一、系统约定

软件源代码包存放位置/usr/local/src

源码包编译安装位置(prefix)/usr/local/software\_name

脚本以及维护程序存放位置/usr/local/sbin

MySQL 数据库位置/usr/local/mysql/data（可按情况设置）

Nginx 网站根目录/usr/local/nginx/html（可按情况设置）

Nginx 虚拟主机日志根目录/usr/local/nginx/logs（可按情况设置）

Nginx 运行账户www:www

使用 yum 程序安装所需开发包（以下为标准的 RPM 包名称）：

# yum -y install ntp make openssl openssl-devel pcre pcre-devel libpng libpng-devel libjpeg-6b libjpeg-devel-6b freetype freetype-devel gd gd-devel zlib zlib-devel gcc gcc-c++ libXpm libXpm-devel ncurses ncurses-devel libmcrypt libmcrypt-devel libxml2 libxml2-devel imake autoconf automake screen sysstat compat-libstdc++-33 curl curl-devel

yum -y install cmake

1. 编译安装MySQL

建立mysql用户

/usr/sbin/groupadd mysql

/usr/sbin/useradd -g mysql mysql

tar zxvf mysql-5.5.30.tar.gz

cd mysql-5.5.30

cmake -DCMAKE\_INSTALL\_PREFIX=/usr/local/mysql -DMYSQL\_DATADIR=/usr/local/mysql/data -DEXTRA\_CHARSETS=all

make

make install

cp /usr/local/src/mysql-5.5.30/support-files/my-huge.cnf /usr/local/mysql/my.cnf

chmod +w /usr/local/mysql

chown -R mysql:mysql /usr/local/mysql

cp /usr/local/src/mysql-5.5.30/support-files/mysql.server /etc/init.d/mysql

chmod +x /etc/init.d/mysql

/sbin/chkconfig --add mysql

设置mysql开机启动

/sbin/chkconfig mysql on

初始化mysql库

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

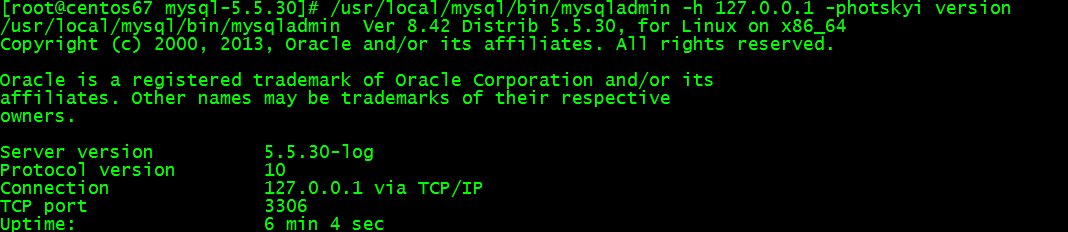
/usr/local/mysql/bin/mysqld\_safe --defaults-file=/usr/local/mysql/my.cnf &

设置mysql库的root的密码

/usr/local/mysql/bin/mysqladmin -u root password 'hotskyi'

查看mysql库的版本

/usr/local/mysql/bin/mysqladmin -h 127.0.0.1 -photskyi version



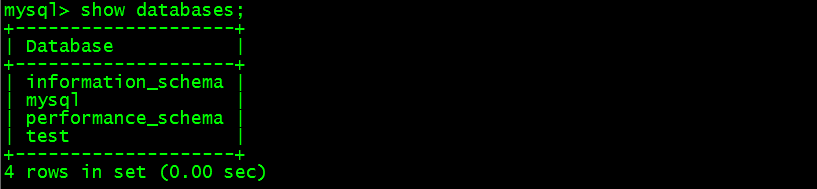
rm -fr /etc/my.cnf

service mysql start



/usr/local/mysql/bin/mysql -u root -p

show databases;



1. 编译安装nginx

cd /usr/local/src/

tar zxvf nginx-1.8.0.tar.gz

cd nginx-1.8.0

./configure --prefix=/usr/local/nginx --conf-path=/usr/local/nginx/conf/nginx.conf --with-http\_realip\_module --with-http\_addition\_module --with-http\_gzip\_static\_module --with-http\_random\_index\_module --with-http\_stub\_status\_module --with-http\_sub\_module --with-http\_dav\_module

make

make install clean

/usr/local/nginx/sbin/nginx

设置nginx开机启动

vim /etc/rc.d/init.d/nginx #编辑启动文件添加下面内容

#!/bin/bash

# nginx Startup script for the Nginx HTTP Server

# it is v.0.0.2 version.

# chkconfig: - 85 15

# description: Nginx is a high-performance web and proxy server.

# It has a lot of features, but it's not for everyone.

# processname: nginx

# pidfile: /var/run/nginx.pid

# config: /usr/local/nginx/conf/nginx.conf

nginxd=/usr/local/nginx/sbin/nginx

nginx\_config=/usr/local/nginx/conf/nginx.conf

nginx\_pid=/usr/local/nginx/logs/nginx.pid

RETVAL=0

prog="nginx"

# Source function library.

. /etc/rc.d/init.d/functions

# Source networking configuration.

. /etc/sysconfig/network

# Check that networking is up.

[ ${NETWORKING} = "no" ] && exit 0

[ -x $nginxd ] || exit 0

# Start nginx daemons functions.

start() {

if [ -e $nginx\_pid ];then

echo "nginx already running...."

exit 1

fi

echo -n $"Starting $prog: "

daemon $nginxd -c ${nginx\_config}

RETVAL=$?

echo

[ $RETVAL = 0 ] && touch /var/lock/subsys/nginx

return $RETVAL

}

# Stop nginx daemons functions.

stop() {

echo -n $"Stopping $prog: "

killproc $nginxd

RETVAL=$?

echo

[ $RETVAL = 0 ] && rm -f /var/lock/subsys/nginx /usr/local/nginx/logs/nginx.pid

}

reload() {

echo -n $"Reloading $prog: "

#kill -HUP `cat ${nginx\_pid}`

killproc $nginxd -HUP

RETVAL=$?

echo

}

# See how we were called.

case "$1" in

start)

start

;;

stop)

stop

;;

reload)

reload

;;

restart)

stop

start

;;

status)

status $prog

RETVAL=$?

;;

\*)

echo $"Usage: $prog {start|stop|restart|reload|status|help}"

exit 1

esac

exit $RETVAL

chmod 775 /etc/rc.d/init.d/nginx #赋予执行权限

chkconfig nginx on #设置开机启动

/etc/rc.d/init.d/nginx restart

service nginx restart

1. 编译安装PHP所需的支持库

Libiconv

# cd /usr/local/src/

# tar zxvf libiconv-1.14.tar.gz

# cd libiconv-1.14/

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

# make

# make install

# cd ..

Libmcrypt

# tar xjvf 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 /usr/local/src/

Mhash

# tar zxvf mhash-0.9.9.9.tar.gz

# cd mhash-0.9.9.9/

# ./configure

# make

# make install

# cd ..

# 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

# ln -s /usr/local/bin/libmcrypt-config /usr/bin/libmcrypt-config

# tar zxvf mcrypt-2.6.8.tar.gz

# cd mcrypt-2.6.8

# /sbin/ldconfig

# ./configure

# make

# make install

1. 编译安装PHP

# cd /usr/local/src

# tar jxvf php-5.5.18.tar.bz2

# cd php-5.5.18

# ./configure --prefix=/usr/local/php --with-config-file-path=/usr/local/etc/cgi --with-mysql=/usr/local/mysql --with-mysqli=/usr/local/mysql/bin/mysql\_config --with-iconv-dir=/usr/local --with-freetype-dir=/usr --with-jpeg-dir=/usr --with-png-dir=/usr --with-zlib --with-libxml-dir=/usr --with-xmlrpc --enable-xml --enable-mbstring --enable-ftp --with-gd --enable-magic-quotes --with-pear --enable-sockets --with-ttf --enable-gd-native-ttf --enable-sysvsem --enable-sysvshm --enable-shmop --enable-force-cgi-redirect --enable-fastcgi --enable-zip --enable-fpm --with-mhash --with-mcrypt --enable-discard-path --enable-bcmath --enable-inline-optimization --with-curl --with-curlwrappers --enable-mbregex --with-openssl

# make ZEND\_EXTRA\_LIBS='-liconv'

# make install

# cp php.ini-production /usr/local/php/etc/php.ini

# cp sapi/fpm/init.d.php-fpm /etc/init.d/php-fpm

# cp sapi/fpm/php-fpm.conf /usr/local/php/etc/php-fpm.conf

修改php.ini文件和php-fpm文件中参数，可以手动更改也可以用下面的sed命令更改

sed -i 's#short\_open\_tag = Off#short\_open\_tag = On#g' /usr/local/php/etc/php.ini

sed -i 's#;pid = run/php-fpm.pid#pid = run/php-fpm.pid#g' /usr/local/php/etc/php-fpm.conf

sed -i 's#pm.max\_children = 5#pm.max\_children = 32#g' /usr/local/php/etc/php-fpm.conf

sed -i 's#pm.start\_servers = 2#pm.start\_servers = 16#g' /usr/local/php/etc/php-fpm.conf

sed -i 's#pm.min\_spare\_servers = 1#pm.min\_spare\_servers = 8#g' /usr/local/php/etc/php-fpm.conf

sed -i 's#pm.max\_spare\_servers = 3#pm.max\_spare\_servers = 32#g' /usr/local/php/etc/php-fpm.conf

sed -i 's#;pm.max\_requests = 500#pm.max\_requests = 120#g' /usr/local/php/etc/php-fpm.conf

#chmod 755 /etc/init.d/php-fpm

#chkconfig --add php-fpm

#chkconfig --level 345 php-fpm on

#/etc/init.d/php-fpm start

修改nginx配置文件让PHP的请求转发到php-fpm所绑定的ip和端口上：

vim /usr/local/nginx/conf/nginx.conf

location ~ \.php$ {

root /usr/local/nginx/html;

fastcgi\_pass 127.0.0.1:9000;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

include fastcgi\_params;

}

代码解释：

当用户的请求URL是以 .php 为结尾的时候

则执行这个区块的内容：

root 指定的目录

pass 转发端口[e]

index 如果是指定目录结尾的时候 自动加上 index.php

param 要执行documnetRoot 里的文件 传递过来的文件名

include 在家配置文件[e]

重启nginx: service nginx restart

验证是否处理请求PHP文件：

vi /usr/local/nginx/html/info.php

<?php

phpinfo();

访问IP网址：

http://172.16.60.247/info.php