

web 平台搭建-LNMP (CentOS-7)

一、准备工作

1.环境要求:

操作系统: CentOS 7.X 64 位

关闭 SELinux 和 iptables 防火墙

此次试验环境使用网络 yum 源, 保证系统能正常连接互联网

2.网络 yum 源:

先将系统自带的 yum 配置文件移除或者删除, 然后下载以下两个配置文件

官方基础: <http://mirrors.163.com/.help/CentOS7-Base-163.repo>

<http://mirrors.aliyun.com/repo/Centos-7.repo>

epel 拓展: <http://mirrors.aliyun.com/repo/epel-7.repo>

下载完成后, 需要使用命令清除掉原有的 yum 缓存, 使用新的配置文件建立缓存

```
# yum clean all    #清除掉原有缓存列表
# yum makecache    #建立新的缓存列表
# yum update       #将所有能更新的软件更新 (非必选)
```

3.安装编译工具和依赖软件包:

```
# yum -y install gcc gcc-c++ pcre-devel openssl openssl-devel zlib-devel ncurses-devel cmake bison
libxml2-devel libpng-devel
```

4. Nginx、MySQL、PHP 三大软件的源码包下载地址:

Nginx: <http://nginx.org/en/download.html>

MySQL: <https://dev.mysql.com/downloads/mysql/>

PHP: <http://www.php.net/>

版本选用:

Nginx: 1.12.* #选用软件的稳定版即可

Mysql: 5.5.* #5.5 以上版本需要 1G 以上的内存, 否则无法安装

PHP: 7.1.* #我们使用的是 php7

注意: 每次安装 LNMP 时, 软件包的小版本都不一样, 官方会对其大版本下的小版本进行覆盖式更新, 本文内部分链接会失效, 切记按照下载版本进行安装。

二、源码软件包安装

1. Nginx

Nginx 是一款轻量级的 Web 服务器/反向代理服务器及电子邮件(IMAP/POP3)代理服务器, 在 BSD-like 协议下发行。其特点是占有内存少, 并发能力强。

1.1 下载 Nginx 源码包

```
# wget http://nginx.org/download/nginx-1.12.2.tar.gz
```

1.2 创建用于运行 Nginx 的用户

```
# useradd -r -s /sbin/nologin nginx
```

1.3 解压缩 Nginx 并安装

```
# ./configure --prefix=/usr/local/nginx --user=nginx --group=nginx --with-http_stub_status_module  
--with-http_ssl_module  
# make  
# make install
```

1.4 上传编写好的 nginx 启动管理脚本（见文本尾部）

2. MySQL

下载: <https://dev.mysql.com/downloads/mysql/>

选择: MySQL Community Server 5.5 »

选择: Select Version: 按照自己要求选择

Select Operating System: Source Code

Select OS Version: Generic Linux

格式: mysql-N.N.NN.tar.gz

```
# wget https://cdn.mysql.com//Downloads/MySQL-5.5/mysql-5.5.62.tar.gz
```

2.1 创建用于运行 Mysql 的用户:

```
# useradd -r -s /sbin/nologin mysql
```

2.2 解压缩 Mysql 并安装:

```
# cmake -DCMAKE_INSTALL_PREFIX=/usr/local/mysql -DMYSQL_UNIX_ADDR=/tmp/mysql.sock  
-DEXTRA_CHARSETS=all -DDEFAULT_CHARSET=utf8 -DDEFAULT_COLLATION=utf8_general_ci  
-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 -DMYSQL_TCP_PORT=3306  
# make  
# make install  
# ln -s /usr/local/mysql/bin/* /usr/local/bin
```

2.3 修改安装后的目录权限

```
# cd /usr/local/mysql  
# chown -R root .  
# chown -R mysql data
```

2.4 生成 Mysql 配置文件

```
# cp -a /lnmp/mysql-5.5.62/support-files/my-medium.cnf /etc/my.cnf
```

2.5 初始化, 生成授权表

```
# cd /usr/local/mysql #一定要先切换到此目录下, 然后再执行下一步。  
# ./scripts/mysql_install_db --user=mysql
```

初始化成功标志：两个 ok

2.6 生成 Mysql 的启动和自启动管理脚本

```
# cd /lnmp/mysql-5.5.62/support-files
切换到 mysql 的源码解压缩目录下的 support-files
# cp -a mysql.server /etc/init.d/mysqld
# chmod +x /etc/init.d/mysqld

# chkconfig --add mysqld
# chkconfig mysqld on
# service mysqld start|stop|restart
```

2.7 给 mysql 的 root 用户设置密码

```
# mysqladmin -uroot password 123456
```

3. PHP

下载：<http://www.php.net/>

```
# wget https://www.php.net/distributions/php-7.1.29.tar.gz
```

3.1 解压缩 PHP 并安装：

```
# ./configure --prefix=/usr/local/php/ --with-config-file-path=/usr/local/php/etc/
--with-mysql=/usr/local/mysql/bin/mysql_config --enable-soap --enable-mbstring=all --enable-sockets
--with-pdo-mysql=/usr/local/mysql --with-gd --without-pear --enable-fpm
# make
# make install
```

报错提示：若遇到 libpng.so not found .报错（老版本的 PHP 会出现此问题）

解决方案：

```
# ln -s /usr/lib64/libpng.so /usr/lib
```

3.2 生成 php 配置文件

```
# cp -a /lnmp/php-7.1.29/php.ini-production /usr/local/php/etc/php.ini
复制源码包内的配置文件到安装目录下，并改名即可
```

3.3 创建软连接，使用 php 相关命令是更方便

```
ln -s /usr/local/php/bin/* /usr/local/bin/
ln -s /usr/local/php/sbin/* /usr/local/sbin/
```

4. 配置 Nginx 连接 PHP（重难点）

4.1 nginx 连接 php 需要启动 php-fpm 服务

```
# cd /usr/local/php/etc/
# cp -a php-fpm.conf.default php-fpm.conf
```

生成 php-fpm 的配置文件，并修改指定参数

```
# vim php-fpm.conf
```

修改指定条目的参数：

```
pid = run/php-fpm.pid
```

```
# cd /usr/local/php/etc/php-fpm.d/
```

```
# cp -a www.conf.default www.conf
```

```
# vim www.conf
```

修改用户和组的指定用户

```
user = nginx
```

```
group = nginx
```

修改 Nginx 启动管理脚本：将 php-fpm 的注释取消掉即可

4.2 修改 Nginx 的配置文件，使其识别.php 后缀的文件

```
# vim /usr/local/nginx/conf/nginx.conf
```

取消下列行的注释，并修改 include 选项的后缀为 fastcgi.conf，并注意每一行结尾的分号和大括号

```
#location ~ \.php$ {
```

```
#    root            html;
```

```
#    fastcgi_pass     127.0.0.1:9000;
```

```
#    fastcgi_index    index.php;
```

```
#    fastcgi_param    SCRIPT_FILENAME    /scripts$fastcgi_script_name;
```

```
#    include          fastcgi_params;    #修改为 fastcgi.conf
```

```
#}
```

测试：

重启 Nginx 服务，创建 php 测试文件，访问并查看是否解析

4.3 修改 Nginx 配置文件，使其默认自动加载 php 文件

```
location / {
```

```
root html;    #Nginx 的默认网页路径:PREFIX/html
```

```
index    index.php index.html;    #设置默认加载的页面，以及优先级
```

```
}
```

附件： 建议使用时先复制到文本文件中查看下是否有字符集问题

```
#####Nginx 启动管理脚本#####
```

```
#!/bin/bash
```

```
#Author: liu
```

```
#chkconfig: 2345 99 33
```

```
#description: nginx server control tools
```

```
ngxc="/usr/local/nginx/sbin/nginx"
```

```
pidf="/usr/local/nginx/logs/nginx.pid"
```

```
ngxc_fpm="/usr/local/php/sbin/php-fpm"
```

```
pidf_fpm="/usr/local/php/var/run/php-fpm.pid"
```

```
case "$1" in
```

```

start)
    $ngxc -t &> /dev/null
    if [ $? -eq 0 ];then
        $ngxc
        $ngxc_fpm
        echo "nginx service start success!"
    else
        $ngxc -t
    fi
    ;;
stop)
    kill -s QUIT $(cat $pidf)
    kill -s QUIT $(cat $pidf_fpm)
    echo "nginx service stop success!"
    ;;
restart)
    $0 stop
    $0 start
    ;;
reload)
    $ngxc -t &> /dev/null
    if [ $? -eq 0 ];then
        kill -s HUP $(cat $pidf)
        kill -s HUP $(cat $pidf_fpm)
        echo "reload nginx config success!"
    else
        $ngxc -t
    fi
    ;;
*)
    echo "please input stop|start|restart|reload."
    exit 1
esac

```