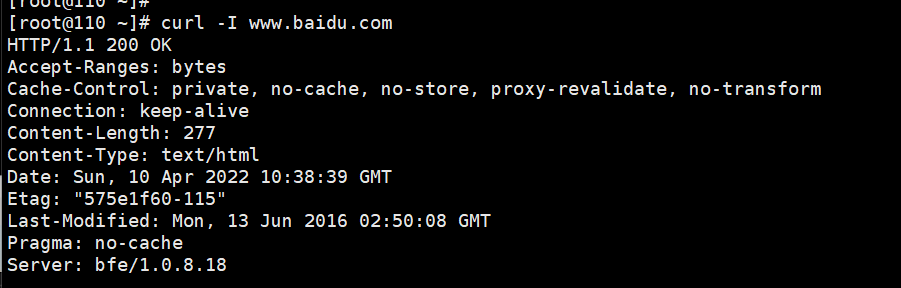
nginx企业级优化

一、配置nginx隐藏版本号

curl -I http://192.168.200.100

-I参数则只显示 http response 的头信息



1.修改源码包

安装依赖

pcre-devel

zlib-devel

openssl-devel

gcc

gcc-c++

make

1.创建用户

useradd -M -s /sbin/nologin nginx

2.修改安装包的源码文件

cd /usr/nginx-1.14.2/

vim src/core/nginx.h

3.配置解压安装

./configure --prefix=/usr/local/nginx --user=nginx --group=nginx && make&&make install

/usr/local/nginx/sbin/nginx

解压，编译安装

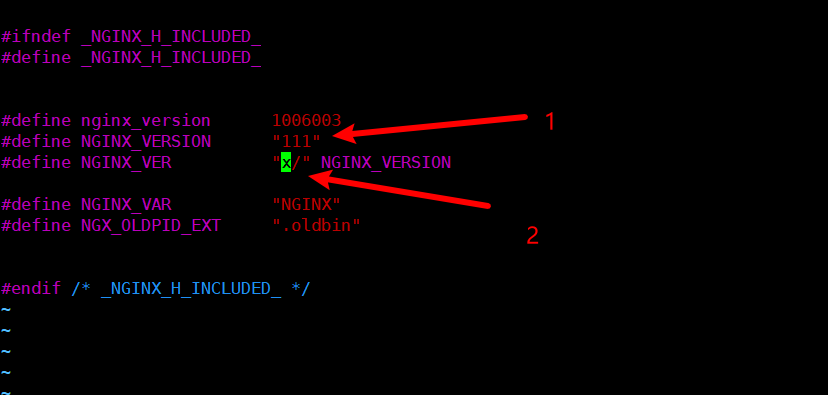
启动nginx

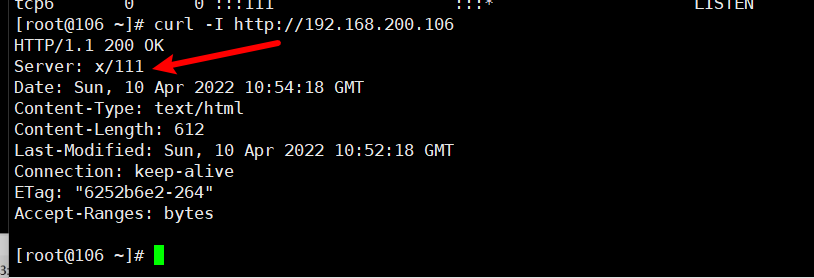
/usr/local/nginx/sbin/nginx

netstat -lnpt

4.测试

curl -I http://192.168.200.106



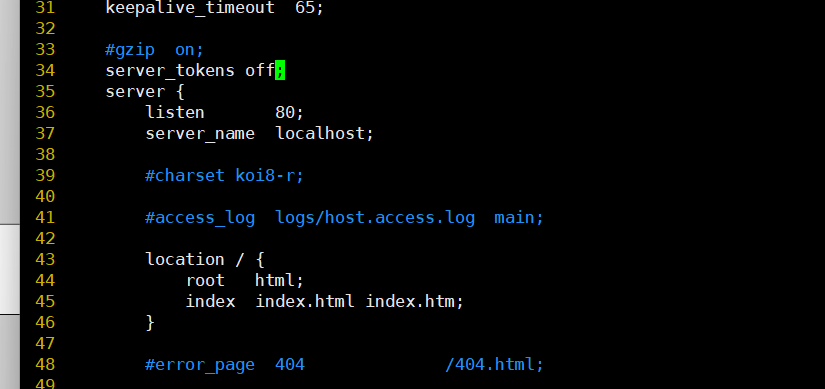


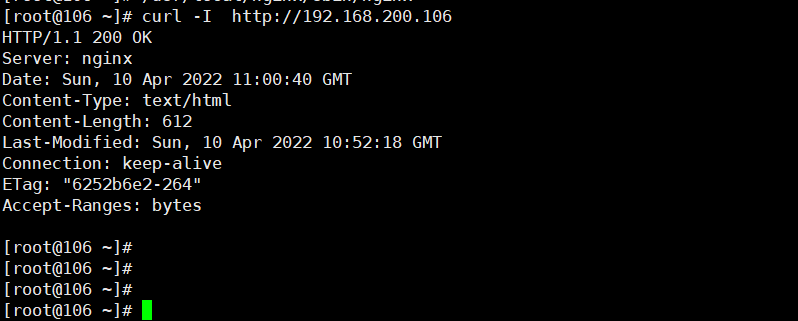
方法2

安装完成后改配置文件，在server前面加一个

server\_tokens off;

不显示版本号





二、修改nginx用户与组

nginx运行时进程需要有用户和组身份的支持，以实现对网站文件读取时进行访问控制。nginx默认使用nobody用户和组账号，一般也要进行修改

方法一:

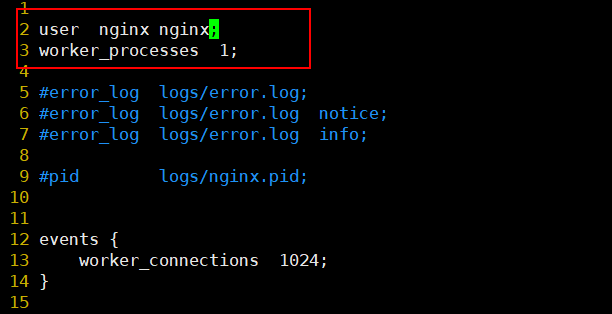
编译安装时进行指定

/configure --prefix=/usr/local/nginx --user=nginx --group=nginx && make &&make install

方法二:

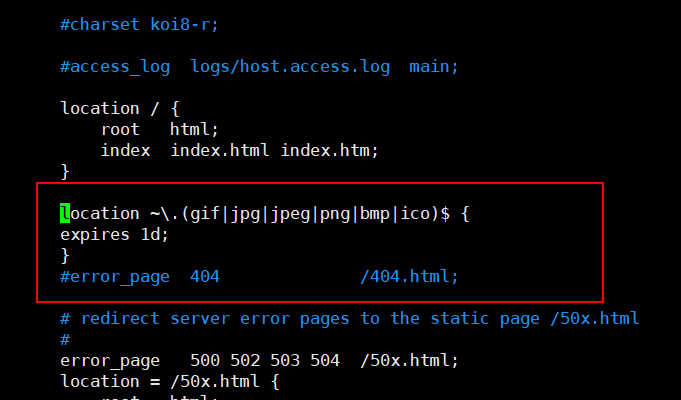
修改配置文件

vim /usr/local/nginx/conf/nginx.conf

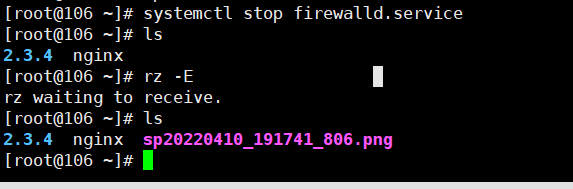


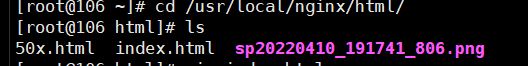
三、配置nginx网页缓存时间

当nginx将页面数据返回给客户端后，可设置资源在客户端缓存时间，以便客户端在后期相对相同内容进行访问时直接返回，加快访问速度，一般针对静态页面，对动态页面不设置缓存时间。可在windows客户端使用fiddler查看网页缓存时间

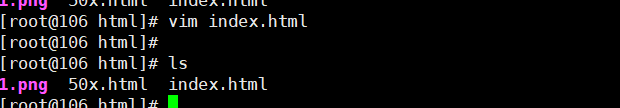
方法：修改配置文件，在http段，server段，location段加入特定内容的过期参数

1.将主页上面加个图片试试

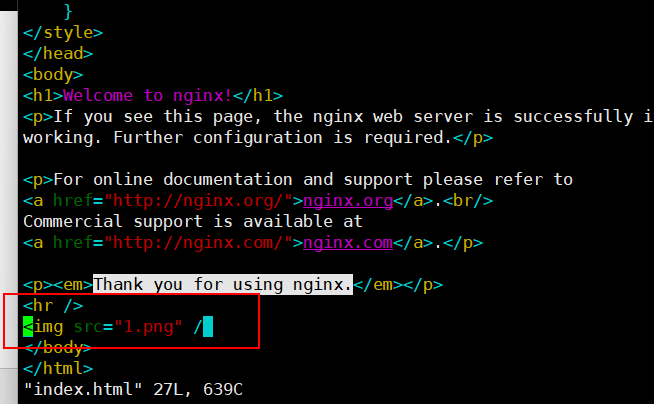




改名

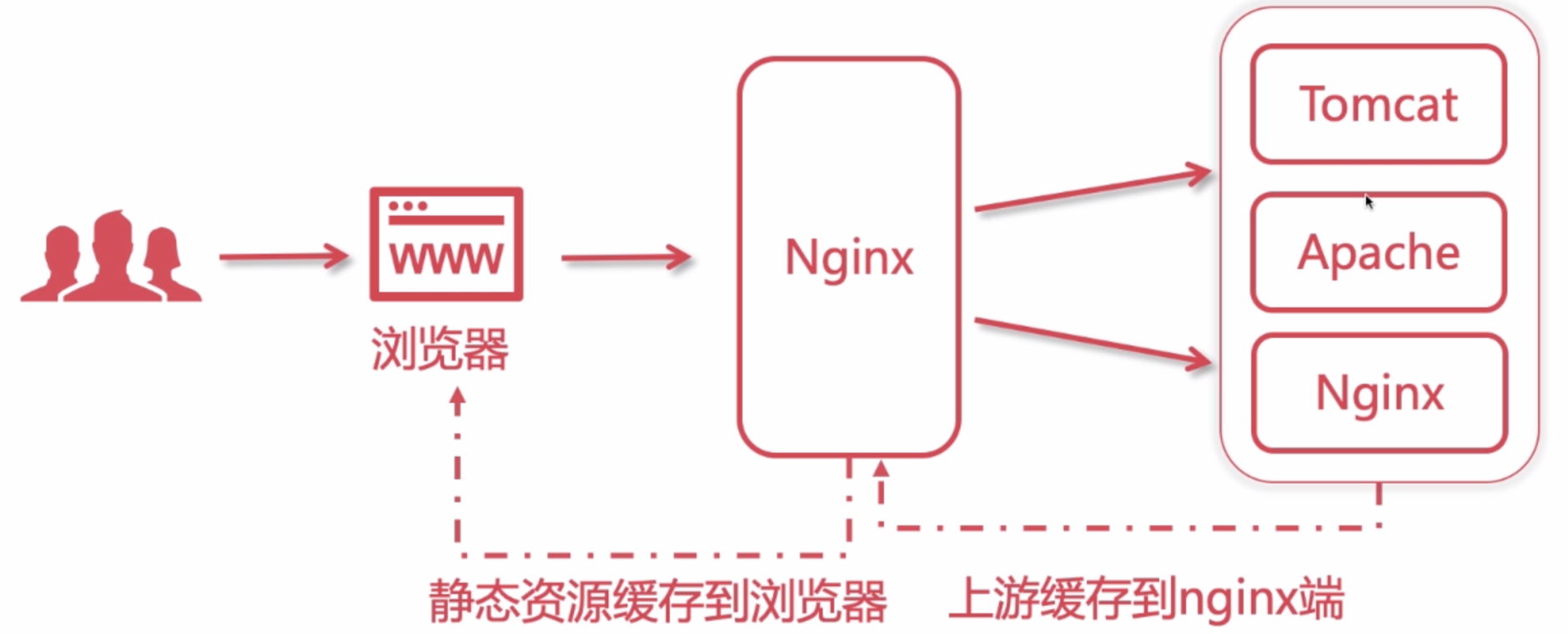


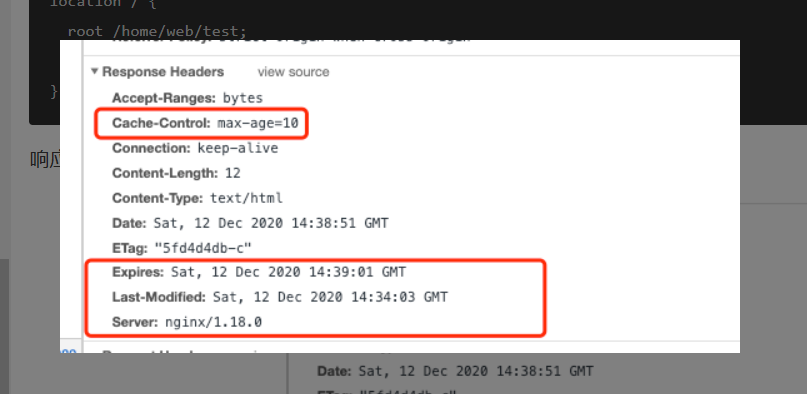
加图



使用抓包软件fiddler查看



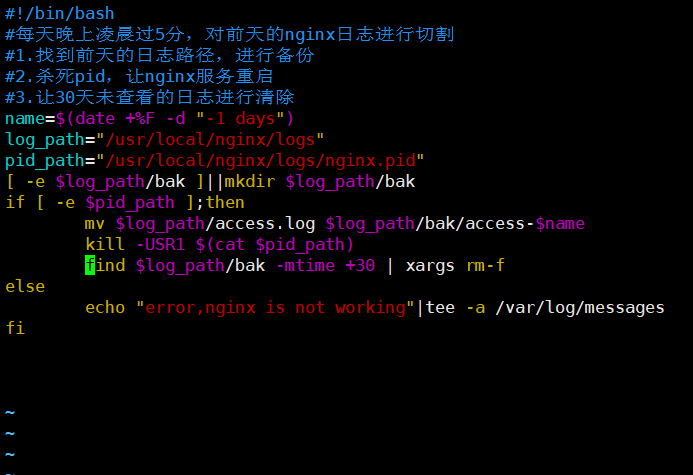




四、实现nginx的日志切割

tee -a 往系统日志里面写东西

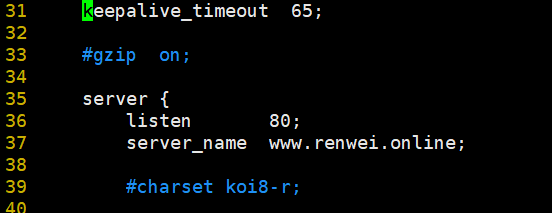
kill -USR1 日志命令重新执行



五、配置nginx实现连接超时

保持连接（长连接）

为避免用户访问多次访问，握手挥手，设置保持长时间连接



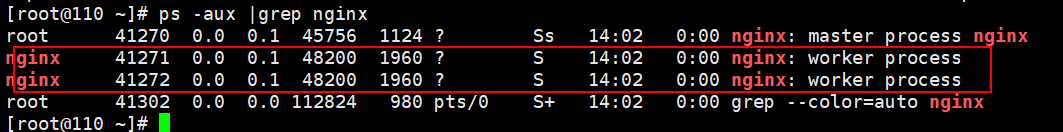
client\_header\_timeout:等待客户端请求头的超时时间

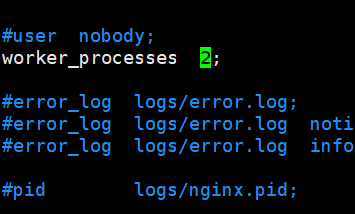
clien\_body\_timeout:等待客户端请求头的超时时间



六、更改nginx工作进程数

高并发的场景，需要启动更多的nginx工作进程以保证快速响应，处理用户响应，避免造成阻塞





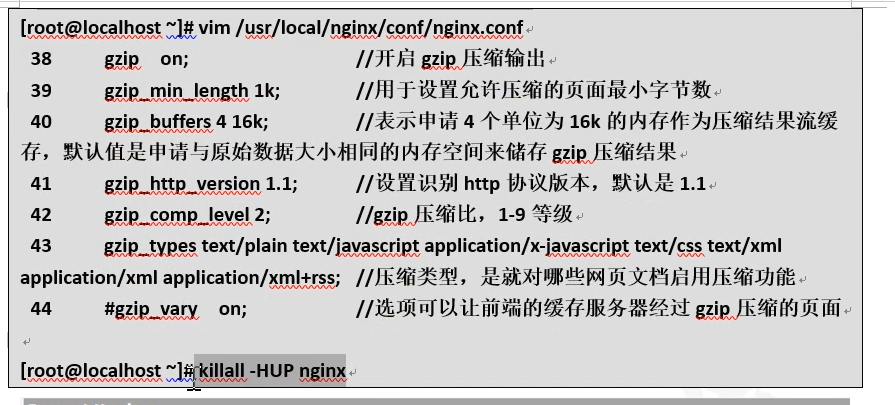
配置cpu亲和性

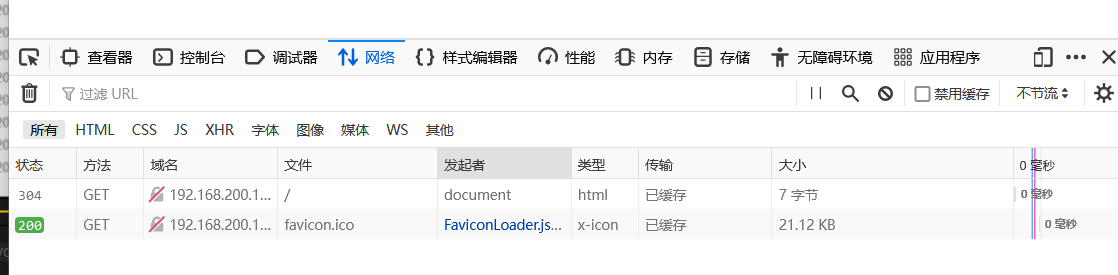
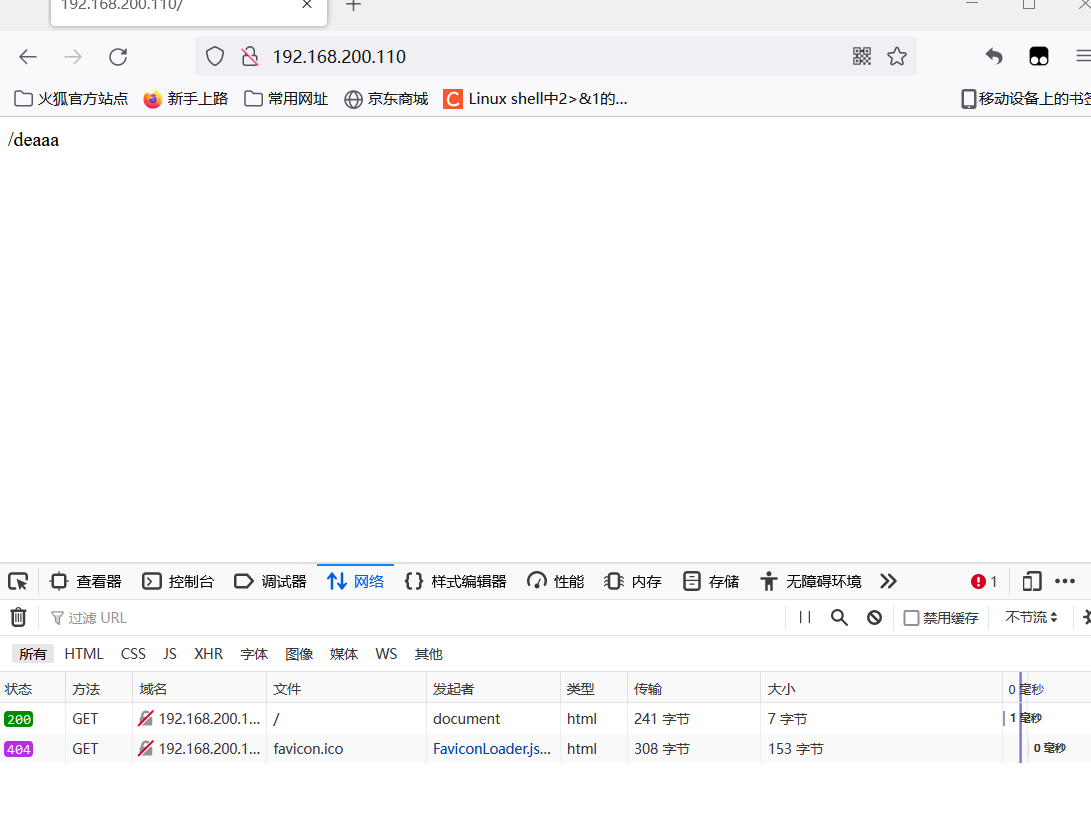
默认nginx的多个进程可能跑在一颗cpu核心上，可以分配不同的进程给不同的cpu进行处理，充分利用硬件多核CPU 。在一台4核物理服务器，进行下面的配置，将进程进行分配

worker\_cpu\_affinity 0001 0010 0100 1000;

七、配置nginx实现网页压缩功能

nginx的nginx\_http\_gzip\_moudle压缩模块提供了对文件内容的压缩的功能，允许nginx服务器将输出内容发送到客户端之前进行压缩，节约网站带宽，提升用户访问体验，模块默认已安装





gzip\_disable regex(特定条件);

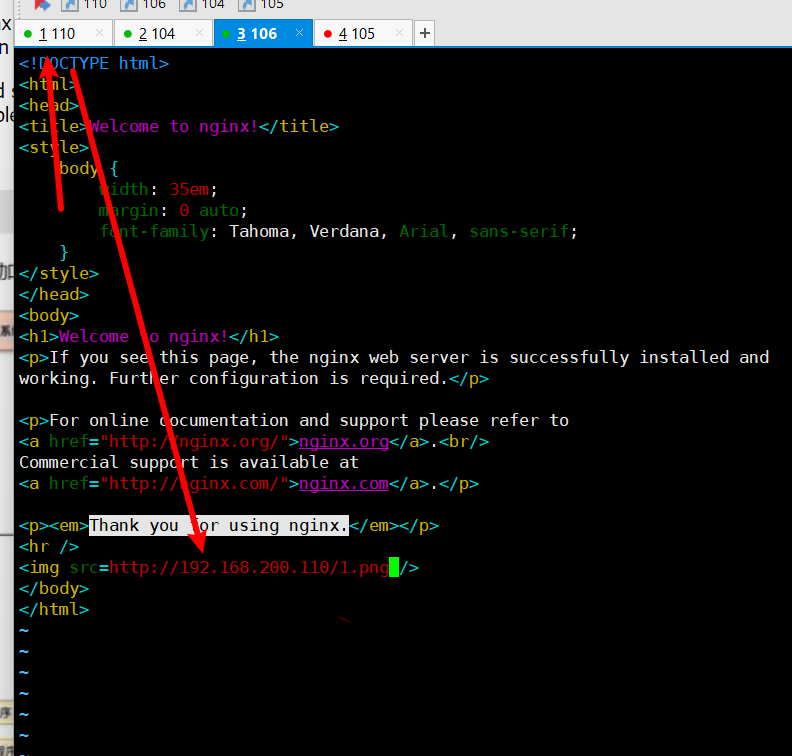
根据客户端的浏览器标志关闭压缩功能

gzip\_disable MSIE [4-6] \.

匹配字符串中包含MSIE 4、5、6的所有浏览器，关闭压缩功能

八、配置nginx实现防盗链功能

1.盗用110的图片地址，放到106页面上



修改域名进行访问，更改hosts文件

添加防盗链

配置说明

none :浏览器中referer为空的情况，就直接在浏览器中访问图片

blocked:referer不为空的情况，但是值被代理或者防火墙删除了，这些值不以http://或https://开头

location ~\*\.(wma|wmv|asf|mp3|zip|rar|jpg|gif|png|swf|flv)$ {

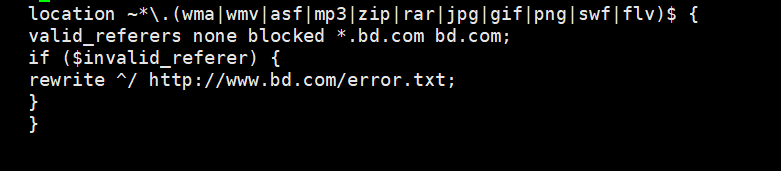
valid\_referers none blocked \*.bd.com bd.com;

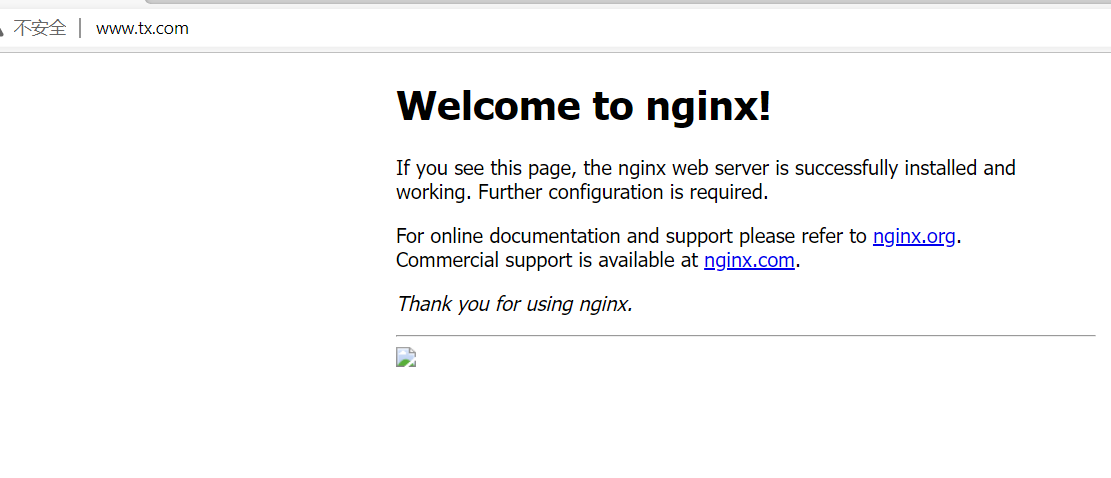
if ($invalid\_referer) {

rewrite ^/ http://www.bd.com/error.txt;

}

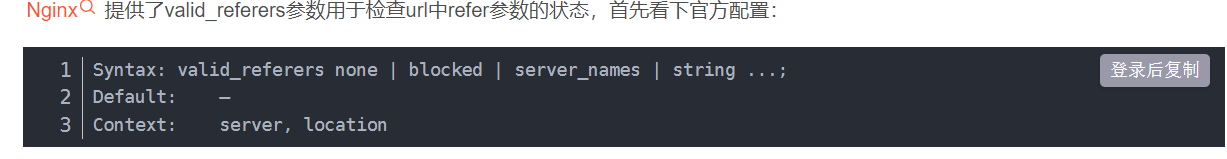
}





防盗链原理

当用户匹配到wma.wmv,png图片时，利用nginx内置的变量，匹配它是否从正确的地址进行访问，如果是，放行；不是，重定向特定页面。



1.none //直接访问url，而不使用任何网站或链接的跳转，$http\_referer变量的值是空的。

2.blocked //请求标头中存在“ Referer”字段，但其值已被防火墙或代理服务器删除；这些值是不以“ http：//”或“ https：//”开头的字符串；

3.字符串 //配置域名列表，支配通配符、正则表达式写法。

这么写表示：如果"refer为空"、或"refer不以http/https开头"、或"域名为example.com的server\_name" 都是有效的refer，也即valid\_referers的值为0（允许访问），除此之外为1（返回403）

九、对FPM模块进行参数优化

nginx的php解析功能

优化参数调整：（根据服务器的内存和服务器负载进行调整）

nginx的php解析功能实现是交由FPM（fastcgi进程管理器）处理的，为了提高PHP的处理速度，可对FPM模块进行参数跳转

pm:使用哪种方式启动fpm进程，static和dynamic，前者将产生固定数量的fpm进程， 后者将以动态的方式产生fpm进程

pm.max\_children static方式下开启的fpm进程数

pm.start\_servers 动态方式下初始的fpm进程数量

pm.min\_spare\_servers 动态方式下最小的fpm空闲进程数

pm.max\_spare\_servers 动态方式下最大的fpm空闲进程数

vim /usr/local/php5/etc/php-fpm.conf

例：服务器为云服务器，运行了个人论坛，内存为1.5G,fpm进程数为20，内存消耗近1G，处理比较慢

vim /usr/local/php5/etc/php-fpm.conf

pm = dynamic

pm.start\_servers = 5

pm.min\_spare\_servers = 2

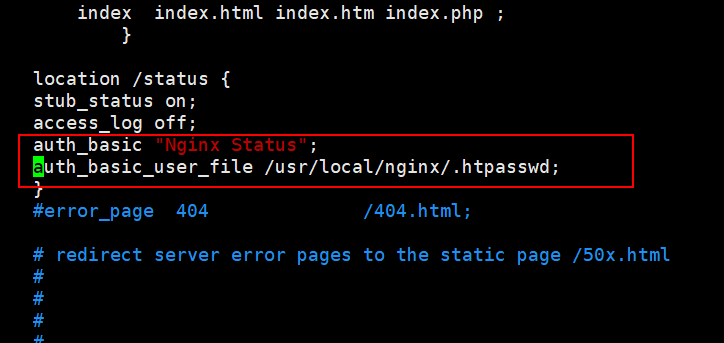
pm.max\_spare\_servers = 8

十、nginx为目录添加访问控制

安装软件

yum -y intall httpd-tools

htpasswd -c /usr/local/nginx/.htpasswd rw



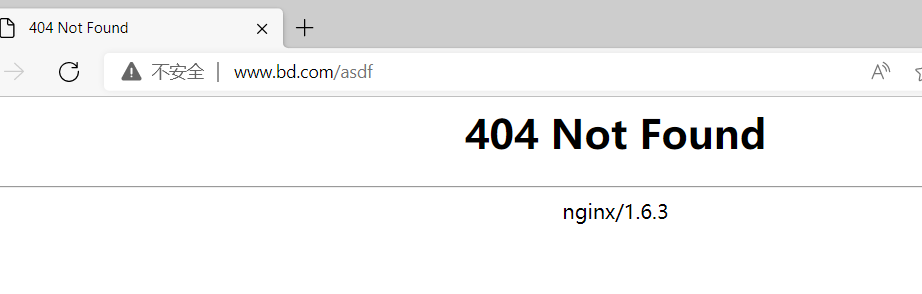
下面还能设置允许拒绝

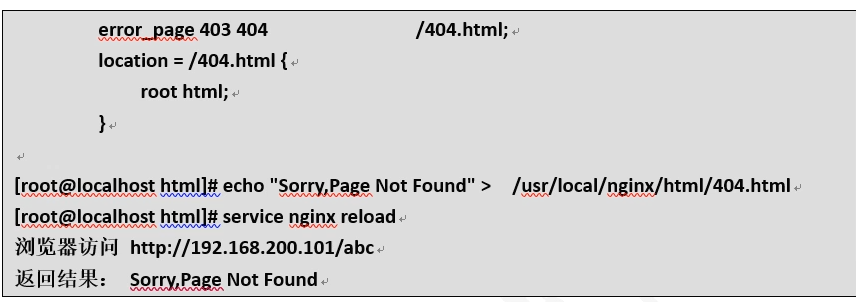
allow 192.168.200.2;

deny 192.168.200.0/24;

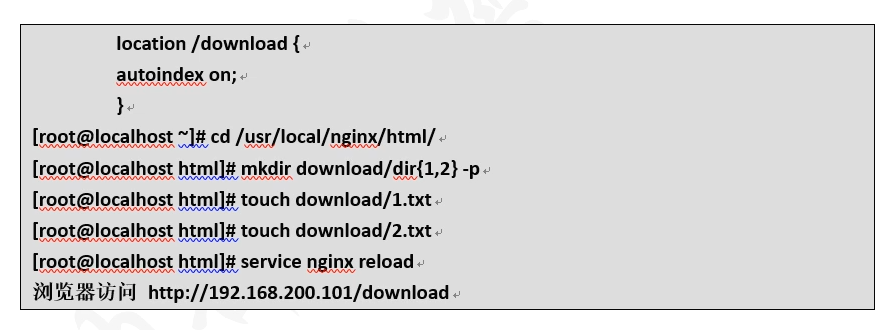
十一、自定义错误页面

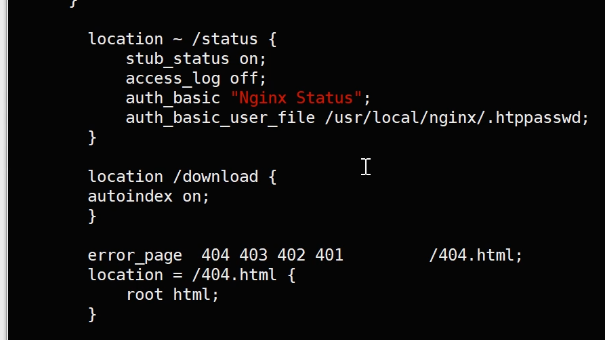
默认显示nginx版本号不安全

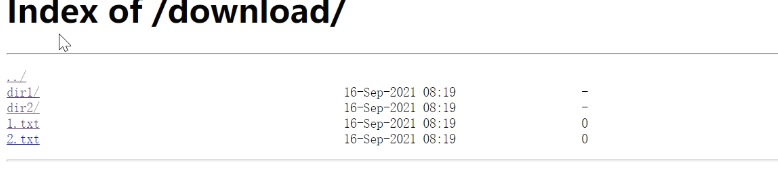




十二、自动索引







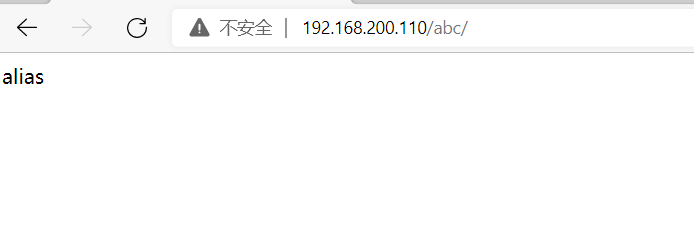
十三、目录别名功能



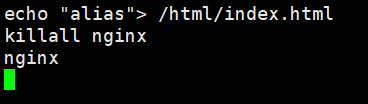
root的处理结果是：root路径＋location路径

alias的处理结果是：使用alias路径替换location路径

mkdir /html







十四、通过UA实现手机端和电脑端的分离

十五、go access可视化日志

1.安装go access

wget http://tar.goaccess.io/goaccess-1.2.tar.gz

tar -xzvf goaccess-1.2.tar.gz

cd goaccess-1.2/

./configure --enable-geoip --enable-utf8 --with-opensslmake

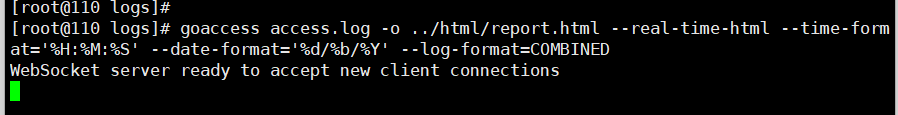
make

make install

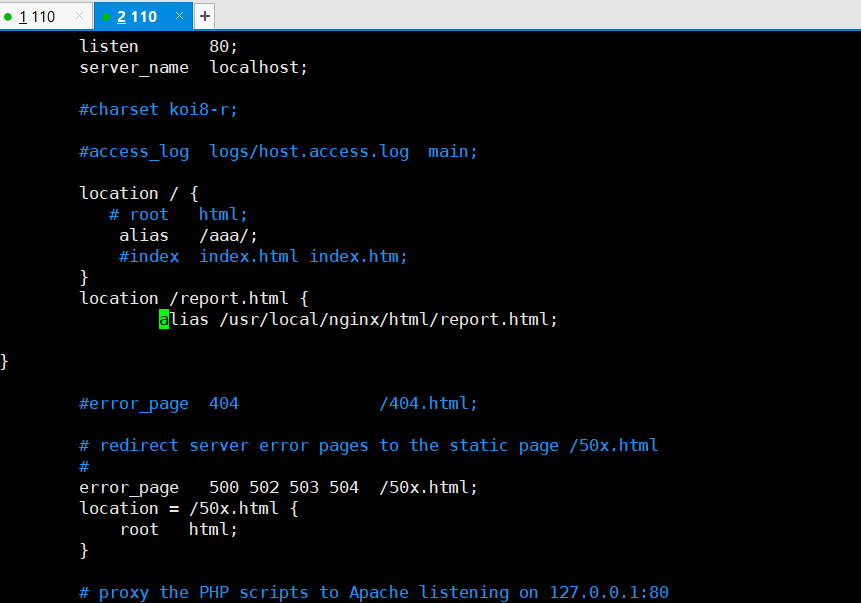
2.进入nginx日志下，输入连接到goaccess命令

goaccess access.log -o ../html/report.html --real-time-html --time-format='%H:%M:%S' --date-format='%d/%b/%Y' --log-format=COMBINED

3.修改日志



4.添加访问控制



5.测试

