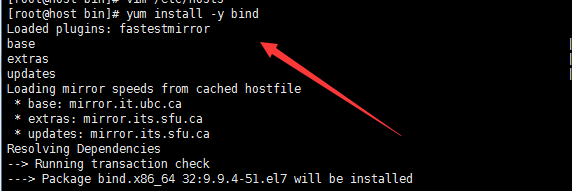
此文章已于 16:34:12 2017/11/25 发布到 HealerJean梦想博客

14、CentOS下搭建DNS服务器

类别 Linux

## 176.122.155.186 为我的服务器的地址 1. 安装bind

### # yum -y install bind\*



## 2. 修改/etc/named.conf配置文件

### vi /etc/named.conf

|  |
| --- |
| //  // named.conf  //  // Provided by Red Hat bind package to configure the ISC BIND named(8) DNS  // server as a caching only nameserver (as a localhost DNS resolver only).  //  // See /usr/share/doc/bind\*/sample/ for example named configuration files.  //  // See the BIND Administrator's Reference Manual (ARM) for details about the  // configuration located in /usr/share/doc/bind-{version}/Bv9ARM.html  options {  listen-on port 53 { any; }; 监听在主机的53端口上。any代表监听所有的主机  listen-on-v6 port 53 { ::1; }; 如果此档案底下有规范到正反解的zone file 档名时，该档名预设应该放置在哪个目录底下  directory "/var/named";  dump-file "/var/named/data/cache\_dump.db";  statistics-file "/var/named/data/named\_stats.txt";  memstatistics-file "/var/named/data/named\_mem\_stats.txt"; allow-query { any; }; // 谁可以对我的DNS服务器提出查询请求。any代表任何人  /\*  - If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.  - If you are building a RECURSIVE (caching) DNS server, you need to enable  recursion.  - If your recursive DNS server has a public IP address, you MUST enable access  control to limit queries to your legitimate users. Failing to do so will  cause your server to become part of large scale DNS amplification  attacks. Implementing BCP38 within your network would greatly  reduce such attack surface  \*/  recursion yes;  dnssec-enable yes;  dnssec-validation yes;  /\* Path to ISC DLV key \*/  bindkeys-file "/etc/named.iscdlv.key";  managed-keys-directory "/var/named/dynamic";  pid-file "/run/named/named.pid";  session-keyfile "/run/named/session.key";  };  logging {  channel default\_debug {  file "data/named.run";  severity dynamic;  };  };  zone "." IN {  type hint;  file "named.ca";  };  include "/etc/named.rfc1912.zones";  include "/etc/named.root.key"; |

## 3. 修改/etc/named.rfc1912.zones文件

### vi /etc/named.rfc1912.zones

|  |
| --- |
| zone "healerjean.top" IN { // 定义要解析主域名  type master;  file "healerjean.top.zone"; // 具体相关解析的配置文件保存在 /var/named/yumaozdy.com.zone 文件中  allow-update { none; };  }; |

## 4、修改解析相关的配置文件

### vi /var/named/healerjean.top.zone

### 解释：

ns.healerjean.top 代表当前dns服务器名称。所以  ns.healerjean.top 一定要解析到自己本身

www IN A 119.75.217.109  // 代表 www. healerjean.top 解析到  119.75.217.109服务器上（我写的是百度）。其他的类似

|  |
| --- |
| $TTL 1D  @ IN SOA ns.healerjean.top. root (  1 ; serial  1D ; refresh  1H ; retry  1W ; expire  3H ) ; minimu  @ IN NS ns.healerjean.top.  ns IN A 176.122.155.186  www IN A 60.135.169.121  名称还可以这样 h.healerjean.top ，如下 (前面的 ns,h必须添加)  $TTL 1D  @ IN SOA h.healerjean.top. root (  1 ; serial  1D ; refresh  1H ; retry  1W ; expire  3H ) ; minimu  @ IN NS h.healerjean.top.  h IN A 176.122.155.186  www IN A 119.75.217.109  test IN A 119.75.217.109  healerjean.top IN A 61.135.169.125  h.healerjan.top IN A 61.135.169.125 |

## 5、修改权限

### chown root:named healerjean.top.zone # 这一步一定要做

## 6、重启服务

### service named restart

## 7、添加域名服务器(必须添加)

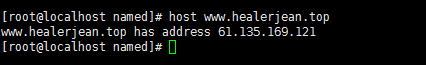
nameserver表示解析域名时使用该地址指定的主机为域名服务器。其中域名服务器是按照文件中出现的顺序来查询的,且只有当第一个nameserver没有反应时才查询下面的nameserver。

### vim /etc/resolv.conf

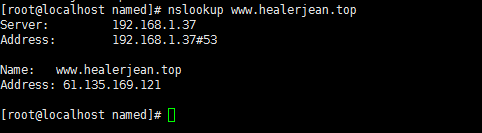
|  |
| --- |
| # Generated by NetworkManager  search localdomain  #nameserver 8.8.8.8  #nameserver 74.82.42.42  #nameserver 8.8.4.4  nameserver 176.122.155.186 |

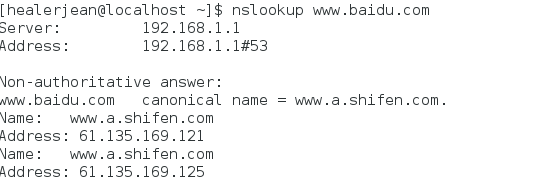
## 8、host 命令

### host [www.healerjean.top](http://www.healerjean.top)

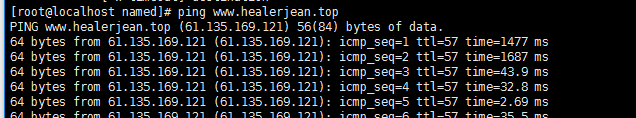


### nslookup 上面的server 为dns的服务器配置地址，下面的为配置的要跳转的地址



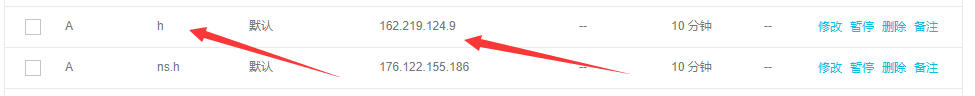


### ping



## 9、阿里云服务器，让自己的dns服务器支持域名

### 1、域名添加A记录

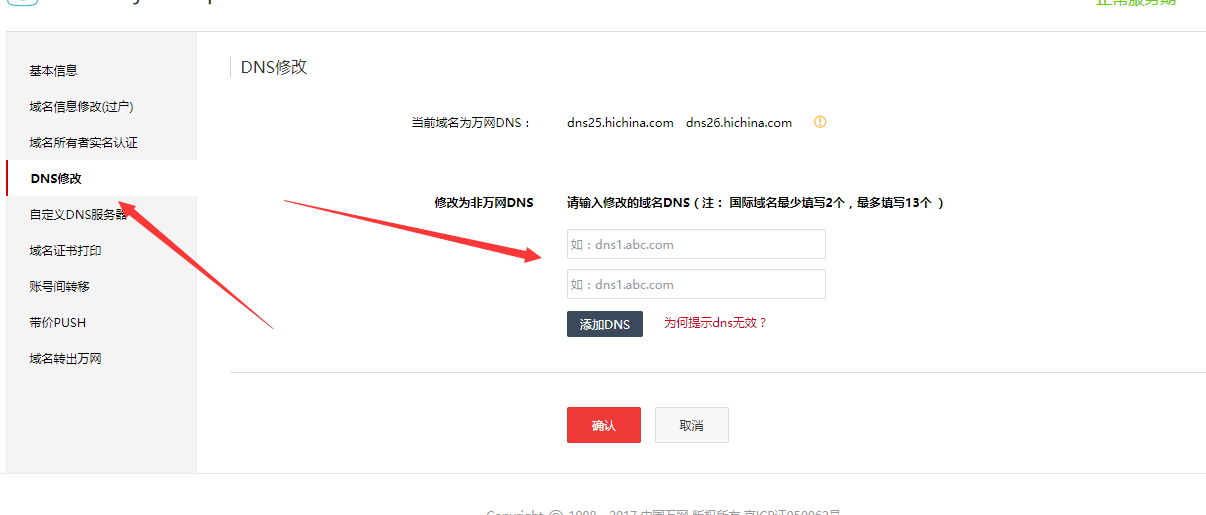


### 2、添加自定义的DNS服务器



### 3、修改dns

**傻叉的事情出现了，这里 需要两个DNS，也就是说一个DNS坏掉了，可以换成另外一个。那么这样的话，我们就需要弄两个主机了，最后，无奈之下我用了两个vps，但是DNS必须要备案注册，所以最终以失败告终了。哎，对不起各位了。但是这种方式在公司内部，是可以实现的，可以用来屏蔽广告，和谷歌等。**



## 10、屏蔽谷歌

|  |
| --- |
| vi /etc/named.rfc1912.zones  // 添加下面这段  zone "google.com" IN {  type master;  file "google.com.zone";  };  [root@ns named]# cp -a yumaozdy.com.zone google.com.zone  [root@ns named]# vi google.com.zone  $TTL 86400  @ IN SOA ns.google.com. root (  1 ; serial  1D ; refresh  1H ; retry  1W ; expire  0 ) ; minimum  @ IN NS ns.google.com.  ns IN A 192.168.1.219  www IN A 192.168.1.11  bbs IN A 192.168.1.46  ttt IN A 192.168.1.68 |

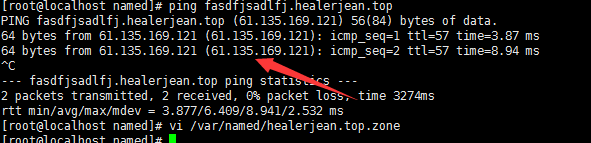
# 二、泛域名解析

## 1、为所有的子域名设置ip地址

|  |
| --- |
| $TTL 1D  @ IN SOA ns.healerjean.top. root (  1 ; serial  1D ; refresh  1H ; retry  1W ; expire  3H ) ; minimu  @ IN NS ns.healerjean.top.  ns IN A 192.169.1.37  www IN A 61.135.169.121  \* IN A 61.135.169.121  hh IN A 204.79.197.200 |

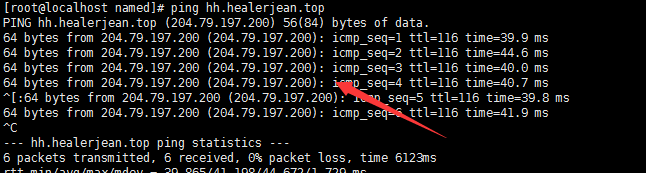
## 2、访问不存在的子域名

### ping fasdfjsadlfj.healerjean.top



## 3、访问存在的子域名

### ping hh.healerjean.top



# 文件

