之前已经编译过nginx，nginx的编译安装参考之前

本次只编译安装php

环境

[root@master sbin]# uname -mosr

Linux 2.6.32-696.el6.x86\_64 x86\_64 GNU/Linux

目录及准备

下载的源码文件和安装文件都会放在家目录下,所以新建立用户phper

[root@master sbin]# useradd phper

[root@master sbin]# passwd phper

将phper加入sudoers

[root@master sbin]# vim /etc/sudoers

切换到phper用户，进入家目录，并创建两个目录src和app，src用来存放下载的源码，app用来存放解压和安装后的文件

[root@master sbin]# su phper

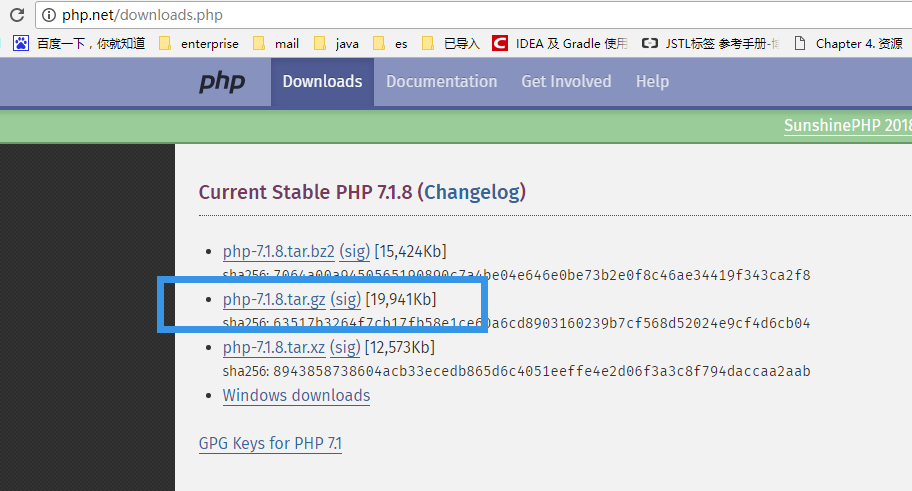
[phper@master sbin]$ cd

[phper@master ~]$ mkdir src app

下载与解压

进入到src目录，下载php源码：

php源码地址：<http://php.net/downloads.php>



下载[php-7.1.8.tar.gz](http://php.net/get/php-7.1.8.tar.gz/from/a/mirror) [(sig)](http://php.net/get/php-7.1.8.tar.gz.asc/from/a/mirror) 版本

点击链接选择和是的源地址，右键将下载地址复制

[phper@master src]$ wget http://php.net/get/php-7.1.8.tar.gz/from/this/mirror

[phper@master src]$ ls

mirror

下载后的压缩包怎么是这个名字？没关系，我们把它好就可以了

[phper@master src]$ mv mirror php-7.1.8.tar.gz

[phper@master src]$ ls

php-7.1.8.tar.gz

名字是不是正确了！

将解压文件放在app目录：

[phper@master src]$ tar -xzvf php-7.1.8.tar.gz -C ../app/

编译和安装

下面进行经典的预编译-编译-安装-配置和启动

首先，我们在app目录下建立bin目录用来作为安装目录

[phper@master app]$ mkdir bin

[phper@master app]$ ls

bin php-7.1.8

进入到php-7.1.8目录

[phper@master app]$ cd php-7.1.8/

我们查看到此目录下存在了configure文件，接下来操作

* 预编译

[phper@master php-7.1.8]$ ./configure --prefix=/home/phper/app/bin

预编译分为了几个部分：

Configuring SAPI modules 为配置SAPI模块

Running system checks 为运行系统检查

General settings 为通用配置

Configuring extensions 为配置扩展

…等等

最后我们看到预编译过程终止了，如果是因为出现类似错误：

checking libxml2 install dir... no

checking for xml2-config path...

configure: error: xml2-config not found. Please check your libxml2 installation.

根据他的错误提示，我们可能没有安装libxml2（至少php找不到），我们使用yum安装libxml2

[phper@master php-7.1.8]$ sudo yum install libxml2

此时yum程序最后可能说我们已经安装了 libxml2，在执行预编译命令还是报错

可能是没有安装libxml2-devel.x86\_64（我机器是64位的）

下面安装此软件：

[phper@master php-7.1.8]$ sudo yum install libxml2-devel.x86\_64

安装成功后再执行预编译命令：

[phper@master php-7.1.8]$ ./configure --prefix=/home/phper/app/bin

如果没有报错，则最后会输出

+--------------------------------------------------------------------+

| License: |

| This software is subject to the PHP License, available in this |

| distribution in the file LICENSE. By continuing this installation |

| process, you are bound by the terms of this license agreement. |

| If you do not agree with the terms of this license, you must abort |

| the installation process at this point. |

+--------------------------------------------------------------------+

Thank you for using PHP.

config.status: creating php7.spec

config.status: creating main/build-defs.h

config.status: creating scripts/phpize

config.status: creating scripts/man1/phpize.1

config.status: creating scripts/php-config

config.status: creating scripts/man1/php-config.1

config.status: creating sapi/cli/php.1

config.status: creating sapi/cgi/php-cgi.1

config.status: creating ext/phar/phar.1

config.status: creating ext/phar/phar.phar.1

config.status: creating main/php\_config.h

config.status: executing default commands

如果编译过程中出现错误，则按照它的错误提示来安装相应的软件

* 编译：

[phper@master php-7.1.8]$ make

…

Generating phar.php

Generating phar.phar

PEAR package PHP\_Archive not installed: generated phar will require PHP's phar extension be enabled.

clicommand.inc

directorytreeiterator.inc

pharcommand.inc

invertedregexiterator.inc

directorygraphiterator.inc

phar.inc

Build complete.

Don't forget to run 'make test'.

如果最后出现上面部分则说明编译成功

它提示说PEAR package PHP\_Archive 没有安装

但是这个并不是错误，我们貌似可以直接安装，而且编译后它提示 Don't forget to run 'make test'.，我们可以运行编译测试命令测试一下

或者我们可以参考其他人的解决方案：<http://www.cnblogs.com/tjxwg/p/3967744.html>

这里我们采用网上的方法：

先clean一下刚才的编译

然后执行预编译的同时指定参数，--without-pear --disable-phar

最后输出：

+--------------------------------------------------------------------+

| License: |

| This software is subject to the PHP License, available in this |

| distribution in the file LICENSE. By continuing this installation |

| process, you are bound by the terms of this license agreement. |

| If you do not agree with the terms of this license, you must abort |

| the installation process at this point. |

+--------------------------------------------------------------------+

Thank you for using PHP.

config.status: creating php7.spec

config.status: creating main/build-defs.h

config.status: creating scripts/phpize

config.status: creating scripts/man1/phpize.1

config.status: creating scripts/php-config

config.status: creating scripts/man1/php-config.1

config.status: creating sapi/cli/php.1

config.status: creating sapi/cgi/php-cgi.1

config.status: creating main/php\_config.h

config.status: executing default commands

编译

[phper@master php-7.1.8]$ make

…

Build complete.

Don't forget to run 'make test'.

此时没有问题提示了，可以直接安装

* 安装

[phper@master php-7.1.8]$ make install

Installing shared extensions: /home/phper/app/bin/lib/php/extensions/no-debug-non-zts-20160303/

Installing PHP CLI binary: /home/phper/app/bin/bin/

Installing PHP CLI man page: /home/phper/app/bin/php/man/man1/

Installing phpdbg binary: /home/phper/app/bin/bin/

Installing phpdbg man page: /home/phper/app/bin/php/man/man1/

Installing PHP CGI binary: /home/phper/app/bin/bin/

Installing PHP CGI man page: /home/phper/app/bin/php/man/man1/

Installing build environment: /home/phper/app/bin/lib/php/build/

Installing header files: /home/phper/app/bin/include/php/

Installing helper programs: /home/phper/app/bin/bin/

program: phpize

program: php-config

Installing man pages: /home/phper/app/bin/php/man/man1/

page: phpize.1

page: php-config.1

Installing PDO headers: /home/phper/app/bin/include/php/ext/pdo/

上面是安装的说明，包括一些文件的地址

* 配置

安装好后，我们进入到预编译指定的安装目录

[phper@master app]$ cd ../bin/

[phper@master bin]$ ls

bin include lib php var

[phper@master bin]$ ll

total 20

drwxrwxr-x 2 phper phper 4096 Aug 17 02:30 bin #二进制执行脚本文件目录

drwxrwxr-x 3 phper phper 4096 Aug 17 02:30 include #

drwxrwxr-x 3 phper phper 4096 Aug 17 02:30 lib

drwxrwxr-x 3 phper phper 4096 Aug 17 02:30 php #配置文件在这个目录下

drwxrwxr-x 4 phper phper 4096 Aug 17 02:30 var

启动/测试

为了方便测试，我们在bin目录下创建conf目录，用来存放配置文件，并将源码包解压目录下的配置文件复制到conf目录一份

[phper@master bin]$ cp ../php-7.1.8/php.ini-development ../php-7.1.8/php.ini-production ./conf

因为存在了配置文件的备份，所以我们直接将php.ini-development名字改为php.ini

[phper@master bin]$ mv ./conf/php.ini-development php.ini

修改nginx的配置文件，开启php的代理，然后启动nginx

在nginx的html目录建立php文件，phpinfo.php 输出php的配置

建立文件phpinfo.php

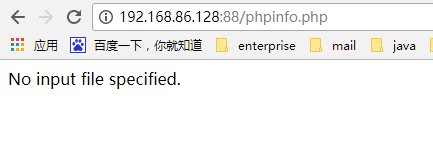
<?php

phpinfo();

启动php-cgi,绑定端口9000，后台运行

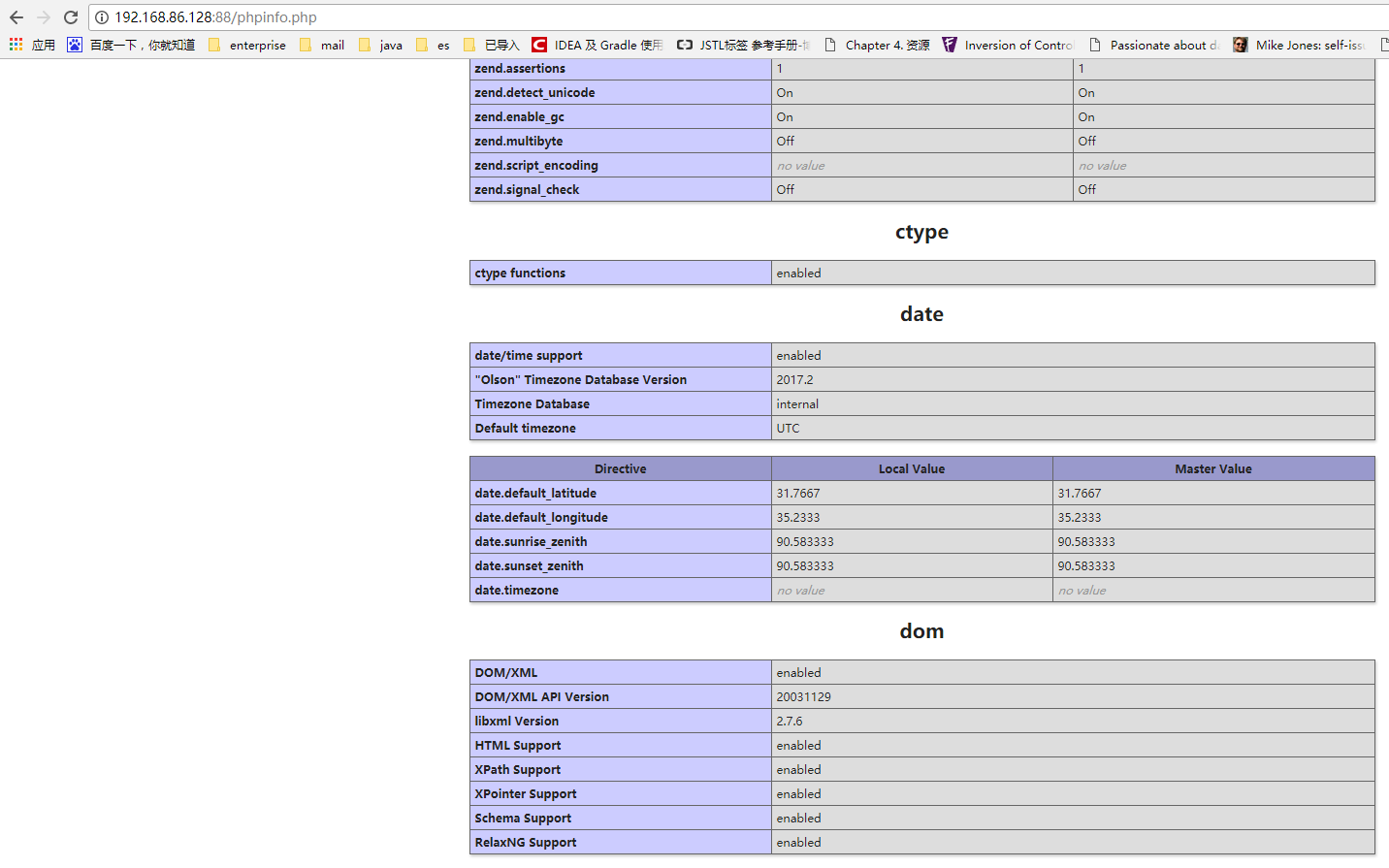
[phper@master bin]$ sudo ./bin/php-cgi -b 127.0.0.1:9000

注：这里之所以加sudo是因为php-cgi进程是由phper用户启动的，它如果对指定根目录没有访问权限的话，访问php文件会输出没有找到文件：



这是如果以sudo启动的话，那么php-cgi进程就可以访问到根目录

此时，浏览器访问192.168.86.128.88/phpinfo.php



配置成功