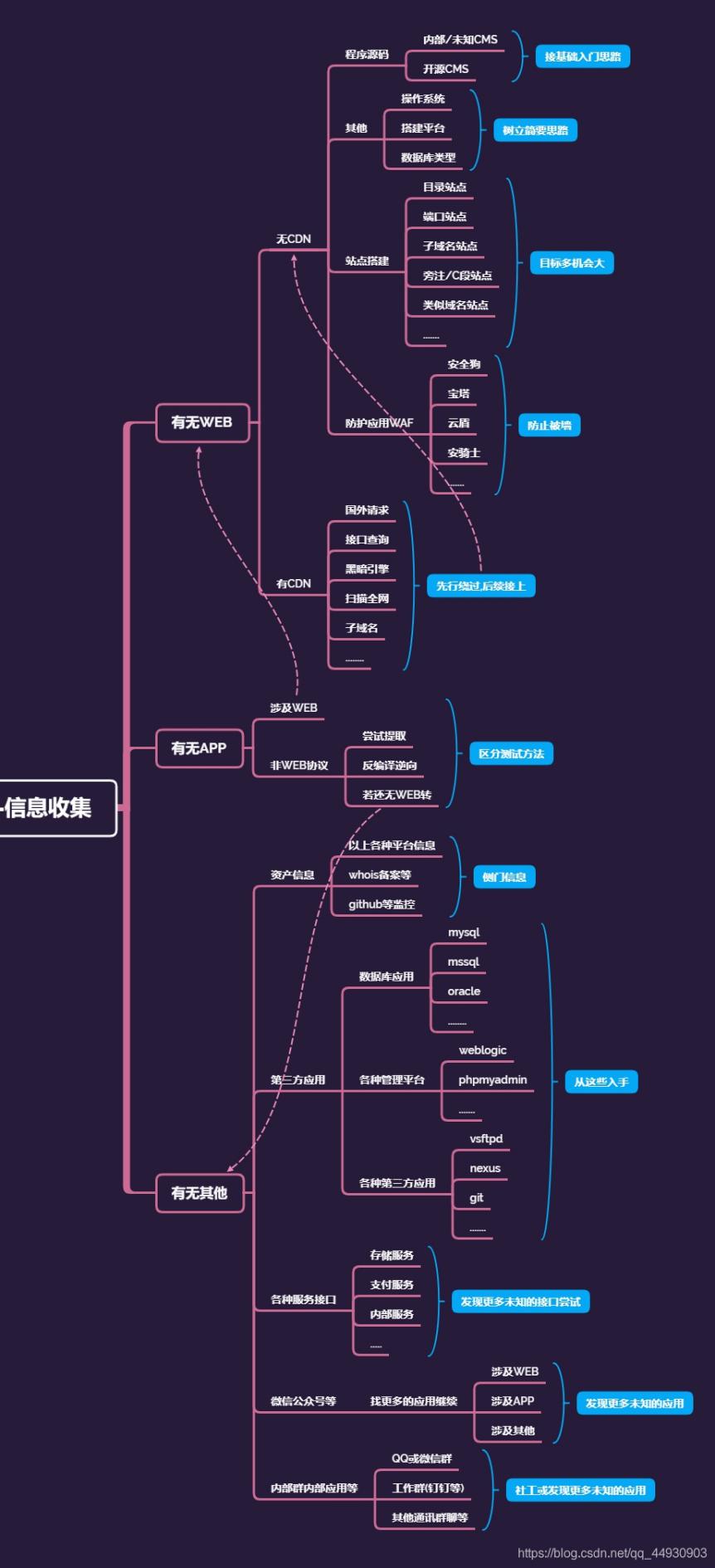
# 10 信息收集-资产监控拓展



#Github监控

便于收集整理最新exp或poc

便于发现相关测试目标的资产

#各种子域名查询

#DNS,备案,证书

#全球节点请求cdn

枚举爆破或解析子域名对应

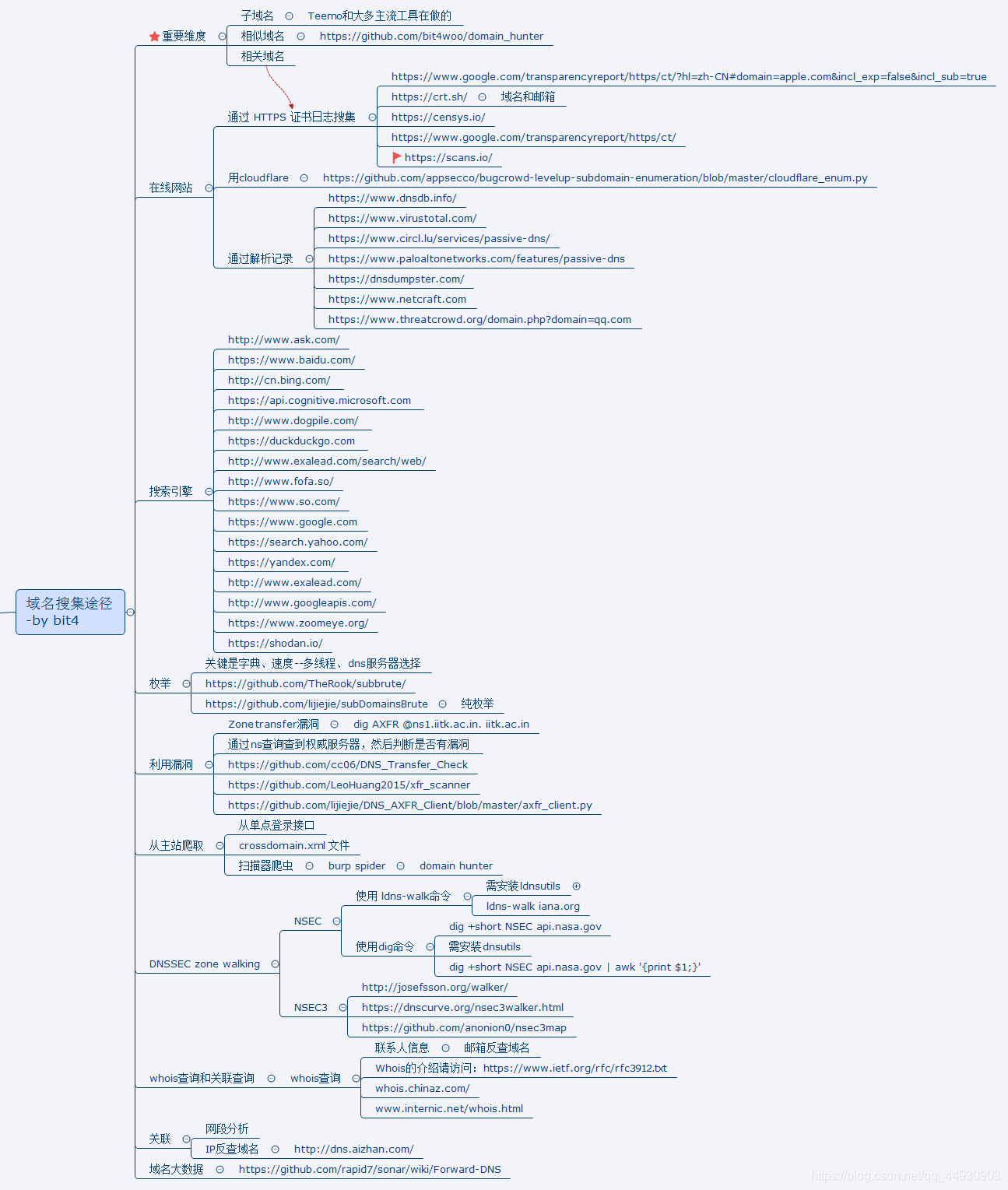
便于发现管理员相关的注册信息

#黑暗引擎相关搜索

fofa，shodan，zoomeye

#微信公众号接口获取

#内部群 内部应用内部接口



演示案例：

✧ 监控最新的EXP发布及其他

# Title: wechat push CVE-2020

# Date:

# Exploit Author:

# Version: 4.0

# Tested on: Linux,windows

# cd /root/sh/git/ && nohup python3 /root/sh/git/git.py &

# coding:UTF-8

import requests

import json

import time

import os

import pandas as pd

time\_sleep = 60 #每隔20秒爬取一次

while(True):

headers1 = {

"User-Agent": "Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.25 Safari/537.36 Core/1.70.3741.400 QQBrowser/10.5.3863.400"}

#判断文件是否存在

datas = []

response1=None

response2=None

if os.path.exists("olddata.csv"):

#如果文件存在则每次爬取10个

df = pd.read\_csv("olddata.csv", header=None)

datas = df.where(df.notnull(),None).values.tolist()#将提取出来的数据中的nan转化为None

requests.packages.urllib3.disable\_warnings()

response1 = requests.get(url="https://api.github.com/search/repositories?q=CVE-2020&sort=updated&per\_page=10",headers=headers1,verify=False)

response2 = requests.get(url="https://api.github.com/search/repositories?q=RCE&ssort=updated&per\_page=10",headers=headers1,verify=False)

else:

#不存在爬取全部

datas = []

requests.packages.urllib3.disable\_warnings()

response1 = requests.get(url="https://api.github.com/search/repositories?q=CVE-2020&sort=updated&order=desc",headers=headers1,verify=False)

response2 = requests.get(url="https://api.github.com/search/repositories?q=RCE&ssort=updated&order=desc",headers=headers1,verify=False)

data1 = json.loads(response1.text)

data2 = json.loads(response2.text)

for j in [data1["items"],data2["items"]]:

for i in j:

s = {"name":i['name'],"html":i['html\_url'],"description":i['description']}

s1 =[i['name'],i['html\_url'],i['description']]

if s1 not in datas:

#print(s1)

#print(datas)

params = {

"text":s["name"],

"desp":" 链接:"+str(s["html"])+"\n简介"+str(s["description"])

}

print("当前推送为"+str(s)+"\n")

#print(params)

requests.packages.urllib3.disable\_warnings()

requests.get("https://sc.ftqq.com/skey信息.send",params=params,headers=headers1,timeout=10,verify=False)

#time.sleep(1)#以防推送太猛

print("推送完成!\n")

datas.append(s1)

else:

pass

print("数据已在!")

pd.DataFrame(datas).to\_csv("olddata.csv",header=None,index=None)

time.sleep(time\_sleep)

✧ 黑暗引擎实现域名端口等收集

✧ 全自动域名收集枚举优秀脚本使用

以xxxxx为例，从标题，域名等收集

以xxxxx为例，全自动脚本使用收集

✧ SRC目标中的信息收集全覆盖

补天上专属SRC上简易测试

✧ 利用其他第三方接口获取更多信息

涉及资源：

https://crt.sh

https://dnsdb.io

http://sc.ftqq.com/3.version

https://tools.ipip.net/cdn.php

https://github.com/bit4woo/teemo

https://securitytrails.com/domain/www.baidu.com/history/a