基于Nginx的负载均衡实验

1. 实验环境

首先需安装Nginx

安装完成后运行，浏览器中输入localhost，若弹出Nginx欢迎界面则安装成功。如图

一張含有 文字 的圖片

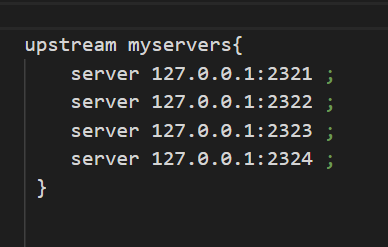
自動產生的描述

在nginx目录下打开conf文件，开启nginx.conf文件，为实现负载均衡，对其部份代码进行如下修改

1、修改location后反向代理路径



2、使用了127.0.0.1：2321 、 127.0.0.1：2322 、127.0.0.1：2323、127.0.0.1：2324四个IP做为服务器



3、配置并运行js文件

js文件创建如下

webserver1

一張含有 文字 的圖片

自動產生的描述

webserver2



同理设置webserver3和webserver4

4、安装node.js软件

在js文件目录下打开cmd，输入node webserver1.js运行http服务器，效果如下

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

也可通过浏览器验证运行效果

一張含有 文字 的圖片

自動產生的描述一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述一張含有 文字 的圖片

自動產生的描述

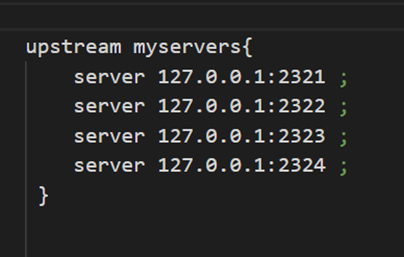
确定HTTP服务器已经启动完成后，开始进行负载均衡实验

二、负载均衡策略

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

nginx.conf文件修改如下



在cmd窗口下执行:nginx -s reload进行重载

输入curl localhost:80 进行测试

一張含有 文字 的圖片

自動產生的描述

可见，以 S1,S2,S3,S4,S1, S2,...的顺序依次进行轮询。

1. 指定权重

指定权重weight，weight和访问比率成正比，用于后端服务器性能不均的情况。

此策略将nginx.conf文件修改如下

一張含有 文字 的圖片

自動產生的描述

重载后进行测试

结果如下

一張含有 文字 的圖片

自動產生的描述

可见，服务器给定权重越大，访问到的几率越大。

1. 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。

修改代码如下

一張含有 文字 的圖片

自動產生的描述

测试结果

