Nginx高级应用

View in Github czHappy 2019.5.2

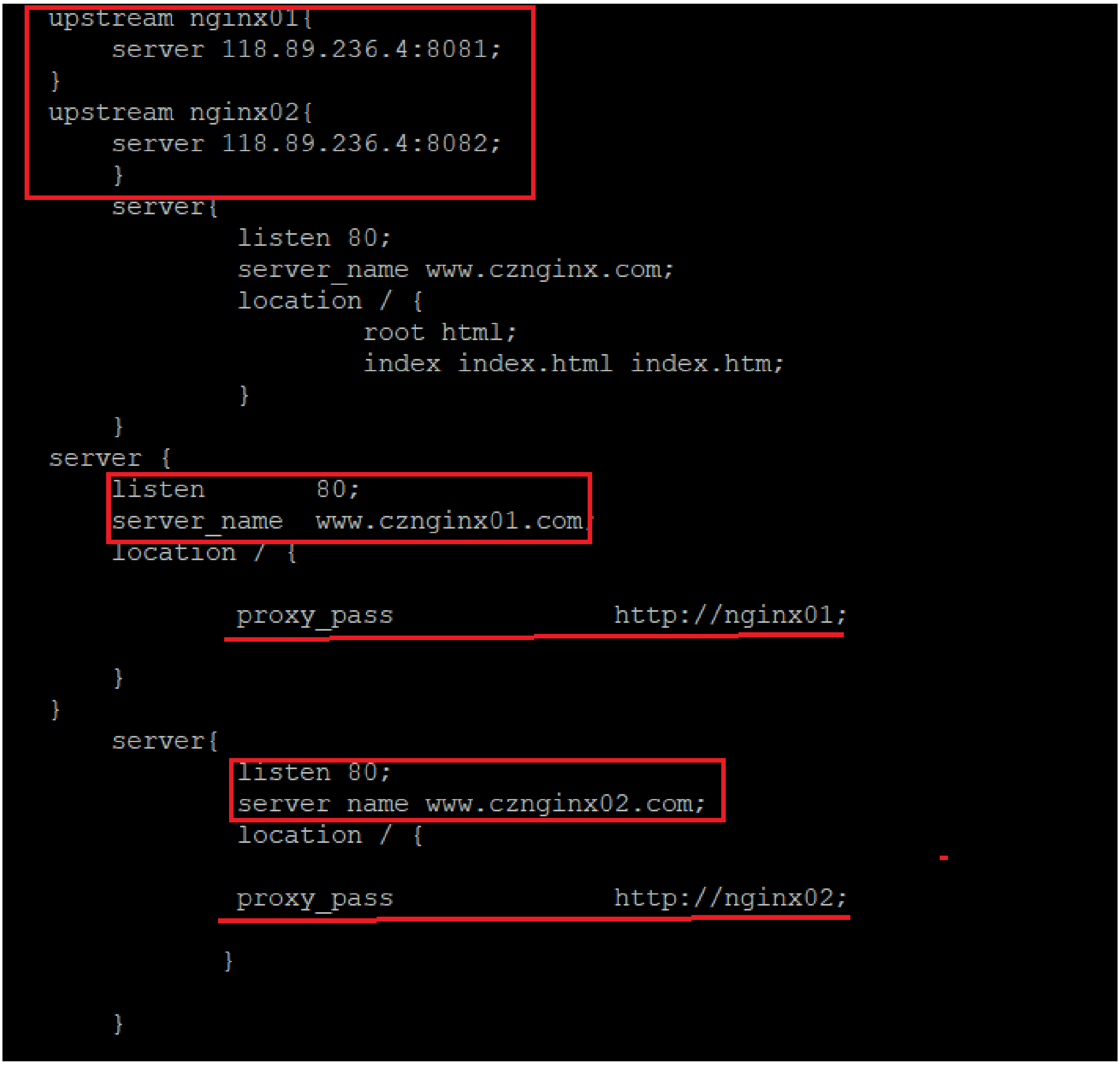
一、配置和实验反向代理

- 配置反向代理服务器的nginx.conf

```
#声明两个upstream 反向代理后端服务器
    upstream nginx01{
        server 118.89.236.4:8081;
    }
    upstream nginx02{
        server 118.89.236.4:8082;
    }

  # 代理服务器80端口设定监听和反向代理
    server {
        listen       80;
        server_name  www.cznginx01.com;
        location / {
            proxy_pass              http://nginx01;
        }
    }
    server{
        listen 80;
        server_name www.cznginx02.com;
        location / {
            proxy_pass              http://nginx02;
        }
    }
```

- 配置后端服务器的nginx.conf

```
       server{
               listen 8081;
               server_name nginx8081.com;
               root html/nginx01;
               index index.html index.htm;
       }
       server{
               listen 8082;
               server_name nginx8082.com;
               root html/nginx01;
               index index.html index.htm;
       }
```

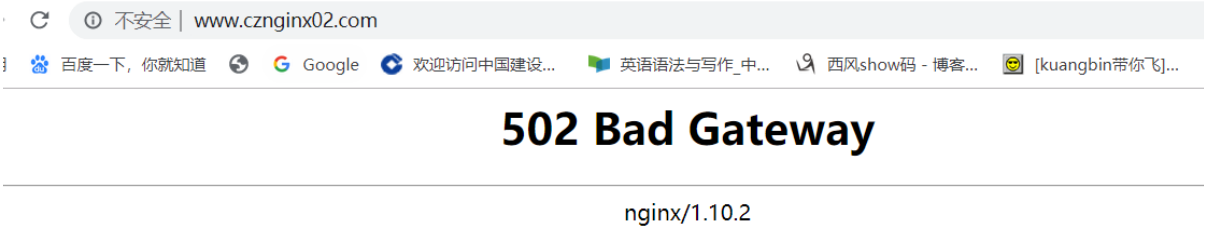


```
upstream nginx01{
    server 118.89.236.4:8081;
}
upstream nginx02{
    server 118.89.236.4:8082;
    }
    server{
            listen 80;
            server_name www.cznginx.com;
            location / {
                    root html;
                    index index.html index.htm;
            }
    }
server {
    listen       80;
    server_name  www.cznginx01.com;
    location / {

            proxy_pass              http://nginx01;

    }
}
    server{
            listen 80;
            server_name www.cznginx02.com;
            location / {

            proxy_pass              http://nginx02;

           }

    }
```

- 在本机（Windows）配置hosts

```
#服务器域名解析
118.89.236.4 www.cznginx01.com
118.89.236.4 www.cznginx02.com
118.89.236.4 www.cznginx.com
```

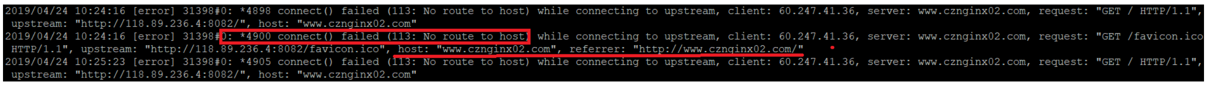
- 实验结果
    - 报错,502



502 Bad Gateway

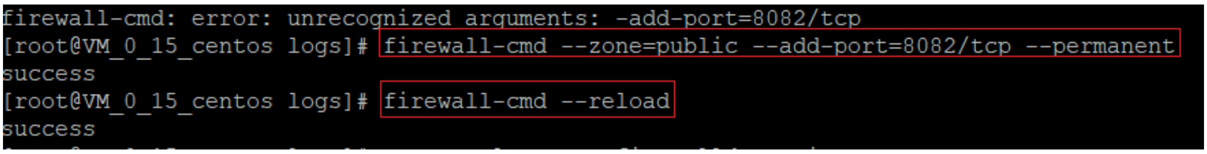
nginx/1.10.2

- 访问8082端口失败，查看error.log日志



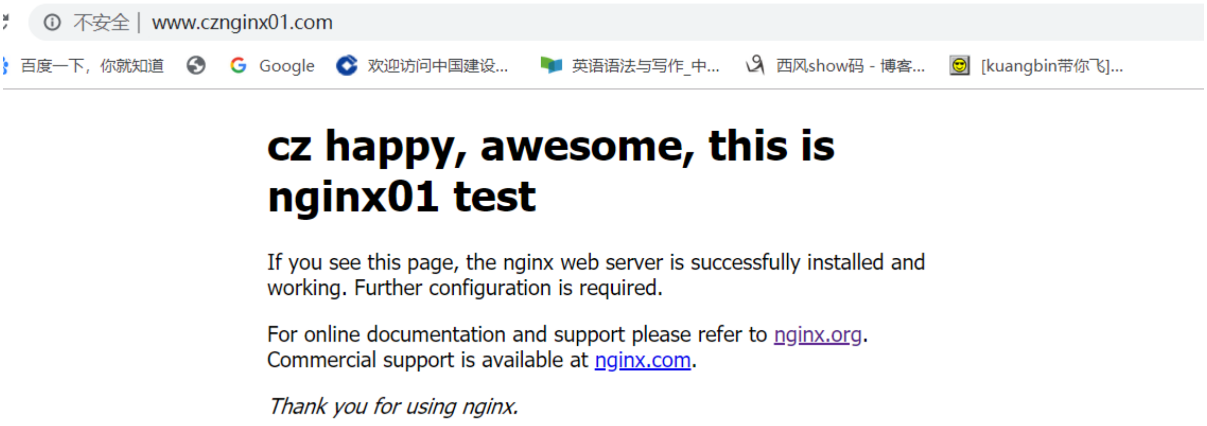
```
2019/04/24 10:24:16 [error] 31398#0: *4898 connect() failed (113: No route to host) while connecting to upstream, client: 60.247.41.36, server: www.cznginx02.com, request: "GET / HTTP/1.1",
 upstream: "http://118.89.236.4:8082/", host: "www.cznginx02.com"
2019/04/24 10:24:16 [error] 31398#0: *4900 connect() failed (113: No route to host) while connecting to upstream, client: 60.247.41.36, server: www.cznginx02.com, request: "GET /favicon.ico
 HTTP/1.1", upstream: "http://118.89.236.4:8082/favicon.ico", host: "www.cznginx02.com", referrer: "http://www.cznginx02.com/"
2019/04/24 10:25:23 [error] 31398#0: *4905 connect() failed (113: No route to host) while connecting to upstream, client: 60.247.41.36, server: www.cznginx02.com, request: "GET / HTTP/1.1",
 upstream: "http://118.89.236.4:8082/", host: "www.cznginx02.com"
```

- 排除upstream的语法错误，开启防火墙8082端口的权限



```
firewall-cmd: error: unrecognized arguments: -add-port=8082/tcp
[root@VM_0_15_centos logs]# firewall-cmd --zone=public --add-port=8082/tcp --permanent
success
[root@VM_0_15_centos logs]# firewall-cmd --reload
success
```

- 成功结果

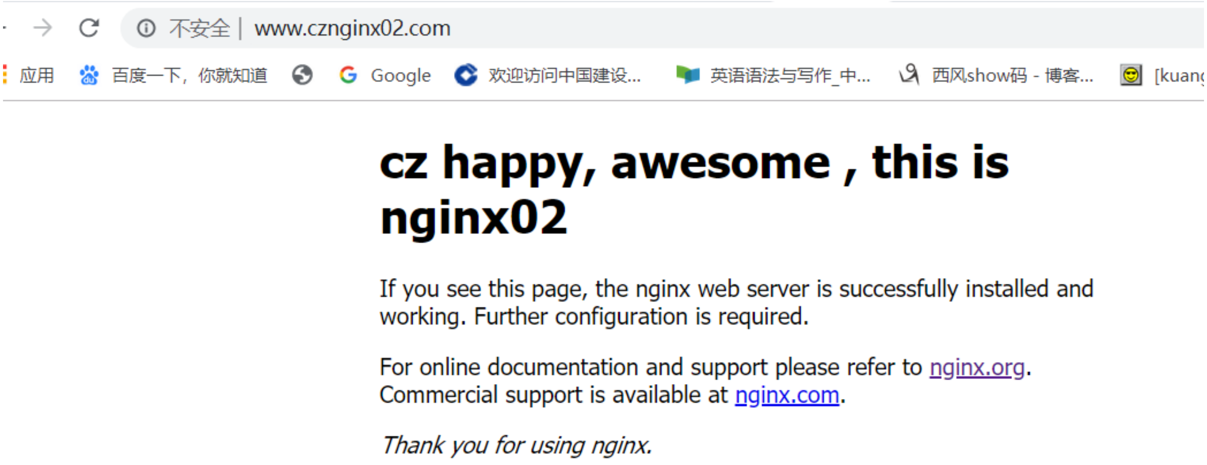


cz happy, awesome, this is nginx01 test

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.



cz happy, awesome , this is nginx02

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

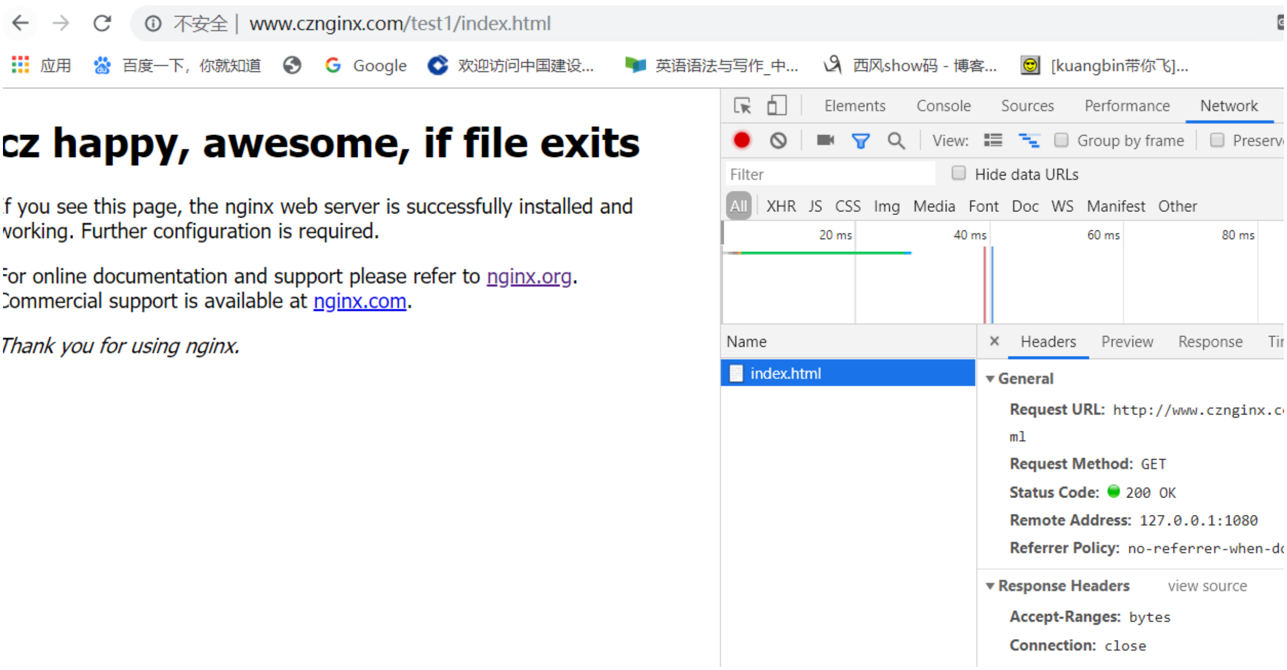
Thank you for using nginx.

二、配置和实验rewrite 以及 redirect

- rewrite和redirect的区别
    - 关于重定向
        - 通过重定向，浏览器知道页面位置发生变化，从而改变地址栏显示的地址。
        - 通过重定向，搜索引擎意识到页面被移动了，从而更新搜索引擎索引，将原来失效的链接从搜索结果中移除
        - 临时重定向(R=302)和永久重定向(R=301)都是亲搜索引擎的，是SEO的重要技术。
        - Redirect是浏览器和服务器发生两次请求，也就是服务器命令客户端“去访问某个页面”；
        - redirect的URL需要传送到客户端。
        - redirect是从一个地址跳转到另一个地址。
    - 关于重写
        - rewrite的URL只是在服务器端
        - Rewrite则是服务器内部的一个接管，在服务器内部告诉“某个页面请帮我处理这个用户的请求”，浏览器和服务器只发生一次交互，浏览器不知道是该页面做的响应，浏览器只是向服务器发出一个请求。
        - URL重写用于将页面映射到本站另一页面，**若重写到另一网络主机（域名），则按重定向处理**。
        - rewrite是把一个地址重写成另一个地址。地址栏不跳转。相当于给另一个地址加了一个别名一样。
- 上述的例子就像用户去买手机，缺货时的两种处理：让用户自己去其他地方买（Redirect）；公司从其他的地方调货（Rewrite）。
- redirect和rewrite操作

```
 server{
        listen 8081;
        server_name 118.89.236.4;
        root html/nginx01;

#  重定向日志以notice级别输出到error.log
    rewrite_log on;
    error_log /usr/local/nginx/logs/error.log notice;

#       地址重写
        location / {
                if (!-f $request_filename){

                        rewrite ^/test1/(.*)$ http://www.baidu.com;
                }
                rewrite ^/test2/(.*)$  /test/index.html;
        }
#  302重定向 临时重定向
        location = /redirect {
                    return 302 http://www.baidu.com;
        }
#  301重定向 永久重定向
        location = /redir {
                return 301  http://118.89.236.4/test/index.html;
        }
#  alias别名
        location /newweb {
                alias /usr/local/nginx/html/nginx01/test/;
        }

            #防盗链
        location ~* \.(gif|jpg|png|swf|flv)$ {
                valid_referers none blocked 118.89.236.4;
                if ($invalid_referer) {
                        return 403;
                }
        }

    }
```

- 对nginx01进行地址重写rewrite操作 如果文件不存在，且第一个参数匹配test2，则跳到test/index.html，如果文件存在，跳到该文件。



cz happy, awesome, if file exits

f you see this page, the nginx web server is successfully installed and vorking. Further configuration is required.

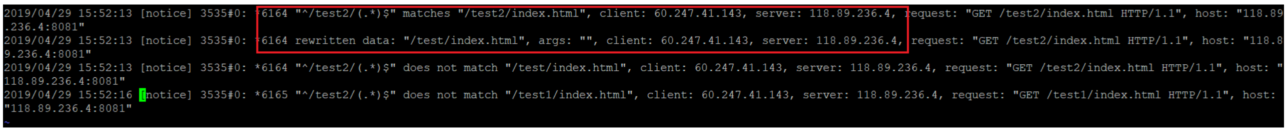
For online documentation and support please refer to nginx.org. Commercial support is available at nginx.com.

Thank you for using nginx.

▾ General
Request URL: http://www.cznginx.c
ml
Request Method: GET
Status Code: ● 200 OK
Remote Address: 127.0.0.1:1080
Referrer Policy: no-referrer-when-do

▾ Response Headers view source
Accept-Ranges: bytes
Connection: close

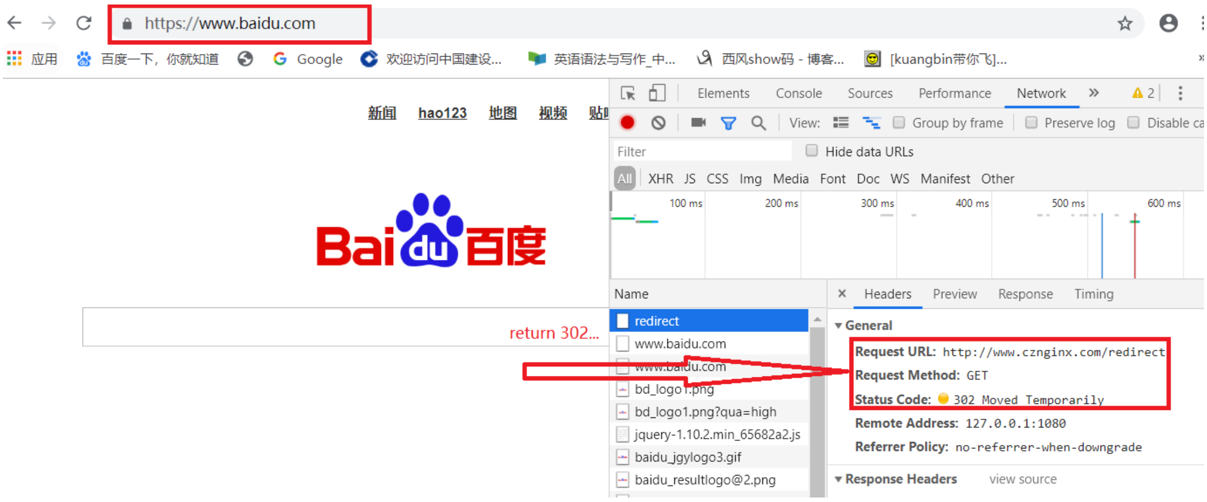
  - 重定向日志



```
2019/04/29 15:52:13 [notice] 3535#0: *6164 "^/test2/(.*)$" matches "/test2/index.html", client: 60.247.41.143, server: 118.89.236.4, request: "GET /test2/index.html HTTP/1.1", host: "118.89
.236.4:8081"
2019/04/29 15:52:13 [notice] 3535#0: *6164 rewritten data: "/test/index.html", args: "", client: 60.247.41.143, server: 118.89.236.4, request: "GET /test2/index.html HTTP/1.1", host: "118.8
9.236.4:8081"
2019/04/29 15:52:13 [notice] 3535#0: *6164 "^/test2/(.*)$" does not match "/test/index.html", client: 60.247.41.143, server: 118.89.236.4, request: "GET /test2/index.html HTTP/1.1", host: "
118.89.236.4:8081"
2019/04/29 15:52:16 [notice] 3535#0: *6165 "^/test2/(.*)$" does not match "/test1/index.html", client: 60.247.41.143, server: 118.89.236.4, request: "GET /test1/index.html HTTP/1.1", host:
"118.89.236.4:8081"
~
```

- 对nginx01进行地址重写redirect操作

- 临时重定向

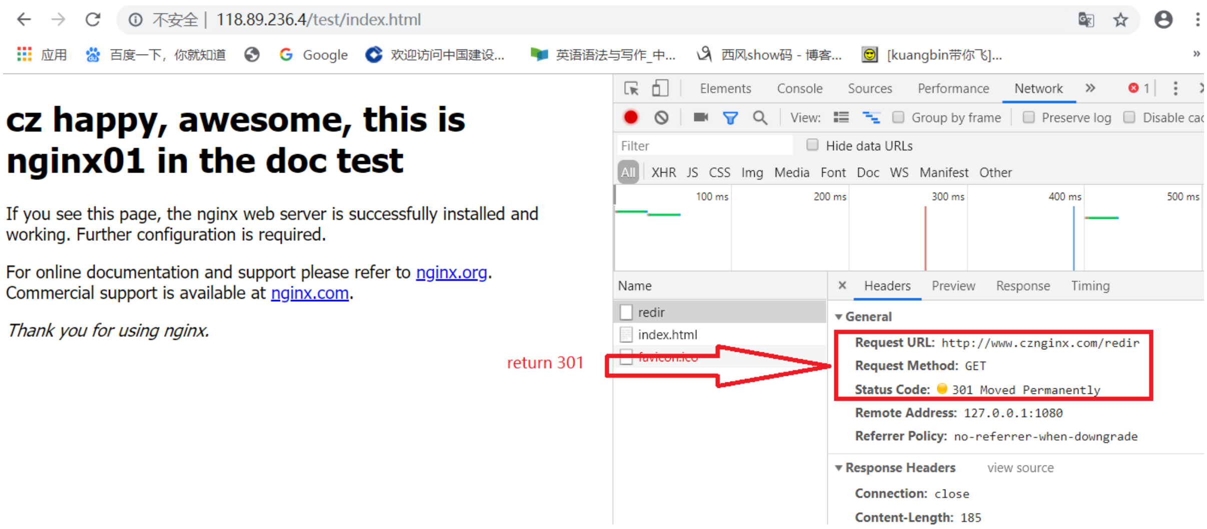


redirect
www.baidu.com
www.baidu.com
bd_logo1.png
bd_logo1.png?qua=high
jquery-1.10.2.min_65682a2.js
baidu_jgylogo3.gif
baidu_resultlogo@2.png

General
Request URL: http://www.cznginx.com/redirect
Request Method: GET
Status Code: 302 Moved Temporarily
Remote Address: 127.0.0.1:1080
Referrer Policy: no-referrer-when-downgrade

Response Headers view source

- 永久重定向



cz happy, awesome, this is nginx01 in the doc test

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

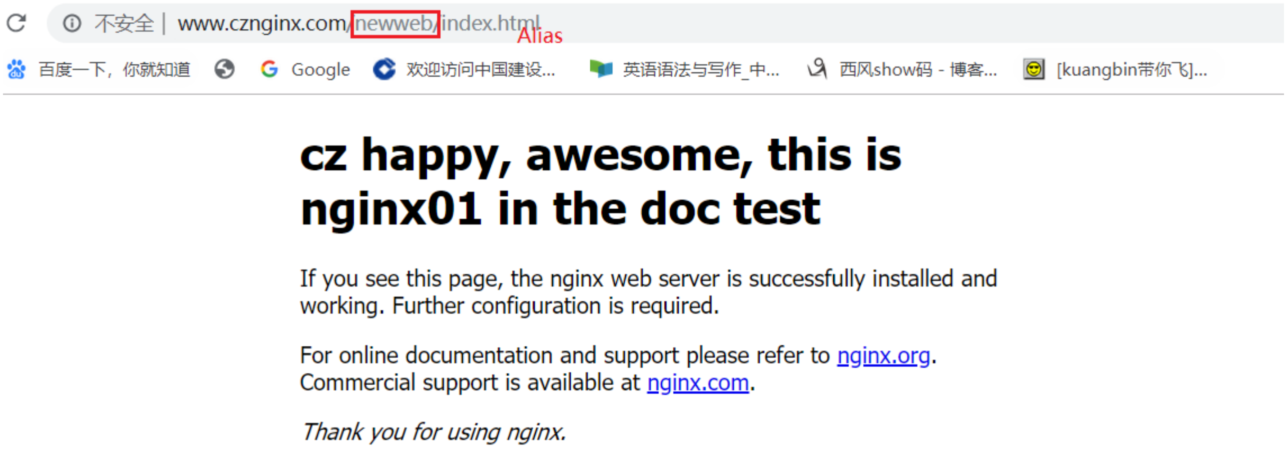
Thank you for using nginx.

redir
index.html
favicon.ico

General
Request URL: http://www.cznginx.com/redir
Request Method: GET
Status Code: 301 Moved Permanently
Remote Address: 127.0.0.1:1080
Referrer Policy: no-referrer-when-downgrade

Response Headers view source
Connection: close
Content-Length: 185

- 配置别名alias



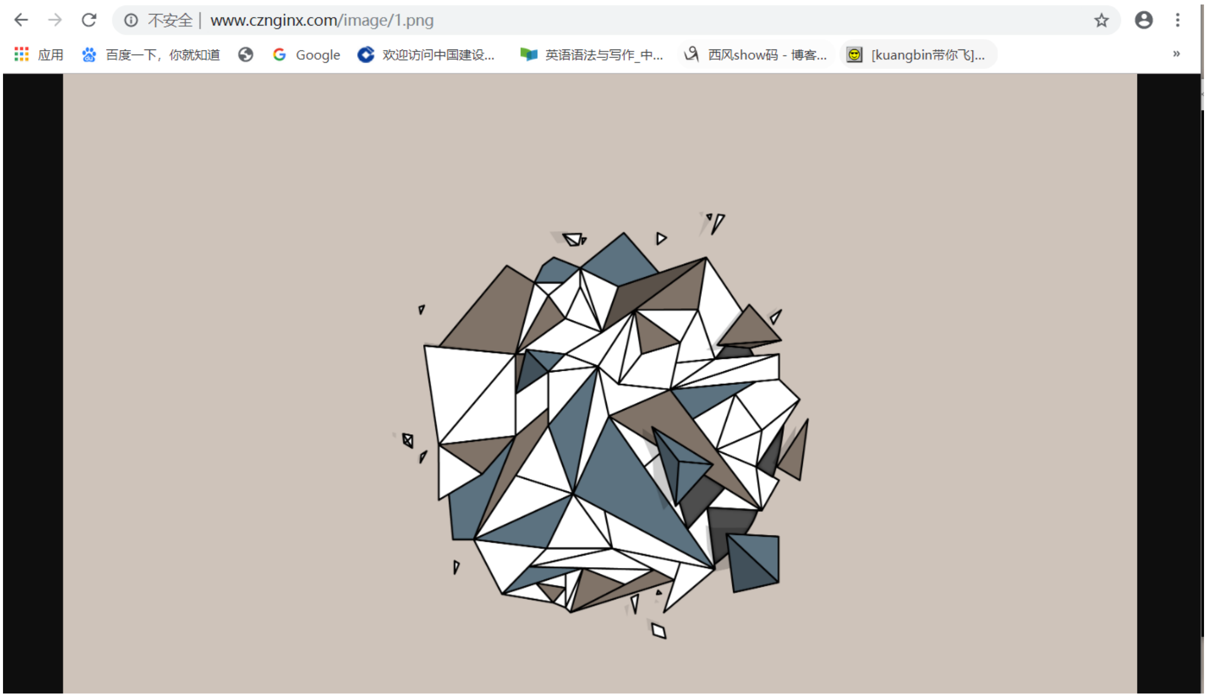
cz happy, awesome, this is nginx01 in the doc test

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

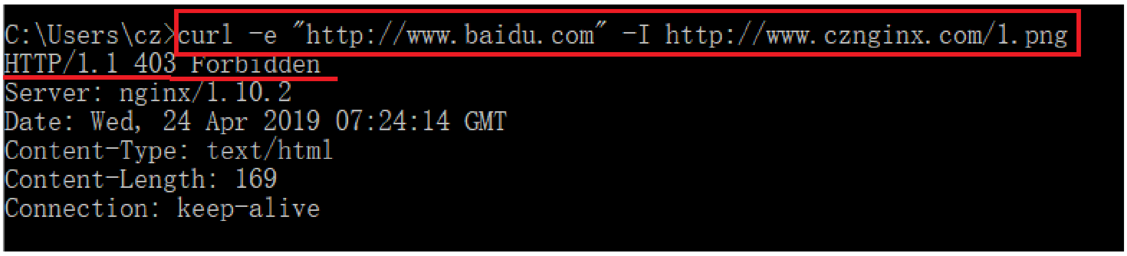
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

- 配置防盗链
  - 设置页面下图片

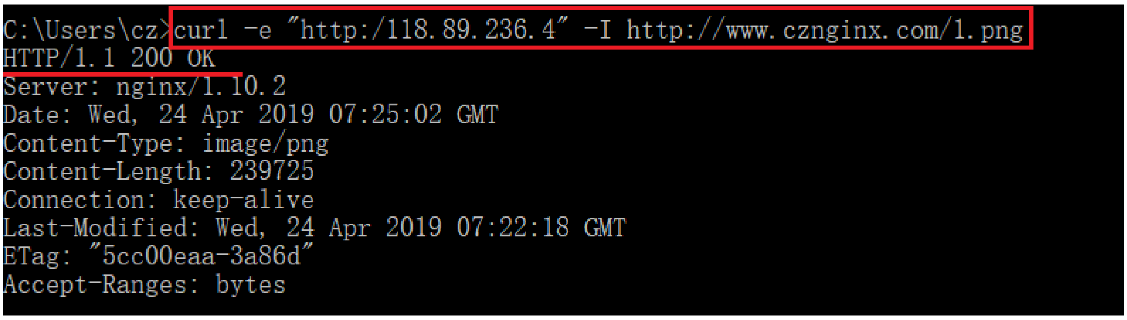


- 配置防盗链
  - 非法



```
C:\Users\cz>curl -e "http://www.baidu.com" -I http://www.cznginx.com/1.png
HTTP/1.1 403 Forbidden
Server: nginx/1.10.2
Date: Wed, 24 Apr 2019 07:24:14 GMT
Content-Type: text/html
Content-Length: 169
Connection: keep-alive
```

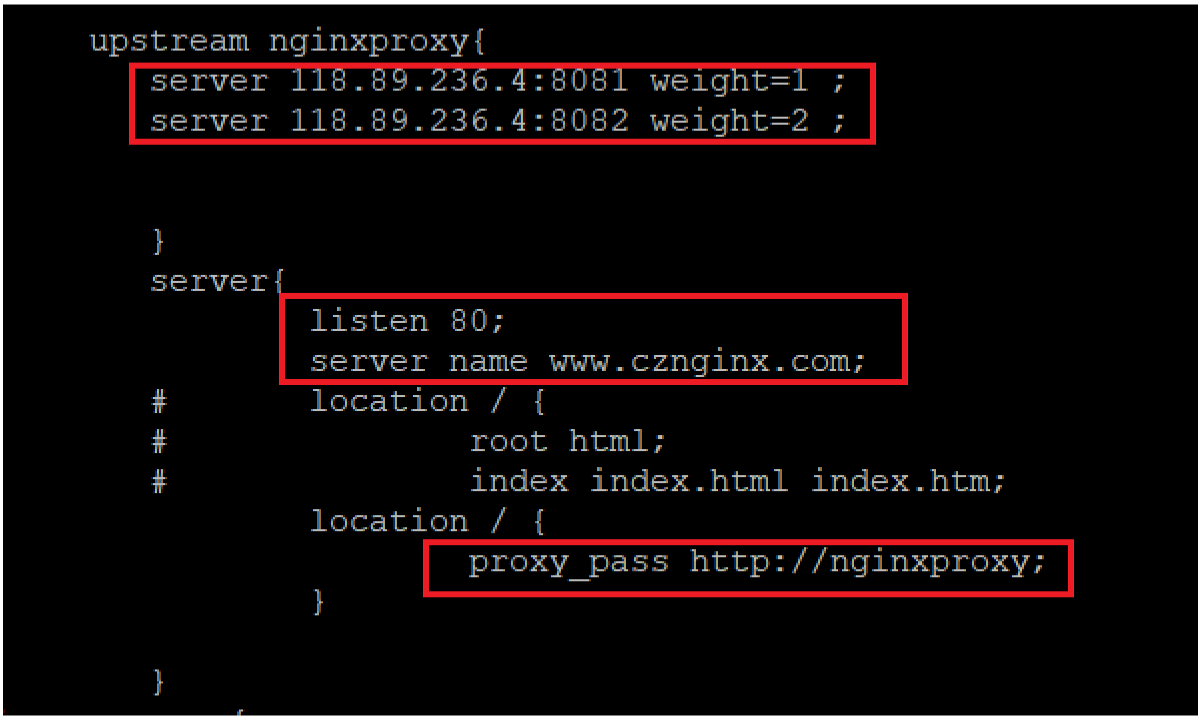
  - 合法



```
C:\Users\cz>curl -e "http:/118.89.236.4" -I http://www.cznginx.com/1.png
HTTP/1.1 200 OK
Server: nginx/1.10.2
Date: Wed, 24 Apr 2019 07:25:02 GMT
Content-Type: image/png
Content-Length: 239725
Connection: keep-alive
Last-Modified: Wed, 24 Apr 2019 07:22:18 GMT
ETag: "5cc00eaa-3a86d"
Accept-Ranges: bytes
```

三、配置负载均衡

```
//把两个upstream合而为一，设置权重
    upstream nginxproxy{
        server 118.89.236.4:8081 weight=1 ;
        server 118.89.236.4:8082 weight=2 ;
    }
    server{
        listen 80;
        server_name www.cznginx.com;
        location / {
                proxy_pass http://nginxproxy;
        }
    }
```



```
upstream nginxproxy{
    server 118.89.236.4:8081 weight=1 ;
    server 118.89.236.4:8082 weight=2 ;


    }
    server{
            listen 80;
            server name www.cznginx.com;
    #       location / {
    #               root html;
    #               index index.html index.htm;
            location / {
                    proxy_pass http://nginxproxy;
            }

    }
```

- 发现

> 配置后每刷新三次，有两次访问到nginx02，一次nginx01，说明该权重设置并不是按概率随机的，而是通过计数器计数的方式决定下一次访问哪一个后端主机的

- 结合负载均衡算法
- 首先测定ip_hash,本机访问服务器时，被锁定在了nginx04

```
upstream nginxproxy{
   #least_conn;
    ip_hash;
    server 118.89.236.4:8081 weight=1 ;
    server 118.89.236.4:8082 weight=20 ;
    server 118.89.236.4:8083 weight=3;
    server 118.89.236.4:8084 weight=44;
    # max_fails=3 fail_timeout=10s;
}
```

- ip_hash和least_conn同用，冲突
- least_conn与weight配合使用,发现并没有什么大用，还是根据权重来，应该是客户端太少了(2台)，而tcp又是长连接，所以相当于每个虚拟主机都连了2个客户端，因此真正起作用的就是weight了。

```
upstream nginxproxy{
    least_conn;
    server 118.89.236.4:8081 weight=1 ;
    server 118.89.236.4:8082 weight=2 ;
    server 118.89.236.4:8083 weight=3;
    server 118.89.236.4:8084 weight=4 max_fails=3 fail_timeout=10s;
}
```

四、配置nginx实现动静分离

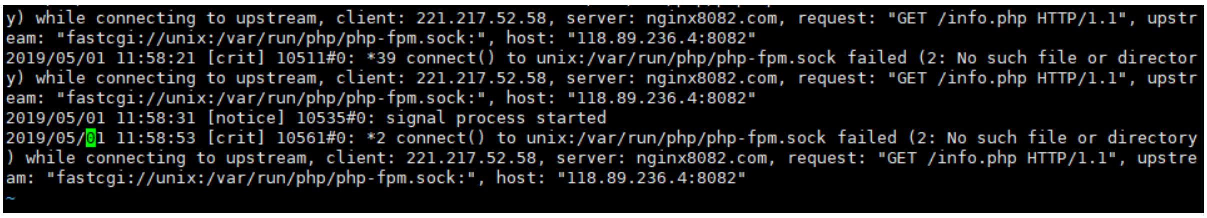
- 基本思路
  - 静态文件如jpg png html 等代理到提供静态资源的后端服务器
  - 动态文件如jsp php 等代理到提供动态资源的后端服务器
- 配置文件

```
server{
    listen 80;
    server_name www.cznginx.com;
    location / {
            proxy_pass http://nginxproxy;
    }
    location ~ .*.(gif|jpg|png|bmp|swf|css|js|html|htm) {
            proxy_pass http://nginx01;
    }
    location ~ .*.(php|asp|jsp|cgi|perl) {
            proxy_pass http://nginx02;
    }
}
```

- 安装php-fpm运行php动态文件
  - 安装 `yum install php php-mysql php-fpm`
  - 启动 `systemctl start php-fpm`
  - 在动态服务器server模块添加

```
 location ~ \.php$ {
    try_files $uri =404;
    fastcgi_pass unix:/var/run/php/php-fpm/php-fpm.sock;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    include fastcgi_params;
}
```

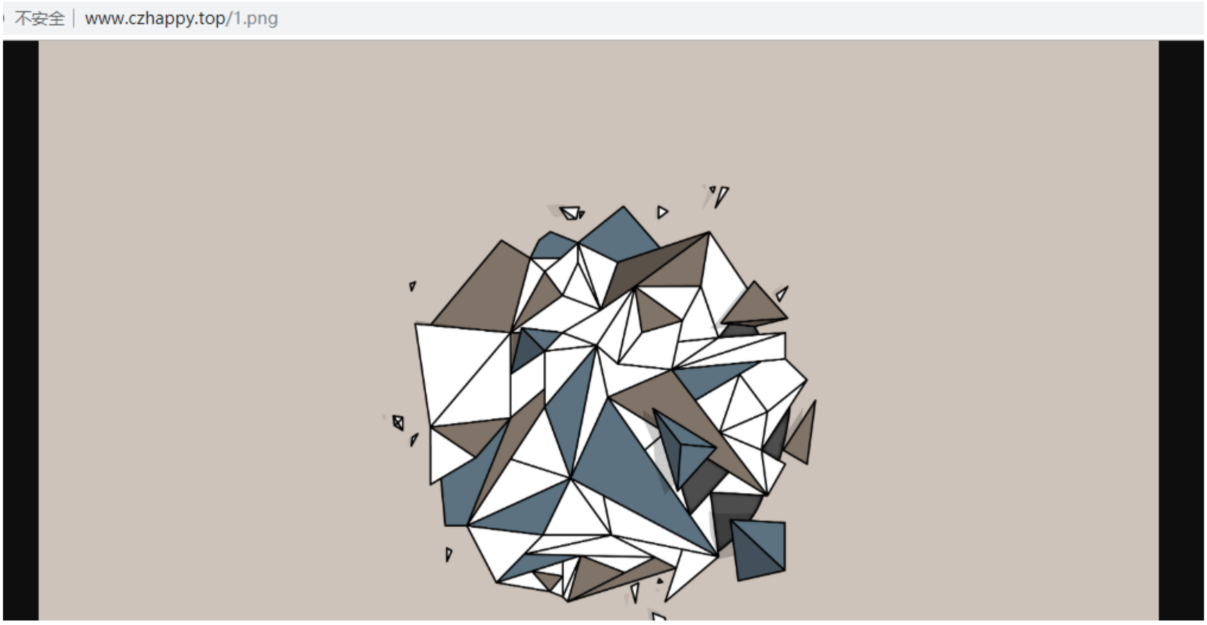
- 访问动态服务器失败，报错502Bad gateway
  - 首先排除防火墙问题，关闭防火墙仍然无效
  - 查看错误日志error.log



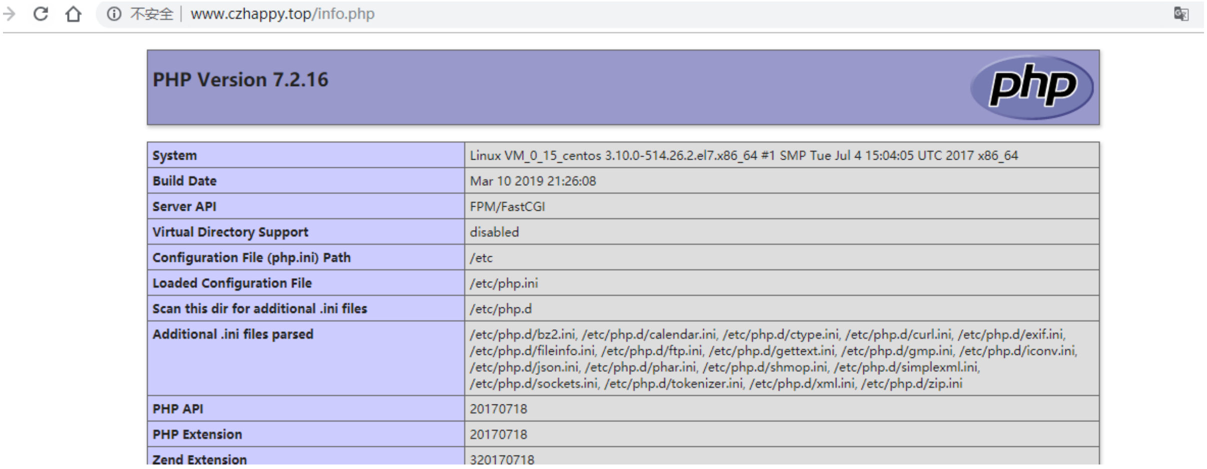
```
y) while connecting to upstream, client: 221.217.52.58, server: nginx8082.com, request: "GET /info.php HTTP/1.1", upstr
eam: "fastcgi://unix:/var/run/php/php-fpm.sock:", host: "118.89.236.4:8082"
2019/05/01 11:58:21 [crit] 10511#0: *39 connect() to unix:/var/run/php/php-fpm.sock failed (2: No such file or director
y) while connecting to upstream, client: 221.217.52.58, server: nginx8082.com, request: "GET /info.php HTTP/1.1", upstr
eam: "fastcgi://unix:/var/run/php/php-fpm.sock:", host: "118.89.236.4:8082"
2019/05/01 11:58:31 [notice] 10535#0: signal process started
2019/05/01 11:58:53 [crit] 10561#0: *2 connect() to unix:/var/run/php/php-fpm.sock failed (2: No such file or directory
) while connecting to upstream, client: 221.217.52.58, server: nginx8082.com, request: "GET /info.php HTTP/1.1", upstre
am: "fastcgi://unix:/var/run/php/php-fpm.sock:", host: "118.89.236.4:8082"
~
```

    - 找到错误，php-fpm.sock路径不正确，找到本机该文件路径/var/run/php-fpm/php-fpm.sock;修改重启服务即可

- 结果
    - 访问静态资源



    - 访问动态资源



PHP Version 7.2.16

System	Linux VM_0_15_centos 3.10.0-514.26.2.el7.x86_64 #1 SMP Tue Jul 4 15:04:05 UTC 2017 x86_64
Build Date	Mar 10 2019 21:26:08
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/bz2.ini, /etc/php.d/calendar.ini, /etc/php.d/ctype.ini, /etc/php.d/curl.ini, /etc/php.d/exif.ini, /etc/php.d/fileinfo.ini, /etc/php.d/ftp.ini, /etc/php.d/gettext.ini, /etc/php.d/gmp.ini, /etc/php.d/iconv.ini, /etc/php.d/json.ini, /etc/php.d/phar.ini, /etc/php.d/shmop.ini, /etc/php.d/simplexml.ini, /etc/php.d/sockets.ini, /etc/php.d/tokenizer.ini, /etc/php.d/xml.ini, /etc/php.d/zip.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718

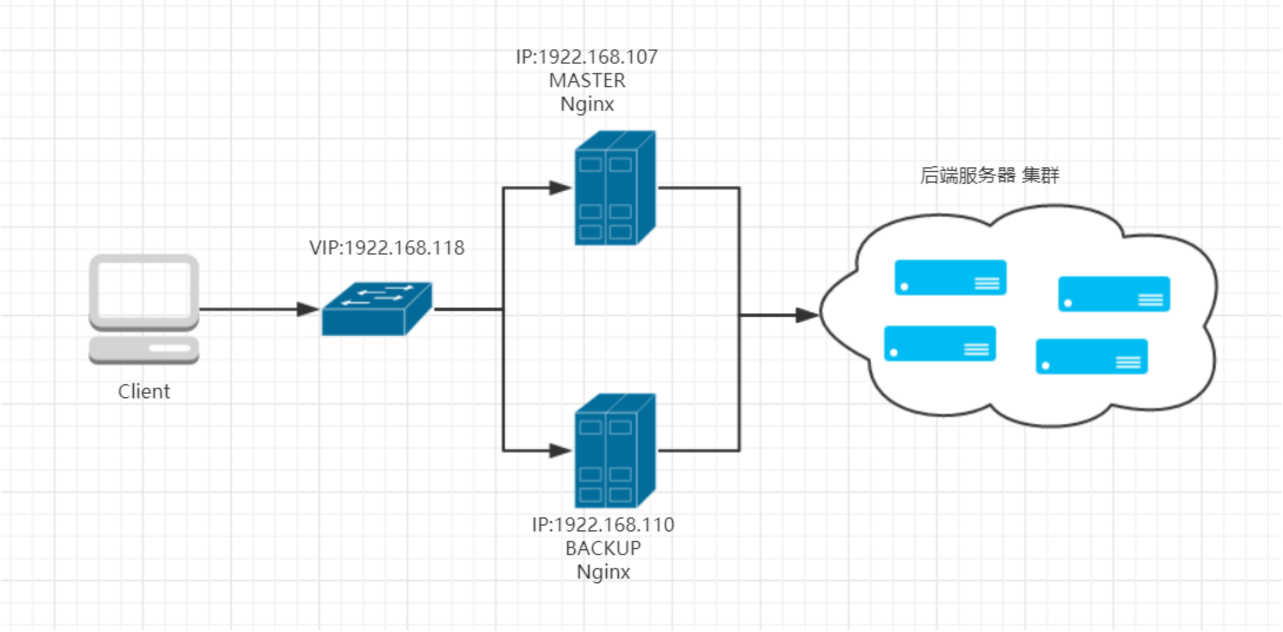
    - 注：这两种资源不在同一个后端服务器上，但是通过正则表达式识别url参数可以代理到相应的后端服务器进行请求。

配置keepalived，实验和观察节点掉线和上线情况。

- keepalived安装

  - `yum -y install keepalived`

- 实验架构图



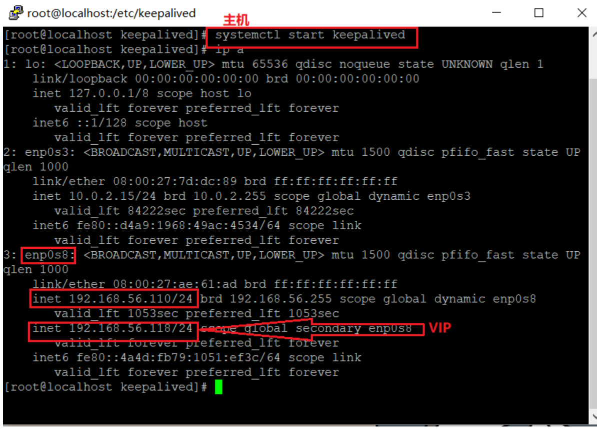
- 在两台结点都修改keepalived的配置文件，下面是备机的keepalived.conf,主机不用改

```
! Configuration File for keepalived

global_defs {
   notification_email {
     acassen@firewall.loc
     failover@firewall.loc
     sysadmin@firewall.loc
   }
   notification_email_from Alexandre.Cassen@firewall.loc
   smtp_server 192.168.200.1
   smtp_connect_timeout 30
   router_id LVS_02 # server id 要改 唯一性
   vrrp_skip_check_adv_addr
   vrrp_strict
   vrrp_garp_interval 0
   vrrp_gna_interval 0
}

vrrp_instance VI_1 {
    state BACKUP # 角色要改
    interface enp0s8 #根据自己本机的网络接口，有的是eth0
    virtual_router_id 51
    priority 80 #优先级小
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1112
    }
    virtual_ipaddress {
        192.168.56.118/24 #自己设定VIP
    }
}
```

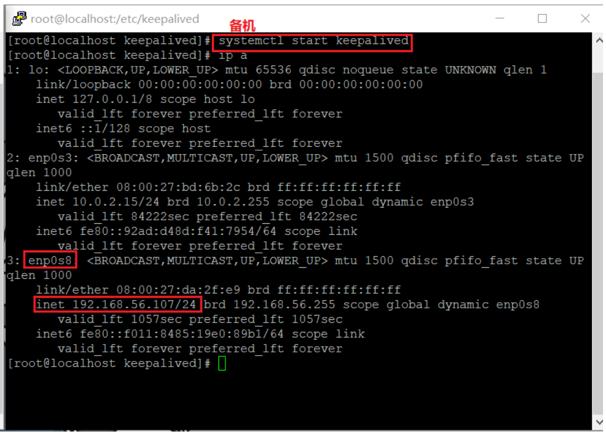
- 实验结果
  - 第一次实验，失败，主备机都存在虚拟IP，怀疑二者不能互通，应该是防火墙的问题，开启vrrp协议

    ```
    firewall-cmd --direct --permanent --add-rule ipv4 filter INPUT 0  --protocol vrrp -j ACCEP

    firewall-cmd --reload
    ```

  - 第二次实验成功，结点1，2都开启keepalived，使用ip a查看网卡enp0s8时只能在主机MASTER看到虚拟IP 192.68.56.118,关闭主机，在备机上ip a,立刻能查看到虚拟IP 192.68.56.118
  - 主备正常情况

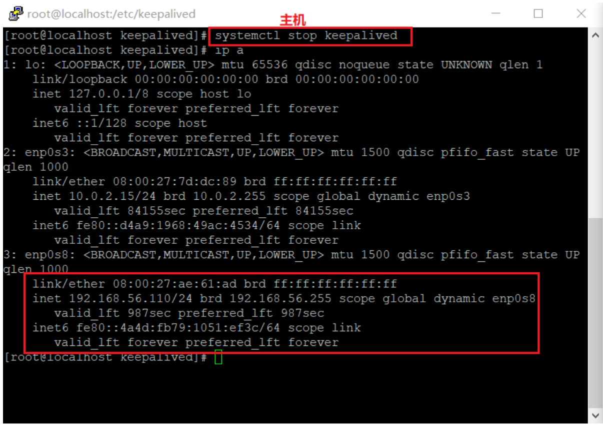


```
[root@localhost keepalived]# systemctl start keepalived
[root@localhost keepalived]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:7d:dc:89 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 84222sec preferred_lft 84222sec
    inet6 fe80::d4a9:1968:49ac:4534/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:ae:61:ad brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.110/24 brd 192.168.56.255 scope global dynamic enp0s8
       valid_lft 1053sec preferred_lft 1053sec
    inet 192.168.56.118/24 scope global secondary enp0s8
       valid_lft forever preferred_lft forever
    inet6 fe80::4a4d:fb79:1051:ef3c/64 scope link
       valid_lft forever preferred_lft forever
[root@localhost keepalived]#
```

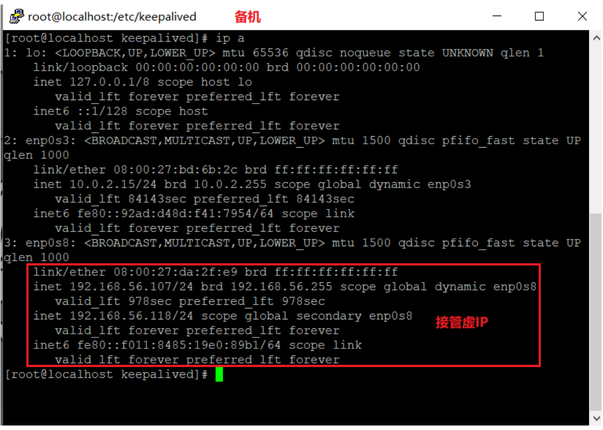


```
[root@localhost keepalived]# systemctl start keepalived
[root@localhost keepalived]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:bd:6b:2c brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 84222sec preferred_lft 84222sec
    inet6 fe80::92ad:d48d:f41:7954/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:da:2f:e9 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.107/24 brd 192.168.56.255 scope global dynamic enp0s8
       valid_lft 1057sec preferred_lft 1057sec
    inet6 fe80::f011:8485:19e0:89b1/64 scope link
       valid_lft forever preferred_lft forever
[root@localhost keepalived]#
```

  - 主机MASTER宕机,备机接管



```
[root@localhost keepalived]# systemctl stop keepalived
[root@localhost keepalived]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:7d:dc:89 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 84155sec preferred_lft 84155sec
    inet6 fe80::d4a9:1968:49ac:4534/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:ae:61:ad brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.110/24 brd 192.168.56.255 scope global dynamic enp0s8
       valid_lft 987sec preferred_lft 987sec
    inet6 fe80::4a4d:fb79:1051:ef3c/64 scope link
       valid_lft forever preferred_lft forever
[root@localhost keepalived]#
```

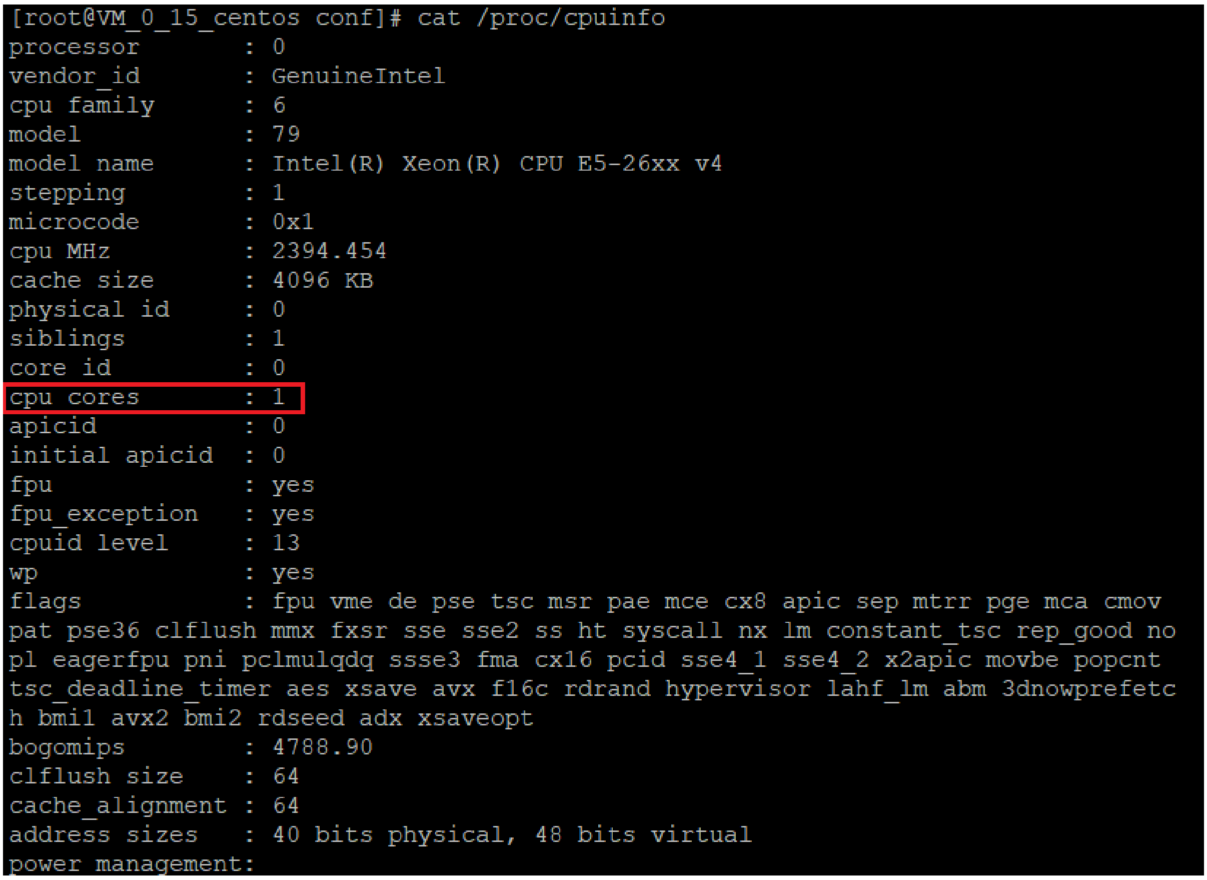


```
[root@localhost keepalived]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:bd:6b:2c brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 84143sec preferred_lft 84143sec
    inet6 fe80::92ad:d48d:f41:7954/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 08:00:27:da:2f:e9 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.107/24 brd 192.168.56.255 scope global dynamic enp0s8
       valid_lft 978sec preferred_lft 978sec
    inet 192.168.56.118/24 scope global secondary enp0s8
       valid_lft forever preferred_lft forever
    inet6 fe80::f011:8485:19e0:89b1/64 scope link
       valid_lft forever preferred_lft forever
[root@localhost keepalived]#
```

五、NGINX配置调优

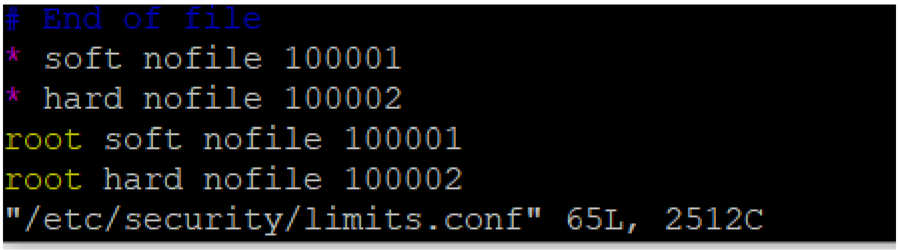
- 全局配置优化

- 系统信息查看



```
[root@VM_0_15_centos conf]# cat /proc/cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 79
model name      : Intel(R) Xeon(R) CPU E5-26xx v4
stepping        : 1
microcode       : 0x1
cpu MHz         : 2394.454
cache size      : 4096 KB
physical id     : 0
siblings        : 1
core id         : 0
cpu cores       : 1
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 13
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx lm constant_tsc rep_good no
pl eagerfpu pni pclmulqdq ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt
tsc_deadline_timer aes xsave avx f16c rdrand hypervisor lahf_lm abm 3dnowprefetc
h bmi1 avx2 bmi2 rdseed adx xsaveopt
bogomips        : 4788.90
clflush size    : 64
cache_alignment : 64
address sizes   : 40 bits physical, 48 bits virtual
power management:
```

- 最大打开文件数查看



```
# End of file
* soft nofile 100001
* hard nofile 100002
root soft nofile 100001
root hard nofile 100002
"/etc/security/limits.conf" 65L, 2512C
```

```
worker_processes  1;#工作进程数 和主机核心数一致
worker_rlimit_nofile 100000;#系统最大打开文件数
```

- event模块优化

```
events {
use epoll;#开启epoll网络模型高效
worker_connections  100000;#增大每个进程最大连接数
multi_accept on;#打开multi_accept
}
```

- http模块优化

  - 安全性

`server_tokens off; #在http 模块当中配置隐藏nginx版本号`

- 日志记录优化(减少磁盘I/O)

  ```
  #access_log  logs/access.log  main buffer=2k;#带缓冲的日志读写 防止写日志大量占用IO
  access_log off;#关闭访问日志
  error_log logs/error.log crit;#只记录严重错误
  ```

- 连接模块优化

  ```
  #connect module
  sendfile  on; #sendfile()是立即将数据从磁盘读到OS缓存。因为这种拷贝是在内核完成的，sendfile()要比组合read()和write()以及打开关闭丢弃缓冲更加有效
  tcp_nopush  on;#tcp_nopush 告诉nginx在一个数据包里发送所有头文件，而不一个接一个的发送
  tcp_nodelay on;#告诉nginx不要缓存数据，而是一段一段的发送-当需要及时发送数据时，就应该给应用设置这个属性，这样发送一小块数据信息时就不能立即得到返回值。
  keepalive_timeout  60;#超时自动断开连接
  client_header_timeout 30;#设置请求头和请求体(各自)的超时时间。
  client_body_timeout 30;
  reset_timedout_connection on;#告诉nginx关闭不响应的客户端连接。这将会释放那个客户端所占有的内存空间
  send_timeout 10;#指定客户端的响应超时时间
  ```

- 限制模块优化

  ```
  #limit module
  limit_conn_zone $binary_remote_addr zone=addr:5m; #limit_conn_zone 设置用于保存各种key（比如当前连接数）的共享内存的参数
  limit_conn addr 100;#limit_conn 为给定的key设置最大连接数。这里key是addr允许每一个IP地址最多同时打开有100个连接
  ```

- Gzip模块优化

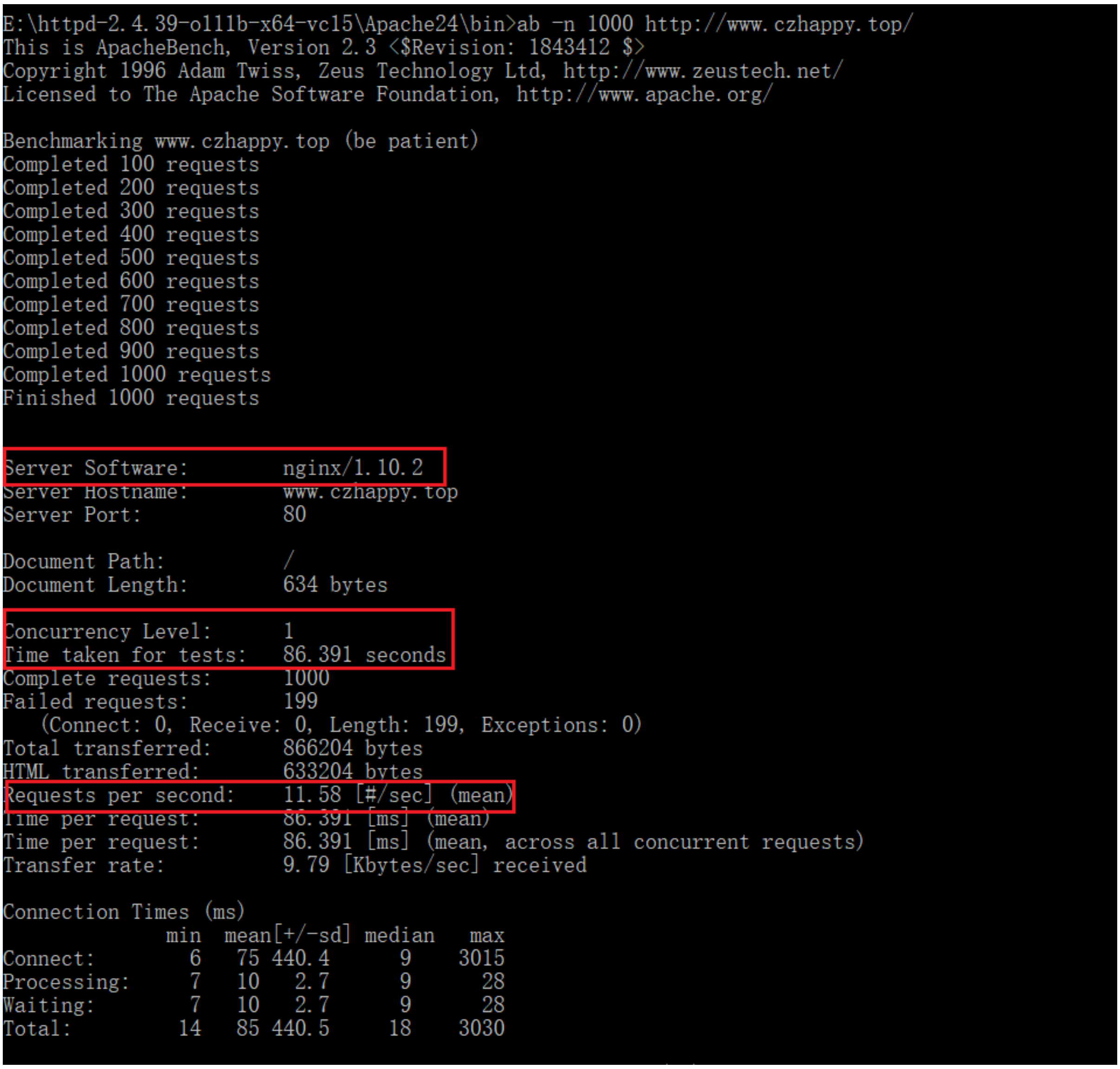
  ```
  #######
  #GZIP module
  gzip  on; #打开
  gzip_min_length 1k;#最小长度
  gzip_buffers 4 4k;#缓存
  gzip_http_version 1.0;
  gzip_comp_level 2;
  gzip_types text/plain text/css application/json application/x-javascript text/xml application/xml application/xml+rss text/javascript;
  gzip_vary on;
  ```

```
gzip_proxied expired no-cache no-store private auth;
gzip_disable "MSIE [1-6]\.";
```

- 缓存模块优化

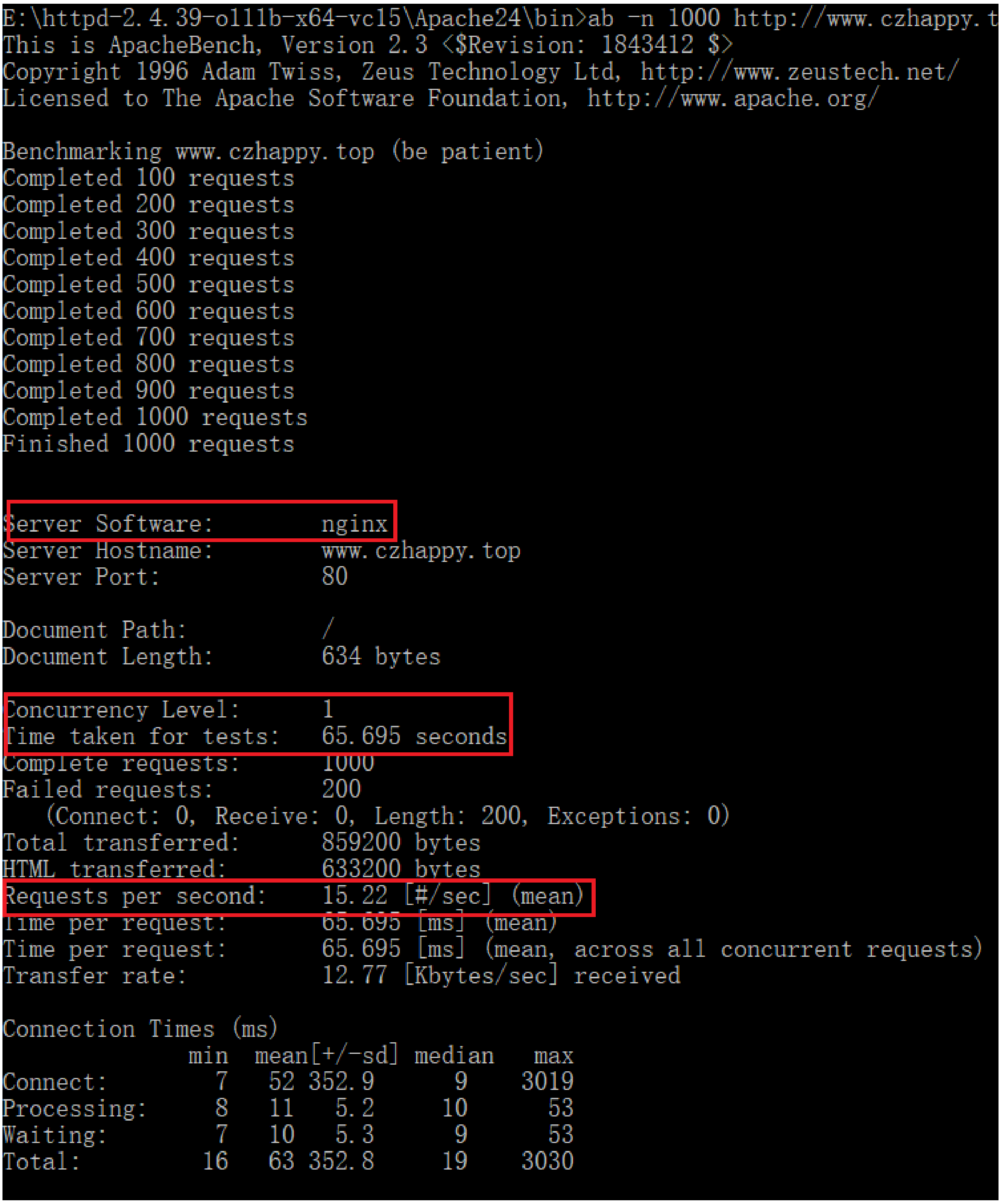
```
#cache module
open_file_cache max=100000 inactive=20s; #打开缓存的同时也指定了缓存最大数目，以及缓存的时间
open_file_cache_valid 30s; #指定检测正确信息的间隔时间。
open_file_cache_min_uses 2; #定义了open_file_cache中指令参数不活动时间期间里最小的文件数。
open_file_cache_errors on;#指定了当搜索一个文件时是否缓存错误信息
```

- 优化前后对比
  - 压测工具Apache24 ab工具
  - 优化前单用户1000次访问



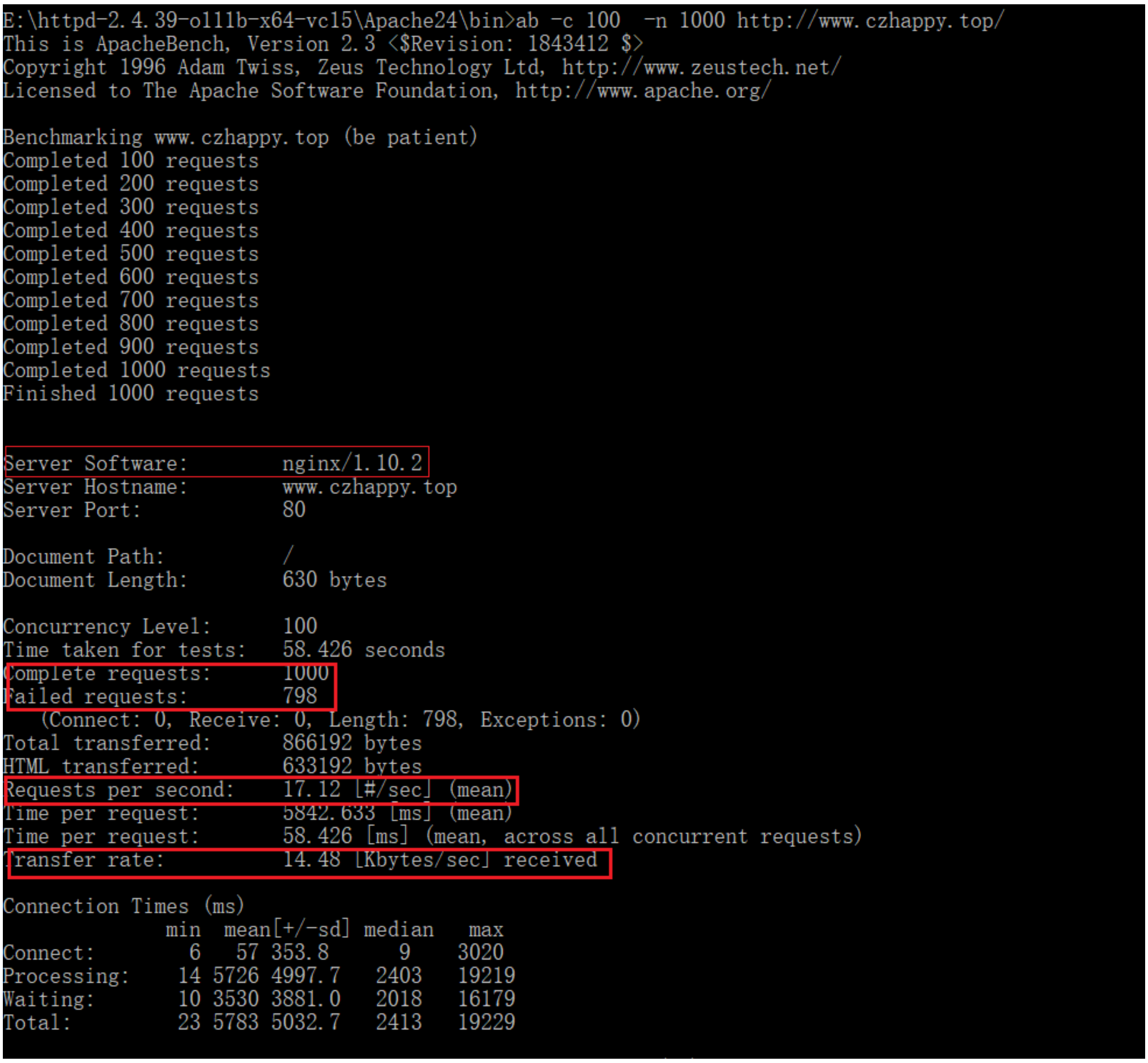
```
E:\httpd-2.4.39-o111b-x64-vc15\Apache24\bin>ab -n 1000 http://www.czhappy.top/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking www.czhappy.top (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests


Server Software:        nginx/1.10.2
Server Hostname:        www.czhappy.top
Server Port:            80

Document Path:          /
Document Length:        634 bytes

Concurrency Level:      1
Time taken for tests:   86.391 seconds
Complete requests:      1000
Failed requests:        199
   (Connect: 0, Receive: 0, Length: 199, Exceptions: 0)
Total transferred:      866204 bytes
HTML transferred:       633204 bytes
Requests per second:    11.58 [#/sec] (mean)
Time per request:       86.391 [ms] (mean)
Time per request:       86.391 [ms] (mean, across all concurrent requests)
Transfer rate:          9.79 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        6   75 440.4      9    3015
Processing:     7   10   2.7      9      28
Waiting:        7   10   2.7      9      28
Total:         14   85 440.5     18    3030
```

  - 优化后单用户1000次访问



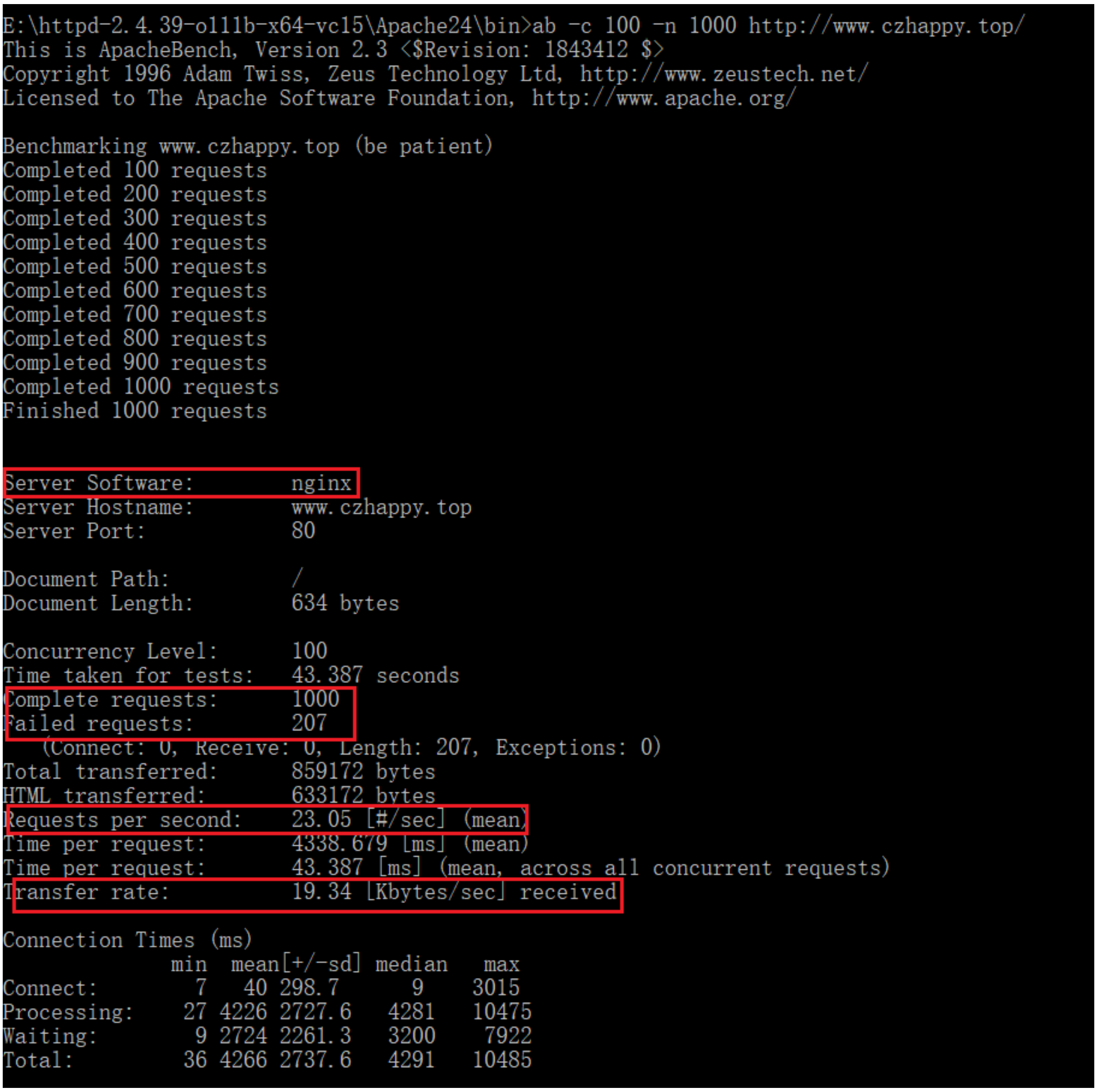
```
E:\httpd-2.4.39-o111b-x64-vc15\Apache24\bin>ab -n 1000 http://www.czhappy.t
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking www.czhappy.top (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests


Server Software:        nginx
Server Hostname:        www.czhappy.top
Server Port:            80

Document Path:          /
Document Length:        634 bytes

Concurrency Level:      1
Time taken for tests:   65.695 seconds
Complete requests:      1000
Failed requests:        200
   (Connect: 0, Receive: 0, Length: 200, Exceptions: 0)
Total transferred:      859200 bytes
HTML transferred:       633200 bytes
Requests per second:    15.22 [#/sec] (mean)
Time per request:       65.695 [ms] (mean)
Time per request:       65.695 [ms] (mean, across all concurrent requests)
Transfer rate:          12.77 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        7   52 352.9      9    3019
Processing:     8   11   5.2     10      53
Waiting:        7   10   5.3      9      53
Total:         16   63 352.8     19    3030
```

- 优化前并发用户数100，单个访问次数1000



```
E:\httpd-2.4.39-o111b-x64-vc15\Apache24\bin>ab -c 100  -n 1000 http://www.czhappy.top/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking www.czhappy.top (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests


Server Software:        nginx/1.10.2
Server Hostname:        www.czhappy.top
Server Port:            80

Document Path:          /
Document Length:        630 bytes

Concurrency Level:      100
Time taken for tests:   58.426 seconds
Complete requests:      1000
Failed requests:        798
   (Connect: 0, Receive: 0, Length: 798, Exceptions: 0)
Total transferred:      866192 bytes
HTML transferred:       633192 bytes
Requests per second:    17.12 [#/sec] (mean)
Time per request:       5842.633 [ms] (mean)
Time per request:       58.426 [ms] (mean, across all concurrent requests)
Transfer rate:          14.48 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        6   57 353.8      9    3020
Processing:    14 5726 4997.7   2403   19219
Waiting:       10 3530 3881.0   2018   16179
Total:         23 5783 5032.7   2413   19229
```

- 优化后并发用户数100，单个访问次数1000



```
E:\httpd-2.4.39-o111b-x64-vc15\Apache24\bin>ab -c 100 -n 1000 http://www.czhappy.top/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking www.czhappy.top (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests


Server Software:        nginx
Server Hostname:        www.czhappy.top
Server Port:            80

Document Path:          /
Document Length:        634 bytes

Concurrency Level:      100
Time taken for tests:   43.387 seconds
Complete requests:      1000
Failed requests:        207
   (Connect: 0, Receive: 0, Length: 207, Exceptions: 0)
Total transferred:      859172 bytes
HTML transferred:       633172 bytes
Requests per second:    23.05 [#/sec] (mean)
Time per request:       4338.679 [ms] (mean)
Time per request:       43.387 [ms] (mean, across all concurrent requests)
Transfer rate:          19.34 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        7   40 298.7      9    3015
Processing:    27 4226 2727.6   4281   10475
Waiting:        9 2724 2261.3   3200    7922
Total:         36 4266 2737.6   4291   10485
```

- 优化前后数据分析
  - 单用户共1000次访问，优化后
    - nginx版本号隐藏，避免access.log文件过于庞大占用硬盘存储空间
    - 连接失败率20%不变
    - 每秒处理请求(吞吐率)从11.58提高到15.22
    - 速度传输率从9.79KB/S提高到12.77KB/S
  - 100并发度共1000次访问，优化后
    - nginx版本号隐藏，避免access.log文件过于庞大占用硬盘存储空间
    - 连接失败率从79.8%降低至20.7%
    - 每秒处理请求(吞吐率)从17.12提高到23.05
    - 速度传输率从14.48KB/S提高到19.34KB/S

实验参考

- 安装nginx php

- centos7防火墙操作
- keepalived 操作
- rewrite和redirect
- yum 使用
- centos7配置网卡
- ab 命令
- nginx 优化配置