

Linux高级运维

NSD OPERATION

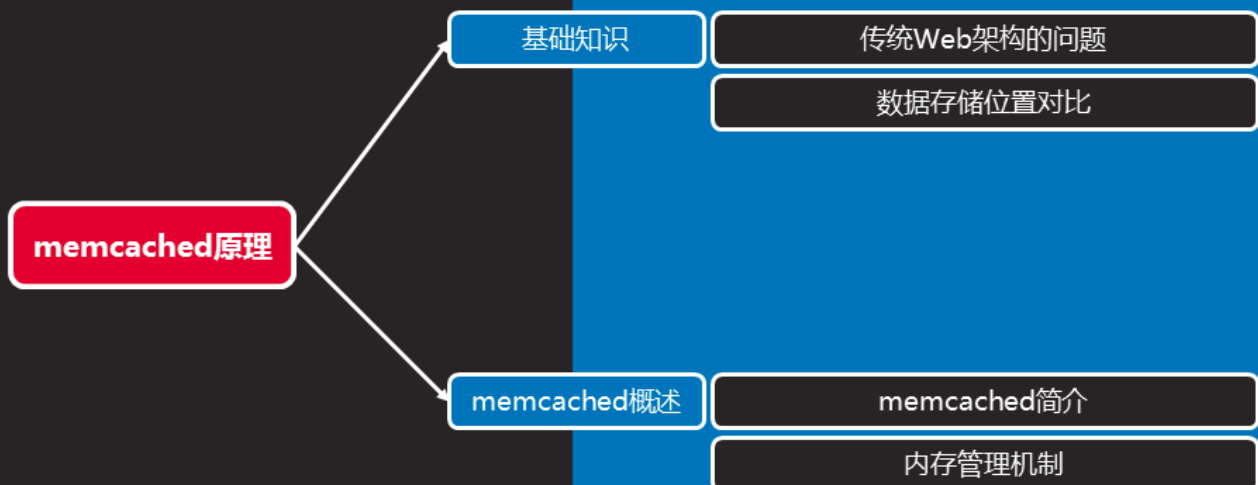
DAY04

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	memcached原理
	10:30 ~ 11:20	部署memcached
	11:30 ~ 12:20	
下午	14:00 ~ 14:50	Session共享
	15:00 ~ 15:50	
	16:00 ~ 16:50	
	17:00 ~ 17:30	总结和答疑



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

知识讲解

- 性能
 - CPU缓存 > 内存 > 磁盘 > 数据库
- 价格
 - CPU缓存 > 内存 > 磁盘 > 数据库

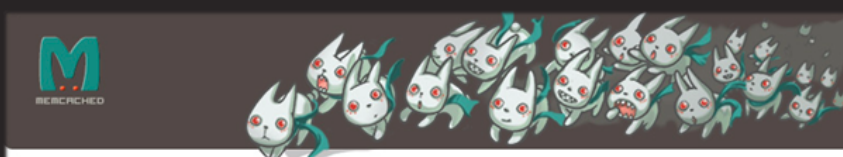


memcached概述

memcached简介

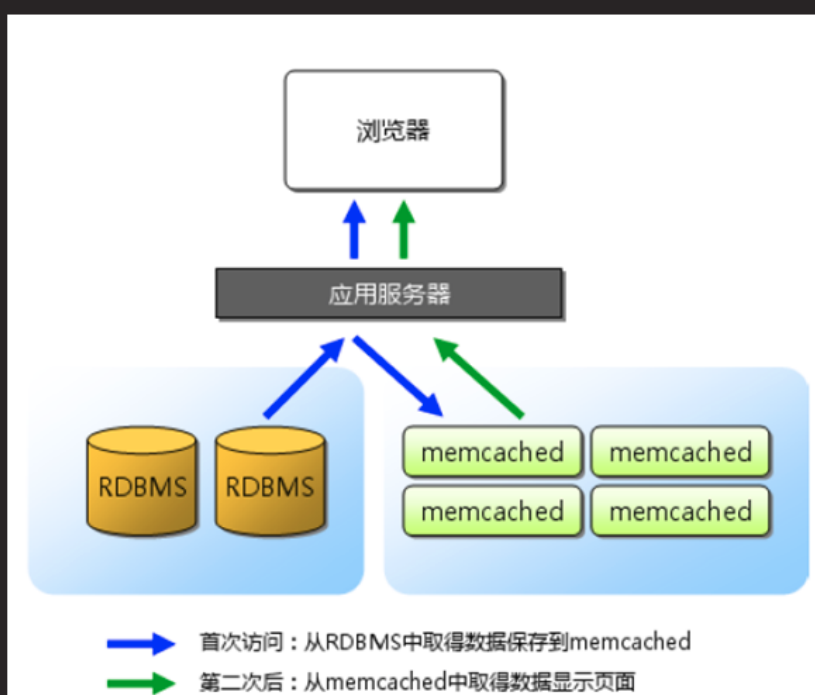
知识讲解

- 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/>



memcached简介（续1）

知识讲解



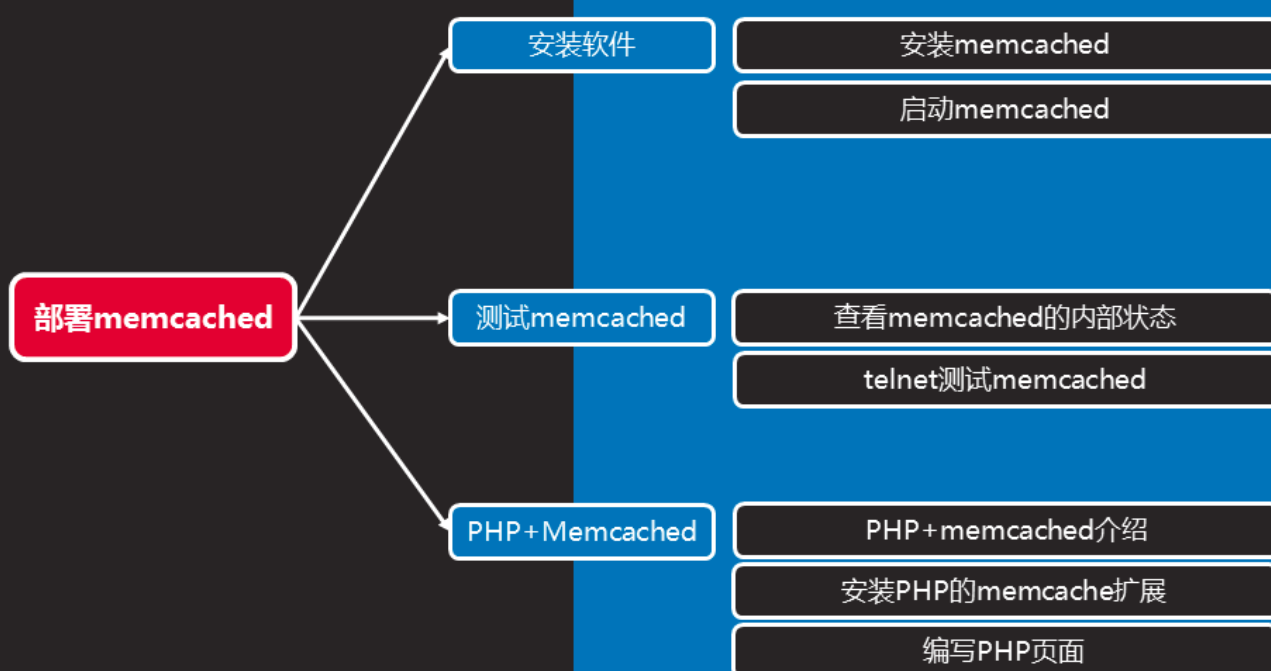
内存管理机制

知识讲解

- 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
 - memcached使用这种内存管理机制，可以提高读写速度



部署memcached



安装软件



安装memcached

```
[root@localhost ~]# yum install memcached  
[root@localhost ~]# rpm -qa memcached  
memcached-1.4.15-10.el7_3.1.x86_64
```



启动memcached

知识讲解

```
[root@localhost memcached]# cat /etc/sysconfig/memcached
PORT="11211"
USER="memcached"
MAXCONN="1024"
CACHESIZE="64"
[root@localhost memcached]# /usr/lib/systemd/system/memcached.service
[Service]
Type=simple
EnvironmentFile=/etc/sysconfig/memcached
ExecStart=/usr/bin/memcached -u $USER -p $PORT -m $CACHESIZE -c
$MAXCONN $OPTIONS
[Install]
WantedBy=multi-user.target
[root@localhost memcached]# systemctl start memcached
[root@localhost memcached]# systemctl status memcached
```



启动memcached (续1)

- 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

知识讲解



测试memcached

查看memcached的内部状态

```
[root@localhost ~]# telnet localhost 11211
```

```
Trying ::1...
```

```
Connected to localhost.
```

```
Escape character is '^['.
```

```
stats
```

```
STAT pid 481
```

```
STAT uptime 16574
```

```
STAT time 1213687612
```

```
STAT version 1.2.5
```

```
STAT pointer_size 32
```

telnet测试memcached

- 参考memcached常用指令

```
[root@localhost bin]# telnet 127.0.0.1 11211
Trying 127.0.0.1...
.....
set name 0 180 3           //定义变量
plj                        //输入变量的值
STORED
get name                  //获取变量的值
VALUE name 0 3           //输出结果
plj
END
quit                      // 退出登录
```

知识讲解



telnet测试memcached (续1)

- memcached常用指令

```
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
```

知识讲解



案例1：构建memcached服务

课堂练习

- 任务要求：
 - 安装memcached软件，并启动服务
 - 使用telnet测试memcached
 - 对memcached进行增、删、改、查等操作



PHP+Memcached

PHP+memcached介绍

知识讲解

- 部署LNMP
 - 使用php页面，测试对memcached的读写操作
 - 具体操作参考第二章节Nginx+FastCGI
- PHP无法直接操作memcached
 - 需要安装memcache扩展



安装PHP的memcache扩展

```
[root@localhost ~]# yum install php-pecl-memcache
[root@localhost ~]# rpm -ql php-pecl-memcache
/etc/php.d/memcache.ini
/usr/lib64/php/modules/memcache.so
```

... ..

```
[root@localhost ~]# systemctl restart php-fpm
```

知识讲解



编写PHP页面

知识讲解

```
[root@localhost html]# cat /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;
?>
[root@localhost html]# elinks --dump http://localhost/mem.php
```



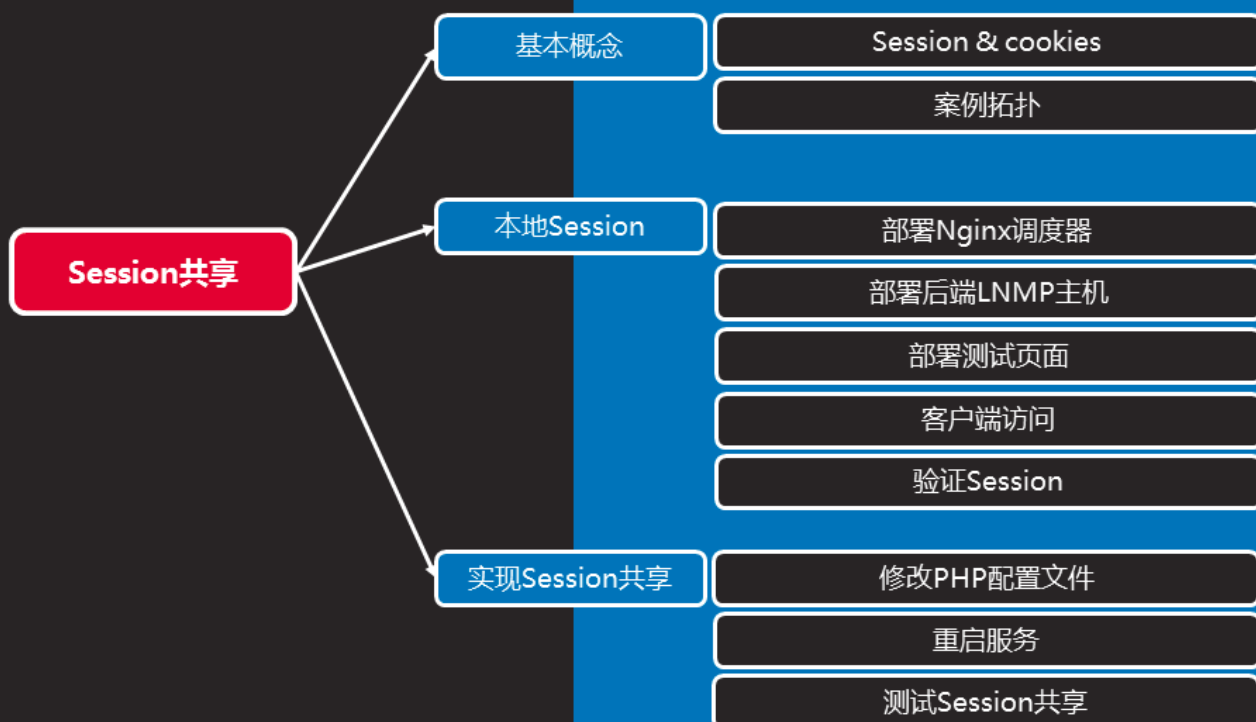
案例2：LNMP+memcached

课堂练习

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



Session共享



基本概念

Session & cookies

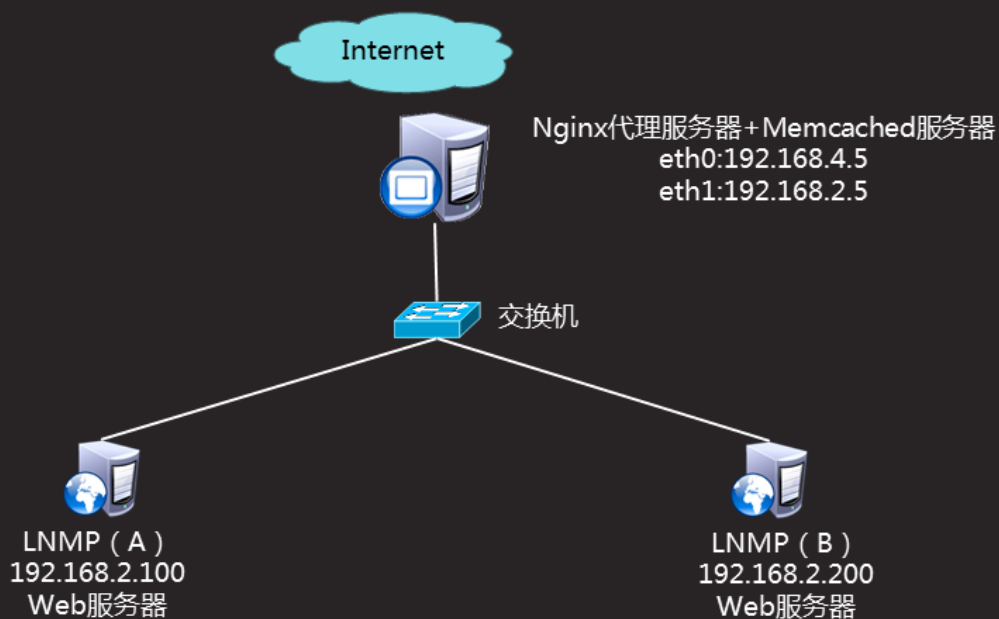
知识讲解

- 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、账号名，过期时间，
路径和域



案例拓扑

知识讲解



本地Session

部署Nginx调度器

- 安装Nginx软件

```
[root@localhost ~]# yum -y install pcre pcre-devel openssl-devel  
[root@localhost ~]# tar -zxvf nginx-1.12.2.tar.gz  
[root@localhost ~]# cd nginx-1.12.2  
[root@localhost nginx-1.12.2]# ./configure [root@localhost nginx-  
1.12.2]# make && make install
```

- 安装memcached软件

```
[root@localhost ~]# yum -y install memcached
```


部署Nginx调度器（续1）

- 启动服务

```
[root@svr1 ~]# vim /usr/local/nginx/conf/nginx.conf
```

```
upstream webs {  
    server 192.168.2.100:80;           //LNMP服务器的IP、端口  
    server 192.168.2.200:80;  
}  
server {  
    ... ..  
    location / {  
        proxy_pass http://webs;  
    }  
}
```

- 修改配置文件

```
[root@svr1 ~]# /usr/local/nginx/sbin/nginx
```

知识讲解



部署后端LNMP主机

- Nginx

```
[root@localhost ~]# yum -y install pcre pcre-devel openssl-devel  
[root@localhost ~]# tar -zxvf nginx-1.12.2.tar.gz  
[root@localhost ~]# cd nginx-1.12.2  
[root@localhost nginx-1.12.2]# ./configure  
[root@localhost nginx-1.12.2]# make && make install
```

- MariaDB

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

- PHP

```
[root@localhost ~]# yum -y install php php-mysql php-pecl-memcache  
[root@localhost ~]# rpm -ivh php-fpm
```

知识讲解



部署后端LNMP主机（续1）

知识讲解

- 启动服务

```
[root@localhost ~]# nginx  
[root@localhost ~]# systemctl restart php-fpm  
[root@localhost ~]# systemctl restart mariadb
```

- 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LNMP主机都操作

```
[root@www ~]# cd lnmp-soft/php_scripts/session/  
[root@www ~]# tar -xf php-memcached-demo.tar.gz  
[root@www ~]# cp * /usr/local/nginx/html
```



部署测试页面

知识讲解

- 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LNMP主机都操作

```
[root@www ~]# cd lnmp-soft/php_scripts/session/  
[root@www ~]# tar -xf php-memcached-demo.tar.gz  
[root@www ~]# cd php-memcached-demo/  
[root@www ~]# cp * /usr/local/nginx/html
```



客户端访问

知识讲解

- 浏览器直接访问后端的LNMP主机2.100
`[root@www ~]# firefox http://192.168.2.100`
- 浏览器直接访问后端的LNMP主机2.200
`[root@www ~]# firefox http://192.168.2.200`
- 浏览器访问调度器，刷新页面后需要登陆两次
`[root@www ~]# firefox http://192.168.4.5`



验证Session

知识讲解

- 登陆后端两台LNMP主机分别查看
`[root@www ~]# ls /var/lib/php/session`

```
[root@localhost session]# ls
sess_256j87ffmf41p0hcald3apkri2  sess_jk67qc5fa89c8hkvn168fttin7
sess_2cgb98ck4mnkib1bd8omlb09s3  sess_qqek1tmel07br8f63d6v9ch401
sess_4fd9n5lb3tli51rk9qb1fcnu96  sess_s7n6cgfisbcbg1ika8edjkn5e4
sess_ahilcq9bquot0vqsjtd84k7244  sess_up8tkb4ne638d5oip2j4e1ln36
sess_ep4j8a020plekh4adcgtvn1tk0
```



案例3：PHP的本地Session信息

课堂练习

- 部署Nginx为前端调度服务器
- 调度算法设置为轮询
- 部署后端LNMP服务器
- 部署测试页面，查看本机Session信息



实现Session共享

修改PHP配置文件

知识讲解

```
[root@localhost session]# 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://192.168.2.5:11211"
```



重启服务

```
[root@localhost ~]# systemctl restart php-fpm
```

知识讲解



测试Session共享

知识讲解

- 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**切换服务器后**，账户信息还在
- 两台后端服务器使用的是同一个账户

```
[root@localhost ~]# firefox http://192.168.4.5
```



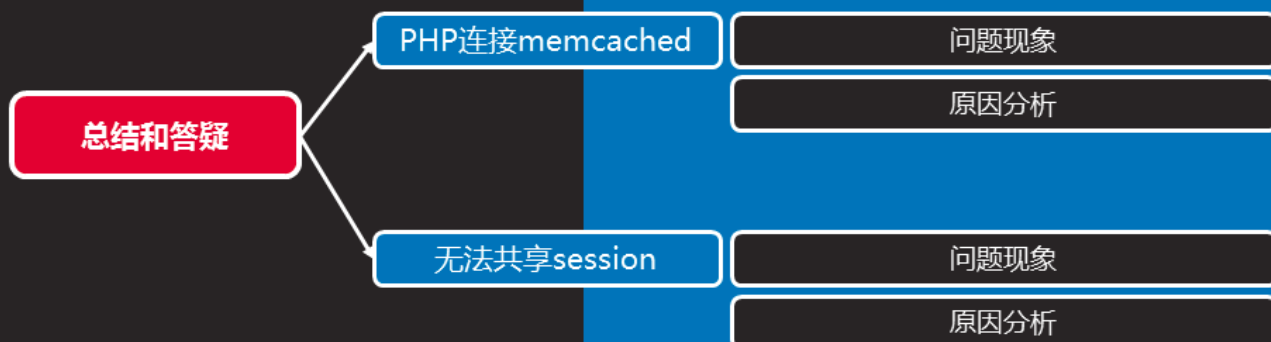
案例4：PHP实现session共享

课堂练习

- 参照案例步骤完成相关实验：
 - 配置PHP支持memcached
 - 客户端访问两台不同的后端Web服务器时，Session信息一致



总结和答疑



PHP连接memcached

问题现象

- 故障错误信息

`[root@svr5 nginx-1.8.0]# firefox http://192.168.2.100`

编写PHP代码，连接测试memcached数据库服务器，返回空白页
或者直接出现下载窗口

知识讲解



原因分析

- 分析故障
 - 是否正确配置nginx
 - 是否正确配置PHP
 - 是否正确配置memcached
- 分析故障原因
 - nginx没有配置fastcgi_pass，将返回下载窗口
 - php-fpm没有启动，或nginx没有设置正确的FPM端口
 - PHP没安装memcache扩展，或没有在配置文件中加载该扩展
 - Memcached服务没有启动，或者没有监听正确的IP和端口

知识讲解



无法共享session

问题现象

- 故障错误信息

[root@svr5 nginx-1.8.0]# firefox http://192.168.4.5

访问调度器后，刷新页面session（账户信息）依然有变化

知识讲解



原因分析

知识讲解

- 分析故障
 - PHP是否添加了memcached扩展库
 - PHP-fpm是否修改了www.conf配置
- 分析故障原因
 - 未安装php-pecl-memcache的话，无法连接memcached
 - 没有修改配置文件的话，默认php会在本地存储session

