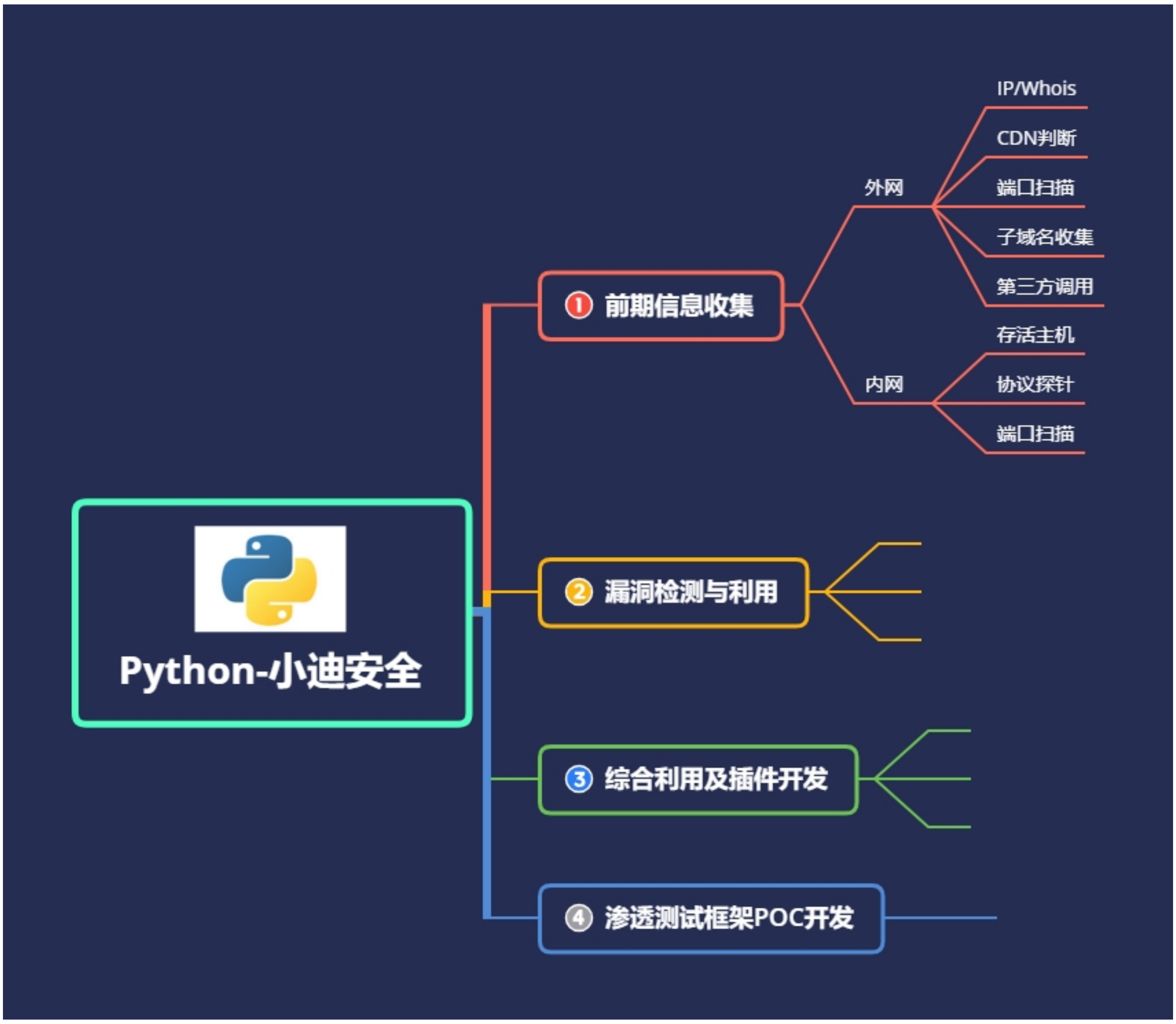
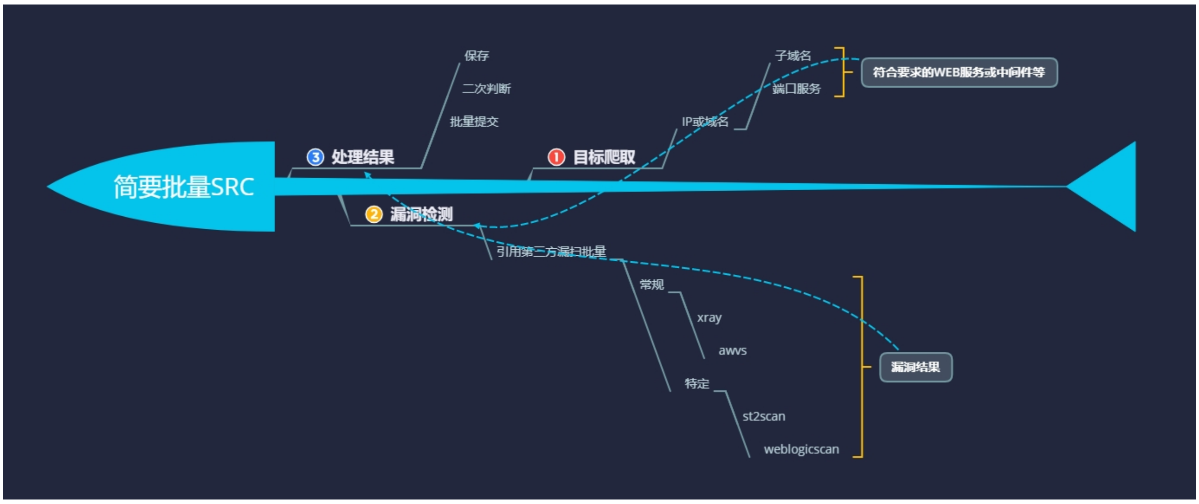
Day76　Python开发-内外网收集 Socket&子域名&DNS





76.1 外网信息收集

76.1.1 域名反查IP功能

```python
#ip查询
def ip_check(url):
    ip=socket.gethostbyname(url)
    print(ip)
```

whois查询：

```python
#whois查询
def whois_check(url):
    data=whois(url)
    print(data)
```

> 我们进行信息收集的时候，会遇到CDN，使用nslookup找到该url下的IP数目信息，进行判断

76.1.2 识别目标是否存在CDN

```python
#CDN判断-利用返回IP条数进行判断
def cdn_check(url):
    ns="nslookup "+url
    #data=os.system(ns)
    #print(data) #结果无法读取操作
    data=os.popen(ns,"r").read()
    if data.count(".")>8:
        print("存在CDN")
    else:
        print("不存在CDN")
```

76.1.3 端口扫描

自写socket协议tcp,udp扫描：

```python
#端口扫描
#1.自写socket协议tcp,udp扫描
#2.调用第三方masscan,nmap等扫描
def port_check(url):
    ip = socket.gethostbyname(url)
    ports={'21','22','135','443','445','80','1433','3306',"3389",'1521','8000','7002','7001','8080',"9090",'8089',"4848"}
    server = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
    for port in ports:
        try:
            data=server.connect_ex((ip,int(port)))
            if data==0:
                print(ip+":"+port+"|open")
            else:
                print(ip+":"+port+"|close")
                pass
        except Exception as err:
                pass
```

调用第三方模块masscan,nmap等扫描：

https://www.cxyzjd.com/article/u012206617/90753823

76.1.4 子域名查询

利用字典加载爆破

```python
#子域名查询-
#1.利用字典记载爆破进行查询
#2.利用bing或第三方接口进行查询
def zym_list_check(url):
    url=url.replace("www.","")
    for zym_list in open("dic.txt"):
        zym_list=zym_list.replace("\n","")
        zym_list_url=zym_list+"."+url
        try:
            ip=socket.gethostbyname(zym_list_url)
            print("SUCCESS:"+zym_list_url+"->"+ip)
            time.sleep(0.1)
        except Exception as e:
            time.sleep(0.1)
```

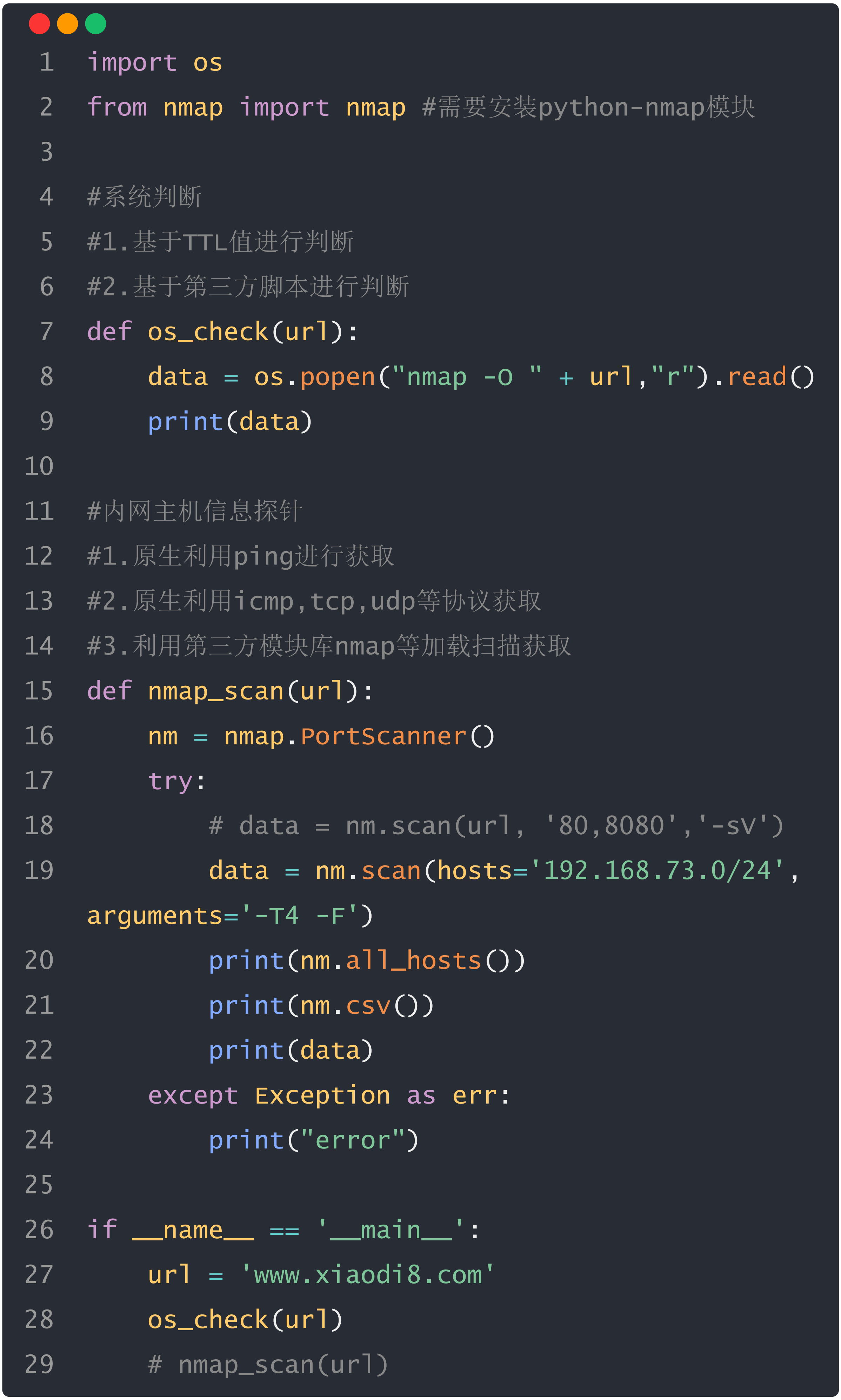
利用bing或第三方接口进行查询

后续具体展开。

76.2 内网信息收集

nmap使用

> 首先python通过`pip install python-nmap`命令去安装nmap模块
> 将本地的nmap配置到环境变量中
> python通过nmap模块去调用本地的nmap



```python
import os
from nmap import nmap #需要安装python-nmap模块

#系统判断
#1.基于TTL值进行判断
#2.基于第三方脚本进行判断
def os_check(url):
    data = os.popen("nmap -O " + url,"r").read()
    print(data)

#内网主机信息探针
#1.原生利用ping进行获取
#2.原生利用icmp,tcp,udp等协议获取
#3.利用第三方模块库nmap等加载扫描获取
def nmap_scan(url):
    nm = nmap.PortScanner()
    try:
        # data = nm.scan(url, '80,8080','-sV')
        data = nm.scan(hosts='192.168.73.0/24', arguments='-T4 -F')
        print(nm.all_hosts())
        print(nm.csv())
        print(data)
    except Exception as err:
        print("error")

if __name__ == '__main__':
    url = 'www.xiaodi8.com'
    os_check(url)
    # nmap_scan(url)
```

76.2.1 主机存活

```
import nmap
nm=nmap.PortScanner()
try:
nm.scan(hosts='192.168.2.0/24',arguments='-T4 -F')
#查看当前存活主机
print(nm.all_hosts())
#查看当前存活主机的详细信息
print(nm.csv())
except Exception as err:
print('errpr')
```

76.2.2 端口扫描

```
import nmap
nm=nmap.PortScanner()
data=nm.scan('ip.......','80,8888','-sV')
print(data)
```

资源：

```
https://www.jb51.net/softs/598504.html
https://www.cnblogs.com/csnd/p/11807823.html
https://pan.baidu.com/s/13y3U6jX3WUYmnfKnXT8abQ 提取码：xiao https://pan.baidu.com/s/1tQS1mUelmEh3I68AL7yXGg 提取码：xiao
```