# 说明

**反向代理：指的是代理外网用户的请求到内部的指定web服务器**

**正向代理：没有隐藏用户请求**

# 安装位置

# 参数说明

## 变量

$query\_string ： 查询字符串

## 命令

**root** : 指定文件所在的位置

**expires max** 有效期最大

**proxy\_pass** ：反向代理

proxy\_pass <http://www.baidu.com>

proxy\_pass 127.0.0.1:8000

**proxy\_pass\_header**:传递请求头信息

proxy\_pass\_header Server //Server:服务器信息

**proxy\_set\_header**:设置头信息

1. proxy\_set\_header x-Real-IP $remote\_addr;传送用户真实IP
2. proxy\_set\_header x-Scheme $scheme ;用户请求的协议

**upstream**:组

upstream tornadoes {

server 127.0.0.1:8000

server 127.0.0.1:8001

server 127.0.0.1:8002

}

# 实例

## 负载均衡

1. 配置组

upstream tornadoes { // tornadoes是自定义名称

server 127.0.0.1:8000

server 127.0.0.1:8001

server 127.0.0.1:8002

}

1. 配置location

location /cat { //cat路径是自定义的

proxy\_pass <http://tornadoes> //这是最主要命令

}

## 响应websocket

1. 配置组（这不是必须的)

upstream websocket {

server 127.0.0.1:8000

}

1. 配置locateion

location /chat {

// websocket指定是nginx中的一个组

proxy\_pass [http://websocket/chat](http://websocket/chat;)

proxy\_http\_version 1.1

proxy\_set\_header Upgrade $http\_upgrade // 升级命令

proxy\_set\_header connection “upgrade” //升级命令

}

## 反向代理