PHP-SESSION共享

环境：centos7.4虚拟机4台，关闭防火墙、SELinux，清空iptables规则，搭建好yum源

IP地址规划： proxy：192.168.1.1 nginx

Backend1:192.168.1.2 lnp

Backend2:192.168.1.3 lnp

Cache：192.168.1.4 memcached、redis

#PHP本地session

[root@backend1 ~]# cat /usr/local/nginx/html/index.php

<?php

$i="backend1";

echo $i;

echo "\n"

?>

[root@backend1 ~]# curl localhost

backend1

[root@backend1 ~]#

[root@backend2 ~]# cat /usr/local/nginx/html/index.php

<?php

$i="backend2";

echo $i;

echo "\n"

?>

[root@backend2 ~]# curl localhost

backend2

[root@backend2 ~]#

[root@proxy ~]# curl localhost

backend1

[root@proxy ~]# curl localhost

backend2

[root@proxy ~]#

#backend1和backend2设置php解析，proxy设置反向代理，基础架构搭建成功

[root@backend1 html]# vim index.php

[root@backend1 html]# cat index.php | grep bgcolor

<body bgcolor='red'>

[root@backend1 html]# vim home.php

[root@backend1 html]# cat home.php | grep bgcolor

<body bgcolor='red'>

[root@backend1 html]#

[root@backend2 ~]# rm -rf /usr/local/nginx/html/index.php

[root@backend2 ~]# tar -xf php-memcached-demo.tar.gz

[root@backend2 ~]# cd php-memcached-demo/

[root@backend2 php-memcached-demo]# cp -a \* /usr/local/nginx/html/

[root@backend2 php-memcached-demo]# cd /usr/local/nginx/html/

[root@backend2 html]# vim index.php

[root@backend2 html]# cat index.php | grep bgcolor

<body bgcolor='blue'>

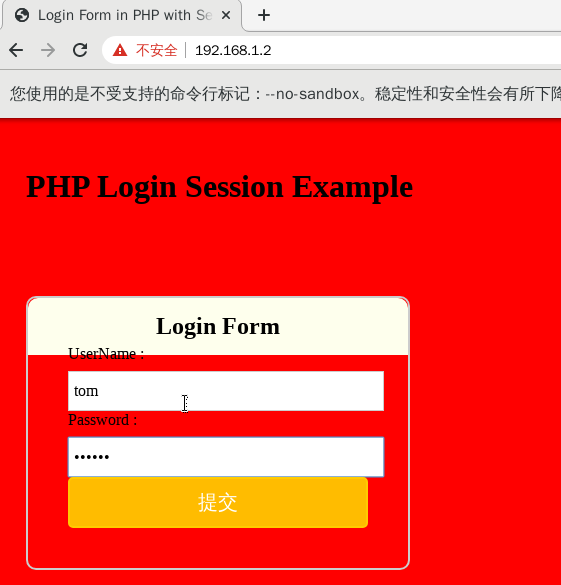
[root@backend2 html]# vim home.php

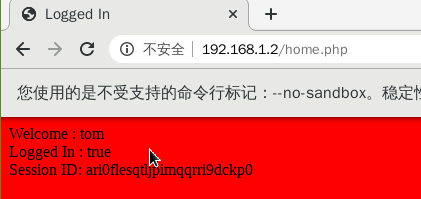
[root@backend2 html]# cat home.php | grep bgcolor

<body bgcolor='blue'>

[root@backend2 html]#

#测试访问

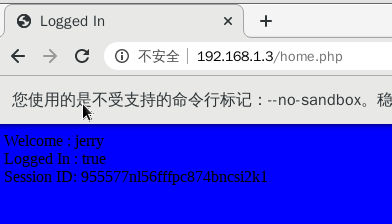




[root@backend1 ~]# ls /var/lib/php/session/

sess\_ari0flesqtljpimqqrri9dckp0





[root@backend2 ~]# ls /var/lib/php/session/

sess\_955577nl56fffpc874bncsi2k1

#架构构建成功，session存放在本地

#测试memcached

[root@cache ~]# yum -y install memcached

[root@cache ~]# systemctl start memcached.service

[root@backend1 ~]# yum -y install php-pecl-memcache.x86\_64

[root@backend1 ~]# systemctl restart php-fpm.service

[root@backend1 ~]# vim /usr/local/nginx/html/mem.php

[root@backend1 ~]# cat /usr/local/nginx/html/mem.php

<?php

$memcache=new Memcache;

$memcache->connect('192.168.1.4',11211) or die ('could not connect!! ');

$memcache->set('key', 'test');

$get\_values=$memcache->get('key');

echo $get\_values;

echo "\n";

?>

[root@backend1 ~]# curl localhost/mem.php

test

[root@backend2 ~]# yum -y install php-pecl-memcache.x86\_64

[root@backend2 ~]# vim /usr/local/nginx/html/mem.php

[root@backend2 ~]# cat /usr/local/nginx/html/mem.php

<?php

$memcache=new Memcache;

$memcache->connect('192.168.1.4',11211) or die ('could not connect!! ');

$memcache->set('key', 'test');

$get\_values=$memcache->get('key');

echo $get\_values;

echo "\n";

?>

[root@backend2 ~]# systemctl restart php-fpm.service

[root@backend2 ~]# curl localhost/mem.php

test

[root@backend2 ~]#

#测试使用memcached实现session共享

[root@backend1 ~]# vim /etc/php-fpm.d/www.conf

[root@backend1 ~]# tail -2 /etc/php-fpm.d/www.conf

php\_value[session.save\_handler] = memcache

php\_value[session.save\_path] = "tcp://192.168.1.4:11211"

[root@backend1 ~]# systemctl restart php-fpm.service

[root@backend1 ~]#

[root@backend2 ~]# vim /etc/php-fpm.d/www.conf

[root@backend2 ~]# tail -2 /etc/php-fpm.d/www.conf

php\_value[session.save\_handler] = memcache

php\_value[session.save\_path] = "tcp://192.168.1.4:11211"

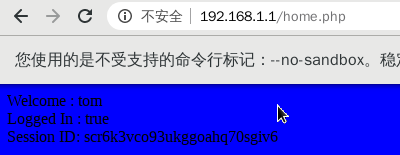
[root@backend2 ~]# systemctl restart php-fpm.service

[root@backend2 ~]#

#验证共享







#刷新页面背景色发生变化，但是session\_id不变，成功实现

#redis实现session共享

#安装redis

[root@cache ~]# ls

redis-5.0.7.tar.gz

[root@cache ~]# tar -xf redis-5.0.7.tar.gz

[root@cache ~]# yum -y install gcc

[root@cache ~]# cd redis-5.0.7/

[root@cache redis-5.0.7]# make

[root@cache redis-5.0.7]# make install PREFIX=/usr/local/redis/

[root@cache redis-5.0.7]# cp redis.conf /usr/local/redis/bin/

[root@cache redis-5.0.7]# cd /usr/local/redis/bin/

[root@cache bin]# vim redis.conf

[root@cache bin]# cat redis.conf | grep -v "^$" | grep -v "^#"

bind 0.0.0.0

protected-mode yes

port 6379

tcp-backlog 511

timeout 0

tcp-keepalive 300

daemonize yes

supervised no

pidfile /var/run/redis\_6379.pid

loglevel notice

logfile "/usr/local/redis/bin/redis.log"

databases 16

always-show-logo yes

save 900 1

save 300 10

save 60 10000

stop-writes-on-bgsave-error yes

rdbcompression yes

rdbchecksum yes

dbfilename dump.rdb

dir /usr/local/redis/bin/

replica-serve-stale-data yes

replica-read-only yes

repl-diskless-sync no

repl-diskless-sync-delay 5

repl-disable-tcp-nodelay no

replica-priority 100

lazyfree-lazy-eviction no

lazyfree-lazy-expire no

lazyfree-lazy-server-del no

replica-lazy-flush no

appendonly no

appendfilename "appendonly.aof"

appendfsync everysec

no-appendfsync-on-rewrite no

auto-aof-rewrite-percentage 100

auto-aof-rewrite-min-size 64mb

aof-load-truncated yes

aof-use-rdb-preamble yes

lua-time-limit 5000

slowlog-log-slower-than 10000

slowlog-max-len 128

latency-monitor-threshold 0

notify-keyspace-events ""

hash-max-ziplist-entries 512

hash-max-ziplist-value 64

list-max-ziplist-size -2

list-compress-depth 0

set-max-intset-entries 512

zset-max-ziplist-entries 128

zset-max-ziplist-value 64

hll-sparse-max-bytes 3000

stream-node-max-bytes 4096

stream-node-max-entries 100

activerehashing yes

client-output-buffer-limit normal 0 0 0

client-output-buffer-limit replica 256mb 64mb 60

client-output-buffer-limit pubsub 32mb 8mb 60

hz 10

dynamic-hz yes

aof-rewrite-incremental-fsync yes

rdb-save-incremental-fsync yes

[root@cache bin]# ./redis-server redis.conf

[root@cache bin]# netstat -antpu | grep redis

tcp 0 0 0.0.0.0:6379 0.0.0.0:\* LISTEN 5629/./redis-server

[root@cache bin]# ./redis-cli info server

# Server

redis\_version:5.0.7

redis\_git\_sha1:00000000

redis\_git\_dirty:0

redis\_build\_id:a75a13aa197e2e91

redis\_mode:standalone

os:Linux 3.10.0-693.el7.x86\_64 x86\_64

arch\_bits:64

multiplexing\_api:epoll

atomicvar\_api:atomic-builtin

gcc\_version:4.8.5

process\_id:5629

run\_id:fcab6f7c335ad9802ff6fa1e7a346b893385e527

tcp\_port:6379

uptime\_in\_seconds:39

uptime\_in\_days:0

hz:10

configured\_hz:10

lru\_clock:15778207

executable:/usr/local/redis/bin/./redis-server

config\_file:/usr/local/redis/bin/redis.conf

#安装php-redis扩展

[root@backend2 ~]# ls phpredis-2.2.4.tar.gz

phpredis-2.2.4.tar.gz

[root@backend2 ~]# yum -y install php-devel

[root@backend2 ~]# tar -xf phpredis-2.2.4.tar.gz

[root@backend2 ~]# cd phpredis-2.2.4/

[root@backend2 phpredis-2.2.4]# phpize

Configuring for:

PHP Api Version: 20100412

Zend Module Api No: 20100525

Zend Extension Api No: 220100525

[root@backend2 phpredis-2.2.4]# ./configure --with-php-config=/usr/bin/php-config

[root@backend2 phpredis-2.2.4]# make && make install

[root@backend2 phpredis-2.2.4]# ls /usr/lib64/php/modules/redis.so

/usr/lib64/php/modules/redis.so

[root@backend2 phpredis-2.2.4]# cd

[root@backend2 ~]# php –m | grep redis

[root@backend2 ~]# vim /etc/php.d/redis.ini

[root@backend2 ~]# cat /etc/php.d/redis.ini

extension = "redis.so"

[root@backend2 ~]# systemctl restart php-fpm.service

[root@backend2 ~]# php -m | grep redis

redis

[root@backend2 ~]#

[root@backend1 ~]# ls phpredis-2.2.4.tar.gz

phpredis-2.2.4.tar.gz

[root@backend1 ~]# yum -y install php-devel

[root@backend1 ~]# tar -xf phpredis-2.2.4.tar.gz

[root@backend1 ~]# cd phpredis-2.2.4/

[root@backend1 phpredis-2.2.4]# phpize

Configuring for:

PHP Api Version: 20100412

Zend Module Api No: 20100525

Zend Extension Api No: 220100525

[root@backend1 phpredis-2.2.4]# ./configure --with-php-config=/usr/bin/php-config

[root@backend1 phpredis-2.2.4]# make && make install

[root@backend1 phpredis-2.2.4]# ls /usr/lib64/php/modules/redis.so

/usr/lib64/php/modules/redis.so

[root@backend1 phpredis-2.2.4]# cd

[root@backend1 ~]# php -m | grep -i redis

[root@backend1 ~]# vim /etc/php.d/redis.ini

[root@backend1 ~]# cat /etc/php.d/redis.ini

extension = "redis.so"

[root@backend1 ~]# systemctl restart php-fpm.service

[root@backend1 ~]# php -m | grep -i redis

redis

[root@backend1 ~]#

#测试php连接redis

[root@backend1 ~]# vim /usr/local/nginx/html/redis.php

[root@backend1 ~]# cat /usr/local/nginx/html/redis.php

<?php

$redis = new Redis();

$redis->connect('192.168.1.4', 6379); //连接Redis

$redis->select(2);//选择数据库2

$redis->set( "testKey" , "Hello Redis"); //设置测试key

echo $redis->get("testKey");//输出value

echo "\n";

?>

[root@backend1 ~]# curl localhost/redis.php

Hello Redis

[root@backend1 ~]#

[root@backend2 ~]# vim /usr/local/nginx/html/redis.php

[root@backend2 ~]# cat /usr/local/nginx/html/redis.php

<?php

$redis = new Redis();

$redis->connect('192.168.1.4', 6379); //连接Redis

$redis->select(2);//选择数据库2

$redis->set( "testKey" , "Hello Redis"); //设置测试key

echo $redis->get("testKey");//输出value

echo "\n";

?>

[root@backend2 ~]# curl localhost/redis.php

Hello Redis

[root@backend2 ~]#

#验证redis实现session共享

[root@backend1 ~]# vim /etc/php-fpm.d/www.conf

[root@backend1 ~]# cat /etc/php-fpm.d/www.conf | tail -2

php\_value[session.save\_handler] = redis

php\_value[session.save\_path] = "tcp://192.168.1.4:6379"

[root@backend1 ~]# systemctl restart php-fpm.service

[root@backend1 ~]#

[root@backend2 ~]# vim /etc/php-fpm.d/www.conf

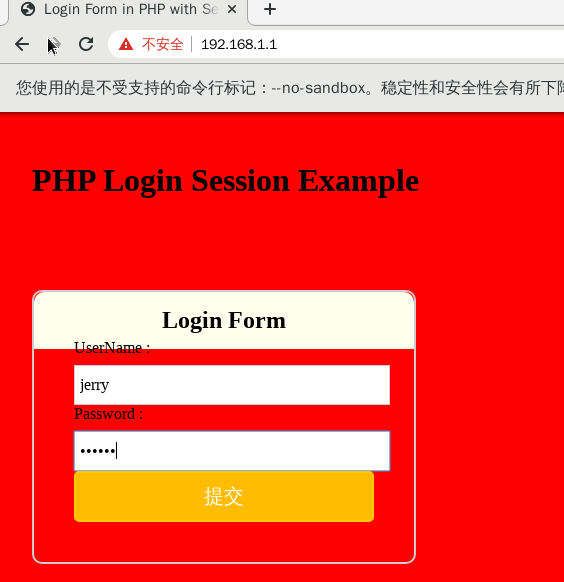
[root@backend2 ~]# cat /etc/php-fpm.d/www.conf | tail -2

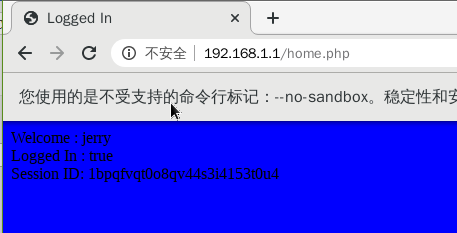
php\_value[session.save\_handler] = redis

php\_value[session.save\_path] = "tcp://192.168.1.4:6379"

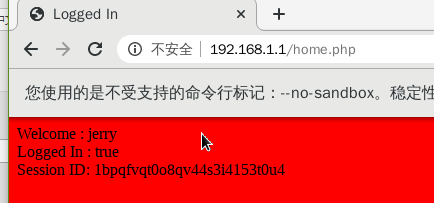
[root@backend2 ~]# systemctl restart php-fpm.service

[root@backend2 ~]#





#刷新页面



#成功实现使用redis实现session共享