



链滴

nginx 实现对 HTTP 和 TCP 的反向代理

作者: [leekeggs](#)

原文链接: <https://ld246.com/article/1544358385695>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Nginx从1.9.0开始发布ngx_stream_core_module模块，该模块支持tcp代理及负载均衡。ngx_stream_core_module模块默认不启用，需要在编译时通过指定--with-stream参数来激活这个模块。

1. 编译安装Nginx

下载地址：<https://nginx.org/download/>。

在编译安装nginx之前，要确保nginx依赖的一些组件已经安装了。

1.1 检查安装zlib

```
rpm -qa | grep zlib
```

如果未安装，可使用yum安装

```
yum install zlib zlib-devel -y
```

1.2 检查安装openssl

```
rpm -qa | grep openssl
```

如果未安装，可使用yum安装

```
yum install openssl-devel openssl -y
```

1.3 检查安装pcre

```
rpm -qa | grep pcre
```

如果未安装，可使用yum安装

```
yum install pcre pcre-devel -y
```

1.4 安装nginx

```
$ tar -zxvf nginx-1.9.15.tar.gz
$ ./configure --prefix=/opt/nginx/ \
--with-http_stub_status_module \
--with-http_ssl_module \
--with-pcre \
--with-stream
$ make
$ make install
```

编译参数说明

--prefix 指定nginx安装路径

--with-http_ssl_module 用于构建一个支持https协议的模块，需要OpenSSL库，该模块默认不启用

-with-pcre 强制使用PCRE库

--with-stream 构建一个支持TCP/UDP代理和负载均衡的模块，该模块默认不启用

更多编译参数说明参考[nginx编译参数详解](#)

2. 配置http代理

假设有3台主机node1、node2和node3，node1和node2上部署tomcat，node3上部署nginx，如：

主机名	IP	服务	安装路径
node1 opt/tomcat	192.168.1.101	tomcat	
node2 opt/tomcat	192.168.1.102	tomcat	
node3 opt/nginx	192.168.1.103	nginx	

在node3上修改nginx配置，`vim nginx.conf`，内容如下

```
#user nobody;
worker_processes 1;

#error_log logs/error.log;
#error_log logs/error.log notice;
#error_log logs/error.log info;

#pid logs/nginx.pid;

events {
    worker_connections 1024;
}

http{
    upstream backend{
        server 192.168.1.101:8080 weight=7 max_fails=3 fail_timeout=5s;
        server 192.168.1.102:8080 weight=3 max_fails=3 fail_timeout=5s;
    }
    server{
        listen 8001;
        location /{
            proxy_pass http://backend;
        }
    }
}
```

配置文件结构说明参考[nginx.conf详解](#)

其中`upstream`用于配置负载均衡，这里配置了两个服务器做负载均衡，发往192.168.1.103:8001的请求会被转发到node1和node2的8080端口。

`upstream`负责均衡策略有：轮询（默认），权重`weight`，`hash`和`ip_hash`。上面的配置文件中采用权重策略。关于`upstream`的详细说明参考[upstream详解](#)

node1和node2上启动tomcat，node3上启动nginx

/opt/nginx/sbin/nginx

使用浏览器或其它方式访问<http://192.168.1.103> ,观察tomcat的访问日志localhost_access_log, 可看到每次请求都会被转发到node1和node2的其中一台上。

3. 配置TCP代理

修改node3上的nginx配置文件, `vim nginx.conf`, 增加如下内容

```
stream{
  upstream ssh {
    hash $remote_addr consistent;
    server 192.168.1.101:22 weight=5;
  }

  server{
    listen 2233;
    proxy_pass ssh;
  }
}
```

`stream`块用于配置TCP代理, 在`upstream`中使用了`hash`作为负责均衡策略, `consistent`参数的作用: 指定使用ketama一致性哈希方法, 该方法确保在向组中添加服务器或从组中删除服务器时, 只有少数密钥将重新映射到不同的服务器。这有助于为缓存服务器实现更高的缓存命中率。

重新加载nginx配置文件

/opt/nginx/sbin/nginx -s reload

我们来测试一下, 使用ssh连接node3, `-p`选项指定端口2233, 如下:

```
ssh -p 2233 wxyuan@node3
```

最后你会看到, 实际连接的主机是node1