

nginx知识回顾

- 1.nginx是什么和作用
- 2.nginx的配置技巧
- 3.nginx的高可用集群
- 4.nginx进程模型和worker的抢占机制
- 5.nginx并发量计算
- 6.nginx默认首页解析过程
- 7.nginx事件处理
- 8.nginx配置文件迷糊点
- 9.nginx配置花式玩法
- 10.配置多台server虚拟主机外部文件导入
- 11.nginx中的nginx.pid失效异常解决
- 12.nginx中的一些常用命令技巧
- 13.域名解析和Nginx的跨域问题解决

1.nginx是什么和作用

nginx是一个高性能处理http请求的反向代理服务器,具有很强的并发能力;它的主要作用有反向代理,负载均衡,动静分离;

- 反向代理: 反向代理就是客户端在访问服务端的时候,并不知道服务端的真实ip地址,通过虚拟ip访问到真实服务器;与之对应的还有正向代理,正向代理就是客户端知道服务端的ip地址;
- 负载均衡: 客户端的大量请求通过nginx被不同的负载均衡算法发送到不同的服务器上,轮询,权重,ip_hash,时间等;
- 动静分离: 值得是我们的静态资源原本在tomcat目录下,用户发送一个请求需要加载很多静态资源导致访问tomcat服务器的压力过大,而使用nginx式当,用户请求静态资源,可由nginx直接响应给用户;

2.nginx的配置技巧

1. nginx安装完成后的位置:

- bin目录: /usr/local/nginx/sbin
- 配置目录: /usr/local/nginx/nginx.conf

2. nginx.conf

- 精准匹配:意思就是说,你访问了80-----我直接就转发到proxy_pass中

▼ 精准匹配

Nginx | 复制代码

```
1 //全局块
2 //events块
3 http块
4 server{
5     listen 80;
6     server_name ip;
7     location /{
8         proxy_pass http://127.0.0.1:8080;
9     }
10 }
```

- 正则匹配: 这种匹配的意思是我只替换了ip:端口

▼ 正则匹配

Nginx | 复制代码

```
1 localhost ~ /edu/{
2     proxy_pass:http://127.0.0.1:8080;
3 }
4 //举例
5 http://虚拟机ip:80/edu/index.html
6 替换成---http://127.0.0.1:80/edu/index.html
7 //规则如此, 重要的是正则规则
8 location [=|~|~*|^~|] /edu/{
9 }
10 =代表使用正则, 要严格匹配      ~正则区分大小写, ~*不区分大小写
```

- 动静分离

▼ 动静分离

Nginx | 复制代码

```
1 localhost /www/{
2     root /data;           //如果你访问的是http://ip:80/www/a.jpg----->默认拼
                             劲/data/www/a.jpg
3     autoindex on;         //代表可以访问 http://ip:80/www/  下载
4 }
```

3.nginx的高可用集群

看文档搭建即可

4.nginx进程模型和worker的抢占机制

nginx在工作的时候一般有两个进程，master主进程和worker从进程,当然我说的是进程，不是线程,所有请求到达nginx,master向worker发送信号，worker就去争抢这些请求去处理，一个master多个worker的优势，当我们对nginx进行热部署的时候，并不是所有的worker都要去更新，只有没有争抢到请求的才去热更新,有请求处理任务的先去处理，等处理结束后，再去更新，这样就可以24小时不间断工作，当有一个worker挂掉了，我们的nginx还可以正常工作.

5.nginx并发量计算

我的配置全局块中work_connections连接数1024,worker_processor几个工人4,一个master,求支持的最大请求数

计算

JSON | 复制代码

```
1 解:
2 如果请求静态资源: 占用2个连接
3 如果请求动态资源: 占用4个连接
4 一个工人最大连接数为1024
5 1024/2*4:代表的最大静态资源请求数
6 1024/4*4:代表的最大动态资源请求数
```

6.nginx默认首页解析过程

浏览器输入 http://ip:port/-----nginx服务器-----找到某一个server去找对应的html

```
server {
    listen      88;
    server_name localhost;

    location / {
        root    html;
        index   index.html index.htm;
    }
    error_page  500 502 503 504  /50x.html;
    location = /50x.html {
        root    html;
    }
}
```

可以是内网ip,也可以是公网ip, 还有你配置好的域名

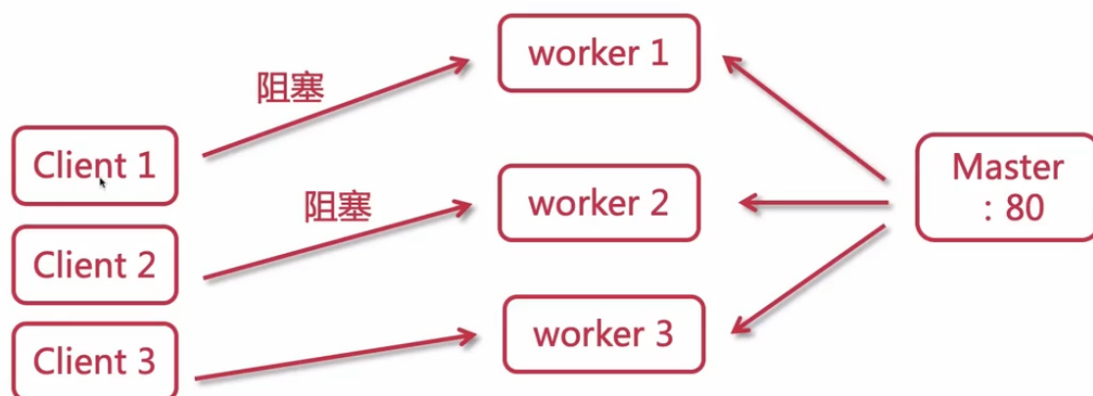
没有加/---代表的是相对目录, 即和conf的同级目录

默认的面配置

精准匹配: 相对目录下去找

7.nginx事件处理

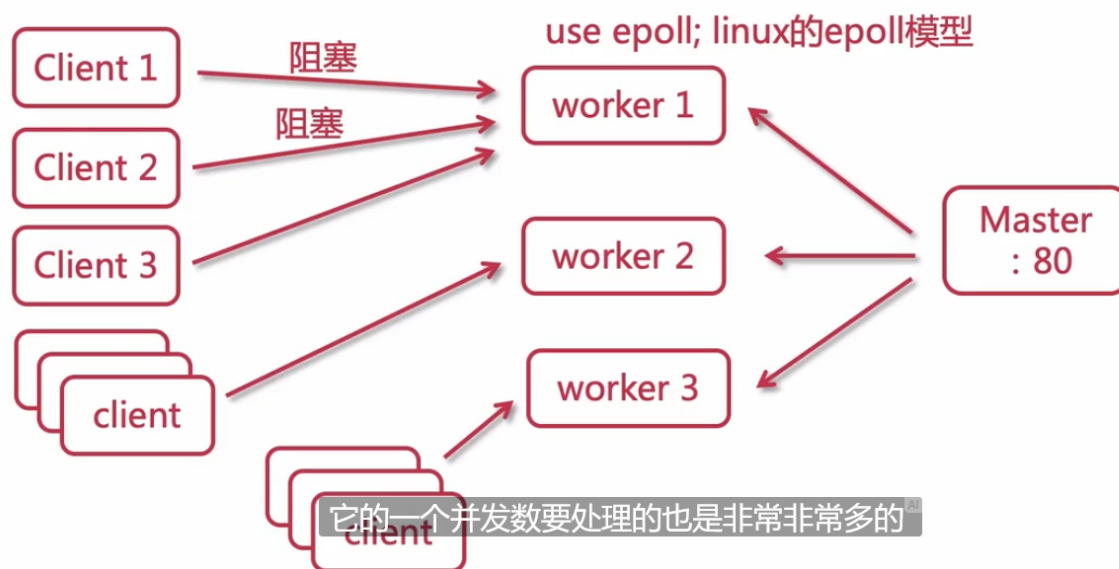
- 传统的服务器事件处理



传统的服务器事件处理采用了阻塞io的模型去处理事件，当我们开启一个worker进程之后，worker去争抢到client事件要经过，处理响应等过程，在某一过程中阻塞了，这个worker就不能再去处理其他的client事件,所以很不高效

- nginx的事件处理

Nginx事件处理



nginx采用了nio模型的事件异步通讯去处理,epoll模型，我们大多数都在linux系统搭建nginx，nginx在linux系统默认使用这种处理方式，在不同的系统处理方式不同，当然我们可以修改。很高效的原因

8.nginx配置文件迷糊点

```

user root;    创建worker进程的用户,可以改为root
worker_processes 2;    几个worker进程, 建议根据cpu算

# debug info notice warn error crit
#error_log logs/error.log;
#error_log logs/error.log notice;    日志级别,位置查看
#error_log logs/error.log info;

#pid logs/nginx.pid;

```

```

./configure \
--prefix=/usr/local/nginx \    安装目录
--pid-path=/var/run/nginx/nginx.pid \
--lock-path=/var/lock/nginx.lock \
--error-log-path=/var/log/nginx/error.log \    默认位置
--http-log-path=/var/log/nginx/access.log \
--with-http_gzip_static_module \
--http-client-body-temp-path=/var/temp/nginx/cli
--http-proxy-temp-path=/var/temp/nginx/proxy \
--http-fastcgi-temp-path=/var/temp/nginx/fastcgi
--http-uwsgi-temp-path=/var/temp/nginx/uwsgi \
--http-scgi-temp-path=/var/temp/nginx/scgi

```

```

events {
    # 默认使用epoll
    use epoll;
    # 每个worker允许连接的客户端最大连接数
    worker_connections 10240;
}

```

```

sendfile on;
#tcp_nopush on;    开启数据包发送, 这两个同时开启可以提高性能

gzip on    开启这个可以让静态资源打包
            发送, 发送更快, 但会消耗系统资源

#keepalive_timeout 0;
keepalive_timeout 65;    让客户保持连接多少秒

```

9.nginx配置花式玩法

```
server {
    listen      89;
    server_name localhost;

    location / {
        root    html;
        index   imooc.html index.htm;
    }
}

server {
    listen      91;
    server_name localhost;

    #精确匹配
    location = / {
        root    html;
        index   imooc.html index.htm;
    }
}
```

6-19 location 的匹配规则

- 空格：默认匹配，普通匹配

```
location / {
    root /home;
}
```

- 精确匹配

```
location = /imooc/img/face1.png {
    root /home;
}
```

- 匹配正则表达式，不区分大小写

```
#符合图片的显示
location ~ \.(GIF|jpg|png|jpeg) {
    root /home;
}
```

- 匹配正则表达式，区分大小写

```
#GIF必须大写才能匹配到
location ~ \.(GIF|jpg|png|jpeg) {
    root /home;
}
```

- ~: 以某个字符路径开头

```
location ~ /imooc/img {
    root /home;
}
```

10.配置多台server虚拟主机外部文件导入

include abc.conf-----很好玩

想要了解更多玩法 go!go!go!

```
include imooc.conf;
```

这里没加/代表同级目录,conf文件
夹下
导入外部配置

11.nginx中的nginx.pid失效异常解决

```
-rwxr-xr-x. 1 root root 3857192 9月 21 22:07 nginx
[root@centos7-basic sbin]# ./nginx -s reload
nginx: [error] open() "/var/run/nginx/nginx.pid" failed (2: No such file or directory)
[root@centos7-basic sbin]# cd
```

```
[root@centos7-basic sbin]# ./nginx -s reload
nginx: [error] open() "/var/run/nginx/nginx.pid" failed (2: No such file or directory)
[root@centos7-basic sbin]# cd /var/run/nginx
-bash: cd: /var/run/nginx: 没有那个文件或目录
[root@centos7-basic sbin]# mkdir /var/run/nginx ① 创建这个文件
[root@centos7-basic sbin]# cd /var/run/nginx
[root@centos7-basic nginx]# cd /usr/local/nginx/
[root@centos7-basic nginx]# cd sbin/
[root@centos7-basic sbin]# ./nginx -t
nginx: the configuration file /usr/local/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/nginx/conf/nginx.conf test is successful
[root@centos7-basic sbin]# ./nginx -s reload ② 重新加载
nginx: [error] invalid PID number "" in "/var/run/nginx/nginx.pid"
[root@centos7-basic sbin]# ./
```

```

nginx: [error] invalid PID number "" in "/var/run/nginx/nginx.pid"
[root@centos7-basic sbin]# ./nginx -h 1 查看nginx配置命令
nginx version: nginx/1.16.1
Usage: nginx [-?hvVtTq] [-s signal] [-c filename] [-p prefix] [-g directives]

Options:
  -?, -h      : this help
  -v          : show version and exit
  -V          : show version and configure options then exit
  -t          : test configuration and exit
  -T          : test configuration, dump it and exit
  -q          : suppress non-error messages during configuration testing
  -s signal   : send signal to a master process: stop, quit, reopen, reload
  -p prefix   : set prefix path (default: /usr/local/nginx/)
  -c filename : set configuration file (default: conf/nginx.conf)
  -g directives : set global directives out of configuration file

[root@centos7-basic sbin]# ./nginx -c /usr/local/nginx/conf/nginx.conf 2 重新指定nginx配置
在reload即可

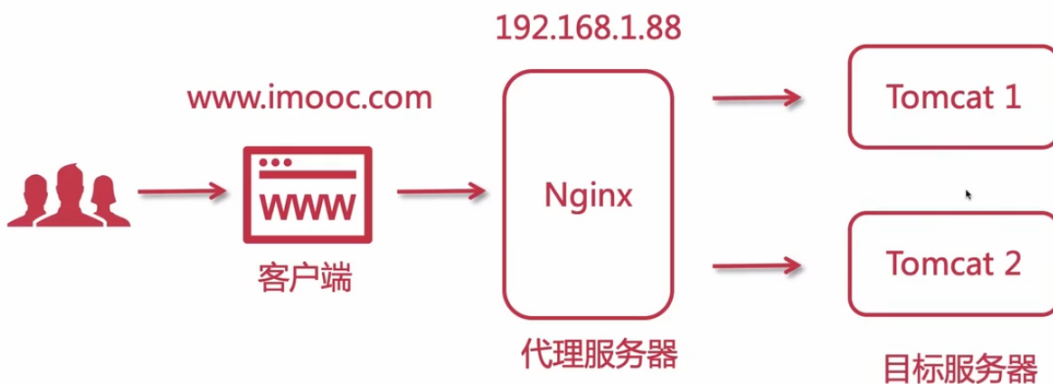
```

12.nginx中的一些常用命令技巧

- nginx -t ---加载配置之前查看文件是否有错误
- nginx -v ----查看nginx版本号,最关键的是可以找到许多文件的安装位置
- nginx -? -----nginx的一些帮助命令

13.域名解析和nginx的跨域问题解决

1. 域名解析



无论什么系统都可以设值: 它的hosts文件配置域名: `[root@centos7-basic conf]# vim /etc/hosts`

2. 跨域问题:


```
server {  
    listen      90;  
    server_name localhost;  
  
    #允许跨域请求的域，*代表所有  
    add_header 'Access-Control-Allow-Origin' *;  
    #允许带上 cookie请求  
    add_header 'Access-Control-Allow-Credentials' 'true';  
    #允许请求的方法，比如 GET/POST/PUT/DELETE  
    add_header 'Access-Control-Allow-Methods' *;  
    #允许请求的 header  
    add_header 'Access-Control-Allow-Headers' *;
```

nginx.conf中配置

总结：笔记缺乏之处：正则匹配的方式玩的不是很溜，欢迎大佬积极展示自己的正则匹配技巧，我也想学习一下.