# Linux DNS域名服务器

## 简介

DNS是Domain Name System（域名系统）的简称。DNS可以为计算机、服务以及接入互联网或局域网的任何资源提供分层的名称解析功能。DNS提供了很多功能，其中最主要的功能就是进行域名与IP地址之间的解析。在互联网中标记唯一一台计算机使用的IP地址，通过合法的IP地址，我们可以与全世界任何一台主机进行通信。然而在当今计算机如此普及的情况下，以人类现有的智慧与记忆很难将大量的IP域名背诵下来，这是使用域名系统就可以将难以记忆的数字IP地址与容易记忆的域名建立映射关系，用户通过输入域名，计算机会寻找指定的DNS服务器，请求服务器帮助解析该域名对应的IP地址，成功解析后，将获得该域名对应的真实IP地址，然后使用该IP地址与对方通信。

域名是分级的，一般分为：主机名.三级域名.二级域名.顶级域名.。（注意：最后有一个点代表的是根域，是所有域名的起点）计算机域名中最后的点则是根，其次是根下的顶级域名，然后是二级域名等。

典型的域名树状结构如下：

DNS树状结构图

根

arpa cn .... edu com net org

In-addr stanford mit yahoo

www www news

例如：斯坦福大学的域名为www.mit.edu.，代表的是根域下有edu子域，edu子域下面有mit子域，mit子域下有主机www。注意，一般情况下，我们通过浏览器输入网址域名时，最后一个根域是不需要输入的。

一般顶级域名代表国家或组织形式，如cn代表中国；edu代表教育机构；com代表商业公司等。二级域名代表组织或公司名称，三级域名代表组织或公司内部的主机名称。最后通过完全合格的域名（FQDN）可以定位全球唯一的一台主机。这种分层管理机制的优势在于根域服务器不需要管理全世界所有的域名信息，它只需要管理顶级域信息即可，而顶级域服务器只需要管理二级域信息即可。依此类推，实现分层管理，这类似于国家的行政管理机制。

域名查询分为递归查询与迭代查询。举例说明，如果客户端准备访问斯坦福大学的网址，客户端首先会检查本地缓存中是否有之前的查询记录，如果有，直接读取结果即可，如果没有相关缓存记录，则向本地DNS服务器发送查询请求，也就是所谓的递归查询，本地DNS服务器如果有答案，就会将答案直接返回给客户端，但本地的DNS服务器没有正确的答案时，它就需要向根服务器查询，不是询问www.stanford.com对应的IP是多少，根服务器仅管理顶级域名，而且所有的顶级域名都属于根的管理范畴，所以此时本地服务器向根查询的结果是：

根服务器会将它所管理的com域对应的IP提供给DNS服务器，本地DNS服务器得到com域服务器的IP后，会再向com查询，然而com也没有最终的答案，com域服务器会将它所管理的stanford域服务器对应的IP提供给本地DNS服务器，最后本地DNS服务器再向stanford域服务器查询，询问该域下主机名称为www的计算机对应的IP地址，由于www主机却是直接属于stanford域的管理范畴，所以stanford会将最终的正确答案返回给本地DNS服务器，最后本地DNS服务器将得到的结果返回给客户端，同时本地DNS服务器还会将结果缓存起来，当下次再有相同查询请求时，本地DNS服务器就可以直接从缓存中找到结果返回给客户端。

## 安装DNS软件

提供DNS服务的软件有很多，而伯克利加州大学研发的BIND（Berkeley Internet Name Domain）是目前应用最广泛的DNS服务系统。该项目软件除了BIND主程序外，在Linux平台下还提供了chroot与utils软件包，bind-chroot软件包的主要功能是使BIND软件可以运行在chroot模式下，这样BIND运行在相对路径的根路径，而不是Linux系统真正的根路径，即使有人对BIND软件进行攻击与破坏，影响的也仅仅是操作系统中的一个子目录，不会影响整个操作系统平台。

通过yum安装bind-chroot后，对bind而言，/var/named/chroot/目录就是根路径，所有BIND软件的配置文件都在根目录的某某路径下。bind-utils软件包提供了一些DNS查询公交，如：dig、host、nslookup等。

[root@centos7 ~]# yum -y install bind bind-chroot bind-utils

## 配置文件解析

BING配置文件主要分为主配置文件与域数据记录文件，主配置文件包括很多使用花括括号引起来的定义语句，在定义语句中可以设置多个选项，主配置文件的核心就是定义域，以及告知计算机到哪里可以找到相应域的数据记录文件。而域数据记录文件则存储具体的域名与IP之间的解析记录，DNS通过读取域数据记录文件来解答客户端的查询请求。

主配置文件一般为/etc/named.conf文件，但当我们安装了bind-chroot软件包后，在后面的内容中提到的一些路径默认指的都是一个虚拟路径，都是相对虚拟根路径而言的，虚拟根路径默认为/var/named/chroot/目录。如果主配置文件是/etc/namd.conf，那么最终主配置文件在本机的绝对路径应该为/var/named/chroot/etc/named.conf，修改主配置文件时一定要注意，如果语法格式有问题，则DNS服务是无法正常启动的。

下面介绍/etc/named.conf文件中常用的定义语句：

acl（Access Control List）语句允许我们预定义一组主机，从而控制是允许还是拒绝他人访问域名服务器，BING预定义了一些主机访问控制列表，其中，any可以匹配任意IP地址，localhost可以匹配本地系统上的所有IP，localnets匹配本地系统所连接的任意网络，none将不匹配任何IP地址。下面通过实例自定义两个访问控制列表，一个为黑名单，一个为白名单。

**acl black {**

**127.16.0.0/16;**

**192.168.0.12;**

**192.168.0.18;**

**};**

**acl white {**

**10.0.0.0/8;**

**192.168.0.0/24;**

**};**

**options {**

**allow-query { white; };**

**blackhole { black; };**

**};**

options语句用来定义全局配置选项，在全局配置中至少需要定义一个工作路径，默认的工作路径为/var/named/，具体常见参数如下：

选项 描述

**directory 设置域名服务的工作目录，默认为/var/namd**

**dump-file 运行rndc dumpdb备份缓存资料后保存的文件路径与名称**

**statistics-file 运行rndc stats后，统计信息的保存路径与名称**

**listen-on port 指定监听的IPv4网络接口**

**allow-query-cache 指定哪些主机可以查询服务器的权威解析记录**

**blackhole 指定哪些主机可以通过服务器查询非权威解析数据，如递归查**

**询数据**

**recursion 设置拒绝哪些主机的查询请求**

**forwards 指定一个IP地址，所有对本服务器的查询将转发到该IP进行解**

**析**

**max-cache-size 设置缓存文件的最大容量**

zone语句用来定义域及下相关选项，定义域也就意味着你希望维护自己公司的域名解析。该语句的重点选项有type 与 file，zone语句内常用的选项及描述如下：

选项 描述

**type 设置域类型，类型可以是**

**hint：当本地找不到相关解析后，可以查询根域名服务器**

**master：定义权威域名服务器**

**slave：定义辅助域名服务器**

**forward：定义转发域名服务器**

**file 定义域数据文件，文件保存在directory所定义的目录下**

**notify 当域数据资料更新后是否主动通知其他域名服务器**

**masters 定义主域名服务器IP地址，当type设置为slave后此选项才有效**

**allow-update 允许哪些主机动态更新域数据**

**allow-transfer 哪些服务器可以从主服务器下载数据文件**

下面是一个简单的zone语句实例，example.com是定义的域，type定义本机为example.com域的主域名服务器，该域的解析文件保存在example.com.zone文件中，该文件被保存在/var/namd/目录下。当定义反向解析域时，需要将网络地址段反过来输入，并以加固的.in-addr.arpa格式结尾。通过定义类型为hint的根域，这样当客户端查询不属于自己维护的域名时，可以寻找根服务器进行迭代查询，最终返回正确的结果，全球的根域名服务器信息保存在named.ca文件中，该文件是在安装完BIND软件后自动生成的模板文件，我们可以在/usr/share/doc/bind-9.9.4/sample/var/named/目录下找到它。

**zone "." IN {**

**type hint;**

**file "named.ca";**

**};**

**zone "example.com" IN {**

**type master;**

**file "example.com.zone"**

**};**

**zone "0.168.192.in-addr.arpa" IN {**

**type master;**

**file "192.168.0.zone";**

**};**

在BIND软件的主配置文件中，如果定义了zone语句，还需要额外创建域数据文件，默认域数据文件被存储在/var/named目录下，文件名称由zone语句中的file选项设定。

数据文件分为正向解析数据文件与方向解析数据文件，正向解析文件保存了域名到IP地址的映射记录，反响解析文件保存了IP地址到域名的映射记录。以下列出了常用的记录类型及描述：

记录类型 描述

SOA记录 域权威记录，说明本机服务器为该域的管理服务器

NS记录 域名服务器记录

A 记录 正向解析记录，域名到IP地址的映射

PTR记录 反向解析记录，IP地址到域名的映射

CNAME记录 别名记录，为主机添加别名

MX记录 邮件记录，指定域内的邮件服务器，需要指定优先级

[下面通过一个简短的正向解析的域数据文件，说明各种记录的语法格式，配置文件中TTL的值为DNS记录的缓存时间，这个值是其他域名服务器将数据存放在缓存中的时间，1D代表一天。SOA记录后面的root.example.com.代表域的权威服务器，jacob.google.com是管理员的邮箱，由于@在数据文件中代表特殊含义，设置使用.来代表](mailto:夏敏通过一个简短的正向解析的域数据文件，说明各种记录的语法格式，配置文件中TTL的值为DNS记录的缓存时间，这个值是其他域名服务器将数据存放在缓存中的时间，1D代表一天。SOA记录后面的root.example.com.代表域的权威服务器，jacob.goole.com是管理员的邮箱，由于@在数据文件中代表特殊含义，设置使用.来代表@fuhao)@符号，所以实际的邮箱应该是jacob@google.com，SOA记录可以跨行输入，跨越多行时使用括号引用。NS记录代表域名服务器的记录，如果公司有多个域名服务器，可以添加多条NS记录，但每个NS记录在下面都需要有对应的A记录。A记录为正向解析记录，格式为在域名后面输入相应的IP地址，IN代表Internet，A代表正向解析记录。在使用MX记录指定邮件服务器时，我们给mail.example.com服务器设置优先级为10，最后CNAME记录为别名，可以使用web.example.com来代表www.example.com。具体配置如下：

**$TTL 1D # DNS缓存时间为1天**

**@ IN SOA root.example.com. jacob.google.com. (**

**0**

**1D**

**1H**

**1W**

**3H)**

**IN NS root.example.com.**

**IN MX 10 mail.example.com.**

**root.example.com. IN A 172.16.0.254**

**[www.example.com](http://www.example.com). IN A 172.16.0.200**

**IN A 172.16.0.201**

**ftp.example.com. IN A 172.16.0.100**

**mail IN A 172.16.0.25**

**web IN C NAME www**

关于配置文件的简写，在BING主配置文件中，zone语句后定义的域对数据文件的简写来说很重要，如果区数据文件中不以“.”结尾的名称后会被自动加上该域名称，如zone语句定义了example.com，而数据文件中有一个mail没有以“.”结尾，那么系统会自动追加example.com到mail后，有时候管理员将A记录错误的写成www.example.com这种形式，由于没有以“.”结尾，实际系统识别的应该是www.example.com.example.com.。此外，在数据文件中使用@符号也同样代表zone所定义的域名称。最后，如果在数据文件中某条记录的名称为空格或制表符，那么BIND系统会直接使用上一条记录的名称，当我们为同一个域名设置多个A记录，则客户端请求该域名解析时，服务器会以轮询的方式将结果返回给客户端，可以从一定程度上实现负载均衡的功能。

## 部署主域名服务器

服务器列表：

服务器IP 服务器名称 功能描述

172.16.0.254 dns1.abc.com 主域名服务器

172.16.0.253 dns2.abc.com 从域名服务器

127.16.0.100 filesever.abc.com 文件服务器

127.16.0.101 printserver.abc.com 打印机服务器

127.16.0.200 [www.abc.com](http://www.abc.com) 网站服务器

127.16.0.201 [www.abc.com](http://www.abc.com) 网站服务器

127.16.0.25 mail.abc.com 邮件服务器

127.16.0.22 ntp.abc.com 时间服务器

1，安装软件

[root@centos7 ~]# yum -y install bind bind-chroot bind-utils

2，修改主配置文件

默认chroot之后的虚拟根目录/var/named/chroot/etc 下面并没有现成的配置文件，需要在/usr/share/doc/bind-9.9.4/samole/etc 目录下找到配置文件模板，复制该文件至/var/named/chroot/etc 目录下，并根据自己的实际情况修改该配置文件。

[root@centos7 ~]# cd /usr/share/doc/bind-9.9.4/sample/etc/

[root@centos7 etc]# cp named.conf /var/named/chroot/etc/

[root@centos7 etc]# chown root.named /var/named/chroot/etc/named.conf

[root@centos7 etc]# vim /var/named/chroot/etc/named.conf

**options**

**{**

**directory "/var/named";**

**dump-file "data/cache\_dump.db";**

**statistics-file "data/named\_stats.txt";**

**memstatistics-file "data/named\_mem\_stats.txt";**

**listen-on port 53 { any; };**

**allow-query { any; };**

**allow-query-cache { any; };**

**recursion yes;**

**};**

**acl secondserver {**

**172.16.0.253;**

**};**

**zone "." IN {**

**type hint;**

**file "named.ca";**

**};**

**zone "abc.com" IN {**

**type master;**

**allow-transfer { secondserver; };**

**file "abc.com.zone";**

**};**

**zone "16.172.in-addr.arpa" IN {**

**type master;**

**allow-transfer { secondserver; };**

**file "172.16.zone";**

**}**

3，创建区数据文件

主配置文件仅是对zone域的定义，冠以域内主机的具体记录的解析，还需要依赖数据文件的内容，常见的域名解析有正向解析记录、反向解析记录、CNAME记录以及MX记录等。在上一步完成了主配置文件中关于zone域的配置后，就需要根据模板创建具体的区数据解析文件，这些文件的创建可以根据软件包中提供的named.localhost文件作为模板。下面将创建一个用于正向解析的abc.com.zone域数据文件，再创建一个用于进行反向解析的172.16.zone域数据文件。

[root@centos7 etc]# cd /usr/share/doc/bind-9.9.4/sample/var/named/

[root@centos7 named]# cp named.ca /var/named/chroot/var/named/

[root@centos7 named]# chown root.named /var/named/chroot/var/named/named.ca

[root@centos7 named]# cp named.localhost /var/named/chroot/var/named/abc.com.zone

[root@centos7 named]# cp named.localhost /var/named/chroot/var/named/172.16.zone

[root@centos7 named]# chwon root.named /var/named/chroot/var/named/abc.com.zone

[root@centos7 named]# chwon root.named /var/named/chroot/var/named/172.16.zone

[root@centos7 named]# vim /var/named/chroot/var/named/abc.com.zone

**$TTL 1D**

**@ IN SOA dns1.abc.com. jacob.abc.com. (**

1. **; serial**

**1D ; refresh**

**1H ; retry**

**1W ; expire**

**3H ) ; minimum**

**NS dns1.abc.com.**

**NS dns2.abc.com.**

**MX 10 mail.abc.com.**

**dns1 IN A 172.16.0.254**

**dns2 IN A 172.16.0.253**

**ntp.abc.com. IN A 172.16.0.22**

**mail.abc.com. IN A 172.16.0.25**

**fileserver IN A 172.16.0.100**

**printserver IN A 172.16.0.101**

**www IN A 172.16.0.200**

**IN A 172.16.0.201**

[root@centos7 named]# vim /var/named/chroot/var/named/abc.com.zone

**$TTL 1D**

**@ IN SOA dns1.abc.com. jacob.abc.com. (**

1. **; serial**

**1D ; refresh**

**1H ; retry**

**1W ; expire**

**3H ) ; minimum**

**NS dns1.abc.com.**

**NS dns2.abc.com.**

**254.0 IN PTR dns1.abc.com**

**253.0 IN PTR dns2.abc.com**

**22.0 IN PTR ntp.abc.com**

**25.0 IN PTR mail.abc.com**

**100.0 IN PTR fileserver.abc.com**

**101.0 IN PTR printserver.abc.com**

**200.0 IN PTR www.abc.com**

**201.0 IN PTR [www.abc.com](http://www.abc.com)**

4，服务管理

在主服务部署完成后，如果不禁用防火墙或通过防火墙特定的端口，客户端主机是无法进行查询工作的。使用Linux自带的防火墙firewalld来开启DNS服务器所需要使用的53端口，其中，TCP的53端口用于主从复制，UDP的53端口用于数据查询。

[root@centos7 ~]# firewall-cmd --premanent --add-port=53/tcp

[root@centos7 ~]# firewall-cmd --permanent --add-port=53/udp

[root@centos7 ~]# firewall-cmd --reload

[root@centos7 ~]# systemctl start named

[root@centos7 ~]# systemctl enable named

5，客户端验证

客户端正确配置DNS服务器后，查询工具比较多，常用的有nslookup、dig、host。

例如：

[root@centos7 ~]# nslookup [www.abc.com](http://www.abc.com)

[root@centos7 ~]# nslookup 172.16.0.100

[root@centos7 ~]# dig [www.abc.com](http://www.abc.com)

[root@centos7 ~]# dig abc.com MX

[root@centos7 ~]# host [www.abc.com](http://www.abc.com)

部署从域名服务器

部署从域名服务器的作用是防止出现单点故障或实现负载均衡。如果只有一台服务器，当该服务器宕机时，将导致所有客户端的地址解析出现文件。另外，为了满足大规模的查询请求，我们可以创建多台DNS服务器实现负载均衡。然而如果所有的DNS都作为主服务器，会有大量的配置需要我们重复进行，再有就是当解析记录发生改变后，各个服务器之间的区数据文件的版本比较混乱（不方便统一），所以我们需要部署从域名服务器，从域名服务器会从主域名服务器上下载数据文件，只要主服务器修改了数据文件中的记录，从服务器就可以自动同步数据记录。

1，安装软件

[root@centos7 ~]# yum -y install bind bind-chroot bind-utils

2，修改配置文件

与主服务器一样，我们需要复制模板配置文件，并修改named.comf配置文件与主服务器配置文件。有所不同的是，所有从服务器配置文件中除根域以外的所有zone域类型均为slave，依次声明自己是从服务器，并使用masters语句指定哪台主服务器进行数据同步。

[root@centos7 ~]# cd /usr/share/doc/bind-9.9.4/sample/etc/

[root@centos7 ~]# cp named.conf /var/named/chroot/etc/

[root@centos7 ~]# chown root.named /var/named/chroot/etc/named.conf

[root@centos7 ~]# vim /var/named/chroot/etc/named.conf

**options**

**{**

**directory "/var/named";**

**dump-file "data/cache\_dump.db";**

**statistics-file "data/named\_stats.txt";**

**memstatistics-file "data/named\_mem\_stats.txt";**

**listen-on port 53 { any; };**

**allow-query { any; };**

**allow-query-cache { any; };**

**recursion yes;**

**};**

**acl secondserver {**

**172.16.0.254;**

**};**

**zone "." IN {**

**type hint;**

**file "named.ca";**

**};**

**zone "abc.com" IN {**

**type slave;**

**masters { 172.16.0.254; };**

**file "abc.com.zone";**

**};**

**zone "16.172.in-addr.arpa" IN {**

**type slave;**

**masters { 127.16.0.254; };**

**file "172.16.zone";**

**}**

3，同步数据文件

从服务器进行同步操作前，需要创建一个DING软件读取操作的目录，以便将主服务器的数据文件保存至该目录。

在从服务器上，我们仅需要简单设置一下BING主配置即可，当从服务器的主配置文件修改完成后，通过启动服务，DING会自动根据配置文件中的masters语句寻找主域名服务器，并将主服务器上的数据文件下载至服务器本地。完成第一次的数据同步后，从服务器会根据同步过来的数据文件中的SOA记录选项，决定下次同步数据的时间，本例中主域名服务器的SOA记录括号中有五个选项。

第一个选项10是序列号，从服务器会根据这个序列号来决定是否进行同步操作，只有当主服务器中数据文件的序列号大于从服务器中数据文件的序列号，从服务器才会真正与主服务器进行数据同步，该序列号建议使用时间格式，如20121212001，表示2012年12月12日的第一次修改（序列号只要是数字即可，BIND不强制要求具体数字格式）。

第二个选项1D，代表从服务器多久与主服务器进行序列号的对比（仅在主服务器序列号大于从服务器的序列号时，才进行数据同步），D代表Day，这里为1天。

第三个选项1H，代表如果从服务器请求连接主服务器时，由于网络延迟、主服务器故障等原因，暂时无法连接到主服务器，那么从服务器会每隔一个小时再试一次。

第四个选项1W，代表如果从服务器尝试一周后还是未能连接到主服务器，则不再进行连接。

第五个选项3H，代表的是缓存的时间为3小时。

[root@centos7 ~]# setsebool -P named\_write\_master\_zones=1

[root@centos7 ~]# mkdir -p /var/named/chroot/var/named/slaves/

[root@centos7 ~]# chown root.named /var/named/chroot/var/named/slaves/

[root@centos7 ~]# chmod 775 /var/named/chroot/var/named/slaves/

[root@centos7 ~]# systemctl start named

[root@centos7 ~]# systemctl enable named

## 更多参考

<http://blog.csdn.net/wangquannetwork/article/details/46576971>

<http://blog.csdn.net/zzw45/article/details/43591>

<http://blog.csdn.net/tCDPYh6sA3/article/details/60961669>