

Centos6.2_X86_64_LNMP 安装全程实录

2012-05-04 01:56:35

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 、作者信息和本声明。

否则将追究法律责任。 <http://itnihao.blog.51cto.com/1741976/851727>

操作系统版本 Centos6.2X_86_64

系统环境的设置

```
for i in `rpm -q gcc gcc-c++ autoconf 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 krb5-devel libidn
libidn-devel openssl openssl-devel bison bison-devel readline readline-devel libmcrypt
libmcrypt-devel net-snmp-devel|grep 'not installed' | awk '{print $2}'`;
do
    yum -y install $i;
done
```

mysql 的安装

```
wget http://mysql.ntu.edu.tw/Downloads/MySQL-5.5/mysql-5.5.23.tar.gz
groupadd -g 27 mysql
useradd -g 27 -s /sbin/nologin mysql
cd mysql
tar xvf cmake-2.8.8.tar.gz
cd cmake-2.8.8
./configure
make && make install 2>./lnpp_install_error.log
if [ $? != 0 ]
then
exit 1
fi
cd ..
echo "install cmake is succeed"
Cmake . -DCMAKE_INSTALL_PREFIX=/usr/local/mysql -DSYSCONFDIR=/etc
-DINSTALL_MANDIR=/usr/share/man -DENABLED_PROFILING=ON -DMY
SQL_DATADIR=/usr/local/mysql/data/ -DWITH_EXTRA_CHARSETS=all
-DWITH_READLINE=ON -DWITH_LIBWRAP=on -DWITH_SSL=system -DWITH_DE
BUG=0 -DWITH_MYISAM_STORAGE_ENGINE=1
-DWITH_INNOBASE_STORAGE_ENGINE=1 -DWITH_PARTITION_STORAGE_ENGINE=1
-DWITH_PERFSCHEMA_STO
RAGE_ENGINE=1 -DWITH_FEDERATED_STORAGE_ENGINE=1
-DENABLED_LOCAL_INFILE=1

make && make install 2> ./lnpp_install_error.log
cp support-files/my-medium.cnf /etc/my.cnf
cp support-files/mysql.server /etc/init.d/mysqld
chmod 755 /etc/init.d/mysqld
chown -R mysql:mysql /usr/local/mysql/
```

```
#sed -i '/^\[mysqld\]$/a set-variable=max_connections=2000' /etc/my.cnf
cp extra/my_print_defaults /usr/local/mysql/bin/
chkconfig --add mysqld
chkconfig mysqld on
ln -s /usr/local/mysql/lib/libmysqlclient.so.18 /usr/lib/
ln -s /usr/local/mysql/lib/libmysqlclient.so.18 /usr/lib64/
ln -s /usr/local/mysql/bin/mysql /bin/mysql
/usr/local/mysql/scripts/mysql_install_db --user=mysql --basedir=/usr/local/mysql
--datadir=/usr/local/mysql/data
/usr/local/mysql/bin/mysqld_safe &
service mysqld restart
```

php 的安装

```
#wget http://downloads.sourceforge.net/project/mcrypt/Libmcrypt/2.5.8/libmcrypt-2.5.8.tar.gz
tar xvf libmcrypt-2.5.8.tar.gz
cd libmcrypt-2.5.8
./configure --prefix=/usr/local/
make && make install
cd ..

#wget http://downloads.sourceforge.net/project/mhash/mhash/0.9.9.9/mhash-0.9.9.9.tar.gz
tar xvf mhash-0.9.9.9.tar.gz
cd mhash-0.9.9.9
./configure --prefix=/usr/local/
make && make install
cd ..

#wget http://downloads.sourceforge.net/project/mcrypt/MCrypt/2.6.8/mcrypt-2.6.8.tar.gz
tar xvf mcrypt-2.6.8.tar.gz
cd mcrypt-2.6.8
LD_LIBRARY_PATH=/usr/local/lib ./configure --prefix=/usr/local/
make && make install
cd ..

#wget http://ftp.gnu.org/pub/gnu/libiconv/libiconv-1.14.tar.gz tar xvf libiconv-1.14.tar.gz cd
libiconv-1.14 ./configure --prefix=/usr/local make && make install cd . ln -s /usr/local/mysql/lib/*
/usr/lib ln -s /usr/local/mysql/lib/* /usr/lib64 echo "/usr/local/lib" >>/etc/ld.so.conf
groupadd -g 28 www useradd -g 28 -u 28 -M -s /sbin/nologin www mkdir /etc/php.d
#wget http://cn2.php.net/distributions/php-5.3.11.tar.gz
tar xvf php-5.3.11.tar.gz
cd php-5.3.11
./configure --prefix=/usr --exec-prefix=/usr --bindir=/usr/bin --sbindir=/usr/sbin
--libexecdir=/usr/libexec --datadir=/usr/share --sysconfdir=/etc --sharedstatedir=/var/lib
--localstatedir=/var --libdir=/usr/lib64 --includedir=/usr/include --infodir=/usr/share/info
--mandir=/usr/share/man --with-libdir=/lib64 --enable-fpm --with-fpm-user=www
--with-fpm-group=www --with-curl --with-config-file-path=/etc --with-config-file-scan-dir=/etc/php.d
```

```
--with-exec-dir=/usr/bin --disable-ipv6 --with-libxml-dir --with-openssl --with-pcre-regex
--with-zlib --with-bz2 --with-curl --with-curlwrappers --with-pcre-dir --with-jpeg-dir --with-png-dir
--with-zlib-dir --with-xpm-dir --with-freetype-dir --enable-gd-native-ttf
--enable-gd-jis-conv --with-mhash --with-mcrypt --with-mysql
--with-mysql-sock=/etc/mysql.sock --with-mysqli=/usr/local/mysql/bin/mysql_config
--with-zlib-dir --with-zlib-dir --with-readline --enable-shmop --with-iconv-dir=/usr/local
--with-libxml-dir=/usr --with-snmp --with-openssl-dir --enable-soap --enable-sockets --enable-zip
sed -i "s/-lm -lxml2 -lz -lm -lcrypt/-lm -lxml2 -lz -lm -lcrypt -liconv/g" Makefile
make && make install
cp php.ini-development /etc/php.ini
cp /etc/php-fpm.conf.default /etc/php-fpm.conf sed -i "s;/date.timezone =;/date.timezone =Asia/Chongqing/g" /etc/php.ini
cd ./
```

pcre 的安装

```
wget http://downloads.sourceforge.net/project/pcre/pcre/8.30/pcre-8.30.tar.gz
tar xvf pcre-8.30.tar.gz
cd pcre-8.30
./configure
make && make install
```

nginx 的安装

```
wget http://nginx.org/download/nginx-1.2.0.tar.gz
tar xvf nginx-1.2.0.tar.gz
cd nginx-1.2.0
mkdir /etc/nginx/
mkdir /var/log/nginx
mkdir /var/tmp/nginx
./configure --prefix=/usr --sbin-path=/usr/sbin --conf-path=/etc/nginx/nginx.conf
--error-log-path=/var/log/nginx/error.log --http-log-path=/var/log/nginx/access.log
--pid-path=/var/log/nginx/nginx.pid --lock-path=/var/lock/nginx.lock --user=www --group=www
--with-http_ssl_module --with-http_flv_module --with-http_mp4_module
--with-http_gzip_static_module --with-http_stub_status_module
--with-google_perftools_module --http-proxy-temp-path=/var/tmp/nginx/proxy
--http-fastcgi-temp-path=/var/tmp/nginx/fcgi --http-client-body-temp-path=/var/tmp/nginx/client
--http-uwsgi-temp-path=/var/tmp/nginx/uwsgi --http-scgi-temp-path=/var/tmp/nginx/scgi
make && make install
```

安装 google_perftools 工具，可以进一步提 gao

```
wget http://download.savannah.gnu.org/releases/libunwind/libunwind-1.0.1.tar.gz
wget http://gperftools.googlecode.com/files/gperftools-2.0.tar.gz
tar xvf libunwind-1.0.tar.gz
cd libunwind-1.0
./configure
make
```

```
make install
cd ../
tar xvf gperftools-2.0.tar.gz
cd gperftools-2.0
./configure --disable-cpu-profiler --disable-heap-profiler --disable-heap-checker --enable-minimal
--disable-dependency-tracking
make && make install
echo "/usr/local/lib" >> /etc/ld.so.conf.d/usr_local_lib.conf
echo "/usr/local/include" >> /etc/ld.so.conf
ldconfig
echo -e "<?php\n phpinfo() \n ?>" >/usr/html/test.php
#more information see http://wiki.nginx.org/NginxGooglePerftoolsModule
mkdir /var/tmp/nginx/tcmalloc chmod 0777 /var/tmp/nginx/tcmalloc
sed -i 's/pid logs/nginx.pid;/agoogole_perftools_profiles \\\var\\tmp\\nginx\\tcmalloc\\tcmalloc;'
nginx.conf
chown www.root /var/tmp/nginx/tcmalloc lsof -n | grep tcmalloc
cp nginx.sh /etc/init.d/
cp php-fpm.sh /etc/init.d/php-fpm
chmod 755 /etc/init.d/nginx
chmod 755 /etc/init.d/php-fpm
chkconfig --add nginx chkconfig --add php-fpm chkconfig --levels 345 nginx on chkconfig
--levels 345 php-fpm on
service nginx start
service php-fpm start
cp nginx_log.sh /home/nginx_log
chmod 755 /home/nginx_log
echo "1 0 * * * cd /home;/home/nginx_log" >/tmp/nginx_log.tmp
crontab -l >>/tmp/nginx_log.tmp
crontab -u root /tmp/nginx_log.tmp
rm -f /tmp/nginx_log.tmp
```

nginx 的脚本如下 (nginx.sh)

```
#!/bin/sh
#
# nginx - this script starts and stops the nginx daemon
#
# chkconfig:   - 85 15
# description:  Nginx is an HTTP(S) server, HTTP(S) reverse \
#               proxy and IMAP/POP3 proxy server
# processname: nginx
# config:      /etc/nginx/nginx.conf
# config:      /etc/sysconfig/nginx
# pidfile:     /var/run/nginx.pid
# Source function library.
. /etc/rc.d/init.d/functions
```

```
# Source networking configuration.
. /etc/sysconfig/network
# Check that networking is up.
[ "$NETWORKING" = "no" ] && exit 0
nginx="/usr/sbin/nginx"
prog=$(basename $nginx)
NGINX_CONF_FILE="/etc/nginx/nginx.conf"
[ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx
lockfile=/var/lock/subsys/nginx
make_dirs() {
    # make required directories
    user=`$nginx -V 2>&1 | grep "configure arguments:" | sed 's/^[^]*--user=\([^ ]*\).*/\1/g' -`
    if [ -z "`grep $user /etc/passwd`" ]; then
        useradd -M -s /bin/nologin $user
    fi
    options=`$nginx -V 2>&1 | grep 'configure arguments:'`
    for opt in $options; do
        if [ `echo $opt | grep '.*-temp-path'` ]; then
            value=`echo $opt | cut -d "=" -f 2`
            if [ ! -d "$value" ]; then
                # echo "creating" $value
                mkdir -p $value && chown -R $user $value
            fi
        fi
    done
}
start() {
    [ -x $nginx ] || exit 5
    [ -f $NGINX_CONF_FILE ] || exit 6
    make_dirs
    echo -n "Starting $prog: "
    daemon $nginx -c $NGINX_CONF_FILE
    retval=$?
    echo
    [ $retval -eq 0 ] && touch $lockfile
    return $retval
}
stop() {
    echo -n "Stopping $prog: "
    killproc $prog -QUIT
    retval=$?
    echo
    [ $retval -eq 0 ] && rm -f $lockfile
    return $retval
}
```

```
}
restart() {
    configtest || return $?
    stop
    sleep 1
    start
}
reload() {
    configtest || return $?
    echo -n $"Reloading $prog: "
    killproc $nginx -HUP
    RETVAL=$?
    echo
}
force_reload() {
    restart
}
configtest() {
    $nginx -t -c $NGINX_CONF_FILE
}
rh_status() {
    status $prog
}
rh_status_q() {
    rh_status >/dev/null 2>&1
}
case "$1" in
    start)
        rh_status_q && exit 0
        $1
        ;;
    stop)
        rh_status_q || exit 0
        $1
        ;;
    restart|configtest)
        $1
        ;;
    reload)
        rh_status_q || exit 7
        $1
        ;;
    force-reload)
        force_reload

```

```
;;
status)
    rh_status
;;
condrestart|try-restart)
    rh_status_q || exit 0
;;
*)
    echo                                $"Usage:                                $0
{start|stop|status|restart|condrestart|try-restart|reload|force-reload|configtest}"
    exit 2
esac
```

php-fpm 脚本内容如下

```
#!/bin/bash
#
# Startup script for the PHP FastCGI server.
#
# chkconfig: 345 85 15
# description: PHP is an HTML-embedded scripting language
# processname: php
# config: /etc/php.ini

# Source function library.
. /etc/rc.d/init.d/functions

PHPFCGI="/usr/sbin/php-fpm"
FCGIPORT="9000"
FCGIADDR="127.0.0.1"
FCGI_WEB_SERVER_ADDRS="127.0.0.1"
PHP_FCGI_CHILDREN=3
PHP_FCGI_MAX_REQUESTS=1000
ALLOWED_ENV="PATH USER"
PHPUSER=www
PIDFILE=/var/run/php-fpm.pid
PHP_FILE=etc/php.ini
PHP_FPM_FILE=/etc/php-fpm.conf
if [ -z "$PHP_FCGI_CHILDREN" ]; then
    PHP_FCGI_CHILDREN=5
fi
#ALLOWED_ENV="$ALLOWED_ENV                                PHP_FCGI_CHILDREN
PHP_FCGI_MAX_REQUESTS FCGI_WEB_SERVER_ADDRS"
case "$1" in
    start)
        PHPFCGI_START=$"Starting ${NAME} service: "
```

```
echo -n $PHPFCGI_START

# check for $PHPUSER, create if non-existent
if [ -z "`id -u $PHPUSER 2> /dev/null`" ]; then
    useradd -s /sbin/nologin $PHPUSER
fi

# clean environment
E=
for i in $ALLOWED_ENV; do E="$E ${i}"; done
# daemon --user $PHPUSER --pidfile $PIDFILE "env - $E $PHPFCGI -q -b
$FCGIADDR:$FCGIPORT &> /dev/null &"
daemon $PHPFCGI -y $PHP_FPM_FILE -c $PHP_FILE
pid=`pidof php-fpm`
if [ -n "$pid" ]; then
    echo $pid > $PIDFILE
    success $PHPFCGI_START
else
    failure $PHPFCGI_START
fi
echo
;;
stop)
    echo -n "Stopping php-fpm: "
    killproc -p $PIDFILE php-fpm.pid
    echo
    ;;
status)
    status php-fpm
    ;;
restart)
    $0 stop
    $0 start
    ;;
*)
    echo "Usage: $0 {start|stop|status|restart}"
    exit 1
esac

exit 0
```

nginx 日志脚本(nginx_log.sh)

```
#!/bin/bash
nginx_log_path=/var/log/nginx
nginx_log_back_path=/var/log/nginx/back_log
```



```
mkdir -p $nginx_log_back_path/${date +%Y}/${date +%m}
mv $nginx_log_path/access.log $nginx_log_back_path/${date +%Y}/${date +%m}/access.log.${date +%Y%m%d}
mv $nginx_log_path/error.log $nginx_log_back_path/${date +%Y}/${date +%m}/error.log.${date +%Y%m%d}
kill -USER1 `cat /var/log/nginx/nginx.pid`
系统调优
kernel_set.sh
#!/bin/bash
echo -e "*               soft    nofile          20480\n*               hard    nofile          65535" >>/etc/security/limits
.conf
source /etc/profile
cat >>/etc/sysctl.conf<< EOF
fs.file-max = 65536
kernel.core_uses_pid = 1
kernel.msgmax = 65536
kernel.msgmnb = 65536
kernel.shmall = 268435456
kernel.shmmax = 4294967295
net.core.netdev_max_backlog = 262144
net.core.rmem_default = 1746400
net.core.rmem_max = 3492800
net.core.somaxconn = 262144
net.core.wmem_default = 1746400
net.core.wmem_max = 3492800
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1
net.ipv4.ip_forward = 1
net.ipv4.ip_local_port_range = 1024    65000
net.ipv4.netfilter.ip_conntrack_max = 655350
net.ipv4.netfilter.ip_conntrack_tcp_timeout_established = 1200
net.ipv4.tcp_fin_timeout = 1
net.ipv4.tcp_keepalive_time = 30
net.ipv4.tcp_max_orphans = 262144
net.ipv4.tcp_max_syn_backlog = 819200
net.ipv4.tcp_max_tw_buckets = 500000
net.ipv4.tcp_mem = 78643200 104857600 157286400
net.ipv4.tcp_rmem = 873200 1746400 3492800
net.ipv4.tcp_synack_retries = 2
net.ipv4.tcp_syncookies = 1
net.ipv4.tcp_tw_recycle = 1
net.ipv4.tcp_tw_reuse = 1
```

It 你好 linux 学习文档之 Centos6.2_X86_64_LNMP 安装全程实录 QQ 972322817
itnihao@qq.com

```
net.ipv4.tcp_wmem = 873200 1746400 3492800
EOF
sysctl -p
```

以上由 itnihao 发布，转载请注明出处，谢谢合作

----- itnihao 2012 年 5 月 4 号于成都