
    };
```

各地子公司如果在自己的域名服务器中查不到的域名，统一转发给总公司的域名服务器来处理。

（2）建立正解区域及反解区域，即在/etc/named.conf 中添加如下字段：

```
zone "chengdu.hope.com" {
type master;
file "dns.chengdu.hope.com";
};

zone "20.135.140.in-addr.arpa" {
```

```
type master;
file "140.135.20";
};
```

4．创建域名数据库文件。/var/named/dns.chengdu.hope.com，文件的内容如下：

```
$TTL 86400
@           IN      SOA     ns.chengdu.hope.com.    root.ns.chengdu.hope.com
(
                                199802151; serial
                                28800 ; refresh
                                14400; retry
                                3600000; expire
                                86400）; minimum
            IN      NS          ns.chengdu.hope.com.
ns              IN      A           140.135.20.1
www         IN      A       140.135.20.2
ftp         IN      A       140.135.20.2
……
```

140.135.20 文件的内容如下：

```
$TTL 86400
@       IN      SOA     ns.chengdu.hope.com. root.ns.chengdu.hope.com（
                        199802151; serial
                        28800; refresh
                        14400; retry
                        3600000; expire
                        86400）; minimum
        IN      NS      ns.chengdu.hope.com.
1       IN      PTR     ns.chengdu.hope.com.
```

5．启动 DNS 服务，执行/etc/rc.d/init.d/named start 命令。

6．测试 DNS 服务器：方法与总公司的 DNS 服务器测试相同，这里不现赘述。

按照上面的方法，分别为其他分公司建立域名服务器。

三、总公司备份服务器的建立

1．安装服务器软件（参看实例一的步骤 1）。

2．编辑/etc/named.conf 文件，在该文件中添加如下内容：

```
zone "hope.com"{
type slave;
file "dns.hope.com";
masters {140.135.10.1;};
allow-update { none; };
};
```

3．修改主服务器上的/etc/named.conf 文件，在 options 段中添加如下字段：

notify-source 140.135.10.2;

并将区域“hope.com”更改为如下：

```
zone "hope.com"{
type master;
file "dns.hope.com";
allow-transfer {140.135.10.2;};
allow-update { none; };
};
```

4．重新启动服务器，即可看到/var/named 目录多了文件 dns.hope.com，说明从服务器配置成功。