# Linux Squid代理服务器

## 简介

Squid是一个支持HTTP、HTTPS、FTP等服务的Web缓存代理软件，它可以通过缓存页面实现降低带宽占用并优化页面响应时间。此外，Squid还具有请打的访问控制功能。Squid可以运行在各种系统平台上，包括Windows。Squid会将页面缓存在内存及硬盘中，所以Squid对内存及硬盘的要求比较高，更大的内存及硬盘意味着更多的缓存及更高的缓存命中率。Squid确保返回给用户的数据不是过时的数据，所以我们需要根据实际情况不定期地清空缓存数据。

Squid代码请求流程为：为客户端访问Squid代理服务器，由代理服务器代表客户访问后端Real Server（真实服务器），Real Server将响应数据返回给代理服务器，代理服务器将响应数据返回给哭护短，同时将页面缓存在本地内存及硬盘中，当下次客户端再有相同的数据请求时，代理服务器将直接从本地缓存中提取数据返回给客户端。

## 安装Squid服务

Squid在大多数的操作系统平台下都有特定的二进制软件包，采用二进制软件包安装时部署及运行Squid最快速的方式，每种操作系统都有自己特定的软件包管理器，可以使用系统自带的软件包管理器来维护二进制软件包。另外，Squid还提供了源码包，这中方式允许以自定义的方式安装Squid。Squid官网：<http://www.squid-cache.org/Versions/>，提供了所有的二进制数据及源码的下载链接。

**[root@squid ~]# yum -y install squid**

**[root@squid ~]# systemctl start squid**

## 常见的代理服务器类型

代理服务器有多重类型，一般意义上常分为标准正向代理服务器、透明代理服务器、反向代理服务器。

正向代理服务器主要应用于内部网络希望访问外部网络时缓存页面数据，由于目前IP地址稀缺，企业内部成百上千台计算机不可能同事都直接连接到Internet上，目前的解决方案是通过一个统一的网络接口链接Internet。而Squid就可以提供这么一个连接Internet接口。所有的客户端通过设置代理服务器连接至Squid，通过Squid代理上网。在此模型下，Squid主要负责提供缓存加速及访问控制的功能。正向代理是对用户的一种代理操作。

透明代理与传统的正向代理类似，区别在于传统的正向代理需要每个客户端进行代理服务器的设置，而透明代理对用户是同名的，不需要终端用户进行特殊设置。透明代理需要结合网关进行部署，所有的操作均由管理员在网关服务器及代理服务器进行设置，这些对用户是透明的。

反向代理结合智能DNS即可实现基本的CND（内容分发网络）框架。通过DNS的视图功能，我们可以为来自不同地区的DNS请求解析到不同的结果返回给客户端。例如：当北京地区的客户请求DNS解析，DNS服务器将解析的结果指向北京地址的Squid代理服务器，上海地区的用户请求DNS解析的结果则是上海地区的Squid代理服务器。而此时的代理服务器则代表真实的后端服务器，用户冠绝不到自己是在访问代理服务器，而且由于使用了缓存技术，当大的并发访问量到达时，Squid也可以更快的为用户返回响应数据。

Squid配置文件解析

使用Yum/rpm方式安装Squid时，默认的配置文件为/etc/squid.suqid.conf。该配置文件由配置指令及配置选项构成，配置文件使用#符号作为注释。

配置项 描述

**http\_port** 设置Squid精简同的网络端口，默认为3128端口，使用低于1024

端口时，需要使用root启动Squid,ISP一般使用 8080端口。

vhost参数实现方向代理功能。

**cache\_dir** 设置缓存存储位置及大小，配置文件中可以使用多个cache\_dir

设置多个存储位置。

cache\_dir ufs /usr/local/squid/var/cache 100 16 256

缓存路径为/usr/local/squid/var/cache，100代表容量，单位MB，

16及256代表子目录个数，用于分类分层管理缓存数据。

**access\_log** 设置日志路径及日志格式。

如 access\_log /var/log/squid/access.log combined

**acl** 定义访问控制列表（ACL）对象，可以控制源地址、端口、时间

等，如 acl localent src 192.168.0.0/255.255.255.0，设置名称为

localnet 的 ACL 列表，控制对象为源地址网段192.168.0.0/24

**http\_access** 访问控制，允许使用ACL指令定义的localnet效果如下：

http\_access\_allow locanet

**cache\_peer** Squid支持在一组代理之间进行缓存请求转发，组内的成员代理

之间称为邻居，邻居关系分为父子和兄弟。

cache\_peer用来设置可以连接的其他代理服务器，其格式如下：

cache\_peer hostname type http-port icp-port [option]

hostname 为邻居的主机名或IP；

type类型为父亲(parent) 、兄弟(sibling) 、广播(multicast)；

http-port 指定邻居http端口号；

icp-port 指定ICP端口号；

option（选项）有以下可选值：

proxy-only 实现不缓存来自邻居的任何响应

weight=n 设置服务器权重

ttl=n 该选项对广播邻居有效，设置广播TTL值

round-robin 仅在设置两个以上父Cache才有效，可以实现轮询

的负载均衡

max-conn=n 设置连接邻居Cache的并发量

originserver 设置邻居为源代理服务器，做反向代理使用

**visible\_hostname** 设置主机名，默认Squid将自动检测系统主机名

## Squid应用案例

配置Squid实现反向代理，结构如下：

**|-----------------------------------------------------------------------------|**

**| |**

**| |----------web1 |**

**client-------Internet-------|--------Squid-------转换器------------| |**

**| |----------web2 |**

**| |**

**|-----------------------------------------------------------------------------|**

终端用户请求Web服务时，DNS将请求的域名解析为反向代理服务器的IP地址，代理服务器首先检查本地是否有缓存，如果有则直接将相应数据包返回给客户端，否则代理服务器转发客户端请求给后端真实源服务器。

服务器网络配置值如下：

服务器名称 网络配置

squid.example eno16777736:10.10.10.10

eno33554984:192.168.0.254

web1.example.com eno16777736:192.168.0.1

web2.example.com eno16777736:192.168.0.2

client.example.com eno16777736.10.10.10.100

Squid一般仅缓存静态页面，比如HTML网页，图片，歌曲等。默认不缓存CGI脚本程序或者PHP、JSP。Squid可以根据HTTP头信息来决定如何缓存数据，以及缓存周期，如Expires(缓存有效期)、Cache-Control(是否进行缓冲)。

### 1，代理服务器设置

首先，需要配置网络环境参数。

**[root@squid ~]# vim /etc/sysconfig/network-script/ifcfg-eno16777736**

**DEVICE="eno16777736"**

**BOOTPROTO="static"**

**ONBOOT="yes"**

**IPADDR=10.10.10.10**

**PREFIX=8**

**TYPE="Ethernet"**

**[root@squid ~]# vim /etc/sysconfig/network-script/ifcfg-eno33554984**

**DEVICE="eno33554984"**

**BOOTPROTO="static"**

**ONBOOT="yes"**

**IPADDR=192.168.0.254**

**PREFIX=24**

**TYPE="Ethernet"**

**[root@squid ~]# systemctl restart network**

其次，还需要安装部署Squid代理服务。

**[root@squid ~]# yum -y install squid**

**[root@squid ~]# vim /etc/squid/squid.conf**

# 定义缓存管理ACL

acl manager proto cache\_object

# 定义源地址为本地回环地址的ACL

acl locatlhost src 127.0.0.1/32 ::1

# 定义源地址为10.0.0.0/8的ACL

acl locanet dst 127.0.0.0/8 0.0.0.0/32 ::1

# 定义源地址为10.0.0.0/8的ACL

acl localnet src 10.0.0.0/8 # RFC1918 possible internal network

# 定义源地址ACL

acl localnet src 172.16.0.0/12 # RFC1918 possible internal network

acl localnet src 192.168.0.0/16 # RFC1918 possible internal network

acl localnet src fc00::/7 # RFC 4193 local private network range

acl localnet src fe80::/10 # RFC 4291 link-local (directly plugged) machines

# 定义安全端口为443的ACL

acl SSL\_ports port 443

# 定义安全端口为80的ACL

acl Safe\_ports port 80 # http

# 定义安全端口为21的ACL

acl Safe\_ports port 21 # ftp

acl Safe\_ports port 443 # https

acl Safe\_ports port 70 # gopher

acl Safe\_ports port 210 # wais

acl Safe\_ports port 1025-65535 # unregistered ports

acl Safe\_ports port 280 # http-mgmt

acl Safe\_ports port 488 # gss-http

acl Safe\_ports port 591 # filemaker

acl Safe\_ports port 777 # multiling http

# 定义连接方式为CONNECT 的 ACL

acl CONNECT method CONNECT

# 拒绝所有非Safe\_ports的连接，Safe\_ports为配置文件中所有定义的若干端口

http\_access deny !Safe\_ports

http\_access deny CONNECT !SSL\_ports

# 仅允许本机进行缓存管理

http\_access allow localhost manager

# 拒绝其他主机的所有缓存管理

http\_access deny manager

http\_access allow localnet

http\_access allow localhost

# 允许所有

http\_access allow all

# 设置主机名称

visible\_hostname squid.example.com

# 内存缓存总容量

cache\_mem 1024 MB

# 内存可以缓存的单个文件最大容量为4MB

maximum\_object\_size\_in\_memory 4096 KB

# 磁盘可以缓存的但个文件最大容量为 4MB

maximum\_object\_size 4096 KB

# cache\_dir定义硬盘缓冲目录为/var/spool/squid，缓存最大容量为800MB，Squid将在缓存目录下

# 创建16个一级子目录和256个二级子目录，这些目录用来分类管理缓存数据

cache\_dir ufs /var/spool/squid 800 16 256

# 定义错误报告文件的存放目录

error\_directory /usr/share/squid/errors/zh-cn

# 缓存日志文件

cache\_log /var/log/squid/cache.log

# 管理员邮箱

cache\_mgr admin@test.com

# 代理服务器监听的端口，accel设置squid为加速模式，vhost实现反向代理

http\_port 80 accel vhost

# Squid normally listens to port 3128

# 同时监听代理的3128端口，方便使用squidclient工具对缓存进行管理

http\_port 3128

# Uncomment and adjust the following to add a disk cache directory.

#cache\_dir ufs /var/spool/squid 100 16 256

# 配置后台源服务器，originserver模拟本纪服务器为源服务器；80为HTTP端口；

# 设置ICP端口为0（源服务器不支持ICP查询）；no-query表示禁止使用ICP对源服务器进行查询；

# round-robin可以让代理轮询多台源服务器，可以使用weight=N为源服务器指定权重；

# name用来设置源服务器的唯一名称，代理转发数据至相同主机的不同端口非常有用

cache\_peer 192.168.0.108 parent 80 0 no-query originserver round-robin name=server1

cache\_peer 192.168.0.112 parent 80 0 no-query originserver round-robin name=server2

# 禁止缓存CGI脚本

hierarchy\_stoplist cgi-bin ?

# refersh\_pattern 应用在没有过期时间的数据，squid顺序检查refresh\_pattern并确定数据是否过期，格式如下：

# refersh\_pattern [-i] regexp min percent max [option]

# refersh\_pattern 使用正则表达是匹配数据对象（-i选项表示不区分大小写）

# min 为过期的最低时间（单位为分钟），即对象在缓存中至少要存放的时间

# max 为最大时间

# percent 为百分比，如果squid最后修改系数算法低于该百分比则数据不过期

refresh\_pattern ^ftp: 1440 20% 10080

refresh\_pattern ^gopher: 1440 0% 1440

refresh\_pattern -i (/cgi-bin/|\?) 0 0% 0

refresh\_pattern . 0 20% 4320

**[root@squid ~]# firewall-cmd --set-default-zone=trusted**

**[root@squid ~]# systemctl restart squid**

**[root@squid ~]# systemctl enable squid**

### 2，两台源服务器设置

设置正确的网络参数，安装并启动httpd服务。

**[root@web1 ~]# vim /etc/sysconfig/network-scripts/ifcfg-eno16777736**

**DEVICE="eno16777736"**

**BOOTPROTO="static"**

**ONBOOT="yes"**

**IPADDR=192.168.0.1**

**PREFIX=24**

**TYPE="Ethernet"**

**[root@web1 ~]# firewall-cmd --set-default-zone=trusted**

**[root@web1 ~]# systemctl restart network**

**[root@web1 ~]# yum -y install httpd**

**[root@web1 ~]# echo "192.168.0.1" > /var/www/html/index.html**

**[root@web1 ~]# systemctl restart httpd**

**[root@web1 ~]# systemctl enable httpd**

**[root@web2 ~]# vim /etc/sysconfig/network-scripts/ifcfg-eno16777736**

**DEVICE="eno16777736"**

**BOOTPROTO="static"**

**ONBOOT="yes"**

**IPADDR=192.168.0.2**

**PREFIX=24**

**TYPE="Ethernet"**

**[root@web2 ~]# systemctl restart network**

**[root@web2 ~]# firewall-cmd --set-default-zone=trusted**

**[root@web2 ~]# yum -y install httpd**

**[root@web2 ~]# echo "192.168.0.2" > /var/www/html/index.html**

**[root@web2 ~]# systemctl restart httpd**

**[root@web2 ~]# systemctl enable httpd**

### 3，客户端验证

客户端配置正确的网络参数后，通过浏览器访问代理服务器：http://10.10.10.10。由于我们在代理服务器中设置了两台源服务器，并且使用的是轮询查询，所有客户端刷新后可以轮询查看到两个不同的页面。