**漏洞扫描部署**

**前期准备**

* 0.检查Nginx是否安装： `

service nginx status

若出现：**nginx: unrecognized service** 。说明Nginx未安装，首先安装Nginx。

若出现： **nginx is running** 或 **nginx is not running** ，说明nginx已安装。

* 1.安装Nginx: `

apt-get install build-essential

apt-get install libtool

apt-get update

apt-get install libpcre3 libpcre3-dev

apt-get install zlib1g-dev

apt-get install openssl

apt-get install nginx

检查安装： `

nginx status

若出现：**nginx is not running** 。说明Nginx已安装成功。

* 2.安装Mysql： `

sudo apt-get install mysql-server

sudo apt-get install mysql-client

sudo apt-get install libmysqlclient-dev

注：数据库root密码为123

* 3.安装Python2 `

sudo apt-get install python

查看安装： `

python --version

若出现：**Python 2.7.12** 。说明python2安装成功（或者其它python2版本均可）。

* 4.安装Python3 `

sudo apt-get install python3

查看安装： `

python3 --version

若出现：**Python 3.5.2** 。说明python3安装成功（或者其它python3版本均可）。

* 5.安装pip `

sudo apt-get install python-pip

查看安装： `

pip --version

若出现：**pip 8.1.1 from /usr/lib/python2.7/dist-packages (python 2.7)** 。说明pip安装成功（或者其它类似版本均可）。

* 6.安装pip3 `

sudo apt-get install python3-pip

查看安装： `

pip3 --version

若出现：**pip 8.1.1 from /usr/lib/python3/dist-packages (python 3.5)** 。说明pip3安装成功（或者其它类似版本均可）。

**正式部署**

* 1.Nginx配置:  
  对于刚刚安装的Nginx，可选择将原有的 **/etc/nginx/nginx.conf** 备份,将本文档所在目录下的 **nginx.conf** 复制到 **/etc/nginx/**目录下。  
  若已存在Nginx，将下端代码复制到 **/etc/nginx/nginx.conf** 中：  
  `

server {

listen 8080;

server\_name scanning;

#charset koi8-r;

#access\_log logs/host.access.log main;

location / {

root /opt/security-system-web/app;

autoindex on;

}

location /api/ {

proxy\_pass http://127.0.0.1:9090;

}

location /static/ {

proxy\_pass http://127.0.0.1:9090;

}

#error\_page 404 /404.html;

# redirect server error pages to the static page /50x.html

#

#error\_page 500 502 503 504 /50x.html;

#location = /50x.html {

# root html;

#}

# proxy the PHP scripts to Apache listening on 127.0.0.1:80

#

#location ~ \.php$ {

# proxy\_pass http://127.0.0.1;

#}

# pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000

#

#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;

#}

# deny access to .htaccess files, if Apache's document root

# concurs with nginx's one

#

#location ~ /\.ht {

# deny all;

#}

` 具体未知可参考本目录下的 **/etc/nginx/nginx.conf**。

* 2.创建数据库： `

mysql -uroot -p123

create database ics\_scan character set utf8;

create database ics character set utf8;

* 3. 安装Django及相关插件： `

cd /opt/security-system-server/security\_system/

pip install --upgrade pip

sudo pip install Django==1.10.5

sudo pip install mysqlclient==1.3.9

sudo pip install django-cors-headers==2.1.0

sudo pip install djangorestframework==3.6.3

* 4. 创建项目目录 首先将创建**/opt**目录，将所有压缩文件上传至该目录下。 `

sudo mkdir /opt/

unzip ics*scanning*project.zip

unzip security-system-server.zip

unzip security-system-web.zip

gunzip ics\_201707081521.sql.gz

gunzip ics*scan*201707071531.sql.gz

* 5. 安装zoomeye插件 `

sudo apt-get install curl

sudo apt-get install libcurl4-openssl-dev

sudo apt-get install libssl-dev

cd /opt/ics*scanning*project/src/spider/pycurl-7.43.0/

python setup.py install --curl-config=/usr/bin/curl-config

cd /opt/ics*scanning*project/src/spider/zoomeye-master/

python setup.py install

查看安装： `

python /opt/ics*scanning*project/src/spider/getZoomeye.py

若程序正常运行，则说明zoomeye插件安装成功。

否则如果系统是Ubuntu 14.04.4 LTS的话，执行： `

ls -l /usr/local/lib/libcurl.so.4

如果显示：**lrwxrwxrwx 1 root root 16 Aug 16 21:15 /usr/local/lib/libcurl.so.4 -> libcurl.so.4.2.0**

执行： `

sudo rm -rf /usr/local/lib/libcurl.so.4

sudo ln -s /usr/lib/x86\_64-linux-gnu/libcurl.so.4.3.0 /usr/local/lib/libcurl.so.4

* 6. 安装shodan插件 `

sudo pip install shodan

cd /usr/local/lib/python2.7/dist-packages/shodan

sudo vim client.py

如本目录下图chang\_shodan.png，修改134、157、159代码。

如本目录下图chang\_shodan2.png，修改304/336代码。

* 7. 修改linux任务计划 `

sudo vim /etc/crontab

在行尾添加如下内容：

\*/1 \* \* \* \* root /opt/icscrontab\_DispatchAll.sh >> /opt/DispatchAll.log

\*/1 \* \* \* \* root /opt/icscrontab\_getShodan.sh >> /opt/getShodan.log

\*/1 \* \* \* \* root /opt/icscrontab\_getZoomeye.sh >> /opt/getZoomeye.log

\*/1 \* \* \* \* root /opt/icscrontab\_getDitecting.sh >> /opt/getDitecting.log

* 8. 导入数据库 进入mysql命令行，导入两个sql文件。 `

use ics\_scan;

source /opt/ics*scan*201707071531.sql

use ics;

source /opt/ics\_201707081521.sql

* 9. 修改代码数据库配置

需要配置爬虫、扫描引擎、后台的数据库密码 `

vim /opt/ics*scanning*project/src/scan/common/settings.py

修改PASSWORD字段为当前数据库密码。如本目录下图dbpw1.png。 `

vim /opt/ics*scanning*project/src/spider/common/settings.py

修改PASSWORD字段为当前数据库密码。如本目录下图dbpw2.png。 `

vim /opt/security-system-server/security*system/security*system/settings.py

如本目录下图dbpw3.png，修改86、94行代码均为当前数据库密码。

* 10. 启动管理端 `

cd opt/security-system-server

sudo chmod 777 restart\_django.sh

sudo ./restart\_django.sh