一　memcache

非关系型数据库

验证时需要客户端主机安装telnet，远程memcached来验证服务器的功能：

add name 0 180 10 //变量不存在则添加

set name 0 180 10 //添加或替换变量

replace name 0 180 10 //替换

get name //读取变量

append name 0 180 10 //向变量中追加数据

delete name //删除变量

stats //查看状态

flush\_all //清空所有

提示：0表示不压缩，180为数据缓存时间，10为需要存储的数据字节数量。

１　构建memcached服务

1）使用yum安装软件包memcached

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

[root@proxy ~]# rpm -qa memcached

memcached-1.4.15-10.el7\_3.1.x86\_64

2) memcached配置文件（查看即可，不需要修改）

[root@proxy ~]# vim /usr/lib/systemd/system/memcached.service

ExecStart=/usr/bin/memcached -u $USER -p $PORT -m $CACHESIZE -c $MAXCONN $OPTIONS

[root@proxy ~]# vim /etc/sysconfig/memcached

PORT="11211"

USER="memcached"

MAXCONN="1024"

CACHESIZE="64"　　　　　　　　　　　　　＃单位m

OPTIONS=""

3）启动服务并查看网络连接状态验证是否开启成功：

netstat命令可以查看系统中启动的端口信息，该命令常用选项如下：

-a显示所有端口的信息

-n以数字格式显示端口号

-t显示TCP连接的端口

-u显示UDP连接的端口

-l显示服务正在监听的端口信息，如httpd启动后，会一直监听80端口

-p显示监听端口的服务名称是什么（也就是程序名称）

注意：在RHEL7系统中，使用ss命令可以替代netstat，功能与选项一样。

[root@proxy ~]# systemctl start memcached

[root@proxy ~]# systemctl status memcached

[root@proxy ~]# netstat -anptu | grep memcached

tcp 0 0 0.0.0.0:11211 0.0.0.0:\* LISTEN 2839/memcached

tcp 0 0 :::11211 :::\* LISTEN 2839/memcached

udp 0 0 0.0.0.0:11211 0.0.0.0:\* 2839/memcached

udp 0 0 :::11211 :::\* 2839/memcached

[root@proxy ~]# setenforce 0

[root@proxy ~]# firewall-cmd --set-default-zone=trusted

2 使用telnet访问memcached服务器

1）使用yum安装telnet

[root@proxy ~]# yum -y install telnet

2)使用telnet连接服务器测试memcached服务器功能，包括增、删、改、查等操作。

[root@proxy ~]# telnet 192.168.4.5 11211

Trying 192.168.4.5...

……

##提示：0表示不压缩，180为数据缓存时间,单位秒，3为需要存储的数据字节数量。

set name 0 180 3 //定义变量，变量名称为name

plj //输入变量的值，值为plj

STORED

get name //获取变量的值

VALUE name 0 3 //输出结果

plj

END

##提示：0表示不压缩，180为数据缓存时间，3为需要存储的数据字节数量。

add myname 0 180 10 //新建，myname不存在则添加，存在则报错

set myname 0 180 10 //添加或替换变量

replace myname 0 180 10 //替换，如果myname不存在则报错

get myname //读取变量

append myname 0 180 10 //向变量中追加数据

delete myname //删除变量

stats //查看状态

flush\_all //清空所有

quit //退出登录

3　LNMP+memcached

1）使用yum安装基础依赖包

[root@proxy ~]# yum -y install gcc openssl-devel pcre-devel zlib-devel

.. ..

2）源码安装Nginx

[root@proxy ~]# tar -xf nginx-1.12.2.tar.gz

[root@proxy ~]# cd nginx-1.12.2

[root@proxy nginx-1.12.2]# ./configure \

> --with-http\_ssl\_module

[root@proxy nginx-1.12.2]# make && make install

3）安装MariaDB数据库

[root@proxy ~]# yum -y install mariadb mariadb-server mariadb-devel

4）安装PHP

[root@proxy ~]# yum -y install php php-mysql

[root@proxy ~]# yum -y install php-fpm-5.4.16-42.el7.x86\_64.rpm

5）修改Nginx配置文件

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

location / {

root html;

index index.php index.html index.htm;

}

location ~ \.php$ {

root html;

fastcgi\_pass 127.0.0.1:9000;

fastcgi\_index index.php;

# fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

include fastcgi.conf;

}

4　创建PHP页面，使用PHP语言测试memcached服务

1）部署测试页面

创建PHP首页文档/usr/local/nginx/html/index.php，测试页面可以参考lnmp\_soft/php\_scripts/mem.php：

[root@proxy ~]# vim /usr/local/nginx/html/test.php

<?php

$memcache=new Memcache; //创建memcache对象

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

$memcache->set('key','test'); //定义变量

$get\_values=$memcache->get('key'); //获取变量值

echo $get\_values;

?>

2）客户端测试（结果会失败）

客户端使用浏览器访问服务器PHP首页文档，检验对memcached的操作是否成功：

[root@client ~]# firefox http://192.168.4.5/test.php

注意：这里因为没有给PHP安装扩展包，默认PHP无法连接memcached数据库，需要给PHP安装扩展模块才可以连接memcached数据库。

3）为PHP添加memcache扩展

[root@proxy ~]# yum -y install php-pecl-memcache

[root@proxy ~]# systemctl restart php-fpm

4）客户端再次测试（结果会成功显示数据结果）

[root@client ~]# firefox <http://192.168.4.5/test.php>

5　PHP的本地Session信息

通过Nginx调度器负载后端两台Web服务器，实现以下目标：

部署Nginx为前台调度服务器

调度算法设置为轮询

后端为两台LNMP服务器

部署测试页面，查看PHP本地的Session信息

6　PHP实现session共享

沿用练习三，通过修改PHP-FPM配置文件，实现session会话共享，本案例需要在练习三的基础上实现：

配置PHP使用memcached服务器共享Session信息

客户端访问两台不同的后端Web服务器时，Session 信息一致

1）安装Memcached服务（如果192.168.4.5上已经有本软件包，此步骤可以忽略）

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

2）启动服务并查看网络连接状态验证是否开启成功：

[root@proxy ~]# systemctl restart memcached

[root@proxy ~]# netstat -anptu | grep memcached

tcp 0 0 0.0.0.0:11211 0.0.0.0:\* LISTEN 2839/memcached

tcp 0 0 :::11211 :::\* LISTEN 2839/memcached

udp 0 0 0.0.0.0:11211 0.0.0.0:\* 2839/memcached

udp 0 0 :::11211 :::\* 2839/memcached

3）关闭SELinux、防火墙

[root@proxy ~]# setenforce 0

[root@proxy ~]# firewall-cmd --set-default-zone=trusted

步骤二：在后端LNMP服务器上部署Session共享

注意：这些操作在两台后端Web服务器上均需要执行，以下操作以Web1（192.168.2.100）服务器为例。

1）为PHP添加memcache扩展

注意，因为后端两台web服务器(web1,web2)都需要连接memcached数据库，所以两台主机都需要安装PHP扩展模块(下面也web1为例)。

[root@web1 ~]# yum -y install php-pecl-memcache

2）修改PHP-FPM配置文件，并重启服务

注意，因为后端两台web服务器(web1,web2)都需要修改配置文件(下面也web1为例)。

[root@web1 ~]# vim /etc/php-fpm.d/www.conf //修改该配置文件的两个参数

//文件的最后2行

修改前效果如下:

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

php\_value[session.save\_path] = /var/lib/php/session

//原始文件，默认定义Sessoin会话信息本地计算机（默认在/var/lib/php/session）

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

修改后效果如下:

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

php\_value[session.save\_path] = "tcp://192.168.2.5:11211"

//定义Session信息存储在公共的memcached服务器上，主机参数中为memcache（没有d）

//通过path参数定义公共的memcached服务器在哪（服务器的IP和端口）

[root@web1 ~]# systemctl restart php-fpm

步骤三：客户端测试

客户端使用浏览器访问两台不同的Web服务器。

操作步骤与练习三一致，最终可以获得相关的Session ID信息。

7