1. 将yum的源替换成国内的源

网易（163）yum源是国内最好的yum源之一 ，无论是速度还是软件版本，都非常的不错，将yum源设置为163yum，可以提升软件包安装和更新的速度，同时避免一些常见软件版本无法找到。具体设置方法如下：

1. 进入yum源配置目录  
   cd /etc/yum.repos.d
2. 备份系统自带的yum源  
   mv CentOS-Base.repo CentOS-Base.repo.bk

mv cobbler-config.repo cobbler-config.repo.bk  
下载163网易的yum源：  
wget http://mirrors.163.com/.help/CentOS6-Base-163.repo

1. 更新完yum源后，执行下边命令更新yum配置，使操作立即生效  
   yum makecache
2. 除了网易之外，国内还有其他不错的yum源，比如中科大和搜狐的，大家可以根据自己需求下载  
   中科大的yum源：wget http://centos.ustc.edu.cn/CentOS-Base.repo  
   sohu的yum源：wget http://mirrors.sohu.com/help/CentOS-Base-sohu.repo
3. 安装php

1．安装基础包和库：

yum -y install gcc gcc-c++ gd-devel libjpeg-devel libpng-devel freetype-devel libxml2-devel curl-devel php-gd\* zlib-devel openssl openssl-devel bzip2bzip2-devel **readline-devel libxslt-devel**

2．libiconv库为需要做转换的应用提供了一个iconv()的函数：

wget http://ftp.gnu.org/pub/gnu/libiconv/libiconv-1.14.tar.gz

tar zxvf libiconv-1.14.tar.gz

cd libiconv-1.14

./configure --prefix=/usr/local/libiconv

make && make install

cd ..

3．libmcrypt是加密算法扩展库：

wget http://iweb.dl.sourceforge.net/project/mcrypt/Libmcrypt/2.5.8/libmcrypt-2.5.8.tar.gz

tar zxvf libmcrypt-2.5.8.tar.gz

cd libmcrypt-2.5.8

./configure

make && make install

cd ..

4．Mhash是基于离散数学原理的不可逆向的php加密方式扩展库，其在默认情况下不开启。 mhash的可以用于创建校验数值，消息摘要，消息认证码，以及无需原文的关键信息保存：

wget http://downloads.sourceforge.net/project/mhash/mhash/0.9.9.9/mhash-0.9.9.9.tar.gz?r=&ts=1457067565&use\_mirror=nchc

tar zxvf mhash-0.9.9.9.tar.gz

cd mhash-0.9.9.9

./configure

make && make install

cd ..

5. mcrypt 是 php 里面重要的加密支持扩展库，Mcrypt扩展库可以实现加密解密功能，就是既能将明文加密，也可以密文还原:

wget http://iweb.dl.sourceforge.net/project/mcrypt/MCrypt/2.6.8/mcrypt-2.6.8.tar.gz

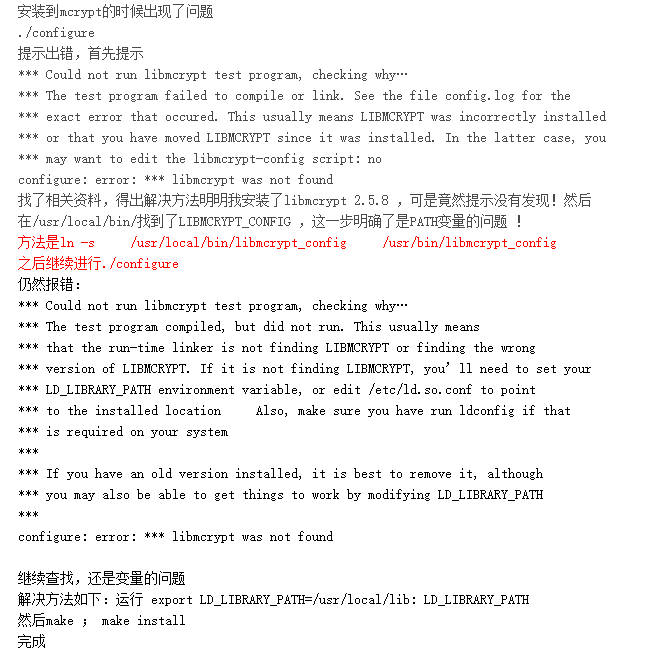
tar zxvf mcrypt-2.6.8.tar.gz

cd mcrypt-2.6.8

./configure

make && make install

cd ..



6. 安装php

wget http://cn2.php.net/get/php-5.5.32.tar.gz/from/this/mirror

tar zxvf php-5.5.32.tar.gz

cd php-5.5.32

./configure --prefix=/opt/local/php --with-config-file-path=/etc --with-mysql -with-xpm-dir --with-vpx-dir --with-t1lib=/opt/t1lib --with-iconv --with-libxml-dir --with-fpm-user=www --with-pdo-mysql --with-mysql-sock=/opt/local/config/mysql/mysql.sock --enable-shmop --enable-sysvsem --enable-inline-optimization --enable-opcache=no --enable-mbregex --enable-fpm --enable-pdo --enable-mbstring --enable-ftp --enable-gd-native-ttf --enable-shared --enable-bcmath --enable-soap --with-openssl --enable-pcntl --enable-sockets --enable-sysvmsg --enable-sysvshm --with-xmlrpc --without-pear --with-pcre-dir --enable-zip --enable-session --with-mcrypt --with-mhash --with-curl --enable-ctype --disable-debug --disable-rpath --with-zlib --with-bz2 --with-readline --enable-cli --with-xsl

./configure --prefix=/opt/local/php --with-config-file-path=/etc -with-xpm-dir --with-iconv --with-libxml-dir --with-apxs2=/usr/local/apache/bin/apxs

--with-fpm-user=www --with-pdo-mysql --with-mysql-sock=/opt/modules/config/mysql/mysql.sock --enable-shmop --enable-sysvsem --enable-inline-optimization --enable-opcache=no --enable-mbregex --enable-fpm --enable-pdo --enable-mbstring --enable-ftp --enable-gd-native-ttf --enable-shared --enable-bcmath --enable-soap --with-openssl --enable-pcntl --enable-sockets --enable-sysvmsg --enable-sysvshm --with-xmlrpc --without-pear --with-pcre-dir --enable-session --with-mcrypt --with-mhash --with-curl --enable-ctype --disable-debug --disable-rpath --with-zlib --with-bz2 --with-readline --enable-cli --with-xsl --with-mysqli

Php 7

./configure --prefix=/usr/local/php --with-config-file-path=/etc --with-mysqli -with-xpm-dir --with-vpx-dir --with-t1lib=/opt/t1lib --with-iconv --with-libxml-dir --with-fpm-user=www --with-pdo-mysql --with-mysql-sock=/var/lib/mysql/mysql.sock --with-apxs2 --enable-module=so --enable-shmop --enable-sysvsem --enable-inline-optimization --enable-opcache=no --enable-mbregex --enable-fpm --enable-pdo --enable-mbstring --enable-ftp --enable-gd-native-ttf --enable-shared --enable-bcmath --enable-soap --with-openssl --enable-pcntl --enable-sockets --enable-sysvmsg --enable-sysvshm --with-xmlrpc --without-pear --with-pcre-dir --enable-zip --enable-session --with-mcrypt --with-mhash --with-curl --enable-ctype --disable-debug --disable-rpath --with-zlib --with-bz2 --with-readline --enable-cli --with-xsl

make && make install

修改php.ini（该步骤可暂时省略，后续再改）

vi /etc/php.ini

修改php-fpm配置：

cd /opt/local/php/etc/

mv php-fpm.conf.default php-fpm.conf

修改该文件

复制php-fpm启动脚本到init.d：

cp /data/tmp/php-5.5.32/sapi/fpm/init.d.php-fpm /etc/init.d/php-fpm

赋予执行权限：

chmod +x /etc/init.d/php-fpm

添加为启动项，并设置开机启动：

vi /etc/init.d/php-fpm

在 #！/bin/sh下面添加以下两行

# chkconfig: 2345 15 95

# description: PHP-FPM (FastCGI Process Manager) is an alternative PHP FastCGI implementation \

# with some additional features useful for sites of any size, especially busier sites.

chkconfig --add php-fpm

chkconfig php-fpm on

给php-fpm创建一个指定的用户和组：

groupadd www //创建群组

useradd -s /sbin/nologin -g www -M www //创建一个用户，不允许登陆和不创建主目录

启动php-fpm：

service php-fpm start

gd库的安装

1. freetype

wget "http://download.savannah.gnu.org/releases/freetype/freetype-2.4.0.tar.bz2"

tar jxvf freetype-2.4.0.tar.bz2

cd freetype-2.4.0

./configure --prefix=/usr/local/freetype && make && make install

2. jpegsrc

wget "http://www.ijg.org/files/jpegsrc.v9.tar.gz"

tar zxvf jpegsrc.v9.tar.gz

cd jpeg-9

CFLAGS="-O3 -fPIC" ./configure --prefix=/usr/local/jpeg && make && make install

mkdir -p /usr/local/jpeg/include

mkdir -p /usr/local/jpeg/lib

mkdir -p /usr/local/jpeg/bin

mkdir -p /usr/local/jpeg/man/man1

3. libpng

wget "http://downloads.sourceforge.net/project/libpng/libpng12/1.2.50/libpng-1.2.50.tar.gz?r=http%3A%2F%2Fwww.libpng.org%2Fpub%2Fpng%2Flibpng.html&ts=1376631135&use\_mirror=nchc"

tar zxvf libpng-1.2.50.tar.gz

cd libpng-1.2.50

CFLAGS="-O3 -fPIC" ./configure --prefix=/usr/local/libpng && make && make install

接下来需要cd到php源码的 [gd](http://www.nowamagic.net/librarys/veda/tag/gd) 目录，否则会报 cannot find config.m4 之类的错误。

cd php-5.3.6

cd ext

cd gd

然后再这个目录执行命令 /usr/local/php/bin/phpize。什么时候需要用到 [phpize](http://www.nowamagic.net/librarys/veda/tag/phpize) 呢？当我们需要再加些模块，又不想重新编译php，这些我们就可以用phpize了。我的PHP安装在 /usr/local/php/ 这个目录里，可以根据个人情况修改。

/usr/local/php/bin/phpize

现在可以进行单独编译安装了：

./configure --with-php-configure=/usr/local/php/bin/php-config --with-jpeg-dir=/usr/local/jpeg --with-png-dir=/usr/local/libpng --with-freetype-dir=/usr/local/freetype

或者

./configure --with-php-config=/usr/local/php/bin/php-config --with-jpeg-dir=/usr/local/jpeg --with-png-dir=/usr/local/libpng --with-freetype-dir=/usr/local/freetype

make && make install

接着修改php.ini文件 在 ;extension=php\_zip.dll 下面添加一行 extension=gd.so

重启 Nginx 之后再看看是否安装成功了。再执行 /usr/local/php/bin/php -m 查看gd模块是否加载成功。

service nginx restart

/usr/local/php/bin/php -m

1. 安装nginx
2. 安装基础包和库：

yum install pcre-devel

1. 下载以下lib包并解压：

wget http://downloads.sourceforge.net/project/pcre/pcre/8.37/pcre-8.37.tar.gz?r=https%3A%2F%2Fsourceforge.net%2Fprojects%2Fpcre%2Ffiles%2Fpcre%2F8.37%2F&ts=1457155584&use\_mirror=nchc

wget http://downloads.sourceforge.net/project/libpng/zlib/1.2.8/zlib-1.2.8.tar.gz?r=http%3A%2F%2Fwww.zlib.net%2F&ts=1457155908&use\_mirror=nchc

wget http://www.openssl.org/source/openssl-1.0.1g.tar.gz

1. 安装nginx：

wget http://nginx.org/download/nginx-1.8.1.tar.gz

tar zxvf nginx-1.8.1.tar.gz

cd nginx-1.8.1

./configure --user=www --group=www --prefix=/opt/local/nginx --with-http\_gzip\_static\_module --with-pcre=/opt/download/pcre-8.37 --with-zlib=/opt/download/zlib-1.2.8 --with-openssl=/opt/download/openssl-1.0.1g

make && make install

4．修改nginx的配置文件：

cd /opt/local/nginx/conf，修改该文件夹下的nginx.conf，并新建conf.d文件夹及下面的文件，具体可以参考已安装完成的那几台服务器

5．把nginx设置为开机自启动程序：

vi /etc/init.d/nginx，具体的脚本内容参考已装好的服务器；

chmod +x /etc/init.d/nginx

chkconfig --add nginx

chkconfig nginx on

6．启动nginx并检查端口

Service nginx start

netstat -lntp

1. 安装mysql

yum install -**y** make apr\* autoconf automake curl curl-devel gcc gcc-c++ gtk+-devel zlib-devel openssl openssl-devel pcre-devel gd kernel keyutils patch perl kernel-headers compat\* cpp glibc libgomp libstdc++-devel keyutils-libs-devel libsepol-devel libselinux-devel krb5-devel libXpm\* freetype freetype-devel freetype\* fontconfig fontconfig-devel libjpeg\* libpng\* php-common php-gd gettext gettext-devel ncurses\* libtool\* libxml2 libxml2-devel patch policycoreutils bison

#Mysql  
yum -y install bison gcc gcc-c++ autoconf automake zlib\* libxml\* ncurses-devel libtool-ltdl-devel\* mysql-devel

1. 下载mysql-5.5.21

wget [http://dev.mysql.com/get/Downloads/MySQL-5.5/mysql-5.5.21.tar.gz/from/http://mysql.mirror.rafal.ca/](http://dev.mysql.com/get/Downloads/MySQL-5.5/mysql-5.5.8.tar.gz/from/http://mysql.mirror.rafal.ca/)

下载cmake wget --no-check-certificate https://cmake.org/files/v3.4/cmake-3.4.1.tar.gz

安装mysql

#cd /opt/download

#tar zxvf cmake-3.4.1.tar.gz

#cd cmake-3.4.1

#./configure --prefix=/opt/local/cmake(配置安装的目录)

#make

#make install

2.Mysql的安装

#添加mysql用户  
#groupadd mysql /mysql组   
#useradd -g mysql mysql /mysql用户

#cd /opt/download

#tar –zxvf mysql-5.5.21.tar.gz

#cd mysql-5.5.21()

编译

cmake -DCMAKE\_INSTALL\_PREFIX=/opt/local/mysql -DMYSQL\_DATADIR=/opt/data/mysql -DMYSQL\_UNIX\_ADDR=/opt/local/config/mysql/mysql.sock -DSYSCONFDIR=/opt/local/config/mysql -DMYSQL\_TCP\_PORT=3306 -DDEFAULT\_CHARSET=utf8 -DDEFAULT\_COLLATION=utf8\_general\_ci -DEXTRA\_CHARSETS=all -DWITH\_EXTRA\_CHARSETS:STRING=utf8 -DWITH\_EMBEDDED\_SERVER=0 -DWITH\_MYISAM\_STORAGE\_ENGINE=1 -DWITH\_INNOBASE\_STORAGE\_ENGINE=1 -DWITH\_MEMORY\_STORAGE\_ENGINE=1 -DWITH\_READLINE=1 -DENABLED\_LOCAL\_INFILE=1 -DMYSQL\_USER=mysql -DWITH\_DEBUG=0 -DWITH\_SSL=yes

export PATH=/opt/local/cmake/bin:$PATH

Make && make install

cp /etc/my.cnf /opt/modules/config/mysql/my.cnf

156 mv /etc/my.cnf /etc/my.cnf.bak

157 cd /opt/modules/

158 cd mysql

159 ll

160 cd scripts

161 ll

162 /opt/modules/mysql/scripts/mysql\_install\_db --defaults-file=/opt/modules/config/mysql/my.cnf --basedir=/opt/modules/mysql --datadir=/opt/data/mysql --user=mysql

163 ll

164 cd ../

165 ll

166 cp support-files/mysql.server /etc/init.d/mysqld

167 chmod +x /etc/init.d/mysqld

168 chkconfig mysqld on

169 cd /etc/init.d/

170 service mysqld start

171 cd /opt/ local /mysql

172 scripts/mysql\_install\_db

173 cd /etc/init.d/

174 service mysqld start

175 vi /opt/local/config/mysql/my.cnf

176 cd /

177 ll

178 cd data

179 ll

180 mkdir log

181 vi /opt/local/config/mysql/my.cnf

182 cd /etc/init.d/

183 service mysqld stop

184 service mysqld start

185 netstat -lupt

186 mysqladmin -u root password 'chinamobile'

cd support-files/

如果还有my.cnf请备份

cp /etc/my.cnf /opt/modules/config/mysql/my.cnf

如果愿意也可以复制配置文件到etc下

cp my-default.cnf /etc/my.cnf(这边是你的mysql配置目录)—上面的我是在/opt/modules/config/mysql

执行初始化配置脚本，创建系统自带的数据库和表，注意配置文件的路径

/opt/modules/mysql/scripts/mysql\_install\_db --defaults-file=/opt/modules/config/mysql/my.cnf --basedir=/opt/modules/mysql --datadir=/opt/data/mysql --user=mysql

拷贝mysql安装目录下support-files服务脚本到init.d目录

#拷贝脚本

cp support-files/mysql.server /etc/init.d/mysqld

#赋予权限

chmod +x /etc/init.d/mysqld

设置开机启动

chkconfig mysqld on

启动MySQL

cd /etc/init.d/

service mysqld start

[root@localhost mysql]# /etc/rc.d/init.d/mysql status  
MySQL is not running, but lock file (/var/lock/subsys/mysql[FAILED]  
[root@localhost mysql]# /etc/rc.d/init.d/mysql start  
Starting MySQL...The server quit without updating PID file (/usr/local/mysql/data/localhost.localdomain.pid).                              [FAILED]

2 原因

没有初始化权限表

3 解决办法

#cd /opt/modules/mysql（进入mysql安装目录）  
scripts/mysql\_install\_db

vi /opt/modules/config/mysql/my.cnf

Log-error=/data/log/mysqld.log

保存后重启数据库

cd /etc/init.d/

service mysqld stop

service mysqld start

设置数据库密码

mysqladmin -u root password ‘chinamobile’

Mysql主从服务器配置

服务器配置

主服务器 10.200.18.74

server-id = 1

binlog-ignore-db = mysql

binlog-ignore-db =information\_schema

sync-binlog = 1

从服务器 10.200.18.77 10.200.18.80

server-id = 2/3

replicate-ignore-db = mysql

replicate-ignore-db = information\_schema

改完配置后重启mysql

进入数据库

命令SHOW MASTER STATUS查看主服务状态

mysql> SHOW MASTER STATUS; +---------------------+----------+--------------+--------------------------+ | File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | +---------------------+----------+--------------+--------------------------+ | mysql-binlog.000001 | 525 | | mysql,information\_schema | +---------------------+----------+--------------+--------------------------+ row in set (0.00 sec)

**主服务器A授权同步账户：**

GRANT REPLICATION SLAVE ON \*.\* TO 'root'@'10.200.18.77(80)' IDENTIFIED BY 'password';

FLUSH PRIVILEGES;

Slave start开始复制

CHANGE MASTER TO MASTER\_HOST='10.200.18.74',MASTER\_USER='admin', MASTER\_PASSWORD='chinamobile',MASTER\_LOG\_FILE=' mysql-bin.000012 ',MASTER\_LOG\_POS=17209;

开始slave同步:mysql>slave start

停止slave同步：mysql>slave stop

在slave服务器上使用show slave status查看slave同步的状态

显示下面的成功

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

如果无法连接的话，查看端口，防火墙；

# 错误Got fatal error 1236

mysql-bin.000010

672997090

flush privileges;

mysql> stop slave;

mysql> change master to master\_log\_file='mysql-bin.000288',master\_log\_pos=141304737;

mysql> start slave;

显示

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

配置完成

wget [http://dev.mysql.com/get/Downloads/MySQL-5.5/mysql-5.5.21.tar.gz/from/http://mysql.mirror.rafal.ca/](http://dev.mysql.com/get/Downloads/MySQL-5.5/mysql-5.5.21.tar.gz/from/http:/mysql.mirror.rafal.ca/)

wget --no-check-certificate https://cmake.org/files/v3.4/cmake-3.4.1.tar.gz

一般性操作都只在主数据库上操作，从数据库上一般不直接操作