

搭建LNMP架构实战（老男孩Linux77期）

1.什么是LNMP?

2010前 LAMP Linux、Apache、MySQL、PHP（1999-2017，兄弟连 2-3 个亿）

2010年后 LNMP, Linux Nginx MySQL(MariaDB) PHP/Python

LEMP, Linux Nginx(engine x) MySQL(MariaDB) PHP/Python

JAVA LNTM, Linux, Nginx, tomcat, MySQL

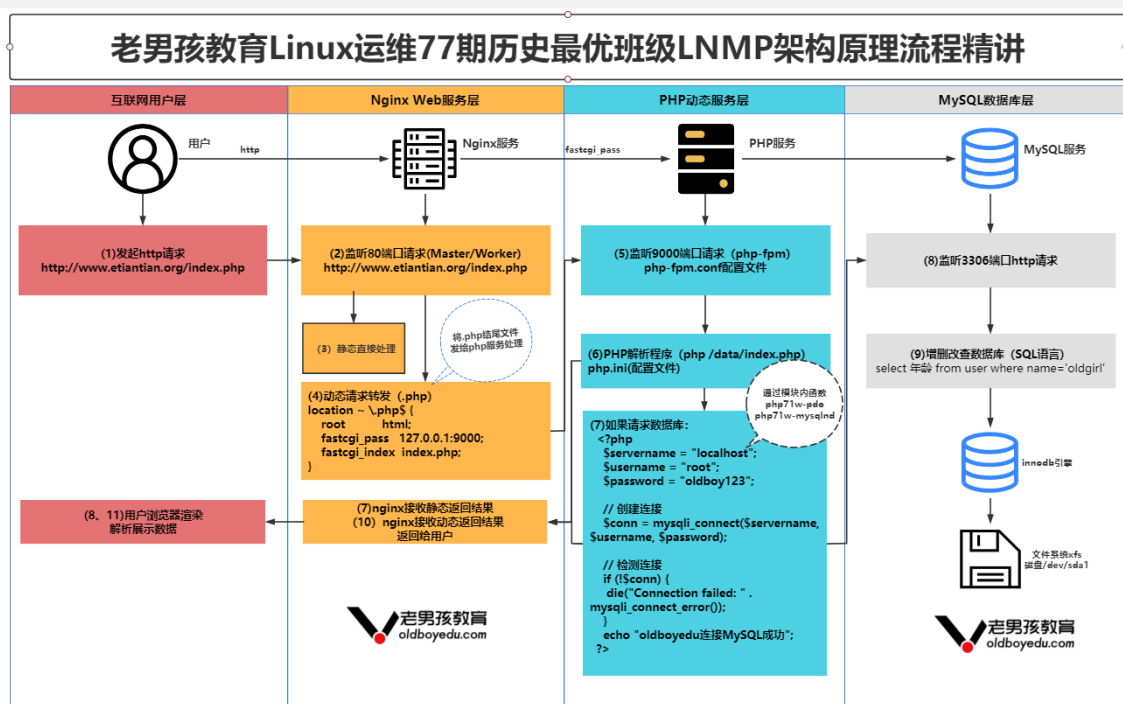
大约在2010年以前，互联网公司最常用的经典Web服务环境组合就是LAMP（即Linux、Apache、MySQL、PHP），近几年随着NginxWeb服务的逐渐流行，又出现了新的Web服务环境组合——LNMP或LEMP，其中LNMP全称为Linux、Nginx、MySQL、PHP等首字母的缩写，而LEMP中的E也表示Nginx，它是取自Nginx名字的发音（engine x）。

2.LNMP架构工作原理流程解析

用户请求 `http://etiantian.org/index.php`，对于Nginx服务而言，是无法处理 `index.php` 这样的动态脚本的，那么Nginx该如何配置，使得支持处理这样的动态请求呢？

当用户发起 Http 请求，请求首先被 Nginx 接收，Nginx 通过 Location 规则进行匹配，将用户请求的动态内容，通过 fastcgi 协议传到给后端的 php

应用服务处理。具体如下图所示。



3.LNMP架构环境安装部署

1) 使用官方仓库安装 Nginx

`#yum install nginx -y` 可能优先使用epel里的源，怎么确保我们配置的官方优先。

```
[root@web01 ~]# rpm -qa yum-plugin-priorities
yum-plugin-priorities-1.1.31-54.el7_8.noarch
```

```
[root@web01 ~]# cat /etc/yum.repos.d/nginx.repo
[nginx-stable]
name=nginx stable repo
baseurl=http://nginx.org/packages/centos/$releasever/$base
arch/
gpgcheck=0
priority=1 #增加优先参数，使得官方源优先epel源。
```

#安装Nginx

```
[root@web01 ~]# yum install nginx -y
```

2) 配置Nginx进程运行的用户

```
[root@web01 ~]# useradd -u1111 www -s /sbin/nologin -M
[root@web01 ~]# sed -i '/^user/c user www;'
/etc/nginx/nginx.conf
[root@web01 ~]# grep "^user" /etc/nginx/nginx.conf
```

3) 启动Nginx, 并将Nginx加入开机自启

```
[root@web01 ~]# systemctl start nginx
[root@web01 ~]# systemctl enable nginx

[root@web01 ~]# lsof -i :80 #查看端口对应的服务
[root@web01 ~]# nginx -v      #查看版本
nginx version: nginx/1.20.1
[root@web01 ~]# curl 10.0.0.7 #welcome to Nginx 表示正常。
```

4) 使用第三方扩展源安装php7.1

```
# (1) 配置PHP安装源
wget https://mirror.webtatic.com/yum/el7/webtatic-
release.rpm --no-check-certificate
rpm -Uvh webtatic-release.rpm #安装PHP第三方
源/etc/yum.repos.d/epel.repo

[root@web01 tools]# rpm -ql epel-release
/etc/yum.repos.d/epel-testing.repo
/etc/yum.repos.d/epel.repo

root@web01 tools]# rpm -ql webtatic-release-7-3
/etc/yum.repos.d/webtatic.repo

#PHP下载地址:
https://us-east.repo.webtatic.com/yum/el7/x86_64/
https://uk.repo.webtatic.com/yum/el7/x86_64/

#如果上述安装源有问题可以使用下面的配置看看
[root@web01 ~]# cat /etc/yum.repos.d/php.repo
[webtatic-php]
name = php Repository
baseurl = http://us-east.repo.webtatic.com/yum/el7/x86_64/
gpgcheck = 0

# (2) 安装PHP及相关插件
```

#2.1) 卸载旧版本的php

```
[root@web01 tools]# rpm -e $(rpm -qa|grep php)
```

```
[root@web01 tools]# rpm -qa|grep php
```

#2.2) 安装PHP及相关插件

```
yum install php71w php71w-cli php71w-common php71w-devel  
php71w-embedded php71w-gd -y
```

```
yum install php71w-mcrypt php71w-mbstring php71w-pdo  
php71w-xml php71w-fpm php71w-mysqlnd -y
```

```
yum install php71w-opcache php71w-pear memcached php71w-  
pear-redis php71w-pear-mongodb -y
```

#2.3) 更多php的插件

```
[root@web01 server]# yum list|grep php71
```

mod_php71w.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-cli.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-common.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-devel.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-embedded.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-fpm.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-gd.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-mbstring.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-mcrypt.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-mysqlnd.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-opcache.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-pdo.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-pear.noarch	1:1.10.4-1.w7
@webtatic-php	
php71w-pear-igbinary.x86_64	2.0.5-1.w7
@webtatic-php	

php71w-pec1-memcached.x86_64	3.0.4-1.w7
@webtatic-php	
php71w-pec1-mongodb.x86_64	1.5.3-1.w7
@webtatic-php	
php71w-pec1-redis.x86_64	3.1.6-1.w7
@webtatic-php	
php71w-process.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-xml.x86_64	7.1.33-1.w7
@webtatic-php	
php71w-bcmath.x86_64	7.1.33-1.w7
webtatic-php	
php71w-dba.x86_64	7.1.33-1.w7
webtatic-php	
php71w-enchanted.x86_64	7.1.33-1.w7
webtatic-php	
php71w-imap.x86_64	7.1.33-1.w7
webtatic-php	
php71w-interbase.x86_64	7.1.33-1.w7
webtatic-php	
php71w-intl.x86_64	7.1.33-1.w7
webtatic-php	
php71w-ldap.x86_64	7.1.33-1.w7
webtatic-php	
php71w-mysql.x86_64	7.1.33-1.w7
webtatic-php	
php71w-odbc.x86_64	7.1.33-1.w7
webtatic-php	
php71w-pdo_dblib.x86_64	7.1.33-1.w7
webtatic-php	
php71w-pec1-apcu.x86_64	5.1.9-1.w7
webtatic-php	
php71w-pec1-apcu-devel.x86_64	5.1.9-1.w7
webtatic-php	
php71w-pec1-geoip.x86_64	1.1.1-1.w7
webtatic-php	
php71w-pec1-igbinary-devel.x86_64	2.0.5-1.w7
webtatic-php	
php71w-pec1-imagick.x86_64	3.4.3-1.w7
webtatic-php	

php71w-pec1-imagick-devel.x86_64	3.4.3-1.w7
webtatic-php	
php71w-pec1-xdebug.x86_64	2.6.1-1.w7
webtatic-php	
php71w-pgsql.x86_64	7.1.33-1.w7
webtatic-php	
php71w-phpdbg.x86_64	7.1.33-1.w7
webtatic-php	
php71w-pspell.x86_64	7.1.33-1.w7
webtatic-php	
php71w-recode.x86_64	7.1.33-1.w7
webtatic-php	
php71w-snmp.x86_64	7.1.33-1.w7
webtatic-php	
php71w-soap.x86_64	7.1.33-1.w7
webtatic-php	
php71w-tidy.x86_64	7.1.33-1.w7
webtatic-php	
php71w-xmlrpc.x86_64	7.1.33-1.w7
webtatic-php	

###企业场景中由开发人员给你编译的参数（他开发的软件）。可以比他多，不能少。

#2.4) 安装php7版本的软件【本地】

[root@web01 ~]# unzip php.zip #提前去官方下载好，然后执行下面的安装

[root@web01 ~]# yum localinstall php/*.rpm -y

5) 配置php-fpm用户与Nginx的运行用户保持一致

```
[root@web01 ~]# sed -i '/^user/c user = www' /etc/php-fpm.d/www.conf
[root@web01 ~]# sed -i '/^group/c group = www' /etc/php-fpm.d/www.conf
[root@web01 blog]# egrep "^user|^group" /etc/php-fpm.d/www.conf
user = www
group = www
```

6) 启动php-fpm，并将其加入开机自启

```

[root@web01 ~]# systemctl start php-fpm
[root@web01 ~]# systemctl enable php-fpm
[root@web01 tools]# lsof -i :9000
COMMAND  PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
php-fpm 1822 root    7u    IPv4  27698      0t0  TCP
localhost:cslistener (LISTEN)
php-fpm 1823 www     9u    IPv4  27698      0t0  TCP
localhost:cslistener (LISTEN)
[root@web01 tools]# ps -ef|grep php-fpm
root      1822      1  0 16:35 ?        00:00:00 php-fpm:
master process (/etc/php-fpm.conf)
www       1823    1822  0 16:35 ?        00:00:00 php-fpm:
pool www
www       1824    1822  0 16:35 ?        00:00:00 php-fpm:
pool www
www       1825    1822  0 16:35 ?        00:00:00 php-fpm:
pool www
www       1826    1822  0 16:35 ?        00:00:00 php-fpm:
pool www
www       1827    1822  0 16:35 ?        00:00:00 php-fpm:
pool www

```

7) 安装Mariadb数据库

```

[root@web01 ~]# yum install mariadb-server mariadb -y

```

8) 启动Mariadb数据库, 并加入开机自动

```

[root@web01 ~]# systemctl start mariadb
[root@web01 ~]# systemctl enable mariadb
[root@web01 blog]# mysql
welcome to the MariaDB monitor.  Commands end with ; or
\.
Type 'help;' or '\h' for help. Type '\c' to clear the
current input statement.
MariaDB [(none)]> quit
Bye

```

9) 给Mariadb配置登陆密码,并是新密码进行登录数据库

```
[root@web01 ~]# mysqladmin password 'oldboy123'  
[root@web01 ~]# mysql -uroot -poldboy123
```

4.LNMP架构配置及整体测试实战

在将Nginx与PHP集成过程中, 需要先了解 Fastcgi代理配置语法

1.设置fastcgi服务器的地址, 该地址可以指定为域名或IP地址以及端口

```
Syntax: fastcgi_pass address;  
Default: -  
Context: location, if in location
```

#语法示例

```
fastcgi_pass localhost:9000;
```

2.设置fastcgi默认的首页文件, 需要结合fastcgi_param一起设置

```
Syntax: fastcgi_index name;  
Default: -  
Context: http, server, location
```

3.通过fastcgi_param设置变量, 并将设置的变量传递到后端的fastcgi服务器

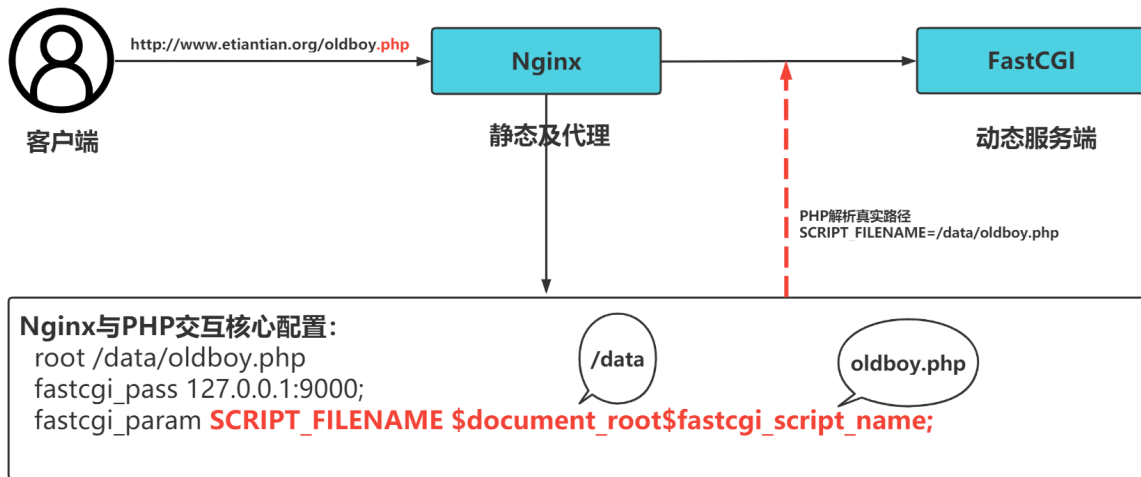
```
Syntax: fastcgi_param parameter value [if_not_empty];  
Default: -  
Context: http, server, location
```

#语法示例

```
fastcgi_index index.php;  
fastcgi_param SCRIPT_FILENAME  
$document_root$fastcgi_script_name;
```

4.通过图形方式展示fastcgi_index与fastcgi_param作用。

Nginx代理PHP的FastCGI配置原理



5.最终Nginx连接Fastcgi服务器配置如下

```
[root@web01 ~]# cat
/etc/nginx/conf.d/blog.etiantian.org.conf
server {
    server_name blog.etiantian.org;
    listen 80;
    root /data/blog;
    index index.php index.html;

    location ~ /\.php$ {
        #root /data/blog;
        fastcgi_pass 127.0.0.1:9000;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME
        $document_root$fastcgi_script_name;
        include fastcgi_params;
    }
}
```

6.在/data目录下创建test_php.php文件

```
[root@web01 ~]# mkdir /data/blog -p
[root@web01 ~]# cat /data/blog/test_php.php
<?php
    phpinfo();
?>
```

#测试能否通过浏览器访问，访问成功如下图

PHP Version 7.1.33

7 在/data目录下创建test_mysql.php文件，填入对应的数据库IP、用户名、密码

```
[root@web01 ~]# cat /data/blog/test_mysql.php
<?php
    $servername = "localhost";
    $username = "root";
    $password = "oldboy123";
    // 创建连接
    $conn = mysqli_connect($servername, $username,
$password);
    // 检测连接
    if (!$conn) {
        die("Connection failed: " .
mysqli_connect_error());
    }
    echo "php连接MySQL数据库成功，oldboy77期";
?>
```

8 最后通过浏览器访问http://blog.etiantian.org/test_mysql.php文件

← → ↻ ⚠ 不安全 | blog.etiantian.org/test_mysql.php

oldboyedu php连接MySQL数据库成功