

# nginx负载均衡五种策略模式

nginx可以根据客户端IP进行负载均衡，在upstream里设置ip\_hash，就可以针对同一个C类地址段中的客户端选择同一个后端服务器，除非那个后端服务器宕了才会换一个。

nginx的upstream目前支持的5种方式的分配

## 1、轮询（默认）

每个请求按时间顺序逐一分配到不同的后端服务器，如果后端服务器down掉，能自动剔除。

```
upstream backserver {  
server 192.168.0.14;  
server 192.168.0.15;  
}
```

## 2、指定权重

指定轮询几率，weight和访问比率成正比，用于后端服务器性能不均的情况。

```
upstream backserver {  
server 192.168.0.14 weight=10;  
server 192.168.0.15 weight=10;  
}
```

## 3、IP绑定 ip\_hash

每个请求按访问ip的hash结果分配，这样每个访客固定访问一个后端服务器，可以解决session的问题。

```
upstream backserver {  
ip_hash;  
server 192.168.0.14:88;  
server 192.168.0.15:80;  
}
```

## 4、fair（第三方）

按后端服务器的响应时间来分配请求，响应时间短的优先分配。

```
upstream backserver {  
server server1;  
server server2;  
fair;  
}
```

## 5、url\_hash（第三方）

按访问url的hash结果来分配请求，使每个url定向到同一个后端服务器，后端服务器为缓存时比较有效。

```
upstream backserver {  
server squid1:3128;  
server squid2:3128;  
hash $request_uri;  
hash_method crc32;  
}
```

在需要使用负载均衡的server中增加

```
proxy_pass http://backserver/;  
upstream backserver{  
ip_hash;  
server 127.0.0.1:9090 down; (down 表示单前的server暂时不参与负载)  
server 127.0.0.1:8080 weight=2; (weight 默认为1.weight越大，负载的权重就越大)  
server 127.0.0.1:6060;  
server 127.0.0.1:7070 backup; (其它所有的非backup机器down或者忙的时候，请求backup机器)  
}
```

`max_fails` : 允许请求失败的次数默认为1. 当超过最大次数时, 返回`proxy_next_upstream` 模块定义的错误

`fail_timeout: max_fails`次失败后, 暂停的时间