

公告

昵称：David_Tang
园龄：7年
粉丝：2040
关注：3
+加关注

< 2019年1月 >						
日	一	二	三	四	五	六
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

搜索

找找看

谷歌搜索

我的标签

- linux(22)
- postgresql(22)
- Packet Tracer(15)
- Oracle笔记(14)
- 微信(12)
- 微信公众号(11)
- zhuojinsz(11)
- 微信公众平台开发(9)
- 微信开发(8)
- 卓锦苏州(6)
- 更多

随笔分类(136)

- Linux(20)
- Linux Server(17)
- Linux Shell(12)
- Mysql(2)
- Network(16)
- Oracle(19)
- PHP(12)
- Postgresql(26)
- 微信开发(12)

随笔档案(115)

- 2014年7月 (2)
- 2014年6月 (4)
- 2014年5月 (1)
- 2014年3月 (1)
- 2014年2月 (1)
- 2013年10月 (2)
- 2013年8月 (2)
- 2013年7月 (3)
- 2013年6月 (3)
- 2013年4月 (8)
- 2013年3月 (5)
- 2013年2月 (1)
- 2013年1月 (4)
- 2012年12月 (2)
- 2012年11月 (3)
- 2012年9月 (16)
- 2012年8月 (18)
- 2012年7月 (23)

随笔-115 文章-0 评论-709

CentOS 6.4下Squid代理服务器的安装与配置

一、简介

代理服务器英文全称是Proxy Server，其功能就是代理网络用户去取得网络信息。

Squid是一个缓存Internet 数据的软件，其接收用户的下载申请，并自动处理所下载的数据。当一个用户想要下载一个主页时，可以向Squid 发出一个申请，要Squid 代替其进行下载，然后Squid 连接所申请网站并请求该主页，接着把该主页传给用户同时保留一个备份，当别的用户申请同样的页面时，Squid 把保存的备份立即传给用户，使用户觉得速度相当快。Squid 可以代理HTTP、FTP、GOPHER、SSL和WAIS等协议并且Squid 可以自动地进行处理，可以根据自己的需要设置Squid，使之过滤掉不想要的东西。

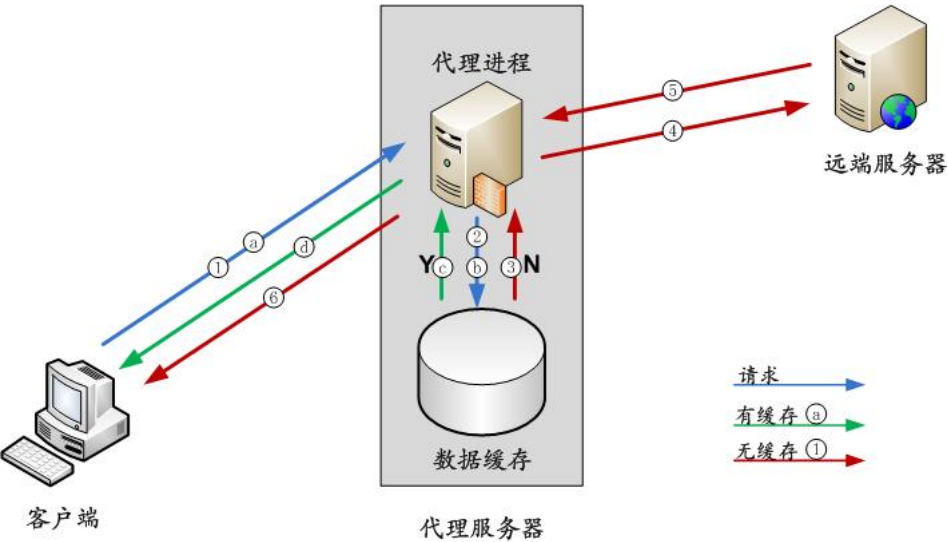
1.1 工作流程

当代理服务器中有客户端需要的数据时：

- a. 客户端向代理服务器发送数据请求；
- b. 代理服务器检查自己的数据缓存；
- c. 代理服务器在缓存中找到了用户想要的的数据，取出数据；
- d. 代理服务器将从缓存中取得的数据返回给客户端。

当代理服务器中没有客户端需要的数据时：

- 1. 客户端向代理服务器发送数据请求；
- 2. 代理服务器检查自己的数据缓存；
- 3. 代理服务器在缓存中没有找到用户想要的的数据；
- 4. 代理服务器向Internet 上的远端服务器发送数据请求；
- 5. 远端服务器响应，返回相应的数据；
- 6. 代理服务器取得远端服务器的数据，返回给客户端，并保留一份到自己的数据缓存中。



Squid代理服务器工作在TCP/IP的应用层。

2012年6月 (9)
2012年5月 (6)
2012年4月 (1)

链接

蘇's 百度
蘇's 网易
蘇's 新浪

积分与排名

积分 - 322061
排名 - 711

最新评论

1. Re:Linux下Nagios的安装与配置

楼主，为啥我Linux下启动了，web页面上显示没在跑

--长夏
2. Re:CentOS 6.5下利用Rsyslog+LogAnalyzer+MySQL部署日志服务器

不错啊。。。谢谢

--冯先生008
3. Re:PostgreSQL学习手册(一) 数据表

终于找到一系列pg的 文章，学起来

--Kevin_zheng
4. Re:Oracle笔记（六）多表查询

博主好厉害啊！思路非常清晰，讲解的非常细致，案例也很精辟到位！谢谢你贡献了这么好的文章，辛苦了

--天主之城的迈
5. Re:PostgreSQL的generate_series函数应用

cool

--Kevin_zheng

阅读排行榜

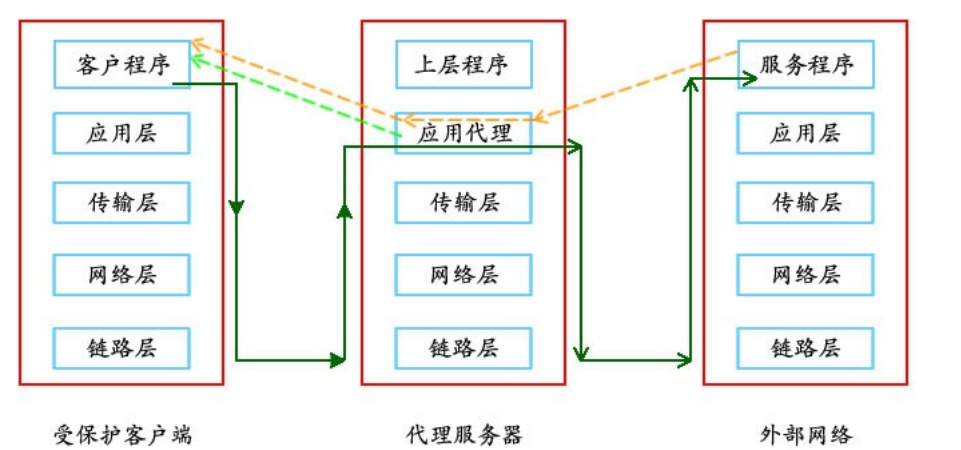
1. CentOS yum 源的配置与使用 (448825)
2. Linux NFS服务器的安装与配置 (388162)
3. linux screen 命令详解(365766)
4. 微信公众平台开发（一）配置接口 (328138)
5. Linux下Nagios的安装与配置 (250779)

评论排行榜

1. Linux下Nagios的安装与配置(92)
2. 微信公众平台开发（一）配置接口 (69)
3. 微信公众平台开发（十）消息回复总结(42)
4. Oracle笔记（六）多表查询(34)
5. CentOS 6.3下Samba服务器的安装与配置(31)

推荐排行榜

1. Linux下Nagios的安装与配置(63)
2. Oracle笔记（六）多表查询(58)
3. CentOS yum 源的配置与使用(56)
4. 微信公众平台开发（一）配置接口 (47)
5. CentOS 6.3下Samba服务器的安装与配置(34)



1.2 Squid 分类

按照代理类型的不同，可以将Squid 代理分为正向代理和反向代理，正向代理中，根据实现方式的不同，又可以分为普通代理和透明代理。

- 普通代理：需要客户机在浏览器中指定代理服务器的地址、端口；
- 透明代理：适用于企业的网关主机（共享接入Internet）中，客户机不需要指定代理服务器地址、端口等信息，代理服务器需要设置防火墙策略将客户机的Web访问数据转交给代理服务程序处理；
- 反向代理：是指以代理服务器来接受internet上的连接请求，然后将请求转发给内部网络上的服务器，并将从服务器上得到的结果返回给internet上请求连接的客户端，此时代理服务器对外就表现为一个服务器。

官方地址：<http://www.squid-cache.org/>

参考文档：<http://www.squid-cache.org/Doc/config/>

二、系统环境

操作系统：CentOS release 6.4 (Final)
Squid版本：squid-3.1.10-20.el6_5.3.x86_64
SELINUX=disabled
HTTP Service: stoped

三、安装Squid服务

- 3.1 检查squid软件是否安装
- # rpm -qa|grep squid
- 3.2 如果未安装，则使用yum 方式安装
- # yum -y install squid

```
Package      Arch      Version      Repository      Size
-----
Installing:
squid        x86_64    7:3.1.10-20.el6_5.3    updates        1.7 M

Transaction Summary
-----
Install      1 Package(s)

Total download size: 1.7 M
Installed size: 5.8 M
Downloading Packages:
squid-3.1.10-20.el6_5.3.x86_64.rpm      | 1.7 MB      00:03
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : 7:squid-3.1.10-20.el6_5.3.x86_64      1/1
  Verifying  : 7:squid-3.1.10-20.el6_5.3.x86_64      1/1

Installed:
squid.x86_64 7:3.1.10-20.el6_5.3

Complete!
[root@SquidServer ~]#
```

- 3.3 设置开机自启动
- # chkconfig --level 35 squid on //在3、5级别上自动运行squid服务

四、squid服务器的配置文件说明

squid 的主配置文件是 /etc/squid/squid.conf，所有squid的设定都是在这个文件里配置，下面我们来讲解一下该文件的配置选项。

```

http_port 3128 //设置监听的IP与端口号

cache_mem 64 MB //额外提供给squid使用的内存，squid的内存总占用为 X * 10+15+"cache_mem"，其中
X为squid的cache占用的容量（以GB为单位），
//比如下面的cache大小是100M，即0.1GB，则内存总占用为0.1*10+15+64=80M，推荐大
小为物理内存的1/3-1/2或更多。
maximum_object_size 4 MB //设置squid磁盘缓存最大文件，超过4M的文件不保存到硬盘

minimum_object_size 0 KB //设置squid磁盘缓存最小文件

maximum_object_size_in_memory 4096 KB //设置squid内存缓存最大文件，超过4M的文件不保存到内存

cache_dir ufs /var/spool/squid 100 16 256 //定义squid的cache存放路径、cache目录容量（单位M）、一
级缓存目录数量、二级缓存目录数量

logformat combined %>a %ui %un [%tl] "%rm %ru HTTP/%rv" %Hs %<st "%{Referer}>h" "%{User-
Agent}>h" %Ss:%Sh //log文件日志格式

access_log /var/log/squid/access.log combined //log文件存放路径和日志格式

cache_log /var/log/squid/cache.log //设置缓存日志

logfile_rotate 60 //log轮循 60天

cache_swap_high 95 //cache目录使用量大于95%时，开始清理旧的cache

cache_swap_low 90 //cache目录清理到90%时停止。

acl localnet src 192.168.1.0/24 //定义本地网段

http_access allow localnet //允许本地网段使用

http_access deny all //拒绝所有

visible_hostname squid.david.dev //主机名

cache_mgr mchina_tang@qq.com //管理员邮箱

```

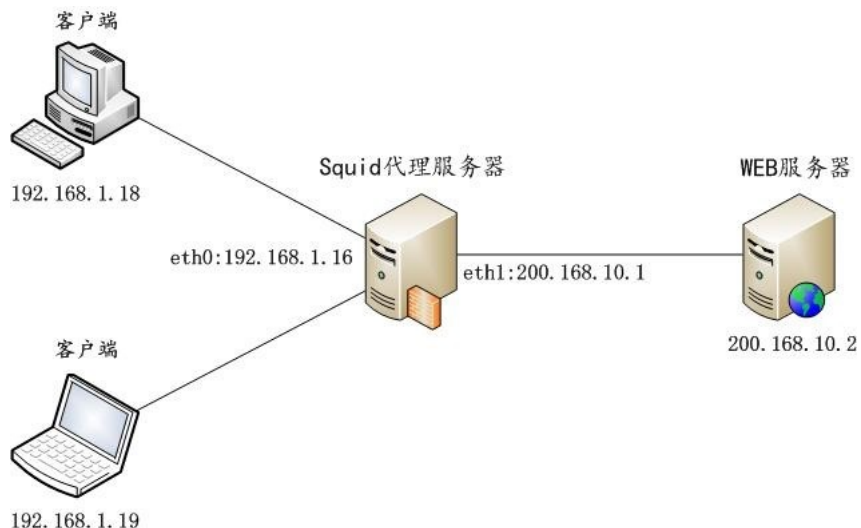
关于ACL的知识，大家可以自行百度查阅，其他更多高级选项，请参考官方文档：<http://www.squid-cache.org/Doc/config/>。

注意：squid2.0 和squid3.0的差别还是很大的，如果配置完，启动squid不正确，请大家多多参考官方文档的相应版本说明。

五、普通代理服务

即标准的、传统的代理服务，需要客户机在浏览器中指定代理服务器的地址、端口。

实验拓扑图如下：



5.1 配置Squid 代理服务器IP地址

将eth1的IP地址修改为200.168.10.1

```
# ifconfig eth1 200.168.10.1
```

```
Last login: Tue Jun 24 15:29:57 2014
[root@SquidServer ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:A4:C1:C2
          inet addr:192.168.1.16  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fea4:c1c2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5170 errors:0 dropped:0 overruns:0 frame:0
          TX packets:133 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:5888864 (5.6 MiB)  TX bytes:14216 (13.8 KiB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:2E:2B:C1
          inet addr:200.168.10.1  Bcast:200.168.10.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe2e:2bc1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5201 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2103 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:5980569 (5.7 MiB)  TX bytes:116762 (114.0 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

[root@SquidServer ~]#
```

5.2 编辑squid 主配置文件/etc/squid/squid.conf

```
http_port 3128
cache_mem 64 MB
maximum_object_size 4 MB
cache_dir ufs /var/spool/squid 100 16 256
access_log /var/log/squid/access.log
acl localnet src 192.168.1.0/24
http_access allow localnet
http_access deny all
visible_hostname squid.david.dev
cache_mgr mchina_tang@qq.com
```

5.3 初始化

```
# squid -z
```

```
[root@SquidServer squid]# ll /var/spool/squid/
total 0
[root@SquidServer squid]# squid -z
2014/06/24 15:54:36 Creating Swap Directories
2014/06/24 15:54:36 /var/spool/squid exists
2014/06/24 15:54:36 Making directories in /var/spool/squid/00
2014/06/24 15:54:36 Making directories in /var/spool/squid/01
2014/06/24 15:54:36 Making directories in /var/spool/squid/02
2014/06/24 15:54:36 Making directories in /var/spool/squid/03
2014/06/24 15:54:36 Making directories in /var/spool/squid/04
2014/06/24 15:54:36 Making directories in /var/spool/squid/05
2014/06/24 15:54:36 Making directories in /var/spool/squid/06
2014/06/24 15:54:36 Making directories in /var/spool/squid/07
2014/06/24 15:54:36 Making directories in /var/spool/squid/08
2014/06/24 15:54:36 Making directories in /var/spool/squid/09
2014/06/24 15:54:36 Making directories in /var/spool/squid/0A
2014/06/24 15:54:36 Making directories in /var/spool/squid/0B
2014/06/24 15:54:36 Making directories in /var/spool/squid/0C
2014/06/24 15:54:36 Making directories in /var/spool/squid/0D
2014/06/24 15:54:36 Making directories in /var/spool/squid/0E
2014/06/24 15:54:36 Making directories in /var/spool/squid/0F
[root@SquidServer squid]# ll /var/spool/squid/
total 64
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 00
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 01
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 02
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 03
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 04
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 05
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 06
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 07
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 08
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 09
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 0A
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 0B
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 0C
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 0D
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 0E
drwxr-x--- 258 squid squid 4096 Jun 24 15:54 0F
[root@SquidServer squid]#
```

5.4 启动Squid

```
# /etc/init.d/squid start
```



```
[root@SquidServer ~]# /etc/init.d/squid start
Starting squid: .
[root@SquidServer ~]#
```

5.5 配置Web 服务器

A. 安装Apache

```
# rpm -qa|grep httpd
```

```
# yum -y install httpd
```

B. 启动Apache并加入开机启动

```
# /etc/init.d/httpd start
```

```
# chkconfig httpd on
```

C. 创建index.html

```
# echo "<h1>Squid-Web1/200.168.10.2</h1>" > /var/www/html/index.html
```

D. 修改Web服务器IP地址

将web服务器的IP地址修改为200.168.10.2

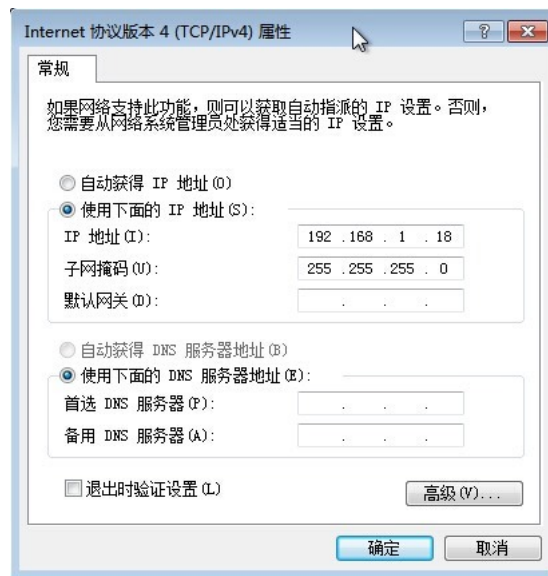
```
# ifconfig eth0 200.168.10.2
```

```
[root@SquidWeb1 ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:A4:C1:C2
          inet addr:200.168.10.2  Bcast:200.168.10.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fea4:c1c2/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5694 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1973 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4283728 (4.0 MiB)  TX bytes:165486 (161.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

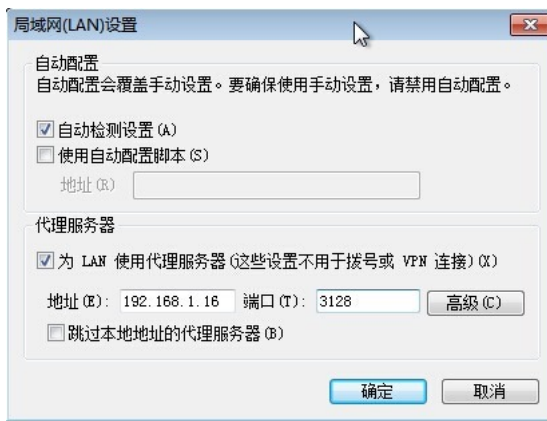
[root@SquidWeb1 ~]#
```

5.6 配置客户端IP地址



5.7 配置浏览器代理

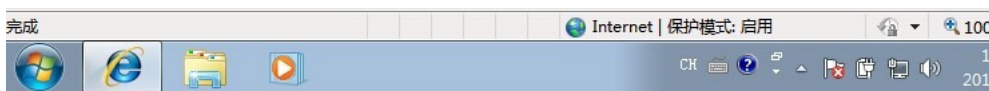
打开浏览器（以IE为例，其他类似），菜单栏 -> 工具 -> Internet 选项 -> 连接 -> 局域网设置 -> 代理服务，按照以下格式设置。



5.8 测试



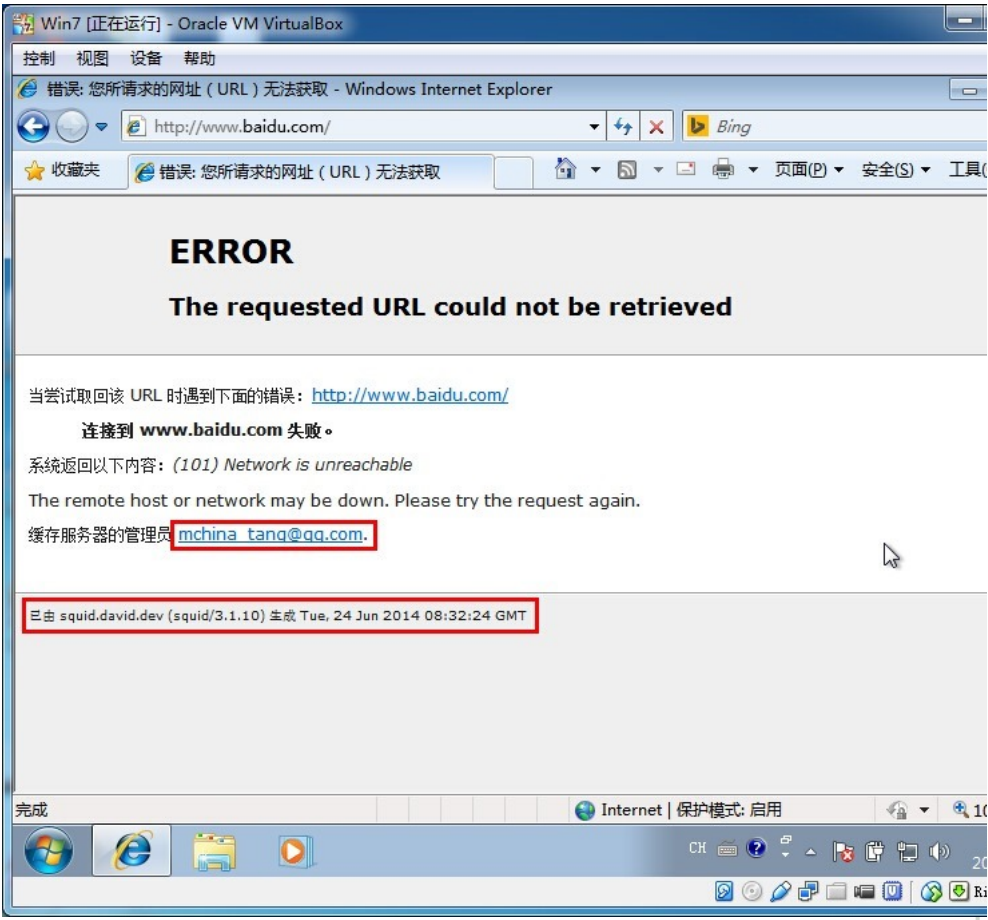
Squid-Web1/200.168.10.2



测试成功。

5.9 测试错误页面

在Oracle VM VirtualBox里，因为上面手动设置了ip地址，导致外网不能访问，刚好可以测试访问出错的页面。

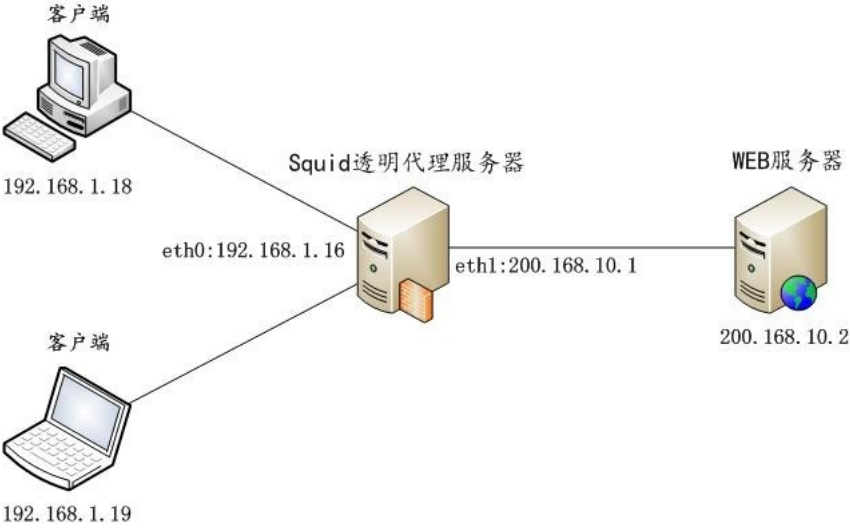


可以看到在squid 配置文件里设置的参数在错误页面里的显示。

六、透明代理服务

适用于企业的网关主机，客户机不需要指定代理服务器地址、端口等信息，通过iptables将客户机的Web访问数据转交给代理服务程序处理。

实验拓扑图如下：



6.1 修改squid 主配置文件/etc/squid/squid.conf

```
http_port 3128 transparent
cache_mem 64 MB
maximum_object_size 4 MB
cache_dir ufs /var/spool/squid 100 16 256
access_log /var/log/squid/access.log
acl localnet src 192.168.1.0/24
http_access allow localnet
http_access deny all
visible_hostname squid.david.dev
cache_mgr mchina_tang@qq.com
```



在http_port 3128 后添加transparent 关键字。

6.2 reload

reload 让上面的配置生效。

```
# /etc/init.d/squid reload
```

6.3 添加iptables规则，把内部的http请求重定向到3128端口

A. 启动iptables 服务

```
# /etc/init.d/iptables start
```

```
[root@SquidServer ~]# /etc/init.d/iptables status
iptables: Firewall is not running.
[root@SquidServer ~]#
[root@SquidServer ~]#
[root@SquidServer ~]# /etc/init.d/iptables start
iptables: Applying firewall rules: [ OK ]
[root@SquidServer ~]#
```

B. 清除现有iptables filter 表规则

```
# iptables -F
```

```
[root@SquidServer squid]# iptables -L -n
Chain INPUT (policy ACCEPT)
target prot opt source destination state RELATED,ESTABLISHED
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 state NEW tcp dpt:22
REJECT all -- 0.0.0.0/0 0.0.0.0/0 reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target prot opt source destination reject-with icmp-host-prohibited
REJECT all -- 0.0.0.0/0 0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
[root@SquidServer squid]#
[root@SquidServer squid]# iptables -F
[root@SquidServer squid]#
[root@SquidServer squid]# iptables -L -n
Chain INPUT (policy ACCEPT)
target prot opt source destination
Chain FORWARD (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
[root@SquidServer squid]#
```

C. 保存iptables 设置

```
# /etc/init.d/iptables save
```

```
[root@SquidServer ~]# /etc/init.d/iptables save
iptables: saving firewall rules to /etc/sysconfig/iptables:[ OK ]
[root@SquidServer ~]#
[root@SquidServer ~]# /etc/init.d/iptables restart
iptables: Flushing firewall rules: [ OK ]
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: unloading modules: [ OK ]
iptables: Applying firewall rules: [ OK ]
[root@SquidServer ~]#
[root@SquidServer ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target prot opt source destination
Chain FORWARD (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
[root@SquidServer ~]#
```

D. 查看nat 表设置

```
# iptables -t nat -L -n
```

```
[root@SquidServer ~]# iptables -t nat -L -n
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
[root@SquidServer ~]#
```

E. 在nat表中新增一条规则

```
# iptables -t nat -I PREROUTING -i eth0 -s 192.168.1.0/24 -p tcp --dport 80 -j REDIRECT --to-port 3128
```



```
[root@SquidServer ~]# iptables -t nat -I PREROUTING -i eth0 -s 192.168.1.0/24 -p tcp --dport 80 -j REDIRECT --to-ports 3128
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
REDIRECT tcp -- 192.168.1.0/24 0.0.0.0/0 tcp dpt:80 redir ports 3128
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
```

F. 保存

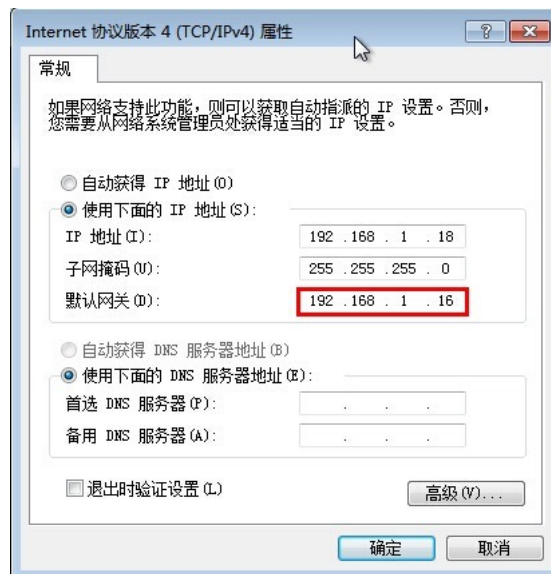
```
[root@SquidServer ~]# /etc/init.d/iptables save
iptables: saving firewall rules to /etc/sysconfig/iptables:[ OK ]
[root@SquidServer ~]# /etc/init.d/iptables restart
iptables: Flushing firewall rules: [ OK ]
iptables: Setting chains to policy ACCEPT: nat filter [ OK ]
iptables: Unloading modules: [ OK ]
iptables: Applying firewall rules: [ OK ]
[root@SquidServer ~]# iptables -t nat -L -n
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
REDIRECT tcp -- 192.168.1.0/24 0.0.0.0/0 tcp dpt:80 redir ports 3128
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
[root@SquidServer ~]#
```

G. 设置iptables 开机启动

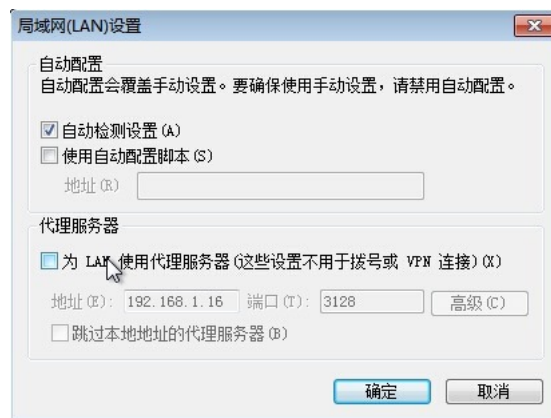
```
# chkconfig iptables on
```

6.4 修改客户端IP地址

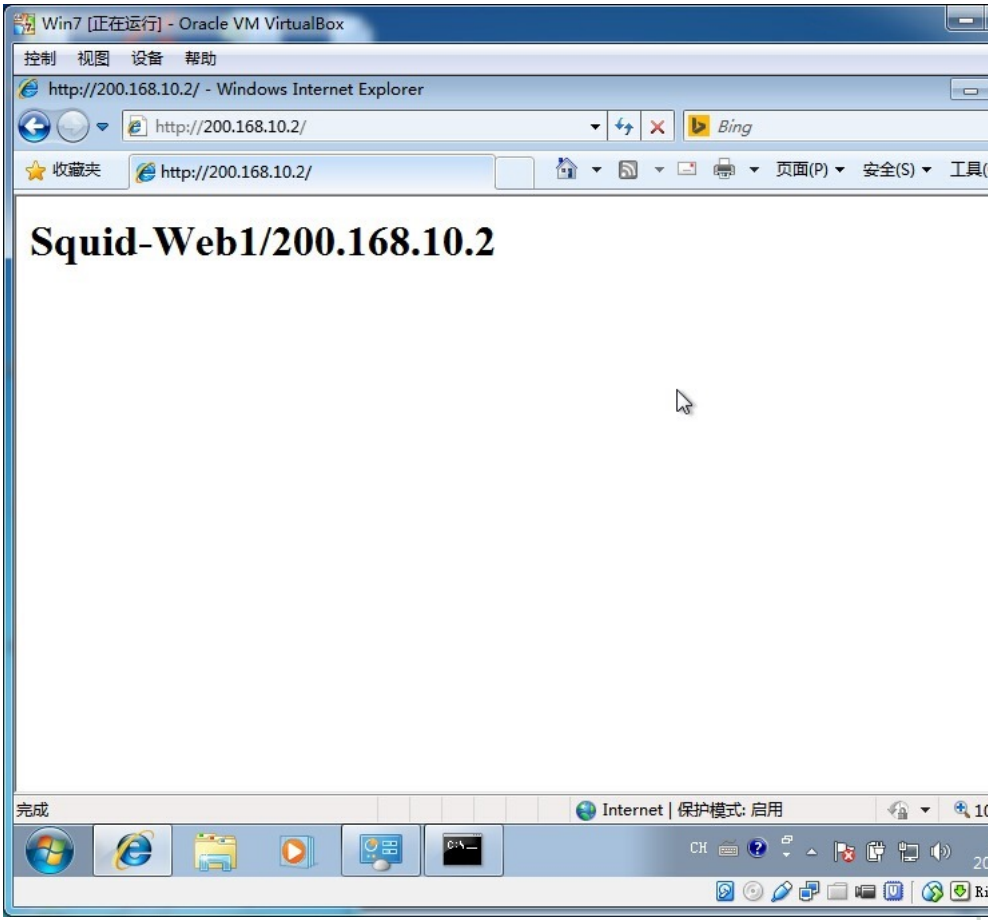
将默认网关设置为squid 服务器的内网ip地址。



6.5 在浏览器中，取消代理设置



6.6 测试

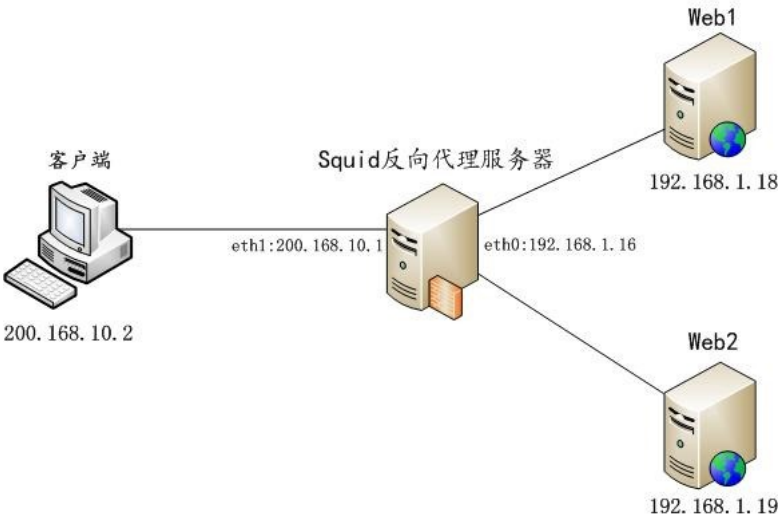


透明代理测试成功。

七、反向代理服务

为Internet用户访问企业Web站点提供缓存加速。

实验拓补图如下：



7.1 关闭防火墙

```
# /etc/init.d/iptables stop
```

7.2 修改Web Server 主页

Web1:

```
# echo "<h1>Squid-Web1/192.168.1.18</h1>" > /var/www/html/index.html
```

Web2:

```
# echo "<h1>Squid-Web1/192.168.1.19</h1>" > /var/www/html/index.html
```

7.3 配置squid

```

http_port 80 accel vhost
http_access allow all
cache_peer 192.168.1.18 parent 80 0 originserver round-robin weight=1
cache_peer 192.168.1.19 parent 80 0 originserver round-robin weight=1
visible_hostname squid.david.dev
cache_mgr mchina_tang@qq.com

```

7.4 启动Squid服务

```

[root@SquidServer ~]# /etc/init.d/squid restart
Stopping squid:
Starting squid: ..... [FAILED]
[root@SquidServer ~]# tail /var/log/messages
Jun 27 11:14:34 SquidServer squid[2091]: Squid Parent: child process 2106 started
Jun 27 11:14:34 SquidServer squid[2091]: Squid Parent: child process 2106 exited with status
Jun 27 11:14:34 SquidServer squid[2091]: Squid Parent: child process 2112 started
Jun 27 11:14:37 SquidServer squid[2091]: Squid Parent: child process 2112 exited with status
Jun 27 11:14:38 SquidServer squid[2091]: Squid Parent: child process 2119 started
Jun 27 11:14:38 SquidServer squid[2091]: Squid Parent: child process 2119 exited with status
Jun 27 11:14:41 SquidServer squid[2091]: Squid Parent: child process 2119 exited with status
Jun 27 11:14:41 SquidServer squid[2091]: Exiting due to repeated, frequent failures
[root@SquidServer ~]#
[root@SquidServer ~]# /etc/init.d/httpd status
httpd (pid 1461) is running...
[root@SquidServer ~]#

```

squid启动失败，因为上面设定了squid的监听端口是80，和系统的http服务冲突，所以要将http服务停掉。

7.5 squid 启动失败，将系统开机自启动的apache服务关闭。

```

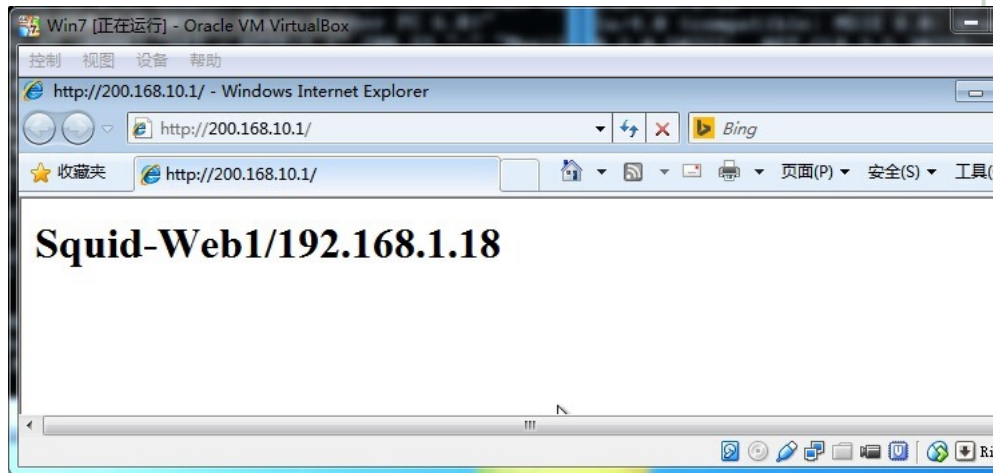
[root@SquidServer ~]# /etc/init.d/httpd stop
Stopping httpd: [ OK ]
[root@SquidServer ~]# /etc/init.d/squid restart
Stopping squid: [ FAILED ]
Starting squid: [ OK ]
[root@SquidServer ~]#

```

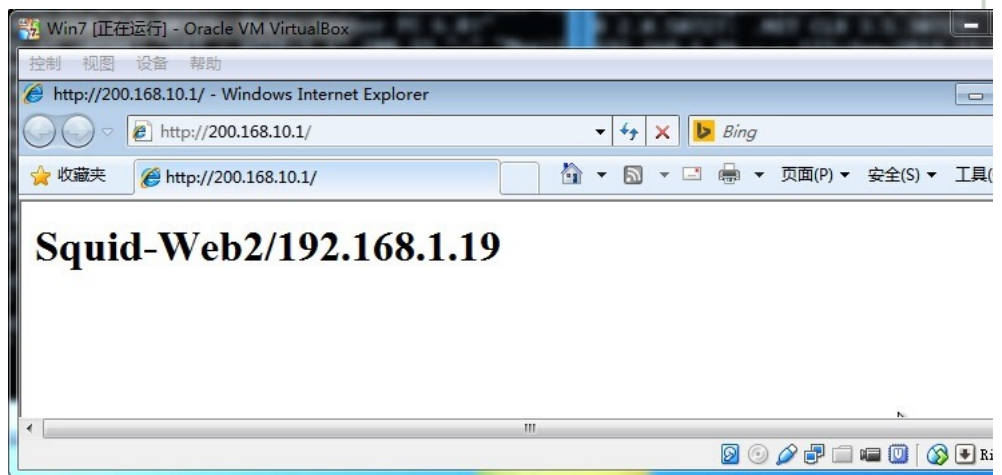
7.6 测试

squid 采用了round-robin，所以客户端的访问将轮询两台web服务器，采用 "Ctrl + F5" 来深度刷新测试。

Web1:



Web2:



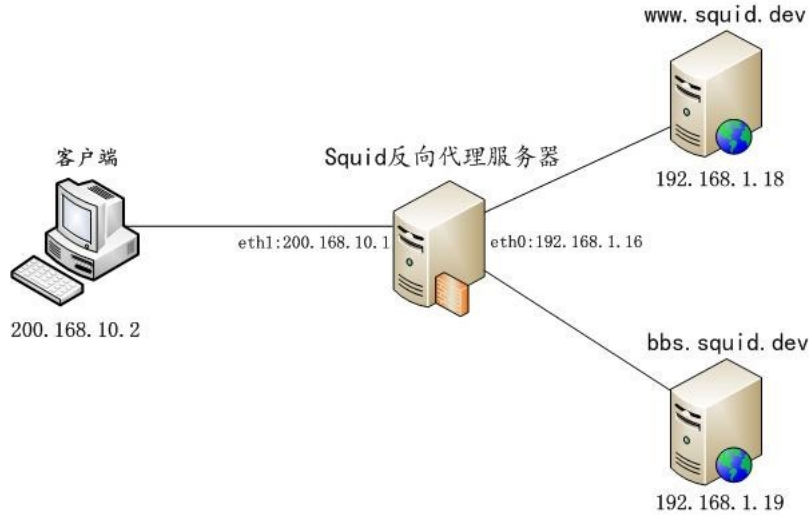
查看squid 的访问日志。

```
[root@SquidServer ~]# tailf /var/log/squid/access.log
1403839960.054 7 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839960.368 7 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839960.537 4 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839960.685 3 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839960.859 5 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839961.215 5 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839961.507 6 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839961.850 6 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839962.153 6 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
1403839963.930 8 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 427 GET http://200.168.10.1/ - ROUNDROBIN_PARENT
```

八、实际应用

下面实验将模拟通过不同的域名访问不同的机器，简单实现企业应用中的负载均衡。客户端在浏览器地址栏中输入www.squid.dev，将访问192.168.1.18这台机器，访问bbs.squid.dev，将访问192.168.1.19这台机器。

实验拓扑图如下：



8.1 修改Web Server 主页

Web1:

```
# echo "<h1>www.squid.dev/192.168.1.18</h1>" > /var/www/html/index.html
```

Web2:

```
# echo "<h1>bbs.squid.dev/192.168.1.19</h1>" > /var/www/html/index.html
```

8.2 配置Squid

```

http_port 80 accel vhost
http_access allow all
cache_peer 192.168.1.18 parent 80 0 originserver name=www
cache_peer 192.168.1.19 parent 80 0 originserver name=bbs
cache_peer_domain www www.squid.dev
cache_peer_domain bbs bbs.squid.dev
visible_hostname squid.david.dev
cache_mgr mchina_tang@qq.com
  
```

8.3 配置客户端

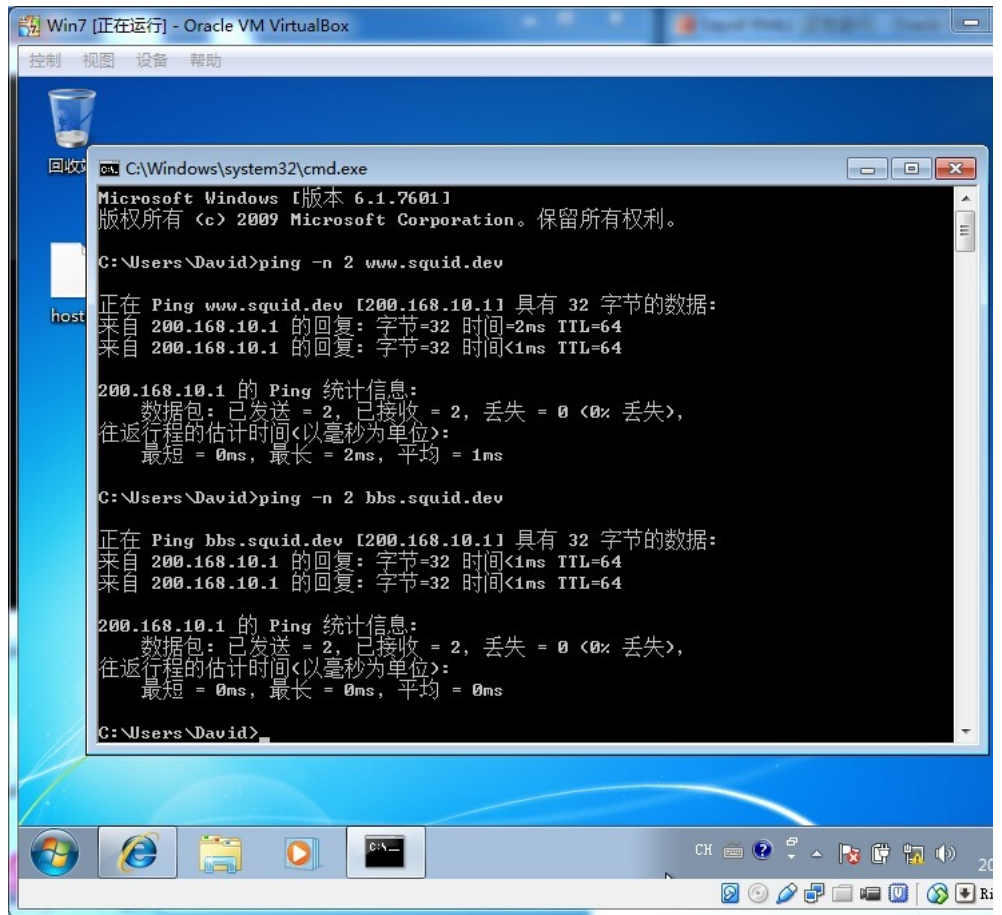
这里可以使用DNS服务来解析，这里我们为了方便，就在hosts文件里直接指定。

修改C:\Windows\System32\drivers\etc\hosts 文件

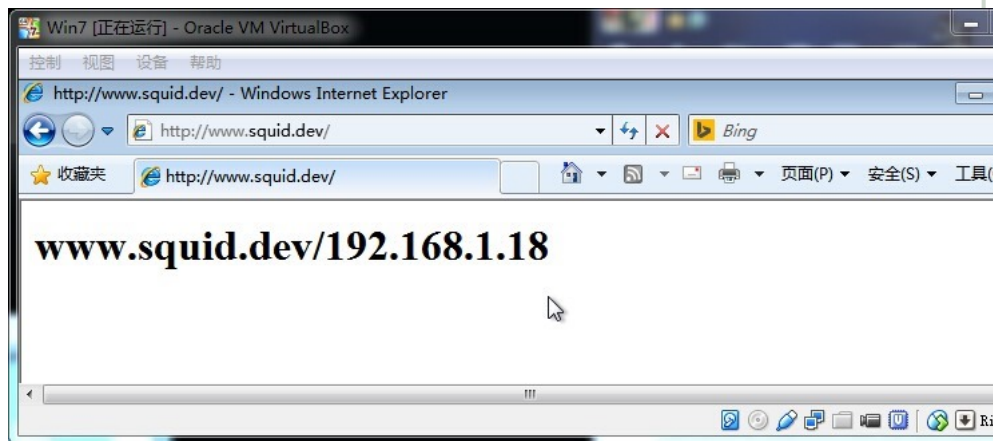
```

# localhost name resolution is handled within DNS itself.
#       127.0.0.1       localhost
#       ::1            localhost
200.168.10.1          www.squid.dev
200.168.10.1          bbs.squid.dev
  
```

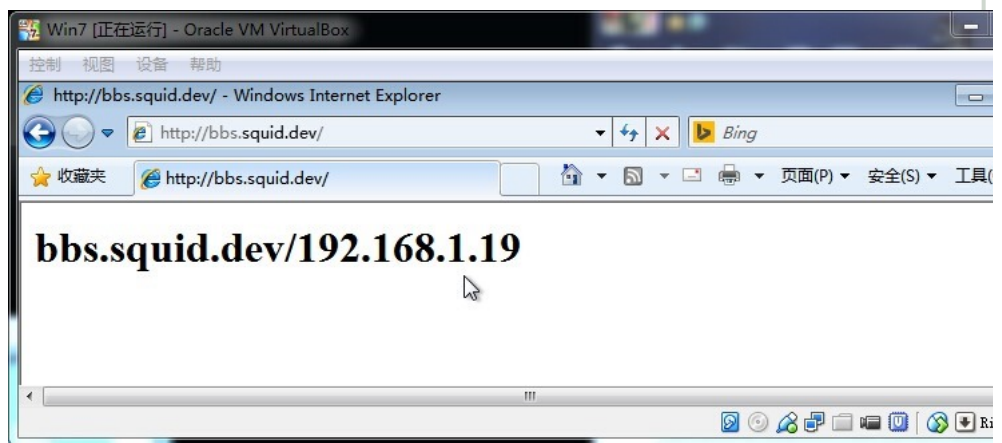
8.4 测试网络情况



8.5 测试www.squid.dev



8.6 测试bbs.squid.dev

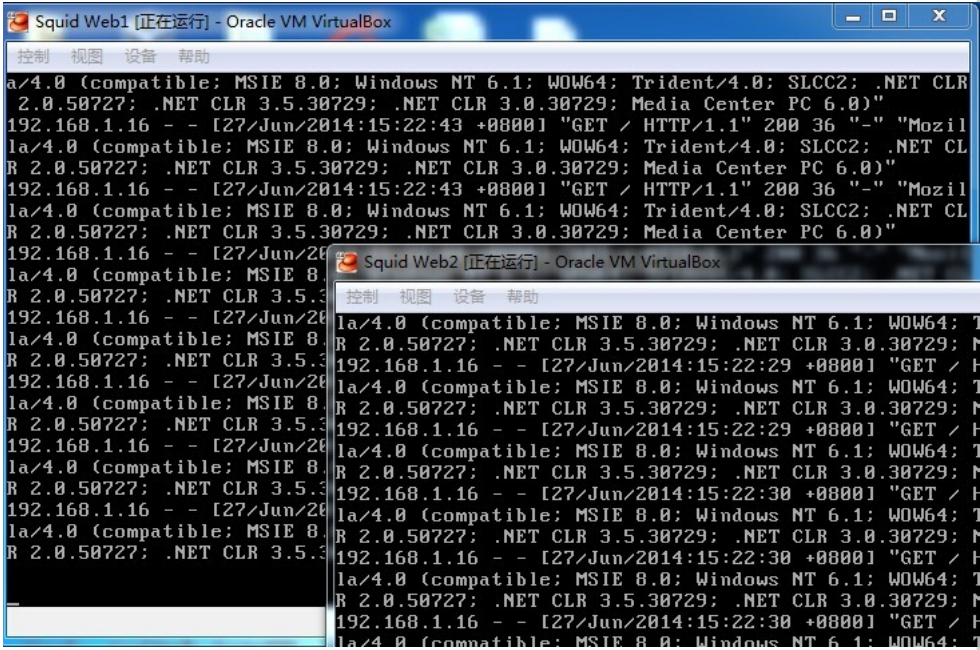


8.7 查看squid 访问日志


```
[root@SquidServer squid]# tailf /var/log/squid/access.log
1403853773.412 3 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 430 GET http://www.squid.dev/ - FIRST_UP
1403853774.608 4 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 430 GET http://www.squid.dev/ - FIRST_UP
1403853776.816 4 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 430 GET http://www.squid.dev/ - FIRST_UP
1403853778.823 0 200.168.10.2 TCP_MEM_HIT/200 437 GET http://www.squid.dev/ - NONE/- text/html
1403853779.261 0 200.168.10.2 TCP_MEM_HIT/200 437 GET http://www.squid.dev/ - NONE/- text/html
1403853785.017 0 200.168.10.2 TCP_MEM_HIT/200 438 GET http://bbs.squid.dev/ - NONE/- text/html
1403853789.076 6 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 430 GET http://bbs.squid.dev/ - FIRST_UP
1403853791.368 4 200.168.10.2 TCP_CLIENT_REFRESH_MISS/200 430 GET http://bbs.squid.dev/ - FIRST_UP
1403853797.694 0 200.168.10.2 TCP_MEM_HIT/200 437 GET http://bbs.squid.dev/ - NONE/- text/html
1403853798.068 0 200.168.10.2 TCP_MEM_HIT/200 437 GET http://bbs.squid.dev/ - NONE/- text/html
```

8.8 查看两台服务器的apache 访问日志

```
# tailf /var/log/httpd/access.log
```



测试成功。

David Camp

- 技术交流，请加QQ群：系统运维技术分享IV：480423035
- 给我写信：mchina_tang@qq.com

我们永远相信，分享是一种美德 | We Believe, Great People Share Knowledge...

分类: [Linux](#), [Linux Server](#)

标签: [Squid](#), [proxy server](#), [代理服务器](#), [反向代理](#), [reverse proxy](#), [代理](#)

[好文置顶](#)[关注我](#)[收藏该文](#)



[David_Tang](#)
关注 - 3
粉丝 - 2040

+加关注

« 上一篇: [CentOS 6.5下利用Rsyslog+LogAnalyzer+MySQL部署日志服务器](#)
» 下一篇: [CentOS 6.3下配置iSCSI网络存储](#)

posted @ 2014-07-03 14:28 David_Tang 阅读(114096) 评论(11) 编辑 收藏

评论列表

- #1楼 2015-03-23 17:45 71standby

文章写得太好了！受益良多！

支持(2) 反对(0)
- #2楼 2015-05-22 11:41 levenlou

很好

支持(0) 反对(0)
- #3楼 2015-07-16 14:27 武仙星团M13

写的很好

支持(0) 反对(0)

#4楼 2015-07-26 15:59 linux_person

请问博主，如何配置使squid同时支持http https透明代理呢？请指教

支持(0) 反对(0)

#5楼 2015-08-13 17:17 fancyjl

请问博主，我按照你的教程想做一个普通代理。
我用的是阿里的云主机，系统为CentOS6.6

在5.1步输入 ifconfig eth1 114.215.83.102 后无法连上云主机了。请问该肿么办？

114.215.83.102是公司的另一台云主机。

在输入之前，我没有输入chkconfig --level 35 squid on 让squid开机启动。请问有影响吗？

谢谢博主！！

支持(0) 反对(0)

#6楼 2015-12-14 12:32 luoshao1

感谢博主，非常好的文章

支持(0) 反对(0)

#7楼 2016-07-13 13:21 alonly

写的非常详细，非常棒，你很牛逼！！

支持(0) 反对(0)

#8楼 2016-08-29 11:35 米米猪

楼主你好，我想问下这个测试环境是如何搭建，全部都用虚拟机吗？

支持(0) 反对(0)

#9楼 2016-10-14 14:03 -Vincent-

楼主问下，每次访问同一个页面，squid 日志都是TCP_MISS，是不是squid是不是没有缓存到呢？怎么解决？

支持(0) 反对(0)

#10楼 2017-10-12 21:56 阳光岛主

非常棒的总结，按照米扑代理的配置步骤，都配置成功了！分享米扑代理的测试方法：

curl 和 wget 测试使用代理，请参考米扑代理的使用示例，总结的非常全面！强烈推荐！

米扑代理的使用示例：<https://proxy.mimvp.com/demo2.php>

这里，直接使用米扑代理的最简单示例：

```
curl -m 30 --retry 3 -x http://58.87.90.149:8888 http://proxy.mimvp.com/exist.php | grep 'color="red"'
<font color="red">58.87.90.149</font>
curl -m 30 --retry 3 -x http://58.87.90.149:8888 -k https://proxy.mimvp.com/exist.php | grep
'color="red"'
<font color="red">58.87.90.149</font>
```

支持(0) 反对(0)

#11楼 2018-04-18 16:35 此时

亲，是否可以转发呢（自己留个备份）
保留原有出处。及作者等所有信息。

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】超50万VC++源码：大型组态工控、电力仿真CAD与GIS源码库！

最新新闻：

- 棋牌外挂年入千万，20秒小视频骗遍宅男，揭秘诱导欺诈
- 摩拜失去了灵魂，现在它又失去了姓名
- 半小时不到，2019支付宝五福已有25700多人集齐

- [苹果自动驾驶项目，黄了吗](#)
- [AI电话营销兴起 你接到的骚扰电话可能不是人打的](#)
- » [更多新闻...](#)