**LVS+keepalived+nginx+apache搭建高可用、高性能php集群**

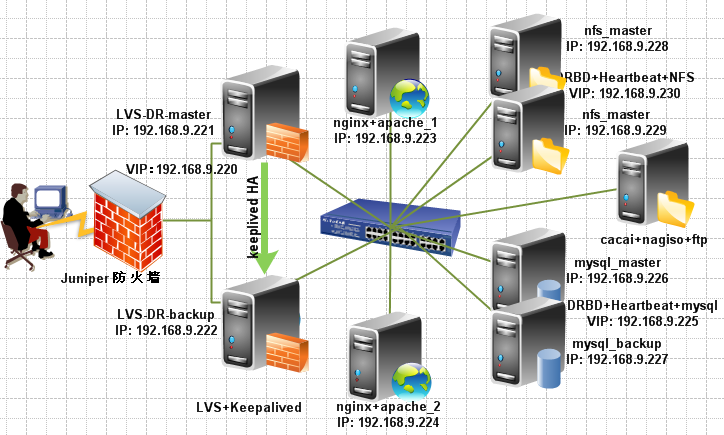
2011-07-01 12:04:30

标签：[php](http://blog.51cto.com/tag-php.html) [mysql](http://blog.51cto.com/tag-mysql.html) [apache](http://blog.51cto.com/tag-apache.html) [集群](http://blog.51cto.com/tag-%E9%9B%86%E7%BE%A4.html) [nginx](http://blog.51cto.com/tag-nginx.html)

原创作品，允许转载，转载时请务必以超链接形式标明文章 [原始出处](http://kerry.blog.51cto.com/172631/600590) 、作者信息和本声明。否则将追究法律责任。<http://kerry.blog.51cto.com/172631/600590>

      基于PHP的运行环境大家用的比较多的是以apache做中间件的LAMP（linux+apache+mysql+php）环境和以nginx做中间件的LNMP(linux+nginx+mysql+php)环境，apache是世界排名第一的web服务器，apache可以整合大多数应用,比如jsp,php,cgi,python等等,但是apache过于臃肿以及对静态文件响应过于缓慢是让人很不满意的；而nginx作为后起之秀，大有超越前辈的势头，从最近很多门户网站选用nginx就可见一斑，nginx速度快，占用资源少,高性能转发，很高的稳定性,支持热部署，杀手级的proxy和rewrite，但nginx对于php处理依托于fastcgi,fastcgi不够稳定,经常出现502错误,生成相对复杂的页面没有优势,反而会使php-cgi进程变为僵尸进程,多用户多站点的安全性不高，对于其它开发语言的整合，apache表现的更好；综上述对比，为了搭建高性能的php运行环境，就要考虑对apache、nginx进行整合，各取所长，nginx在处理静态内容上的效率较apache强很大，那么就用nginx在前端处理静态文件，把基于php的动态文件转发到后端的apache来处理，接下来我们就开始布署LNAMP（linux+nginx+apache+mysql+php）环境

一、网络拓扑:



lvs-master: 192.168.9.221  
LVS-backup: 192.168.9.222  
LVS-VIP: 192.168.9.220

web\_server\_1: 192.168.9.223  
web\_server\_2: 192.168.9.224  
nfs\_server\_1: 192.168.9.228  
nfs\_server\_1: 192.168.9.229  
nfs\_vip: 192.168.9.230  
mysql\_master: 192.168.9.226  
mysql\_backup: 192.168.9.227  
mysql\_vip: 192.168.9.225

目的：搭建高可用、高性能的php集群

二、初始化系统  
#init system  
系统环境: CentOS 5.5 (最小化安装)

./init\_system.sh    #此脚本参见<http://kerry.blog.51cto.com/172631/555535>

#更新组件  
sudo -s  
LANG=C  
yum -y install gcc gcc-c++ bison patch unzip mlocate flex wget automake autoconf gd cpp gettext readline-devel libjpeg \  
libjpeg-devel libpng libpng-devel freetype freetype-devel libxml2 libxml2-devel zlib zlib-devel glibc glibc-devel glib2 \  
glib2-devel bzip2 bzip2-devel ncurses ncurses-devel curl curl-devel e2fsprogs e2fsprogs-devel libidn libidn-devel openldap \  
openldap-devel openldap-clients openldap-servers nss\_ldap expat-devel libtool libtool-ltdl-devel bison

三、LVS+keeplived  
#关于LVS+keeplived的配置请参考我的另一篇博文《CentOS5.5环境下布署LVS+keepalived 》<http://kerry.blog.51cto.com/172631/401253>

四、web\_server安装配置（nginx、apache整合）  
#download software  
cd /opt  
wget <http://nginx.org/download/nginx-1.0.4.tar.gz>  
wget <http://www.apache.org/dist/httpd/httpd-2.2.19.tar.gz>  
wget <http://www.cmake.org/files/v2.8/cmake-2.8.4.tar.gz>  
wget <http://mysql.he.net/Downloads/MySQL-5.5/mysql-5.5.13.tar.gz>  
wget <http://pecl.php.net/get/PDO_MYSQL-1.0.2.tgz>  
#rpaf模块,该模块用于apache做后端时获取访客真实的IP  
wget <http://stderr.net/apache/rpaf/download/mod_rpaf-0.6.tar.gz>  
#字符转换库libiconv  
wget <http://ftp.gnu.org/pub/gnu/libiconv/libiconv-1.13.1.tar.gz>  
wget <http://sourceforge.net/projects/mcrypt/files/Libmcrypt/2.5.8/libmcrypt-2.5.8.tar.bz2/download>  
wget <http://sourceforge.net/projects/mcrypt/files/MCrypt/2.6.8/mcrypt-2.6.8.tar.gz/download>  
wget <http://sourceforge.net/projects/mhash/files/mhash/0.9.9.9/mhash-0.9.9.9.tar.bz2/download>  
wget <http://cn.php.net/distributions/php-5.2.17.tar.bz2>  
#php5-mail-header.patch为php被丁，有助于防止邮件发送被滥用  
wget <http://www.lancs.ac.uk/~steveb/patches/php-mail-header-patch/php5-mail-header.patch>  
#PHP的memcache扩展  
wget <http://pecl.php.net/get/memcache-2.2.6.tgz>  
#eaccelerator加速  
wget <http://bart.eaccelerator.net/source/0.9.6.1/eaccelerator-0.9.6.1.tar.bz2>  
wget <ftp://ftp.imagemagick.org/pub/ImageMagick/ImageMagick.tar.bz2>  
#PHP的imagick扩展  
wget <http://pecl.php.net/get/imagick-3.0.1.tgz>  
#Suhosin是一个PHP程序的保护系统  
wget <http://download.suhosin.org/suhosin-0.9.32.1.tar.gz>  
wget <http://downloads2.ioncube.com/loader_downloads/ioncube_loaders_lin_x86.tar.bz2>  
wget <http://downloads.zend.com/optimizer/3.3.9/ZendOptimizer-3.3.9-linux-glibc23-i386.tar.gz>    #32位  
wget <http://downloads.zend.com/optimizer/3.3.9/ZendOptimizer-3.3.9-linux-glibc23-x86_64.tar.gz>  #64位  
wget <http://monkey.org/~provos/libevent-2.0.11-stable.tar.gz>  
wget <http://sourceforge.net/projects/pcre/files/pcre/8.12/pcre-8.12.tar.bz2/download>  
wget <http://memcached.googlecode.com/files/memcached-1.4.5.tar.gz>  
#PHP的PDO扩展  
wget <http://pecl.php.net/get/PDO_MYSQL-1.0.2.tgz>  
#Tcmalloc 优化nginx、mysql  
wget <http://google-perftools.googlecode.com/files/google-perftools-1.7.tar.gz>

#---------------------------- 使用cmake编译安装mysql ----------------------------------  
#使用Tcmalloc 优化nginx、mysql  
cd /opt  
tar -zxvf google-perftools-1.7.tar.gz  
cd google-perftools-1.7/  
./configure  
make;make install

echo "/usr/local/lib" > /etc/ld.so.conf.d/usr\_local\_lib.conf  
/sbin/ldconfig  
cd ../

tar -zxvf cmake-2.8.4.tar.gz  
cd cmake-2.8.4  
./bootstrap  
gmake  
gmake install  
cd ../

/usr/sbin/groupadd mysql  
/usr/sbin/useradd -g mysql mysql -s /sbin/nologin  
mkdir -p /data/mysql/data  
mkdir -p /data/mysql/binlog  
mkdir -p /data/mysql/relaylog  
mkdir -p /data/mysql/mysql  
chown -R mysql:mysql /data/mysql  
cd /opt  
tar zxvf mysql-5.5.13.tar.gz  
cd mysql-5.5.13/  
rm -rf CMakeCache.txt  
cmake -DCMAKE\_INSTALL\_PREFIX=/usr/local/mysql \  
-DMYSQL\_UNIX\_ADDR=/data/mysql/mysql.sock \  
-DDEFAULT\_CHARSET=utf8 \  
-DDEFAULT\_COLLATION=utf8\_general\_ci \  
-DEXTRA\_CHARSETS=all \  
-DWITH\_MYISAM\_STORAGE\_ENGINE=1 \  
-DWITH\_INNOBASE\_STORAGE\_ENGINE=1 \  
-DWITH\_READLINE=1 \  
-DENABLED\_LOCAL\_INFILE=1 \  
-DMYSQL\_DATADIR=/data/mysql/data \  
-DMYSQL\_TCP\_PORT=3306   
make;make install  
chmod +w /usr/local/mysql  
chown -R mysql:mysql /usr/local/mysql  
ln -s /usr/local/mysql/lib/libmysqlclient.so.18 /usr/lib/libmysqlclient.so.18

#编辑mysql配置文件  
vi /etc/my.cnf

1. [client]
2. port    = 3306
3. socket  = /data/mysql/mysql.sock
5. [mysqld]
6. character\_set\_server = utf8
7. collation-server = utf8\_general\_ci
8. replicate-ignore-db = mysql
9. replicate-ignore-db = test
10. replicate-ignore-db = information\_schema
11. user    = mysql
12. port    = 3306
13. socket  = /data/mysql/mysql.sock
14. basedir = /usr/local/mysql
15. datadir = /data/mysql/data
16. log-error = /data/mysql/mysql\_error.log
17. pid-file = /data/mysql/mysql.pid
18. open\_files\_limit    = 10240
19. back\_log = 600
20. max\_connections = 5000
21. max\_connect\_errors = 6000
22. table\_cache = 512
23. external-locking = FALSE
24. max\_allowed\_packet = 32M
25. sort\_buffer\_size = 6M
26. join\_buffer\_size = 8M
27. thread\_cache\_size = 300
28. thread\_concurrency = 8
29. query\_cache\_size = 512M
30. query\_cache\_limit = 2M
31. query\_cache\_min\_res\_unit = 2k
32. default-storage-engine = MyISAM
33. thread\_stack = 256K
34. transaction\_isolation = READ-COMMITTED
35. tmp\_table\_size = 256M
36. max\_heap\_table\_size = 256M
37. long\_query\_time = 3
38. log-slave-updates
39. log-bin = /data/mysql/binlog/binlog
40. binlog\_cache\_size = 4M
41. binlog\_format = MIXED
42. max\_binlog\_cache\_size = 8M
43. max\_binlog\_size = 100M
44. relay-log-index = /data/mysql/relaylog/relaylog
45. relay-log-info-file = /data/mysql/relaylog/relaylog
46. relay-log = /data/mysql/relaylog/relaylog
47. expire\_logs\_days = 30
48. key\_buffer\_size = 384M
49. read\_buffer\_size = 4M
50. read\_rnd\_buffer\_size = 16M
51. bulk\_insert\_buffer\_size = 64M
52. myisam\_sort\_buffer\_size = 128M
53. myisam\_max\_sort\_file\_size = 10G
54. myisam\_repair\_threads = 1
55. myisam\_recover
57. interactive\_timeout = 120
58. wait\_timeout = 120
60. skip\_external\_locking
61. skip-name-resolve
62. #master-connect-retry = 10
63. slave-skip-errors = 1032,1062,126,1114,1146,1048,1396
65. #master-host     =   192.168.1.2
66. #master-user     =   username
67. #master-password =   password
68. #master-port     =  3306
70. server-id = 1
72. skip-innodb
74. #log-slow-queries = /data/mysql/slow.log
75. #long\_query\_time = 10
77. [mysqldump]
78. quick
79. max\_allowed\_packet = 32M

#初始化mysql  
/usr/local/mysql/scripts/mysql\_install\_db --basedir=/usr/local/mysql --datadir=/data/mysql/data --user=mysql

#利用TCMalloc提高mysql在高并发下的性能  
vi /usr/local/mysql/bin/mysqld\_safe

#在# executing mysqld\_safe的下一行，加上：  
export LD\_PRELOAD=/usr/local/lib/libtcmalloc.so

#使用lsof命令查看tcmalloc是否起效  
/usr/sbin/lsof -n | grep tcmalloc

#设置mysql启动文件  
cp support-files/mysql.server /etc/rc.d/init.d/mysqld  
vi /etc/rc.d/init.d/mysqld  
basedir=/usr/local/mysql  
datadir=/data/mysql/data

chmod 700 /etc/rc.d/init.d/mysqld  
/etc/rc.d/init.d/mysqld start  
/sbin/chkconfig --add mysqld  
/sbin/chkconfig --level 2345 mysqld on  
ln -s /usr/local/mysql/bin/mysql /sbin/mysql   
ln -s /usr/local/mysql/bin/mysqladmin /sbin/mysqladmin

#设置root密码（753951）  
/sbin/mysqladmin -u root password 753951

#配置库文件搜索路径  
echo "/usr/local/mysql/lib/mysql" >> /etc/ld.so.conf  
/sbin/ldconfig

#添加/usr/local/mysql/bin到环境变量PATH中  
export PATH=$PATH:/usr/local/mysql/bin

#添加mysql管理帐户  
#mysql -h localhost -u root -p753951  
#msqyl> use mysql;  
#msqyl> grant all on \*.\* to ['kerry'@'192.168.9.100'](mailto:'kerry'@'192.168.9.100') identified by '852741';  
#msqyl> flush privileges;   
#msqyl> exit;

#----------------------------------- 编译安装apache -------------------------------------  
/usr/sbin/groupadd www  
/usr/sbin/useradd -g www www -s /sbin/nologin  
mkdir -p /data/www/kerry  
mkdir -p /data/logs/kerry  
mkdir -p /data/www/king  
mkdir -p /data/logs/king  
chown -R www:www /data/www  
chown -R www:www /data/logs  
cd /opt  
tar -zxvf httpd-2.2.19.tar.gz  
cd httpd-2.2.19  
cd srclib/apr    
./configure --prefix=/usr/local/apr --enable-threads --enable-other-child --enable-static   
make;make install   
cd ../apr-util   
./configure --prefix=/usr/local/apr-util --with-apr=/usr/local/apr  
make;make install  
cd ../../  
./configure --prefix=/usr/local/apache \  
--enable-deflate \  
--enable-headers \  
--enable-mime-magic \  
--enable-proxy \  
--enable-ssl \  
--enable-so \  
--enable-rewrite \  
--enable-suexec \  
--with-suexec-bin=/usr/sbin/suexec \  
--with-suexec-caller=www \  
--with-apr=/usr/local/apr \  
--with-apr-util=/usr/local/apr-util \  
--with-mpm=prefork \  
--with-ssl=/usr  
make;make install

cp /usr/local/apache/bin/apachectl /etc/init.d/httpd  
vi /etc/init.d/httpd  
#在首行#!/bin/sh下添加  
# Startup script for the Apache Web Server  
#  
# chkconfig: - 85 15  
# description: Apache is a World Wide Web server. It is used to serve \  
# HTML files and CGI.  
# processname: httpd  
# pidfile: /usr/local/apache/logs/httpd.pid  
# config: /usr/local/apache/conf/httpd.conf

#修改apache配置文件  
cd /usr/local/apache/conf/  
mv httpd.conf httpd.conf.bak

vi httpd.conf

1. PidFile logs/httpd.pid
2. LockFile logs/accept.lock
3. ServerRoot "/usr/local/apache"
4. Listen 0.0.0.0:81
5. User www
6. Group www
7. ServerAdmin king\_819@163.com
8. ServerName www.kerry.com
10. Timeout 300
11. KeepAlive Off
12. MaxKeepAliveRequests 100
13. KeepAliveTimeout 5
14. UseCanonicalName Off
15. AccessFileName .htaccess
16. TraceEnable Off
17. ServerTokens ProductOnly
18. FileETag None
19. ServerSignature Off
20. HostnameLookups Off
22. # LoadModule perl\_module modules/mod\_perl.so
24. DocumentRoot "/usr/local/apache/htdocs"
26. **<Directory** "/"**>**
27. Options ExecCGI FollowSymLinks Includes IncludesNOEXEC -Indexes -MultiViews SymLinksIfOwnerMatch
28. Order allow,deny
29. Allow from all
30. AllowOverride All
31. **</Directory>**
33. **<Directory** "/usr/local/apache/htdocs"**>**
34. Options Includes -Indexes FollowSymLinks
35. AllowOverride None
36. Order allow,deny
37. Allow from all
38. **</Directory>**
40. DefaultType text/plain
41. RewriteEngine on
42. AddType text/html .shtml
43. AddHandler cgi-script .cgi .pl .plx .ppl .perl
44. AddHandler server-parsed .shtml
45. **<IfModule** mime\_module**>**
47. TypesConfig conf/mime.types
48. AddType application/perl .pl .plx .ppl .perl
49. AddType application/x-img .img
50. AddType application/x-httpd-php .php .php3 .php4 .php5 .php6
51. AddType application/x-httpd-php-source .phps
52. AddType application/cgi .cgi
53. AddType text/x-sql .sql
54. AddType text/x-log .log
55. AddType text/x-config .cnf conf
56. AddType text/x-registry .reg
57. AddType application/x-compress .Z
58. AddType application/x-gzip .gz .tgz
59. AddType text/html .shtml
60. AddType application/x-tar .tgz
61. AddType application/rar .rar
62. AddType application/x-compressed .rar
63. AddType application/x-rar .rar
64. AddType application/x-rar-compressed .rar
65. AddType text/vnd.wap.wml .wml
66. AddType image/vnd.wap.wbmp .wbmp
67. AddType text/vnd.wap.wmlscript .wmls
68. AddType application/vnd.wap.wmlc .wmlc
69. AddType application/vnd.wap.wmlscriptc .wmlsc
70. **</IfModule>**
72. **<IfModule** dir\_module**>**
73. DirectoryIndex index.html index.htm index.shtml index.php
74. **</IfModule>**
76. **<Files** ~ "^error\_log$"**>**
77. Order allow,deny
78. Deny from all
80. Satisfy All
81. **</Files>**
83. **<FilesMatch** "^\.ht"**>**
84. Order allow,deny
85. Deny from all
86. Satisfy All
87. **</FilesMatch>**
89. ErrorLog "logs/error\_log"
90. LogLevel warn
92. **<IfModule** log\_config\_module**>**
93. LogFormat "%h %l %u %t \"%r\" %**>**s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
94. LogFormat "%h %l %u %t \"%r\" %**>**s %b" common
96. **<IfModule** logio\_module**>**
98. LogFormat "%h %l %u %t \"%r\" %**>**s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O" combinedio
99. **</IfModule>**
100. CustomLog "logs/access\_log" common
101. **</IfModule>**
103. **<IfModule** alias\_module**>**
104. ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"
105. **</IfModule>**
107. **<Directory** "/usr/local/apache/cgi-bin"**>**
108. AllowOverride None
109. Options None
110. Order allow,deny
111. Allow from all
112. **</Directory>**
114. **<IfModule** mpm\_prefork\_module**>**
115. StartServers          3
116. MinSpareServers       3
117. MaxSpareServers       5
118. MaxClients          150
119. MaxRequestsPerChild   1024
120. **</IfModule>**
122. **<IfModule** mod\_headers.c**>**
123. **<FilesMatch** "\.(html|htm|shtml)$"**>**
125. Header set Cache-Control "max-age=3600, must-revalidate"
126. **</FilesMatch>**
127. **</IfModule>**
129. ReadmeName README.html
130. HeaderName HEADER.html
132. IndexIgnore .??\* \*~ \*# HEADER\* README\* RCS CVS \*,v \*,t
134. Include conf/extra/httpd-languages.conf
136. **<Location** /server-status**>**
137. SetHandler server-status
138. Order deny,allow
139. Deny from all
140. Allow from 127.0.0.1
141. **</Location>**
142. ExtendedStatus On
144. **<Location** /server-info**>**
145. SetHandler server-info
146. Order deny,allow
147. Deny from all
148. Allow from 127.0.0.1
150. **</Location>**
152. **<IfModule** ssl\_module**>**
153. Listen 0.0.0.0:443
154. AddType application/x-x509-ca-cert .crt
155. AddType application/x-pkcs7-crl .crl
156. SSLCipherSuite ALL:!ADH:+HIGH:+MEDIUM:-LOW:-SSLv2:-EXP
157. SSLPassPhraseDialog  builtin
158. SSLSessionCache         dbm:/usr/local/apache/logs/ssl\_scache
159. SSLSessionCacheTimeout  300
160. SSLMutex  file:/usr/local/apache/logs/ssl\_mutex
161. SSLRandomSeed startup builtin
162. SSLRandomSeed connect builtin
163. **</IfModule>**
165. #Vhosts
166. NameVirtualHost 127.0.0.1:81
168. **<VirtualHost** 127.0.0.1:81**>**
169. ServerAdmin king\_819@163.com
170. DocumentRoot /data/www/kerry
171. ServerName www.kerry.com
172. ErrorLog logs/kerry-error\_log
173. CustomLog "|/usr/local/apache/bin/rotatelogs /data/logs/kerry/%y\_%m\_%d.access\_log 86400" common
174. **</VirtualHost>**
175. **<VirtualHost** 127.0.0.1:81**>**
176. ServerAdmin king\_819@163.com
177. DocumentRoot /data/www/king
178. ServerName www.king.com
179. ErrorLog logs/king-error\_log
180. CustomLog "|/usr/local/apache/bin/rotatelogs /data/logs/king/%y\_%m\_%d.access\_log 86400" common
181. **</VirtualHost>**

chmod 700 /etc/init.d/httpd  
/etc/init.d/httpd start  
/sbin/chkconfig --add httpd  
/sbin/chkconfig --level 2345 httpd on

#-------------------------------------  编译安装PHP -------------------------------------  
#编译安装相关支持库  
cd /opt  
tar -zxvf libiconv-1.13.1.tar.gz  
cd libiconv-1.13.1/  
./configure  
make;make install

cd /opt  
tar -jxvf libmcrypt-2.5.8.tar.bz2  
cd libmcrypt-2.5.8/  
./configure  
make;make install  
/sbin/ldconfig

cd libltdl/  
./configure --enable-ltdl-install  
make;make install

cd /opt  
tar -jxvf mhash-0.9.9.9.tar.bz2  
cd mhash-0.9.9.9/  
./configure  
make;make install  
ln -s /usr/local/lib/libmcrypt.la /usr/lib/libmcrypt.la  
ln -s /usr/local/lib/libmcrypt.so /usr/lib/libmcrypt.so  
ln -s /usr/local/lib/libmcrypt.so.4 /usr/lib/libmcrypt.so.4  
ln -s /usr/local/lib/libmcrypt.so.4.4.8 /usr/lib/libmcrypt.so.4.4.8  
ln -s /usr/local/lib/libmhash.a /usr/lib/libmhash.a  
ln -s /usr/local/lib/libmhash.la /usr/lib/libmhash.la  
ln -s /usr/local/lib/libmhash.so /usr/lib/libmhash.so  
ln -s /usr/local/lib/libmhash.so.2 /usr/lib/libmhash.so.2  
ln -s /usr/local/lib/libmhash.so.2.0.1 /usr/lib/libmhash.so.2.0.1

cd /opt  
tar -zxvf mcrypt-2.6.8.tar.gz  
cd mcrypt-2.6.8/  
/sbin/ldconfig  
./configure  
make;make install

#编译php,这里我们为php打入补丁.有助于防止邮件发送被滥用(多用户)以及在邮件中提供有价值的信息.补丁介绍信息请点击:http://www.lancs.ac.uk/~steveb/patches/php-mail-header-patch/

cd /opt  
tar -jxvf php-5.2.17.tar.gz  
patch -d php-5.2.17 -p1 < php5-mail-header.patch  
cd php-5.2.17  
./configure --prefix=/usr/local/php \  
--with-config-file-path=/usr/local/php/etc \  
--with-apxs2=/usr/local/apache/bin/apxs \  
--with-mysql=/usr/local/mysql \  
--with-mysqli=/usr/local/mysql/bin/mysql\_config \  
--with-iconv-dir=/usr/local \  
--with-freetype-dir \  
--with-jpeg-dir \  
--with-png-dir \  
--with-zlib \  
--with-libxml-dir=/usr \  
--enable-xml \  
--disable-rpath \  
--enable-discard-path \  
--enable-safe-mode \  
--enable-bcmath \  
--enable-shmop \  
--enable-sysvsem \  
--enable-inline-optimization \  
--with-curl \  
--with-curlwrappers \  
--enable-mbregex \  
--enable-force-cgi-redirect \  
--enable-mbstring \  
--with-mcrypt \  
--with-gd \  
--enable-gd-native-ttf \  
--with-openssl \  
--with-mhash \  
--enable-pcntl \  
--enable-sockets \  
--with-ldap \  
--with-ldap-sasl \  
--with-xmlrpc \  
-enable-zip \  
--enable-soap

make ZEND\_EXTRA\_LIBS='-liconv'  
make install  
cp php.ini-dist /usr/local/php/etc/php.ini

#安装php扩展模块  
cd /opt  
tar -zxvf memcache-2.2.6.tgz  
cd memcache-2.2.6/  
/usr/local/php/bin/phpize  
./configure --with-php-config=/usr/local/php/bin/php-config --with-zlib-dir --enable-memcache  
make;make install

cd /opt  
tar -jxvf eaccelerator-0.9.6.1.tar.bz2  
cd eaccelerator-0.9.6.1/  
/usr/local/php/bin/phpize  
./configure --enable-eaccelerator=shared --with-php-config=/usr/local/php/bin/php-config  
make;make install

cd /opt  
tar -zxvf PDO\_MYSQL-1.0.2.tgz  
cd PDO\_MYSQL-1.0.2/  
/usr/local/php/bin/phpize  
./configure --with-php-config=/usr/local/php/bin/php-config --with-pdo-mysql=/usr/local/mysql  
make;make install

cd /opt  
tar -jxvf ImageMagick.tar.bz2  
cd ImageMagick-6.6.9-8  
./configure  
make;make install

cd /opt  
tar -zxf imagick-3.0.1.tgz  
cd imagick-3.0.1/  
/usr/local/php/bin/phpize  
./configure --with-php-config=/usr/local/php/bin/php-config  
make;make install

cd /opt  
tar -zxf suhosin-0.9.32.1.tgz  
cd suhosin-0.9.32.1  
/usr/local/php/bin/phpize  
./configure --with-php-config=/usr/local/php/bin/php-config  
make;make install

cd /opt  
tar -jxf ioncube\_loaders\_lin\_x86.tar.bz2  
cd ioncube  
mkdir /usr/local/ioncube  
mv ioncube\_loader\_lin\_5.2.so /usr/local/ioncube/

cd /opt  
tar -zxf ZendOptimizer-3.3.9-linux-glibc23-i386.tar.gz  
mkdir -p /usr/local/Zend/lib/Optimizer-3.3.9/php-5.2.17  
cp ZendOptimizer-3.3.9-linux-glibc23-i386/data/5\_2\_x\_comp/ZendOptimizer.so /usr/local/Zend/lib/Optimizer-3.3.9/php-5.2.17/ZendOptimizer.so

#修改php.ini.  
sed -i 's#extension\_dir = "./"#extension\_dir = "/usr/local/php/lib/php/extensions/no-debug-non-zts-20060613/"\nextension = "memcache.so"\nextension = "pdo\_mysql.so"\nextension = "suhosin.so"\nextension = "imagick.so"\n#' /usr/local/php/etc/php.ini  
sed -i 's#output\_buffering = Off#output\_buffering = On#' /usr/local/php/etc/php.ini  
sed -i "s#; always\_populate\_raw\_post\_data = On#always\_populate\_raw\_post\_data = On#g" /usr/local/php/etc/php.ini  
sed -i "s#; cgi.fix\_pathinfo=0#cgi.fix\_pathinfo=0#g" /usr/local/php/etc/php.ini

#配置eAccelerator加速PHP  
mkdir -p /data/eaccelerator\_cache  
chmod -R 777 /data/eaccelerator\_cache  
cat >>/usr/local/php/etc/php.ini<<EOF  
[eaccelerator]  
zend\_extension = "/usr/local/ioncube/ioncube\_loader\_lin\_5.2.so"  
zend\_extension = "/usr/local/Zend/lib/Optimizer-3.3.9/php-5.2.17/ZendOptimizer.so"  
zend\_extension="/usr/local/php/lib/php/extensions/no-debug-non-zts-20060613/eaccelerator.so"  
eaccelerator.shm\_size="64"  
eaccelerator.cache\_dir="/data/eaccelerator\_cache"  
eaccelerator.enable="1"  
eaccelerator.optimizer="1"  
eaccelerator.check\_mtime="1"  
eaccelerator.debug="0"  
eaccelerator.filter=""  
eaccelerator.shm\_max="0"  
eaccelerator.shm\_ttl="3600"  
eaccelerator.shm\_prune\_period="3600"  
eaccelerator.shm\_only="0"  
eaccelerator.compress="1"  
eaccelerator.compress\_level="9"  
eaccelerator.keys = shm  
eaccelerator.sessions = shm  
eaccelerator.content = shm  
EOF

#安装Memcached(可选)  
cd /opt  
tar -xzf libevent-2.0.11-stable.tar.gz  
cd libevent-2.0.11-stable  
./configure  
make;make install  
ln -s /usr/local/lib/libevent-1.4.so.2 /usr/lib

cd /opt  
tar -xzf memcached-1.4.5.tar.gz  
cd memcached-1.4.5  
./configure --prefix=/usr/local/memcached --with-libevent=/usr  
make;make install

#基本使用方法:  
启动:  
/usr/local/memcached/bin/memcached -d -m 64 -p 11211 -u www -l localhost  
关闭:  
killall -9 memcached

#---------------------------------- 安装配置nginx ----------------------------------------  
#安装pcre  
/opt  
tar -jxvf pcre-8.12.tar.gz  
cd pcre-8.12/  
./configure  
make;make install

#安装nginx  
cd /opt  
tar -zxvf nginx-1.0.4.tar.gz  
cd nginx-1.0.4/  
./configure --user=www --group=www --prefix=/usr/local/nginx --with-http\_stub\_status\_module --with-http\_ssl\_module --with-http\_flv\_module --with-http\_gzip\_static\_module --with-google\_perftools\_module  
make;make install  
cd ../

#添加nginx启动脚本  
vi /etc/init.d/nginx

1. #! /bin/sh
2. ulimit -n 65535
3. # Description: Startup script for nginx
4. # chkconfig: 2345 55 25
6. PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
7. DESC="nginx daemon"
8. NAME=nginx
9. DAEMON=/usr/local/nginx/sbin/$NAME
10. CONFIGFILE=/usr/local/nginx/conf/nginx.conf
11. PIDFILE=/usr/local/nginx/logs/$NAME.pid
12. SCRIPTNAME=/etc/init.d/$NAME
14. set -e
15. [ -x "$DAEMON" ] || exit 0
17. do\_start() {
18. $DAEMON -c $CONFIGFILE || echo -n "nginx already running"
19. }
21. do\_stop() {
22. kill -QUIT `cat $PIDFILE` || echo -n "nginx not running"
23. }
25. do\_reload() {
26. kill -HUP `cat $PIDFILE` || echo -n "nginx can't reload"
27. }
29. case "$1" in
30. start)
31. echo -n "Starting $DESC: $NAME"
32. do\_start
33. echo "."
34. /etc/init.d/httpd start
35. ;;
36. stop)
37. echo -n "Stopping $DESC: $NAME"
38. do\_stop
39. echo "."
40. /etc/init.d/httpd stop
41. ;;
42. reload)
43. echo -n "Reloading $DESC configuration..."
44. do\_reload
45. echo "."
46. /etc/init.d/httpd restart
47. ;;
48. restart)
49. echo -n "Restarting $DESC: $NAME"
50. do\_stop
51. sleep 1
52. do\_start
53. echo "."
54. /etc/init.d/httpd restart
55. ;;
56. \*)
57. echo "Usage: $SCRIPTNAME {start|stop|reload|restart}" **>**&2
58. exit 3
59. ;;
60. esac
62. exit 0

#添加nginx配置文件  
mv /usr/local/nginx/conf/nginx.conf /usr/local/nginx/conf/nginx.conf.bak  
vi /usr/local/nginx/conf/nginx.conf

1. user  www www;
2. worker\_processes 8;
3. error\_log  /usr/local/nginx/logs/nginx\_error.log  crit;
4. pid        /usr/local/nginx/nginx.pid;
5. #使用Tcmalloc优化nginx性能
6. google\_perftools\_profiles /var/tmp/tcmalloc;
7. #Specifies the value for maximum file descriptors that can be opened by this process.
8. worker\_rlimit\_nofile 65535;
9. #工作模式及连接数上限
10. events
11. {
12. use epoll;
13. worker\_connections 65535;
14. }
15. #设定http服务器，利用它的反向代理功能提供负载均衡支持
16. http
17. {
18. #设定mime类型
19. include       mime.types;
20. default\_type  application/octet-stream;
21. #charset  gb2312;
22. #设定请求缓冲
23. server\_names\_hash\_bucket\_size 128;
24. client\_header\_buffer\_size 32k;
25. large\_client\_header\_buffers 4 32k;
26. client\_max\_body\_size 30m;
27. sendfile on;
28. tcp\_nopush     on;
29. keepalive\_timeout 60;
30. tcp\_nodelay on;
31. server\_tokens off;
32. client\_body\_buffer\_size 512k;
33. proxy\_connect\_timeout   5;
34. proxy\_send\_timeout      60;
35. proxy\_read\_timeout      5;
36. proxy\_buffer\_size       16k;
37. proxy\_buffers           4 64k;
38. proxy\_busy\_buffers\_size 128k;
39. proxy\_temp\_file\_write\_size 128k;
40. #  fastcgi\_connect\_timeout 300;
41. #  fastcgi\_send\_timeout 300;
42. #  fastcgi\_read\_timeout 300;
43. #  fastcgi\_buffer\_size 64k;
44. #  fastcgi\_buffers 4 64k;
45. #  fastcgi\_busy\_buffers\_size 128k;
46. #  fastcgi\_temp\_file\_write\_size 128k;
47. gzip on;
48. gzip\_min\_length  1k;
49. gzip\_buffers     4 16k;
50. gzip\_http\_version 1.1;
51. gzip\_comp\_level 2;
52. gzip\_types       text/plain application/x-javascript text/css application/xml;
53. gzip\_vary on;
55. #limit\_zone  crawler  $binary\_remote\_addr  10m;
57. ###禁止通过ip访问站点
58. server{
59. server\_name \_;
60. return 404;
61. }
62. server
63. {
64. listen       80;
65. server\_name  www.kerry.com;
66. index index.html index.htm index.php;#设定访问的默认首页地址
67. root  /data/www/kerry;#设定网站的资源存放路径
68. #limit\_conn   crawler  20;
69. if (-d $request\_filename)
70. {
71. rewrite ^/(.\*)([^/])$ http://$host/$1$2/ permanent;
72. }
73. #所有php的页面均交由apache处理
74. location ~ \.(php)?$ {
75. proxy\_set\_header  Host $host;
76. proxy\_set\_header  X-Real-IP  $remote\_addr;
77. proxy\_set\_header   X-Forwarded-For $proxy\_add\_x\_forwarded\_for;
78. proxy\_pass [http://127.0.0.1:81](http://127.0.0.1:81/);#转向apache处理
79. }
80. location ~ .\*\.(htm|html|gif|jpg|jpeg|png|bmp|swf|ioc|rar|zip|txt|flv|mid|doc|ppt|pdf|xls|mp3|wma)$ #设定访问静态文件直接读取不经过apache
81. {
82. expires      30d;
83. }
84. location ~ .\*\.(js|css)?$
85. {
86. expires      1h;
87. }
89. #定义访问日志的写入格式
90. log\_format  wwwlog  '$remote\_addr - $remote\_user [$time\_local] "$request" '
91. '$status $body\_bytes\_sent "$http\_referer" '
92. '"$http\_user\_agent" $http\_x\_forwarded\_for';
93. access\_log  /data/logs/kerry/kerry\_nginx.log wwwlog;#设定访问日志的存放路径
94. }
95. server
96. {
97. listen       80;
98. server\_name  www.king.com;
99. index index.html index.htm index.php;#设定访问的默认首页地址
100. root  /data/www/king;#设定网站的资源存放路径
101. #limit\_conn   crawler  20;
102. if (-d $request\_filename)
103. {
104. rewrite ^/(.\*)([^/])$ http://$host/$1$2/ permanent;
105. }
106. #所有php的页面均交由apache处理
107. location ~ \.(php)?$ {
108. proxy\_set\_header  Host $host;
109. proxy\_set\_header  X-Real-IP  $remote\_addr;
110. proxy\_set\_header   X-Forwarded-For $proxy\_add\_x\_forwarded\_for;
111. proxy\_pass [http://127.0.0.1:81](http://127.0.0.1:81/);#转向apache处理
112. }
113. location ~ .\*\.(htm|html|gif|jpg|jpeg|png|bmp|swf|ioc|rar|zip|txt|flv|mid|doc|ppt|pdf|xls|mp3|wma)$ #设定访问静态文件直接读取不经过apache
114. {
115. expires      30d;
116. }
117. location ~ .\*\.(js|css)?$
118. {
119. expires      1h;
120. }
122. #定义访问日志的写入格式
123. log\_format  bbslog  '$remote\_addr - $remote\_user [$time\_local] "$request" '
124. '$status $body\_bytes\_sent "$http\_referer" '
125. '"$http\_user\_agent" $http\_x\_forwarded\_for';
126. access\_log  /data/logs/king/king\_nginx.log bbslog;#设定访问日志的存放路径
127. }
128. server
129. {
130. listen  80;
131. server\_name  status.www.kerry.com;
132. location / {
133. stub\_status on;
134. access\_log   off;
135. }
136. }
137. }

#将nginx添加到启动服务中  
chmod 700 /etc/init.d/nginx  
/etc/init.d/nginx start  
/sbin/chkconfig --add nginx  
/sbin/chkconfig --level 2345 nginx on

#每天定时切割Nginx日志  
vi /usr/local/nginx/sbin/cut\_nginx\_log.sh

1. #!/bin/bash
2. # This script run at 00:00
3. # The Nginx logs path
4. logs\_path\_kerry="/data/logs/kerry/"
5. logs\_path\_kerry="/data/logs/king/"
6. mv ${logs\_path\_kerry}kerry\_nginx.log ${logs\_path\_kerry}$kerry\_nginx\_$(date -d "yesterday" +"%Y%m%d").log
7. mv ${logs\_path\_king}king\_nginx.log ${logs\_path\_king}king\_nginx\_$(date -d "yesterday" +"%Y%m%d").log
8. kill -USR1 `cat /usr/local/nginx/nginx.pid`

chmod +x /usr/local/nginx/sbin/cut\_nginx\_log.sh  
#添加计划任务，每天凌晨00:00切割nginx访问日志  
crontab -e  
00 00 \* \* \* /bin/bash  /usr/local/nginx/sbin/cut\_nginx\_log.sh

#为apache安装rpaf模块,该模块用于apache做后端时获取访客真实的IP  
#使用apxs安装模块.这里要使用此前apache编译安装后的apxs  
cd /opt  
tar -zxf mod\_rpaf-0.6.tar.gz  
cd mod\_rpaf-0.6  
/usr/local/apache/bin/apxs -i -c -n mod\_rpaf-2.0.so mod\_rpaf-2.0.c

#编辑/usr/local/apache/conf/httpd.conf,添加模块参数,查找LoadModule php5\_module modules/libphp5.so,在下方添加:

LoadModule rpaf\_module modules/mod\_rpaf-2.0.so  
#Mod\_rpaf settings  
RPAFenable On  
#上面出现的192.168.9.9请修改为你本机所监听web服务的ip.多个IP用空格空开  
RPAFproxy\_ips 127.0.0.1 192.168.9.9  
RPAFsethostname On  
RPAFheader X-Forwarded-For

本文出自 “[聆听未来](http://kerry.blog.51cto.com/)” 博客，请务必保留此出处<http://kerry.blog.51cto.com/172631/600590>