信息收集

-----灯塔自动化推荐

Docker 安装

docker 安装参考：*https://docs.docker.com/engine/install/*

shell 脚本*:*

curl -fsSL https://get.docker.com -o get-docker.sh

sudo sh get-docker.sh

灯塔安装

mkdir docker-ARL;cd docker-ARL

curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py;python3 get-pip.py

```
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py;python3 get-pip.py
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 2548k  100 2548k    0     0  11640      0  0:03:44  0:03:44 --:--:-- 21023
ERROR: This script does not work on Python 3.6 The minimum supported Python version is 3.7. Please use https://bootstrap.pypa.io/pip/3
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# ls
get-pip.py
root@VM-24-17-ubuntu:/opt/tools/docker-ARL#
```

pip3 install -i https://pypi.tuna.tsinghua.edu.cn/simple docker-compose

```
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# pip3 install -i https://pypi.tuna.tsinghua.edu.cn/simple docker-compose
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Collecting docker-compose
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/f3/3e/ca05e486d44e38eb495ca60b8ca526b192071717387346ed1031ecf78966/docker_compose-
1 (114 kB)
     |████████████████████████████████| 114 kB 5.7 MB/s
Requirement already satisfied: PyYAML<6,>=3.10 in /home/ubuntu/.local/lib/python3.6/site-packages (from docker-compose) (5.4.1)
Requirement already satisfied: jsonschema<4,>=2.5.1 in /home/ubuntu/.local/lib/python3.6/site-packages (from docker-compose) (3.2.0)
```

wget -O *docker-ARL*/docker.zip https://github.com/TophantTechnology/ARL/releases/download/v2.5.1/docker.zip

```
root@VM-24-17-ubuntu:/opt/tools# wget -O docker-ARL/docker.zip https://github.com/TophantTechnology/ARL/releases/download/v2.5.1/docker.zip
--2022-03-02 11:57:59--  https://github.com/TophantTechnology/ARL/releases/download/v2.5.1/docker.zip
Resolving github.com (github.com)... 20.205.243.166
Connecting to github.com (github.com)|20.205.243.166|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/294338949/09d9bc3e-1b0a-4e1b-be92-f5500c9acdb0?X-Amz-Algorithm=
6&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20220302%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20220302T035904Z&X-Amz-Expires=300&X-Amz-Signature=3306
f59a8709e6559853d93e826b4c3d2e07a07ff4c53701&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=294338949&response-content-disposition=attachment%3
ocker.zip&response-content-type=application%2Foctet-stream [following]
--2022-03-02 11:59:04--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/294338949/09d9bc3e-1b0a-4e1b-be92-f5500c9acdb0?
AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20220302%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20220302T035904Z&X-Amz-Expires=300&X-Amz
7a0e0378f43dd7cf59a8709e6559853d93e826b4c3d2e07a07ff4c53701&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=294338949&response-content-dispositi
%20filename%3Ddocker.zip&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.109.133, 185.199.110.133, 185.199.111.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3974 (3.9K) [application/octet-stream]
Saving to: 'docker-ARL/docker.zip'

docker-ARL/docker.zip                   100%[===================================================================================>]   3.88K  10.0KB/s

2022-03-02 11:59:08 (10.0 KB/s) - 'docker-ARL/docker.zip' saved [3974/3974]
```

unzip -o docker.zip

docker-compose pull

```
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# ls
docker.zip  get-pip.py
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# unzip -o docker.zip
Archive:  docker.zip
 extracting: arl_web.log
 extracting: arl_worker.log
  inflating: config-docker.yaml
  inflating: docker-compose.yml
  inflating: mongo-init.js
  inflating: nginx.conf
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# ls
arl_web.log  arl_worker.log  config-docker.yaml  docker-compose.yml  docker.zip  get-pip.py  mongo-init.js  nginx.conf
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# docker-compose pull
Pulling rabbitmq (rabbitmq:3.8.19-management-alpine)...
3.8.19-management-alpine: Pulling from library/rabbitmq
29291e31a76a: Pull complete
fefcdd48ce32: Pull complete
f6bdc078895e: Pull complete
93387edbdddc: Pull complete
0c4f96331156: Pull complete
8f7f3aaf76a4: Pull complete
36d0b1a5cb2b: Pull complete
f04183ff6f17: Pull complete
cccaa9606d9b: Pull complete
Digest: sha256:5cd381c85eb0b7e13ec3df62e4d6255f0fb2f8dab46a5ee442f0b66c8c0910e3
Status: Downloaded newer image for rabbitmq:3.8.19-management-alpine
Pulling mongodb (mongo:4.0.27)...
4.0.27: Pulling from library/mongo
58690f9b18fc: Pull complete
b51569e7c507: Pull complete
da8ef40b9eca: Pull complete
fb15d46c38dc: Pull complete
8c5b4403b3cc: Pull complete
a336ecd37208: Pull complete
12c733cd45a4: Pull complete
0500d06255ed: Pull complete
166a5a996686: Pull complete
709f9e8f3eb4: Pull complete
22bd5150d072: Pull complete
740523b21eb5: Pull complete
76dcc5baf521: Pull complete
Digest: sha256:58cf38cc566cbfca90626292ed83dba2bd50cfe5283184a8f757c133cbfae1a2
Status: Downloaded newer image for mongo:4.0.27
Pulling scheduler (tophant/arl:latest)...
latest: Pulling from tophant/arl
Digest: sha256:e09669508874a55bdcbcc20dc41ebd61e24cd6307ce4899a907d20f4fcb3824c
Status: Image is up to date for tophant/arl:latest
Pulling worker (tophant/arl:latest)...
latest: Pulling from tophant/arl
Digest: sha256:e09669508874a55bdcbcc20dc41ebd61e24cd6307ce4899a907d20f4fcb3824c
Status: Image is up to date for tophant/arl:latest
Pulling web (tophant/arl:latest)...
latest: Pulling from tophant/arl
Digest: sha256:e09669508874a55bdcbcc20dc41ebd61e24cd6307ce4899a907d20f4fcb3824c
Status: Image is up to date for tophant/arl:latest
root@VM-24-17-ubuntu:/opt/tools/docker-ARL#
```


docker volume create arl_db

docker-compose up -d

```
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# docker volume create arl_db
arl_db
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# docker-compose up -d
Creating network "docker-arl_default" with the default driver
Creating arl_mongodb  ... done
Creating arl_rabbitmq ... done
Creating arl_web      ... done
Creating arl_scheduler ... done
Creating arl_worker    ... done
root@VM-24-17-ubuntu:/opt/tools/docker-ARL#
```

dcoker-compose ps -a # 查看

```
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# docker ps -a
CONTAINER ID   IMAGE                          COMMAND                  CREATED          STATUS          PORTS                                                                                    NAMES
5dd6946da612   tophant/arl:latest             "sh -c 'wait-for-it.…"   26 seconds ago   Up 23 seconds                                                                                    arl_worker
6c338666f4ed   tophant/arl:latest             "sh -c 'wait-for-it.…"   26 seconds ago   Up 22 seconds                                                                                    arl_scheduler
d040b146d403   tophant/arl:latest             "sh -c 'gen_crt.sh; …"   26 seconds ago   Up 24 seconds   0.0.0.0:5003->443/tcp, :::5003->443/tcp                                          arl_web
554279d4649c   rabbitmq:3.8.19-management-alpine  "docker-entrypoint.s…"   28 seconds ago   Up 26 seconds   4369/tcp, 5671-5672/tcp, 15671-15672/tcp, 15691-15692/tcp, 25672/tcp   arl_rabbitmq
ad6e5b8757f1   mongo:4.0.27                   "docker-entrypoint.s…"   28 seconds ago   Up 25 seconds   27017/tcp                                                                        arl_mongodb
root@VM-24-17-ubuntu:/opt/tools/docker-ARL#
```

修改密码

默认密码是 admin/admin

登录端口: https://IP:5003/login

docker exec -ti arl_mongodb mongo -u admin -p admin

use arl

db.user.drop()

db.user.insert({ username: 'admin', password: hex_md5('arlsalt!@#'+'你的密码') })

exit

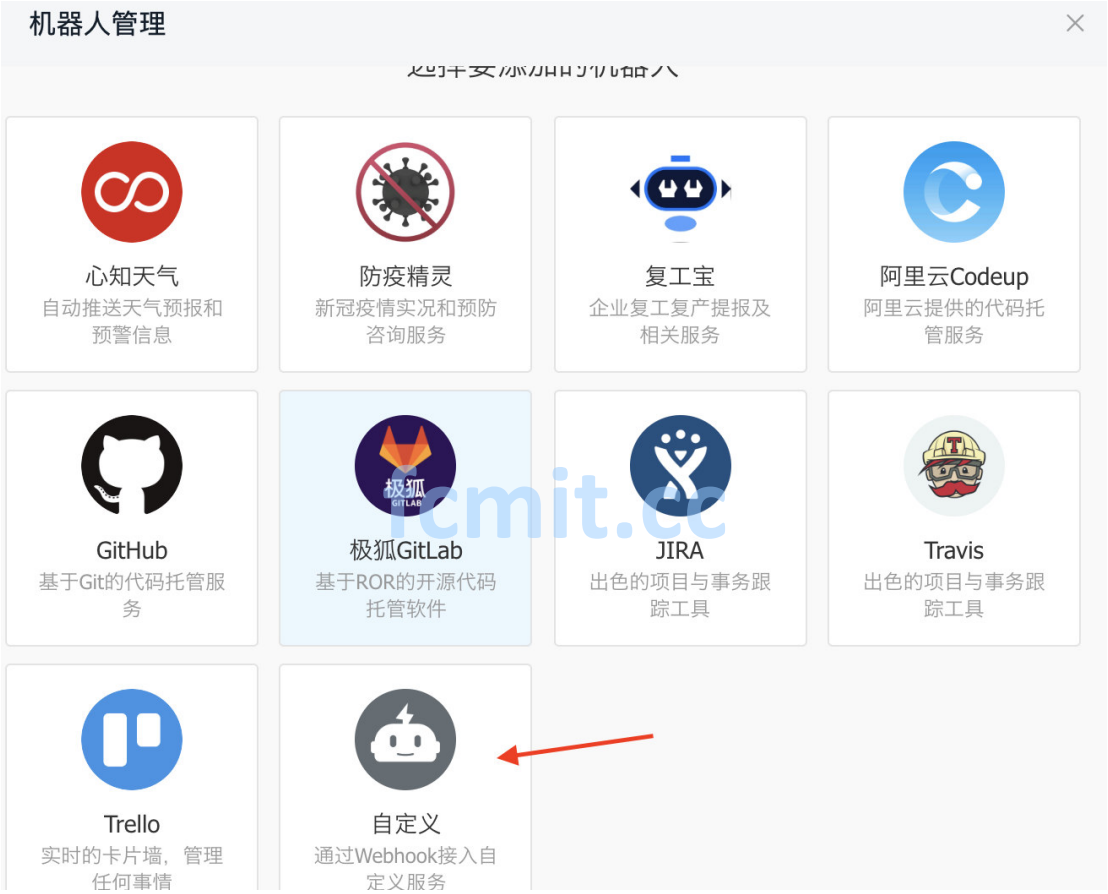
```
root@VM-24-17-ubuntu:/opt/tools/docker-ARL# docker exec -ti arl_mongodb mongo -u admin -p admin
MongoDB shell version v4.0.27
connecting to: mongodb://127.0.0.1:27017/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("9a512b8c-4e6d-480b-bb52-f4fef6c53dfa") }
MongoDB server version: 4.0.27
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        http://docs.mongodb.org/
Questions? Try the support group
        http://groups.google.com/group/mongodb-user
Server has startup warnings:
2022-03-02T04:18:47.214+0000 I STORAGE  [initandlisten]
2022-03-02T04:18:47.214+0000 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
2022-03-02T04:18:47.214+0000 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---

> use arl
switched to db arl
> db.user.drop()
true
> db.user.insert({ username: 'admin',  password: hex_md5('arlsalt!@#'+'mypass') })
WriteResult({ "nInserted" : 1 })
> exit
bye
```

登录

3.配置钉钉机器人

首先自己先创建个群聊，【群聊】-【群设置】-【智能群助手】-【添加更多】-【添加机器人】-【自定义】-【添加】


机器人管理
钉钉机器人可以把你需要的消息及通知，自动推送到钉钉群 了解更多
添加机器人
目前群里最多添加 10 个机器人
...
本群的机器人


机器人管理
心知天气
自动推送天气预报和预警信息
防疫精灵
新冠疫情实况和预防咨询服务
复工宝
企业复工复产提报及相关服务
阿里云Codeup
阿里云提供的代码托管服务
GitHub
基于Git的代码托管服务
极狐GitLab
基于ROR的开源代码托管软件
JIRA
出色的项目与事务跟踪工具
Travis
出色的项目与事务跟踪工具
Trello
实时的卡片墙，管理任何事情
自定义
通过Webhook接入自定义服务

以下两个对应填入 config-docker.yaml 中的 SECRET，ACCESS_TOKEN

配置文件: docker-ARL/config-docker.yaml

```
DINGDING:
  SECRET: " SECRET值 "
  ACCESS_TOKEN: " ACCESS_TOKEN值 "
```


测试是否配置成功

docker-compose exec worker bash

python3.6 -m test.test_utils_push

```
root@VM-24-17-ubuntu:~/docker_arl# docker-compose exec worker bash
[root@858ccfcc98ad code]# python3.6 -m test.test_utils_push
[2022-03-02 13:51:48] [INFO] [MainThread] [fetchSite.py:84] start fetch site 2
/usr/local/lib/python3.6/site-packages/urllib3/connectionpool.py:851: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification
. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning)
[2022-03-02 13:51:49] [INFO] [MainThread] [fetchSite.py:87] end fetch site elapse 0.8097107410430908
[2022-03-02 13:51:49] [INFO] [MainThread] [buildDomainInfo.py:41] start build Domain info 2
[2022-03-02 13:51:49] [INFO] [MainThread] [buildDomainInfo.py:44] end build Domain info 2 elapse 0.20142555236816406
[2022-03-02 13:51:49] [INFO] [MainThread] [push.py:140] push dingding succ
.[2022-03-02 13:51:49] [INFO] [MainThread] [fetchSite.py:84] start fetch site 2
[2022-03-02 13:51:50] [INFO] [MainThread] [fetchSite.py:87] end fetch site elapse 0.6083941459655762
[2022-03-02 13:51:50] [INFO] [MainThread] [buildDomainInfo.py:41] start build Domain info 2
[2022-03-02 13:51:50] [INFO] [MainThread] [buildDomainInfo.py:44] end build Domain info 2 elapse 0.20113205909729004
```

灯塔监测 机器人

大家好！我是 灯塔监测 机器人，很高兴为你们服务。

灯塔机器人推送 机器人 13:51

[灯塔测试域名]新发现域名 10，站点 10

域名 解析类型 记录值

1. www.qq.com CNAME ins-r23tsuuf.ias.tencent-cloud.net
2. www.baidu.com CNAME www.a.shifen.com

站点 标题 状态码 favicon

1. https://www.qq.com/ 腾讯首页 200 1787932733
2. https://www.baidu.com 百度一下，你就知道 200 -1588080585

灯塔机器人推送 机器人

[灯塔测试 IP]新发现 IP 10，站点 10

IP 端口数目 开放端口 组织

监控

灯塔机器人推送 机器人

[监控-百度-baidu.com]新发现域名 3，站点 3

域名 解析类型 记录值

1. 01cdss.baidu.com CNAME cdss01.n.shifen.com
2. 01pda.baidu.com CNAME cdss01.n.shifen.com
3. 01voice.baidu.com CNAME cdss01.n.shifen.com

站点 标题 状态码 favicon

1. https://01voice.baidu.com 404 Not Found 404
2. https://01cdss.baidu.com 404 Not Found 404
3. https://01pda.baidu.com 404 Not Found 404

上述是灯塔的安装，这个是自动检查的资产的好帮手，当你要挖掘的 src 有新的资

产上线的时候，就自动发送到你的钉钉上，这时候，你就可以提前去挖掘，运气好就捡到高危险漏洞。

0x01 常用工具挖掘

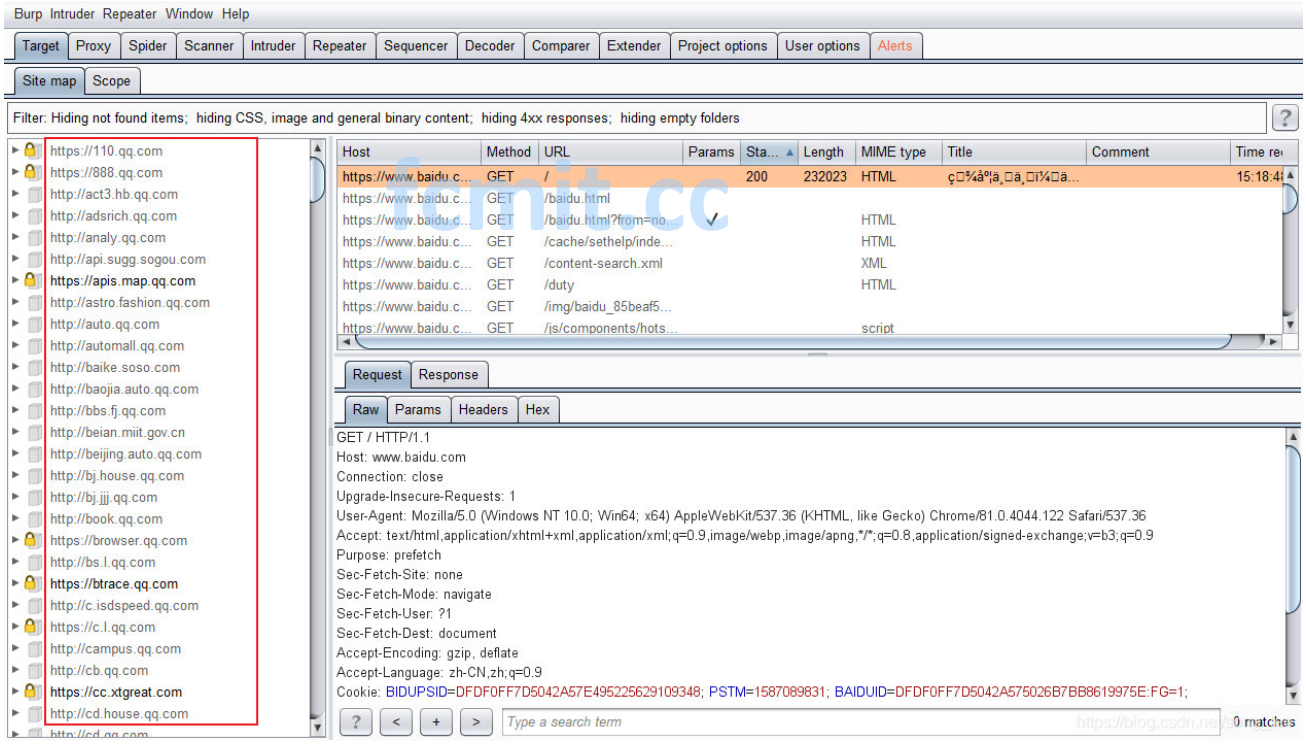
还有一种信息收集收集方式就是直接子域名挖掘，然后一个站点一个站点的查看：

常用工具：Maltego CE，wydomain，subDomainBrute，dnsmaper，Layer 子域名挖掘机 oneforall 子域名收集工具

通过这些工具将相关域名的资产全部找出来，然后慢慢挖掘即可。

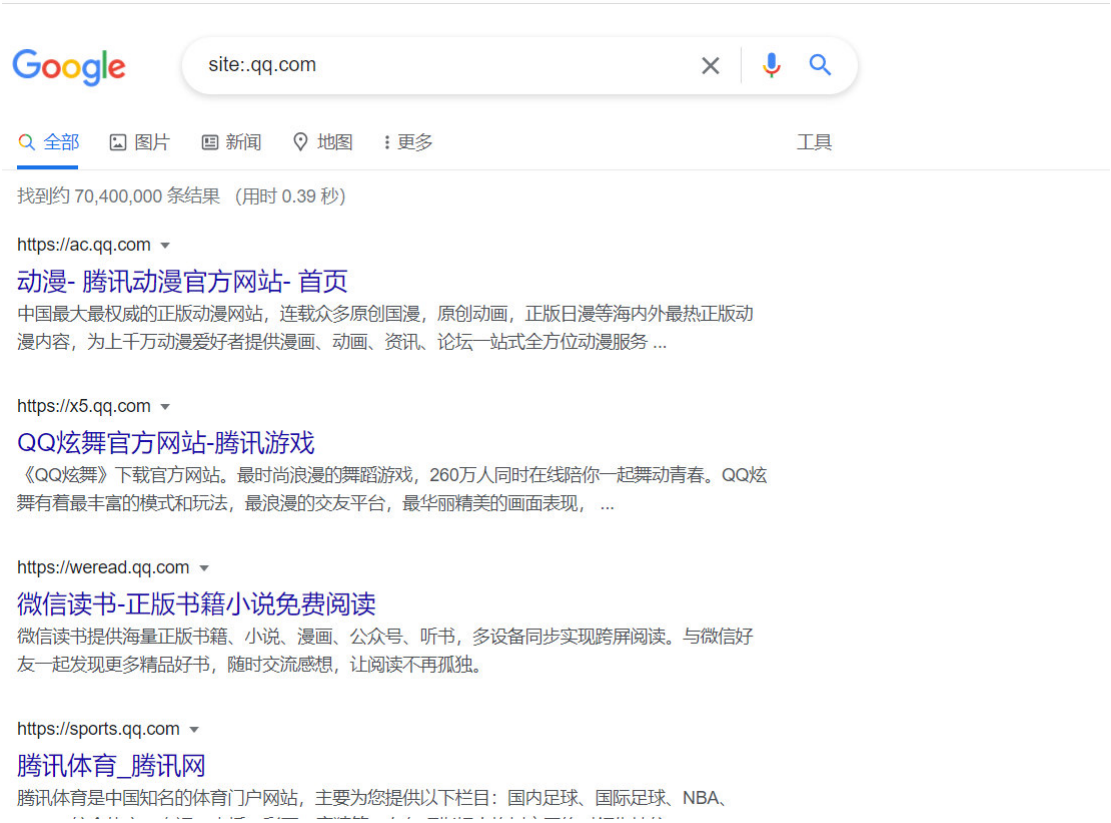
0x02 爬虫提取子域名

这类工具有很多，例如 burpsuite、appscan、awvs 都有爬虫的功能



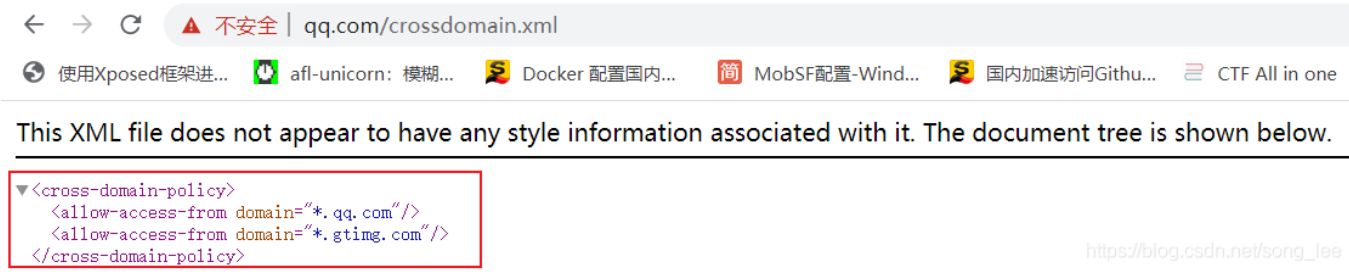
0x03 搜索引擎

搜索引擎提供了一些高级搜索指令，site 就可以查询相关的域名，其实搜索引起收录的网页也是通过爬虫来爬取的。



0x04 站点配置文件

crossdomain.xml，跨域策略配置文件



robots.txt，反爬虫配置文件，Robots 协议用来告知搜索引擎哪些页面能被抓取，哪些页面不能被抓取