系统环境：CentOS7.3.1611-阿里云公共镜像

Nginx：nginx-1.12.2

PHP：php-7.1.11

准备工作：

#setenforce 0 设置SELINUX模式为permissive

#vim /etc/selinux/config 进入配置文件将SELINUX完全关闭

#yum install epel-release 安装一个epel扩展源

1.# yum install cmake gcc gcc-c++ pcre\* zlib\* libxml2 libxml2-devel libjpeg-devel libpng-devel freetype-devel openssl-devel libcurl-devel libmcrypt-devel

安装必须的依赖包

2.#groupadd www

#useradd -g www www 添加www用户

3. #tar -zxf nginx-1.12.2.tar.gz

#cd nginx-1.12.2 解压nginx安装包并进入目录

4.#./configure --prefix=/usr/local/nginx --group=nginx --user=nginx --with-http\_stub\_status\_module --with-http\_ssl\_module --with-stream

#make && make install 编译并安装nginx。

5. #vim /etc/systemd/system/nginx.service

[Unit]

Description=The nginx HTTP and reverse proxy server

After=syslog.target network.target remote-fs.target nss-lookup.target

[Service]

Type=forking

PIDFile=/usr/local/nginx/logs/nginx.pid

ExecStartPre=/usr/local/nginx/sbin/nginx -t

ExecStart=/usr/local/nginx/sbin/nginx

ExecReload=/bin/kill -s HUP $MAINPID

ExecStop=/bin/kill -s QUIT $MAINPID

PrivateTmp=true

[Install]

WantedBy=multi-user.target

添加nginx系统服务管理

6. #tar -zxf php-7.1.11.tar.gz 解压php安装包并进入目录。

#cd /php-7.1.11

7.# yum install libmcrypt-devel hash-devel libxslt-devel 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 krb5-devel libidn libidn-devel openssl openssl-devel gcc gcc-c++

安装依赖包，里面有一些是和nginx重复的。这里也一起列出仅供知晓，安装时会自动跳过已安装的。

8. #./configure --prefix=/usr/local/php7 --with-curl --with-freetype-dir --with-gd --with-gettext --with-iconv-dir --with-kerberos --with-libdir=lib64 --with-libxml-dir --with-mysqli --with-openssl --with-pcre-regex --with-pdo-mysql --with-pdo-sqlite --with-pear --with-png-dir --with-xmlrpc --with-xsl --with-zlib --enable-fpm --enable-bcmath --enable-libxml --enable-inline-optimization --enable-gd-native-ttf --enable-mbregex --enable-mbstring --enable-opcache --enable-pcntl --enable-shmop --enable-soap --enable-sockets --enable-sysvsem --enable-xml --enable-zip --with-jpeg-dir

编译选项，注意检查是否有报错。

9.make && make install 编译并安装

10. #cp php.ini-production /usr/local/php7/lib/php.ini

#cp /usr/local/php7/etc/php-fpm.conf.default /usr/local/php7/etc/php-fpm.conf

#cp /usr/local/php7/etc/php-fpm.d/www.conf.default /usr/local/php7/etc/php-fpm.d/www.conf

拷贝这几个文件，跟PHP5.x路径略有不同。

11. #vim /etc/systemd/system/php7-fpm.service

添加PHP-FPM的系统管理文件。

[Unit]

Description=The PHP FastCGI Process Manager

After=syslog.target network.target

[Service]

Type=simple

PIDFile=/run/php-fpm.pid

ExecStart=/usr/local/php7/sbin/php-fpm --nodaemonize --fpm-config /usr/local/php7/etc/php-fpm.conf

ExecReload=/bin/kill -USR2 $MAINPID

ExecStop=/bin/kill -SIGINT $MAINPID

[Install]

WantedBy=multi-user.target

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

编辑nginx主配置文件（此处仅列举可正常运行的最基本配置）

修改开头处

user www www;

下面一段的注释去掉,并修改为如下（fast\_param后面的）。

location ~ \.php$ {

root html;

fastcgi\_pass 127.0.0.1:9000;

fastcgi\_index index.php;

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

include fastcgi\_params;

}

13.#vim /usr/local/php/etc/php-fpm.d/www.conf 修改PHP配置文件。

user = www

group = www

listen = 127.0.0.1:9001

listen.owner = www

listen.group = www

listen.mode = 0660

14.#systemctl start nginx

#systemctl start php-fpm

#ps -ef | grep nginx

#ps -ef | grep php-fpm

启动服务并检查。

15.在/usr/local/nginx/html/下新建info.php，写入<?php phpinfo(); ?>检测是否正常运行，并查看安装配置。

附:opcache 配置

#vim lib/php.ini

[opcache]

zend\_extension=opcache.so

opcache.memory\_consumption=128

opcache.interned\_strings\_buffer=8

opcache.max\_accelerated\_files=4000

opcache.revalidate\_freq=60

opcache.fast\_shutdown=1

opcache.enable\_cli=1