**memcached原理**

基础知识

传统web架构的问题

许多web应用都将数据保存到RDBMS（关系型数据库mysql sqlserver oracle数据库 数据表 数据列 表和表的关系）中，应用服务器从中读取数据并在浏览器中显示

随着数据量的增大、访问的集中，就会出现RDBMS的负担加重、数据库响应恶化、网站显示延迟等重大影响

数据存储位置对比

性能

CPU缓存>内存>磁盘>数据库

价格

CPU缓存>内存>磁盘>数据库

memcached概述

memcached简介

memcached是高性能的分布式缓存服务器

用来集中缓存数据库查询结果，减少数据库访问次数，以提高动态web应用的响应速度

官方网站：<http://memcached.org/>

首次访问从RDBMS中取得数据保存到memcached中，第二次以后从memcached中取得数据数据显示页面

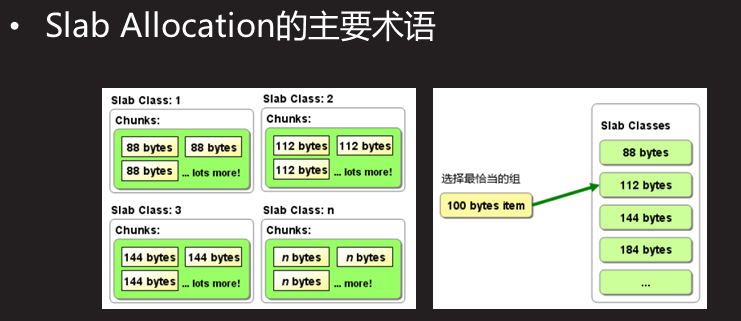
内存管理机制

传统内存分配机制

使用完通过分配的内存后回收内存，这种方式容易产生内存碎片并降低操作系统对内存的管理效率

slab allocation机制

它按照预先规定的大小，将分配的内存分割成特定长度的内存块，再把尺寸相同的内存块分成组，这些内存块不会释放，可以重复利用



memcached使用名为LRU机制来分配空间

删除“最近最少使用”的记录

当memcached的内存空间使用不足时，从最近未被使用的记录中搜索，并将其空间分配给新的记录

-M参数禁止LRU功能，内存用尽时memecached会返回错误，不建议使用memcached -M -m 1024

**部署memcached**

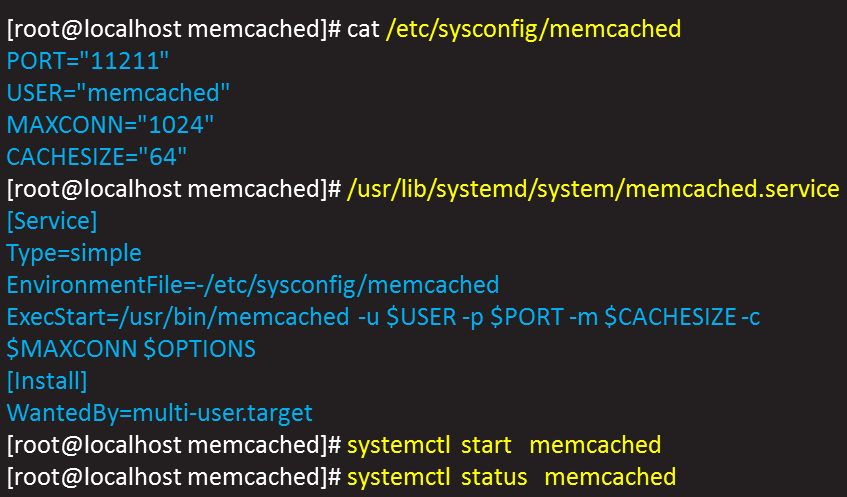
安装软件

安装memcached

yum -y install memcached

rpm -qa memecached

启动memcached



选项

-p指定memcached监听的端口号，默认11211

-u memcached程序运行时使用的用户身份必须是root用户

-m指定使用本机的多少物理内存存数据，默认是64M

-c memcached服务的最大连接数

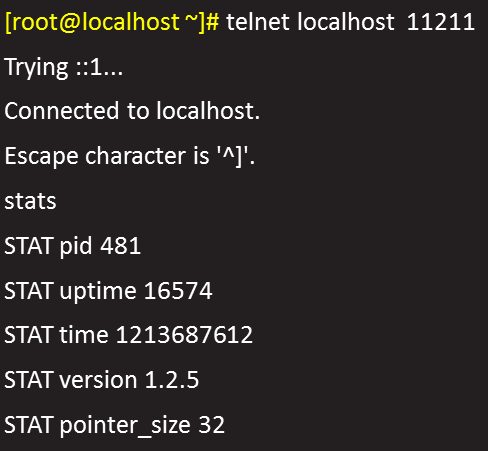
-n chunk size的最小空间是多少，单位为字节

-f chunk size大小增长的倍数，默认1.25倍

-d 在后台启动

测试memcached

查看memcached的内部状态





定义变量：变量名为name 0表示不压缩 180为缓存时间 3需要存储的字节数量

变量值：plj

获取变量的值

输出结果

退出登录



add新建，有了就会执行出错

set添加或执行变量

replace替换变量

get name读取变量

append向变量中追加数据

delete删除变量

stats查看状态

flush\_all清空所有

PHP+memcached

php+memcached介绍

部署LNMP

使用php页面，测试对memcached的读写操作，具体参考nginx+fastcgi

php无法直接操作memcached

需要安装memcache扩展

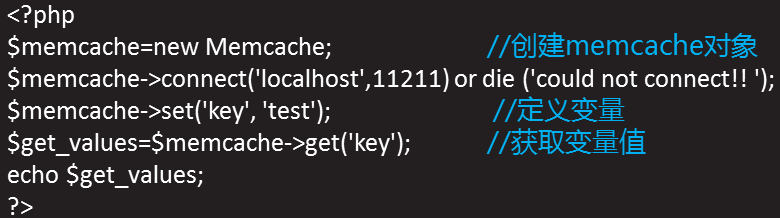
安装php的memcache扩展

yum install php-pecl-memcache

rpm -ql php-pecl-memcache

systemctl restart php-fpm

编写PHP代码进行测试



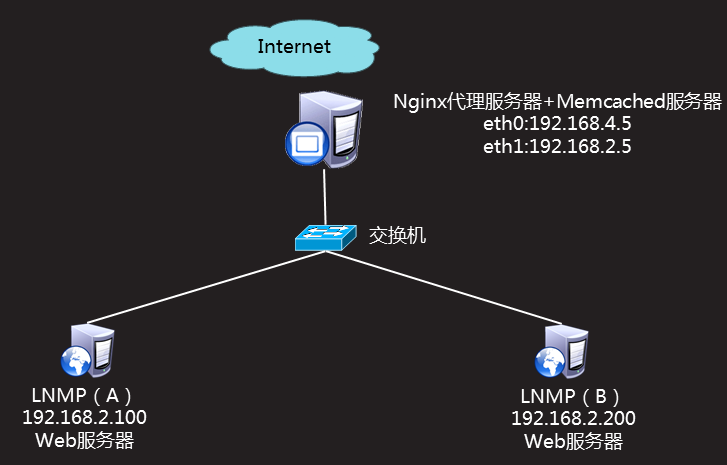
**session共享**

基本概念

session & cookies

session：存储在服务器端，保存用户名、密码等信息

cookies：由服务器下发给客户端，保存在客户端的一个文件里。保存的内容主要包括：sessionid、账户名、过期时间，路径和域



本地session

部署Nginx调度器

安装nginx软件

yum -y install pcre pcre-devel openssl-devel

tar -zxvf nginx-1.12.2.tar.gz

cd nginx-1.12.2.tar.gz

./configure

make && make install

安装memcached软件

yum -y install memcached

修改nginx配置文件并启动

upstream webs{

server 192.168.2.100:80;

server 192.168.2.200:80;

}

server{

location / {

proxy\_pass http://webs;

}

}

nginx -s reload

部署后端LNMP主机

nginx:

yum -y install pcre pcre-devel openssl-devel

tar -zxvf nginx-1.12.2.tar.gz

cd nginx-1.12.2

./configure

make && make install

mariadb

yum -y install mariadb mariadb-server mariadb-devel

php

yum -y install php php-mysql php-pecl-memcache

启动服务

nginx

systemctl restart php-fpm

systemctl restart mariadb

部署测试页面

注意：以上安装LNMP和启动服务需要在后端两台主机都操作

在后端两台主机都操作

cd lnmp-soft/php\_script/session/

tar -xf php-memcached-demo.tar.gz

cd php-memcached-demo

cp \* /usr/local/nginx/html

客户端访问

浏览器直接访问后端的LNMP主机2.100

firefox <http://192.168.2.100>

浏览器直接访问后端的LNMP主机2.200

firefox <http://192.168.2.200>

浏览器访问调度器，刷新页面后需要登录两次

firefox <http://192.168.4.5>

验证session

登录后端两台LNMP主机分别查看

ls /var/lib/php/session

实现session共享

修改PHP配置文件

vim /etc/php-fpm.d/www.conf

修改前：

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

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

修改后：

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

php\_value[session.save\_path]=”tcp://192.168.4.5:11211”



重启服务

systemctl restart php-fpm

测试ssession共享

浏览器访问调度器，刷新页面后，登录账户会被记录在memcached服务器

刷新页面，调度器且护眼服务器后，账户信息还在

两台后端服务器使用的是同一个账户

firefox http://192.168.4.5