在ubuntu下搭建python采集环境

作者：有故事的人 来源：东坡网

本文原地址：http://www.dp1037.com/dpinfo-7-51-0.html

东坡网，为帝国cms加油

在阿里云esc部署一个python采集程序，需要的相关程序有：python及其相关库beautifulsoup、tornado、torndb等，与之配套的nginx、php、mysql，以及辅助工具anyproxy和supervisor等。

系统环境： Ubuntu 14.04.5

一、更新源

```
apt-get update
```

二、安装mysql

```
apt-get install mysql-server
```

安装过程中设置root用户的密码，安装完成之后登录mysql测试

```
mysql -uroot -p密码
```

三、安装nginx

```
apt-get install nginx
```

安装完之后可以访问esc的IP测试是否安装成功

四、安装php

（1）下载php源文件

在php.net网站选择一个版本的php，选择下载地址，然后下载编译，这里选择了5.5.38

```
wget http://hk1.php.net/get/php-5.5.38.tar.gz/from/this/mirror
tar zxf mirror
cd php-5.5.38
```

（2）配置PHP

运行./configure -help 命令可以获得完整的可用选项清单

```
./configure -help
```

根据项目需求，安装以下扩展： --enable-fpm --with-openssl --with-curl --enable-mbstring --with-mysql --with-mysqli --with-gd

```
./configure --enable-fpm --with-openssl --with-curl --enable-mbstring --with-mysql --with-mysqli  --with-gd
```

运行上述命令后开始检查编译，但开始报错，需要一一解决

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

```
apt-get install libxml2-dev
```

error: Cannot find OpenSSL's <evp.h>

```
apt-get install libcurl4-openssl-dev pkg-config
apt-get install libssl-dev
```

error: png.h not found.

```
apt-get install libpng-dev
```

更多错误提示相关解决方案见页面blog.csdn.net/white__cat/article/details/28907535

解决完错误之后，会有提示：Thank you for using PHP. 可以开始构建PHP了

```
make && make install
```

（3）创建配置文件，并将其复制到正确的位置

```
cp php.ini-development /usr/local/lib/php.ini
cp /usr/local/etc/php-fpm.conf.default /usr/local/etc/php-fpm.conf
cp sapi/fpm/php-fpm /usr/local/bin
```

php.ini文件位置，可以由编译时指定，也可以是默认，不知道默认位置时，可以 php -i 命令查看 Configuration File (php.ini) Path 项。

（4）修改配置文件php.ini

```
vim /usr/local/lib/php.ini
```

根据自己需求修改此文件，其中为安全起见，设置cgi.fix_pathinfo=0

（5）修改php-fpm配置文件

```
vim /usr/local/etc/php-fpm.conf
```

修改 php-fpm.conf 配置文件，确保 php-fpm 模块使用 www-data 用户和 www-data 用户组的身份运行。

将 php-fpm.conf 文件中相关位置改为一下内容：

```
; Unix user/group of processes
; Note: The user is mandatory. If the group is not set, the default user's group
;       will be used.
user = www-data
group = www-data
```

启动 php-fpm 服务：

```
/usr/local/bin/php-fpm
```

后期修改过php.ini后都需要重启php-fpm服务，最快是采取kill进程的方法

```
ps -aux | grep php-fpm
```

```
kill 2154
```

（6）配置 Nginx 使其支持 PHP 应用

```
vim /etc/nginx/sites-available/default
```

修改默认的 location 块，使其支持 .php 文件，并将默认网站位置改为/home/www目录

```
root /home/www;
location / {
   index  index.php index.html index.htm;
}
```

下一步配置来保证对于 .php 文件的请求将被传送到后端的 PHP-FPM 模块， 取消默认的 PHP 配置块的注释，并修改为下面的内容：

```
```

```
location ~* \.php$ {
    fastcgi_index   index.php;
    fastcgi_pass    127.0.0.1:9000;
    include         fastcgi_params;
    fastcgi_param   SCRIPT_FILENAME    $document_root$fastcgi_script_name;
    fastcgi_param   SCRIPT_NAME        $fastcgi_script_name;
}
```

重启 Nginx

```
service nginx restart
```

（7）创建php文件测试是否能成功访问

```
echo "<?php phpinfo(); ?>" >> /home/www/index.php
```

可以用php命令行工具查看相关信息

```
php -v
php -m
```

（8）安装phpMyAdmin

```
cd /home/www
wget https://files.phpmyadmin.net/phpMyAdmin/4.6.6/phpMyAdmin-4.6.6-all-languages.tar.gz
tar -zxvf phpMyAdmin-4.6.6-all-languages.tar.gz
mv phpMyAdmin-4.6.6-all-languages apm
rm phpMyAdmin-4.6.6-all-languages.tar.gz
```

（8）将php-fpm加入自动启动

```
vi /etc/rc.local
```

```
/usr/local/bin/php-fpm
```

五、安装anyproxy

AnyProxy是一个开放式的HTTP/HTTPS代理，可以获取特殊参数，用于采集。

（1）安装node

AnyProxy需要环境node.js环境且对版本有要求。我们安装新版本的nodejs，新版nodejs自带npm。

官网网站：nodejs.org/en/，安装V6.9.4

```
cd /root
wget https://nodejs.org/dist/v6.9.4/node-v6.9.4-linux-x64.tar.gz
tar -zxvf node-v6.9.4-linux-x64.tar.gz
```

移动文件夹到通用软件安装目录

```
mv node-v6.9.4-linux-x64 /opt/
```

在系统命令中建立node和npm的软连接

```
ln -s /opt/node-v6.9.4-linux-x64/bin/node /usr/bin/node
ln -s /opt/node-v6.9.4-linux-x64/lib/node_modules/npm/bin/npm-cli.js /usr/bin/npm
```

测试node是否安装成功

```
node -v
npm -v
```

（2）安装anyproxy

```
npm install -g anyproxy
```

（3）开启proxy服务

查看anyproxy所在目录

```
whereis anyproxy
```

按照如上的安装方式，目录在 /opt/node-v6.9.4-linux-x64/bin/anyproxy，开始运行

```
/opt/node-v6.9.4-linux-x64/bin/anyproxy --port 8001
```

可以用 地址:8002 访问实时监控页面。测试成功后用Ctrl + C 退出

六、安装python相关库

（1）python 2.7

ubuntu 14自带python2.7不需要再安装

```
python
exit()
```

（2）python-pip

```
apt-get install python-pip
```

（3）beautifulsoup

```
pip install beautifulsoup
```

（4）beautifulsoup4

```
pip install beautifulsoup4
```

（5）tornado

```
pip install tornado
```

（6）torndb

```
pip install torndb
```

（7）python-mysqldb

```
apt-get install python-mysqldb
```

七、安装supervisor

Supervisor允许用户在类UNIX操作系统上监视和控制多个进程。其官网：supervisord.org

（1）安装

```
apt-get install supervisor
```

（2）生成配置文件

```
echo_supervisord_conf > /home/dpw/etc/supervisord.conf
```

（3）修改配置文件

为方便管理，将项目用到的所有文件放到文件夹/home/dpw/下，相关日志也在此目录下。supervisord.conf文件中logfile参数改到/home/dpw/log目录。其它应用文件的配置放在/home/dpw/etc/conf目录下，在supervisord.conf文件中做相关设置

```
logfile=/home/dpw/log/supervisord.log


[include]
files = conf/*.conf
```

（4）启动supervisord程序

```
supervisord -c /home/dpw/etc/supervisord.conf
```

（5）启动进程监控程序

```
supervisorctl -c /home/dpw/etc/supervisord.conf
```

（6）设置anyproxy的进程监控

```
vi /home/dpw/etc/conf/anyproxy.conf
```

anyproxy的进程监控配置内容如下：

```
[program:anyproxy]
command=/opt/node-v6.9.4-linux-x64/bin/anyproxy --port 8001 --host 0.0.0.0
stdout_logfile=/home/dpw/log/anyproxy.log
redirect_stderr=true
```

（7）重启监控进程

```
supervisorctl -c /home/dpw/etc/supervisord.conf
reload
```

七、部署python爬虫cralwer

以下步骤仅用于记录用，没有通用价值，可以忽略。

（1）部署mysql数据表

```
mysql -uroot -p密码 -e 'creat database 数据库名'
mysql -uroot -p密码 数据库名 < /home/dpw/app/schema.sql
```

（2）新增采集操作mysql用户

```
mysql -uroot -p密码
create user 新用户@'localhost' identified by '新用户密码';
grant all on `允许操作的数据库名%`.* to 新用户@'localhost';
flush privileges;
```

（3）将采集程序加入到supervisor监控中，在/home/dpw/etc/conf/目录下新建.conf文件

site.conf

```
[program:site]
command=/usr/bin/python /home/dpw/app/site.py
stdout_logfile=/home/dpw/log/site.log
autostart=true
autorestart=true
redirect_stderr=true
```

watcher.conf

```
[program:watcher]
command=/usr/local/bin/php /home/dpw/app/console watch /home/dpw/log/anyproxy.log
stdout_logfile=/home/dpw/log/watcher.log
redirect_stderr=true
```

至此环境配置完成。