**[Top](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html" \l "page_top_case)**

**NSD OPERATION DAY05**

1. [案例1：构建memcached服务](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html#case1)
2. [案例2：LNMP+memcached](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html#case2)
3. [案例3：使用Tomcat设置Session](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html#case3)
4. [案例4：Tomcat实现session共享](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html#case4)

**1 案例1：构建memcached服务**

**1.1 问题**

本案例要求先快速搭建好一台memcached服务器，并对memcached进行简单的添、删、改、查操作：

* 安装memcached软件，并启动服务
* 使用telnet测试memcached服务
* 对memcached进行增、删、改、查等操作

**1.2 方案**

使用1台RHEL7虚拟机作为memcached服务器（192.168.4.5）。

在RHEL7系统光盘中包含有memcached，因此需要提前配置yum源，即可直接使用yum安装，客户端测试时需要提前安装telnet远程工具。

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。

**1.3 步骤**

实现此案例需要按照如下步骤进行。

**步骤一：构建memcached服务**

1）使用yum安装软件包memcached

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum -y install memcached
2. [root@svr5 ~]# rpm -qa memcached
3. memcached-1.4.15-9.el7.x86\_64

[root@svr5 ~]# yum -y install memcached

[root@svr5 ~]# rpm -qa memcached

memcached-1.4.15-9.el7.x86\_64

2）启动服务并查看网络连接状态验证是否开启成功：

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# systemctl start memcached
2. [root@svr5 ~]# systemctl status memcached
3. [root@svr5 ~]# netstat -anptu | grep memcached
4. tcp    0    0 0.0.0.0:11211        0.0.0.0:\*        LISTEN        2839/memcached
5. tcp    0    0 :::11211            :::\*                LISTEN        2839/memcached
6. udp    0    0 0.0.0.0:11211        0.0.0.0:\*                    2839/memcached
7. udp    0    0 :::11211            :::\*                            2839/memcached

[root@svr5 ~]# systemctl start memcached

[root@svr5 ~]# systemctl status memcached

[root@svr5 ~]# 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

**步骤二：使用telnet访问memcached服务器**

1）使用yum安装telnet

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum –y install telnet

[root@svr5 ~]# 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。

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# telnet 192.168.4.5 11211
2. Trying 192.168.4.5...
3. ……
4. set name 0 180 3                //定义变量，变量名称为name
5. plj                            //输入变量的值，值为plj
6. STORED
7. get name                        //获取变量的值
8. VALUE name 0 3             //输出结果
9. plj
10. END
11. add myname 0 180 10            //新建，myname不存在则添加，存在则报错
12. set myname 0 180 10            //添加或替换变量
13. replace myname 0 180 10        //替换，如果myname不存在则报错
14. get myname                    //读取变量
15. append myname 0 180 10        //向变量中追加数据
16. delete myname                    //删除变量
17. stats                        //查看状态
18. flush\_all                        //清空所有
19. quit                            //退出登录

[root@svr5 ~]# telnet 192.168.4.5 11211

Trying 192.168.4.5...

……

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

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

**2 案例2：LNMP+memcached**

**2.1 问题**

沿用练习一，部署LNMP+memcached网站平台,通过PHP页面实现对memcached服务器的数据操作，实现以下目标：

1. 部署LNMP实现PHP动态网站架构
2. 为PHP安装memcache扩展
3. 创建PHP页面，并编写PHP代码，实现对memcached的数据操作

**2.2 方案**

使用2台RHEL7虚拟机，其中一台作为memcached及LNMP服务器（192.168.4.5）、另外一台作为测试用的Linux客户机（192.168.4.100），如图-1所示。

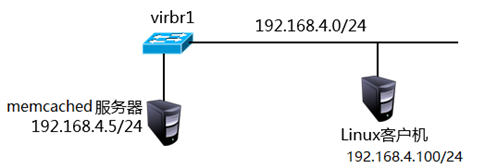


图-1

在RHEL7系统光盘中包含有我们需要的MariaDB、PHP，我们需要使用源码安装Nginx，使用RPM包安装FPM。另外如果希望使用PHP来操作memcached，注意必须要为PHP安装memcache扩展（php-pecl-memcache），否则PHP无法解析连接memcached的指令。客户端测试时需要提前安装telnet远程工具。

**2.3 步骤**

实现此案例需要按照如下步骤进行。

**步骤一：部署LNMP环境**

1）使用yum安装基础依赖包

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum -y install gcc openssl-devel pcre-devel zlib-devel
2. .. ..

[root@svr5 ~]# yum -y install gcc openssl-devel pcre-devel zlib-devel

.. ..

2）源码安装Nginx

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# useradd -s /sbin/nologin www
2. [root@svr5 ~]# tar -zxvf nginx-1.8.0.tar.gz
3. [root@svr5 ~]# cd nginx-1.8.0
4. [root@svr5 nginx-1.8.0]# ./configure \
5. > --user=www --group=www \
6. > --with-http\_ssl\_module
7. [root@svr5 ~]# make && make install

[root@svr5 ~]# useradd -s /sbin/nologin www

[root@svr5 ~]# tar -zxvf nginx-1.8.0.tar.gz

[root@svr5 ~]# cd nginx-1.8.0

[root@svr5 nginx-1.8.0]# ./configure \

> --user=www --group=www \

> --with-http\_ssl\_module

[root@svr5 ~]# make && make install

3）安装MariaDB数据库

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum –y install mariadb mariadb-server mariadb-devel

[root@svr5 ~]# yum –y install mariadb mariadb-server mariadb-devel

4）安装PHP

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum –y install php php-mysql
2. [root@svr5 ~]# yum –y localinstall php-fpm-5.4.16-36.el7\_1.x86\_64.rpm

[root@svr5 ~]# yum –y install php php-mysql

[root@svr5 ~]# yum –y localinstall php-fpm-5.4.16-36.el7\_1.x86\_64.rpm

5）为PHP添加memcache扩展

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum –y install php-pecl-memcache

[root@svr5 ~]# yum –y install php-pecl-memcache

6）修改Nginx配置文件

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# vim /usr/local/nginx/conf/nginx.conf
2. location / {
3. root html;
4. index index.php index.html index.htm;
5. }
6. location ~ \.php$ {
7. root html;
8. fastcgi\_pass 127.0.0.1:9000;
9. fastcgi\_index index.php;
10. # fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;
11. include fastcgi.conf;
12. }

[root@svr5 ~]# 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;

}

**步骤二：启动服务**

1）启动Nginx服务

这里需要注意的是，如果服务器上已经启动了其他监听80端口的服务软件（如httpd），则需要先关闭该服务，否则会出现冲突。

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# systemctl stop httpd                //如果该服务存在，则关闭该服务
2. [root@svr5 ~]# /usr/local/nginx/sbin/nginx
3. [root@svr5 ~]# netstat -utnlp | grep :80
4. tcp    0    0 0.0.0.0:80        0.0.0.0:\*        LISTEN        32428/nginx

[root@svr5 ~]# systemctl stop httpd //如果该服务存在，则关闭该服务

[root@svr5 ~]# /usr/local/nginx/sbin/nginx

[root@svr5 ~]# netstat -utnlp | grep :80

tcp 0 0 0.0.0.0:80 0.0.0.0:\* LISTEN 32428/nginx

2）启动MySQL服务

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# systemctl start mariadb
2. [root@svr5 ~]# systemctl status mariadb

[root@svr5 ~]# systemctl start mariadb

[root@svr5 ~]# systemctl status mariadb

3）启动PHP-FPM服务

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# systemctl start php-fpm
2. [root@svr5 ~]# systemctl status php-fpm

[root@svr5 ~]# systemctl start php-fpm

[root@svr5 ~]# systemctl status php-fpm

**步骤三：创建PHP页面，使用PHP语言测试memcached服务**

创建PHP首页文档/usr/local/nginx/html/index.php：

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# vim /usr/local/nginx/html/test.php
2. <?php
3. $memcache=new Memcache;                //创建memcache对象
4. $memcache->connect(‘localhost’,11211) or die (‘could not connect!!’);
5. $memcache->set(‘key’,‘test’);          //定义变量
6. $get\_values=$memcache->get(‘key’);     //获取变量值
7. echo $get\_values;
8. ?>

[root@svr5 ~]# 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;

?>

**步骤四：客户端测试**

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：

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

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

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

**3 案例3：使用Tomcat设置Session**

**3.1 问题**

通过Tomcat服务器设置Session回话ID，实现以下目标：

1. 部署Nginx为前台调度服务器
2. 调度算法设置为轮询
3. 后端JSP服务器为Tomcat
4. 编写JSP测试页面，显示SessionID信息

**3.2 方案**

使用4台RHEL7虚拟机，其中一台作为Nginx前端调度器服务器（eth0:192.168.4.5,eth1:172.16.2.5）、两台虚拟机部署为Tomcat服务器，分别为Web1服务器（192.168.2.100）和Web2服务器（192.168.2.200），另外一台作为测试用的Linux客户机（172.16.4.100），拓扑如图-2所示。

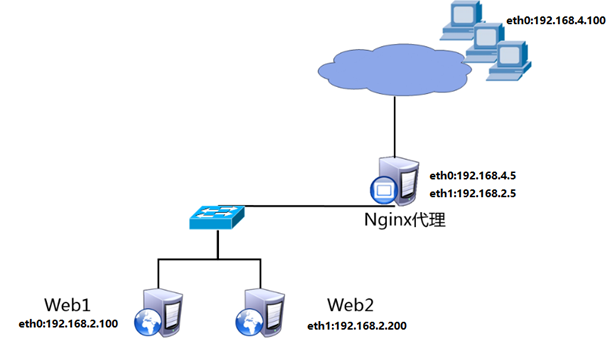


图-2

**3.3 步骤**

实现此案例需要按照如下步骤进行。

**步骤一：部署前端Nginx调度服务器**

1）使用源码安装nginx软件

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum -y install pcre-devel openssl-devel
2. [root@svr5 ~]# tar -zxvf nginx-1.8.0.tar.gz
3. [root@svr5 ~]# cd nginx-1.8.0
4. [root@svr5 nginx-1.8.0]# ./configure
5. [root@svr5 nginx-1.8.0]# make && make install

[root@svr5 ~]# yum -y install pcre-devel openssl-devel

[root@svr5 ~]# tar -zxvf nginx-1.8.0.tar.gz

[root@svr5 ~]# cd nginx-1.8.0

[root@svr5 nginx-1.8.0]# ./configure

[root@svr5 nginx-1.8.0]# make && make install

2）修改Nginx配置文件

Nginx配置文件中，通过upstream定义后端服务器地址池，默认调度策略为轮询，使用proxy\_pass调用upstream定义的服务器地址池：

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# vim /usr/local/nginx/conf/nginx.conf
2. .. ..
3. upstream tomcatgrp {
4. server 192.168.2.100:8080;
5. server 192.168.2.200:8080;
6. }
7. server {
8. listen 80;
9. server\_name localhost;
10. location / {
11. proxy\_pass http://tomcatgrp;
12. root html;
13. index index.html index.htm;
14. }
15. }

[root@svr5 ~]# vim /usr/local/nginx/conf/nginx.conf

.. ..

upstream tomcatgrp {

server 192.168.2.100:8080;

server 192.168.2.200:8080;

}

server {

listen 80;

server\_name localhost;

location / {

proxy\_pass http://tomcatgrp;

root html;

index index.html index.htm;

}

}

**3）启动Nginx服务**

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# /usr/local/nginx/sbin/nginx

[root@svr5 ~]# /usr/local/nginx/sbin/nginx

**步骤二：部署Tomcat**

***注意:以下部署Tomcat的操作，需要在两台后端服务器做相同的操作，下面我们以一台Web1服务器（192.168.2.100）为例，对Web2服务器执行相同操作即可。***

1）安装jdk环境

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr100 ~]# yum –y install java-1.8.0-openjdk

[root@svr100 ~]# yum –y install java-1.8.0-openjdk

2）安装tomcat软件

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr100 ~]# tar -zxvf apache-tomcat-8.0.30.tar.gz
2. [root@svr100 ~]# mv apache-tomcat-8.0.30 /usr/local/tomcat

[root@svr100 ~]# tar -zxvf apache-tomcat-8.0.30.tar.gz

[root@svr100 ~]# mv apache-tomcat-8.0.30 /usr/local/tomcat

3）启动tomcat服务，并查看端口信息

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr205 ~]# /usr/local/tomcat/bin/startup.sh //启动tomcat
2. [root@svr205 ~]# netstat -utnalp | grep :8080
3. tcp    0    0 :::8080        :::\*        LISTEN        3181/java

[root@svr205 ~]# /usr/local/tomcat/bin/startup.sh //启动tomcat

[root@svr205 ~]# netstat -utnalp | grep :8080

tcp 0 0 :::8080 :::\* LISTEN 3181/java

4）创建JSP测试页面，要求页面可以显示Session ID信息

通过在页面代码中添加JSP语句<%String s = session.getId();%>即可：

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# ~]# vim /usr/local/tomcat/webapps/ROOT/test.jsp
2. <html>
3. <body bgcolor="red">                              //指定网页背景颜色
4. <center>
5. <%String s = session.getId();%>     //获取SessionID
6. <%=s%>
7. <h1>tomcatA</h1>            //固定字串信息，Web2的信息应该为tomcatB
8. </center>
9. </body>
10. </html>

[root@svr5 ~]# ~]# vim /usr/local/tomcat/webapps/ROOT/test.jsp

<html>

<body bgcolor="red"> //指定网页背景颜色

<center>

<%String s = session.getId();%> //获取SessionID

<%=s%>

<h1>tomcatA</h1> //固定字串信息，Web2的信息应该为tomcatB

</center>

</body>

</html>

**步骤三：客户端访问测试效果**

1）客户端使用浏览器访问测试页面

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@pc01 ~]# firefox http://192.168.4.5/test.jsp

[root@pc01 ~]# firefox http://192.168.4.5/test.jsp

客户端两次访问的页面，应该对应的是Web1和Web2服务器返回的不同页面，并且会分别显示不同的Session ID信息，效果如图-3所示。



图-3

**4 案例4：Tomcat实现session共享**

**4.1 问题**

沿用练习三，通过为在Tomcat上部署msm（memcached-session-manager）实现session会话共享，本案例需要在练习三的基础上实现：

* 配置Tomcat支持memcached
* 客户端访问两台不同的后端Web服务器时，Session ID信息一致

**4.2 方案**

在练习三拓扑的基础上，Nginx服务器除了承担调度器外，还需要担任memcached数据库的角色，并在两台Tomcat服务器上部署msm实现session会话共享。拓扑结构如图-4所示。

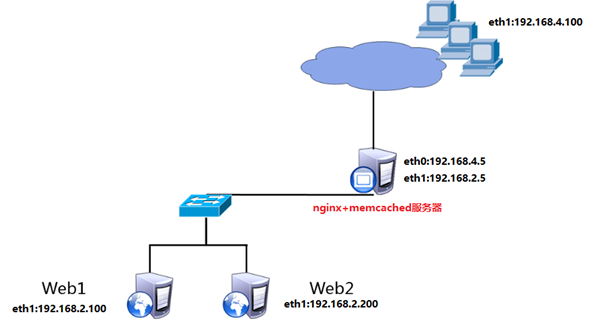


图-4

**4.3 步骤**

实现此案例需要按照如下步骤进行。

**步骤一：构建memcached服务**

1）安装Memcached服务

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# yum –y install memcached

[root@svr5 ~]# 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：

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr5 ~]# systemctl start memcached
2. [root@svr5 ~]# netstat -anptu | grep memcached
3. tcp    0    0 0.0.0.0:11211        0.0.0.0:\*        LISTEN        2839/memcached
4. tcp    0    0 :::11211            :::\*                LISTEN        2839/memcached
5. udp    0    0 0.0.0.0:11211        0.0.0.0:\*                    2839/memcached
6. udp    0    0 :::11211            :::\*                            2839/memcached

[root@svr5 ~]# systemctl start memcached

[root@svr5 ~]# 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

**步骤二：在Tomcat服务器上部署msm**

***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）安装msm

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr100 ~]# cd lnmp-soft/session
2. [root@svr100 session]# ls \*.jar
3. asm-5.1.jar minlog-1.3.0.jar
4. kryo-3.0.3.jar msm-kryo-serializer-1.9.2.jar
5. kryo-serializers-0.34.jar reflectasm-1.11.1.jar
6. memcached-session-manager-1.9.2.jar spymemcached-2.11.1.jar
7. memcached-session-manager-tc8-1.9.2.jar
8. [root@svr100 session]# cp \*.jar /usr/local/tomcat/lib/

[root@svr100 ~]# cd lnmp-soft/session

[root@svr100 session]# ls \*.jar

asm-5.1.jar minlog-1.3.0.jar

kryo-3.0.3.jar msm-kryo-serializer-1.9.2.jar

kryo-serializers-0.34.jar reflectasm-1.11.1.jar

memcached-session-manager-1.9.2.jar spymemcached-2.11.1.jar

memcached-session-manager-tc8-1.9.2.jar

[root@svr100 session]# cp \*.jar /usr/local/tomcat/lib/

2）修改Tomcat配置文件，连接memcached服务器

[copytextpop-up](http://pdf7.tarena.com.cn/tts8_source/ttsPage/LINUX/NSD_V05/OPERATION/DAY05/CASE/01/index.html)

1. [root@svr100 ~]# ~]# vim /usr/local/tomcat/conf/context.xml
2. <Context>
3. <Manager className="de.javakaffee.web.msm.MemcachedBackupSessionManager"
4. memcachedNodes="mem1:192.168.2.5:11211"
5. requestUriIgnorePattern=".\*\.(ico|png|gif|jpg|css|js)$"
6. transcoderFactoryClass="de.javakaffee.web.msm.JavaSerializationTranscoderFactory"/>
7. </Context>
8. [root@localhost ~]# /usr/local/tomcat/bin/shutdown.sh
9. [root@localhost ~]# /usr/local/tomcat/bin/startup.sh

[root@svr100 ~]# ~]# vim /usr/local/tomcat/conf/context.xml

<Context>

<Manager className="de.javakaffee.web.msm.MemcachedBackupSessionManager"

memcachedNodes="mem1:192.168.2.5:11211"

requestUriIgnorePattern=".\*\.(ico|png|gif|jpg|css|js)$"

transcoderFactoryClass="de.javakaffee.web.msm.JavaSerializationTranscoderFactory"/>

</Context>

[root@localhost ~]# /usr/local/tomcat/bin/shutdown.sh

[root@localhost ~]# /usr/local/tomcat/bin/startup.sh

**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。