**用bind9架设自己的智能DNS**

中国的南北网络问题，是许多做网站的人的心病  
除了使用双通或者多通机房以外，还可以通过多台镜像服务器的方法来提高用户的访问速度  
但是，如果使用的双通机房并不是单IP的，或者使用多台镜像的做法，就会面临多个不同的服务器IP的问题  
最早的时候，很多镜像服务器都是使用不同的域名的，比如说www1和www2，或者www和cnc  
这样的做法，对用户相当不友好，而且不方便推广，比如说，你要告诉朋友一个网站，还要问他是什么线路才能给网址吗，或者可能你还不知道这个网站有一个专为网通的线路所设置的镜像  
用统一的域名，然后根据用户的线路自动判断引导到不同的镜像服务器，正是智能DNS的功用（智能DNS其实是CDN的其中一部分，是最前端的部分）

现在还是有一些地方使用多域名，比如说下载网站的下载镜像服务器，因为智能DNS有一定的缺陷（并不一定选择到正确的线路，下面说），也提供给用户自由选择线路的权利

目前也有一些公司提供智能DNS解释服务，把你的域名NS指到他们服务器，然后输入网通和电信服务器的IP就ok了  
也有一些公司提供比较完整的CDN服务，比如比较出名的就是chinacache，不少中型网站是使用他们的服务

最初的bind是不支持通过来源选择不同的ip的，因此只能通过iptables的辅助来实现智能DNS，方法也很简单，在DNS服务器上面启动两 份bind，都不要用默认的端口，两份分别解析不同的线路，然后再用iptables将默认端口的DNS请求根据来源IP发送到不同的bind实例  
这种方法缺点很明显，需要多个bind实例，如果需要更多的线路解释那就还要再开bind实例，另外iptables对来源的解释也会导致iptables太多规则

现在的bind支持了根据来源选择的功能，我不确定是不是bind9的新特性，不过bind9是重写过的，相比漏洞百出的bind8来说，稳定和安全很多，以下的配置以bind9为基础

我使用的是CentOS，由于bind服务没有其他服务（比如apache，squid）要求那么高，所以也没有必要自己编译一份，我是用过yum安装的，版本不是最新的稳定版

装好之后，可以用过/etc/init.d/named start成功启动服务  
有一个文件named.conf出现在/etc下，就是bind的主config文件，我们只要改动他  
（我只说明智能DNS实现部分，其他的bind设置自行修改，呵呵）  
另外一个小技巧，可以通过/usr/sbin/named -g -u named来启动bind，可以立刻看到输出信息，方便调试

我们是通过bind9的view功能来分离判断的  
首先，我们需要一份网通的IP列表，这是一份在网上流传的列表，呵呵

acl “CNC” {  
58.16.0.0/16;  
58.17.0.0/17;  
58.17.128.0/17;  
58.18.0.0/16;  
58.19.0.0/16;  
58.20.0.0/16;  
58.21.0.0/16;  
58.22.0.0/15;  
58.240.0.0/15;  
58.242.0.0/15;  
58.244.0.0/15;  
58.246.0.0/15;  
58.248.0.0/13;  
60.0.0.0/13;  
60.8.0.0/15;  
60.10.0.0/16;  
60.11.0.0/16;  
60.12.0.0/16;  
60.13.0.0/18;  
60.13.128.0/17;  
60.14.0.0/15;  
60.16.0.0/13;  
60.24.0.0/14;  
60.30.0.0/16;  
60.31.0.0/16;  
60.208.0.0/13;  
60.216.0.0/15;  
60.218.0.0/15;  
60.220.0.0/14;  
61.48.0.0/13;  
61.133.0.0/17;  
61.134.96.0/19;  
61.134.128.0/17;  
61.135.0.0/16;  
61.137.128.0/17;  
61.138.0.0/17;  
61.138.128.0/18;  
61.139.128.0/18;  
61.148.0.0/15;  
61.156.0.0/16;  
61.159.0.0/18;  
61.161.0.0/18;  
61.161.128.0/17;  
61.162.0.0/16;  
61.163.0.0/16;  
61.167.0.0/16;  
61.168.0.0/16;  
61.176.0.0/16;  
61.179.0.0/16;  
61.181.0.0/16;  
61.182.0.0/16;  
61.189.0.0/17;  
125.32.0.0/16;  
125.40.0.0/13;  
202.96.0.0/18;  
202.96.64.0/21;  
202.96.72.0/21;  
202.97.128.0/18;  
202.97.224.0/21;  
202.97.240.0/20;  
202.98.0.0/21;  
202.98.8.0/21;  
202.99.64.0/19;  
202.99.96.0/21;  
202.99.128.0/19;  
202.99.160.0/21;  
202.99.168.0/21;  
202.99.176.0/20;  
202.99.208.0/20;  
202.99.224.0/21;  
202.99.232.0/21;  
202.99.240.0/20;  
202.102.128.0/21;  
202.102.224.0/21;  
202.102.232.0/21;  
202.106.0.0/16;  
202.107.0.0/17;  
202.108.0.0/16;  
202.110.0.0/17;  
202.111.128.0/18;  
203.93.8.0/24;  
203.93.192.0/18;  
210.13.128.0/17;  
210.14.160.0/19;  
210.14.192.0/19;  
210.15.32.0/19;  
210.15.96.0/19;  
210.15.128.0/18;  
210.21.0.0/16;  
210.52.128.0/17;  
210.53.0.0/17;  
210.53.128.0/17;  
210.74.96.0/19;  
210.74.128.0/19;  
210.82.0.0/15;  
218.8.0.0/14;  
218.12.0.0/16;  
218.21.128.0/17;  
218.24.0.0/14;  
218.56.0.0/14;  
218.60.0.0/15;  
218.67.128.0/17;  
218.68.0.0/15;  
218.104.0.0/14;  
219.154.0.0/15;  
219.156.0.0/15;  
219.158.0.0/17;  
219.158.128.0/17;  
219.159.0.0/18;  
220.252.0.0/16;  
221.0.0.0/15;  
221.2.0.0/16;  
221.3.0.0/17;  
221.3.128.0/17;  
221.4.0.0/16;  
221.5.0.0/17;  
221.5.128.0/17;  
221.6.0.0/16;  
221.7.0.0/19;  
221.7.32.0/19;  
221.7.64.0/19;  
221.7.96.0/19;  
221.8.0.0/15;  
221.10.0.0/16;  
221.11.0.0/17;  
221.11.128.0/18;  
221.11.192.0/19;  
221.12.0.0/17;  
221.12.128.0/18;  
221.13.0.0/18;  
221.13.64.0/19;  
221.13.96.0/19;  
221.13.128.0/17;  
221.14.0.0/15;  
221.192.0.0/15;  
221.194.0.0/16;  
221.195.0.0/16;  
221.196.0.0/15;  
221.198.0.0/16;  
221.199.0.0/19;  
221.199.32.0/20;  
221.199.128.0/18;  
221.199.192.0/20;  
221.200.0.0/14;  
221.204.0.0/15;  
221.206.0.0/16;  
221.207.0.0/18;  
221.207.64.0/18;  
221.207.128.0/17;  
221.208.0.0/14;  
221.212.0.0/16;  
221.213.0.0/16;  
221.216.0.0/13;  
222.128.0.0/14;  
222.132.0.0/14;  
222.136.0.0/13;  
222.160.0.0/15;  
222.162.0.0/16;  
222.163.0.0/19;  
222.163.32.0/19;  
222.163.64.0/18;  
222.163.128.0/17;  
};

如果觉得有哪些部分不是很准确的话，可以自行修改

由于这个部分太长了，所以写在一个专门的文件然后include进named.conf比较容易维护  
例如，我们把上面这段配置保存为cnc\_acl.conf  
然后在named.conf写入

include “/data/named/cnc\_acl.conf”;

即可

接下来就是view的配置，假设域名是xyz.com，呵呵

view “view\_cnc” {  
match-clients { CNC; };  
zone “xyz.com” {  
type master;  
file “/data/named/cnc\_xyz”;  
};  
};

view “view\_any” {  
match-clients { any; };  
zone “xyz.com” {  
type master;  
file “/data/named/any\_xyz”;  
};  
};

就这样就足够了

然后我们需要cnc\\_xyz和any\_xyz两个具体的zone file  
两个文件基本是一模一样的，就是指向不同的ip

$TTL 2h  
xyz.com. IN SOA ns.xyz.com. root.xyz.com. (  
20071022  
2h  
40m  
1w  
2h  
)

this6.com. IN NS ns.xyz.com.  
this6.com. IN MX 1 mail.xyz.com.  
ns IN A 1.1.1.1  
mail IN A 1.1.1.1

www IN A 1.1.1.1

根据不同的线路把www指向不同的IP即可  
这样，我们就实现了自己的智能DNS

后话，目前许多大型网站都是用这个来实现不同的线路来源访问不同的镜像，甚至还做到地区级别，例如，在佛山机房部署了一台镜像服务器，就将广东省的访问都指向这台服务器  
另外，刚刚提到智能DNS也有缺陷，这里详细描述一下，bind9通过用户的来源IP来返回不同的解释IP，而实际上，并不是实际用户直接访问我们的 DNS，而是用户所使用的DNS（比如说广州ADSL用户的默认DNS）负责访问我们的DNS，网通的DNS就保存有我们网通的IP，如果用户自行修改了 DNS，比如说电信的用户硬要用网通的DNS，或者网通的用户使用国外的DNS（比如opendns之类的），就会导致判断失误。