##### 其他



##### 防盗链

**描述：**应用于location块。配置外域名访问，直接返回文件不存在。

**操作：**

#REFERER-START

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

expires 30d;

#nginx1.15版本

valid\_referers h5.h5-bus.com www.cc007.top;

#nginx1.12版本

#valid\_referers none blocked h5.h5-bus.com www.cc007.top;

if ($invalid\_referer) {

return 403;

}

}

#REFERER-END

##### 伪静态

**描述：**。

**操作：**

##### 限流

**描述：**。

正常开发情况我们不对css、js、gif、png，jpg等进行连接限制，而对除此之外的链接进行限制。

要配http和location两个地方。

（$binary\_remote\_addr这个变量是固定的。绿色的才是自定义变量）

**注意：**

1、limit\_req如果加了burst就一定要加nodelay，否则会造成大量的tcp连接请求等待。

**操作：**

http {

limit\_req\_zone $binary\_remote\_addr zone=yhx\_req\_zone:10m rate=10r/s; #####同一个ip一秒10个请求

limit\_conn\_zone $binary\_remote\_addr zone=yhx\_conn\_zone:10m; #####这里10M能记录32000条记录

limit\_conn\_zone $server\_name zone= yhx\_perserver\_zone :10m;

..........

}

###########################################

location ~ [^/]\.php(/|$) {

..........

limit\_req zone=yhx\_req\_zone; #####此处应用yhx\_req\_zone变量的规则

limit\_conn yhx\_conn\_zone 1; #####同一个ip并发连接数为1

limit\_conn yhx\_perserver\_zone 1000; #####该网站总连接数1000

limit\_rate 300k; #####对每个连接限速300k。（是对连接限速，不是对IP限速。如果一个IP允许两个并发连接，那么这个IP就是限速limit\_rate×2。）

..........

}

**疑问：**。

1. 配了限流之后微信授权就不行了，配到100并发也不行。也不知道微信干了什么？

2、limit\_rate好像没用，我弄成0，网站应该是不能访问了，但是依旧能访问？

##### 反爬虫

**描述：**。

作用域server。

主要通过user\_agent里面的信息作判断，但是这个user\_agent很容易造假，只能尽人事。

**参考：**

<https://blog.51cto.com/9025736/2061947>

**操作：**

###############反爬虫

#禁止Scrapy|curl等工具的抓取

if ($http\_user\_agent ~\* (Scrapy|Curl|HttpClient))

{

return 403;

}

#禁止指定UA及UA为空的访问

if($http\_user\_agent ~ "FeedDemon|JikeSpider|Indy Library|Alexa Toolbar|AskTbFXTV|AhrefsBot|CrawlDaddy|CoolpadWebkit|Java|Feedly|UniversalFeedParser|ApacheBench|Microsoft URL Control|Swiftbot|ZmEu|oBot|jaunty|Python-urllib|lightDeckReports Bot|YYSpider|DigExt|YisouSpider|HttpClient|MJ12bot|heritrix|EasouSpider|Ezooms|^$" )

{

return 403;

}

#禁止非GET|HEAD|POST方式的抓取

if ($request\_method !~ ^(GET|HEAD|POST)$)

{

return 403;

}

##### 客户端缓存

**描述：**。

[HTTP Headers 模块](https://www.nginx.cn/doc/standard/httpheaders.html)

主要就是这个expires。

一般对图片、css、js等资源进行缓存

一般结合防盗链使用。

**操作：**

location ~\* \.(jpg|png|gif)$ {

expires 7d;

}

##### Gzip压缩

**描述：**。

<https://segmentfault.com/a/1190000012571492?utm_source=tag-newest>

**操作：**

##########开启关闭

gzip on;

##########压缩缓冲区大小，和增加步进空间

gzip\_buffers 16 8k;

##########压缩级别，1-9

gzip\_comp\_level 2;

##########设置允许压缩的页面最小字节数，页面字节数从header头中的Content-Length中进行获取。默认值是0，不管页面多大都压缩。建议设置成大于1k的字节数，小于1k可能会越压越大。

gzip\_min\_length 1k;

##########Nginx做为反向代理的时候启用。通过判断后端服务器返回的header中的参数决定是否压缩。具体参数自行百度。

gzip\_proxied expired no-cache no-store private auth;

##########需要压缩的文件类型

gzip\_types text/plain application/javascript application/x-javascript text/javascript text/css application/xml;

##########增加响应头”Vary: Accept-Encoding”，具体不知道什么用

gzip\_vary on;

##########IE5.5和IE6 SP1使用msie6参数来禁止gzip压缩 ，所以通过User-Agents匹配识别浏览器，识别到是不能压缩的浏览器就不压缩。

gzip\_disable "MSIE [1-6]\.";

**参考：**

<https://www.cnblogs.com/xzkzzz/p/9224358.html>

<https://segmentfault.com/a/1190000012571492?utm_source=tag-newest>

<https://www.cnblogs.com/kevingrace/p/10018914.html>

<https://www.jianshu.com/p/cc61d74104e5>

<https://segmentfault.com/a/1190000012694104>

##### 反向代理

#proxy\_pass https://h5.h5-bus.com;

#Proxy Settings

#proxy\_redirect off;

#proxy\_set\_header Host $host;

#proxy\_set\_header X-Real-IP $remote\_addr;

#proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

#proxy\_next\_upstream error timeout invalid\_header http\_500 http\_502 http\_503 http\_504;

#proxy\_max\_temp\_file\_size 0;

#proxy\_connect\_timeout 90;

#proxy\_send\_timeout 90;

#proxy\_read\_timeout 90;

#proxy\_buffer\_size 4k;

#proxy\_buffers 4 32k;

#proxy\_busy\_buffers\_size 64k;

#proxy\_temp\_file\_write\_size 64k;