

# Arch Linux 安装并使用 larmies/theHarvester

Github: <https://github.com/laramies/theHarvester>

依赖:python git pip

## 1. 安装：

```
git clone
```

<https://github.com/laramies/theHarvester.git>

```
cd theHarvester/
```

`pip install -r requirements.txt`(网速慢请自行寻找pip镜像站)

```
python theHarvester.py -h
```

```
*
 * _____      _   _       _   _       _   _    _   _     _   _
 * |  ___ \ / ____| | | |_ _|_ \| |_|_|_||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_*
 * | |___\ V ___ \| |_| |__|_) ||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_*
 * |  __) | |__) | | | |__|_) ||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_||_|_|_|_*
 * |_____\___/\___\_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_*
 *
theHarvester 4.3.0                                     *
Coded by Christian Martorella                          *
Edge-Security Research                                +
cmartorell@edge-security.com                           *
*
*****
usage: theHarvester.py [-h] -d DOMAIN [-l LIMIT] [--START START] [-p] [-s] [--screenshot SCREENSHOT] [-v] [-e [DNS_SERVER]] [-t] [-r [DNS_RESOLVE]] [-n] [-c] [-f FILENAME] [-b SOURCE]

theHarvester is used to gather open source intelligence (OSINT) on a company or domain.

options:
-h, --help                show this help message and exit
-d DOMAIN, --domain DOMAIN Company name or domain to search.
-l LIMIT, --limit LIMIT    Limit the number of search results, default=500.
--START START              Start with result number X, default=0.
-p, --proxies             Take proxies from requests, enter proxies in proxies.yaml.
-s, --shodan               Use Shodan to query discovered hosts.
--screenshot SCREENSHOT   Take screenshots of resolved domains specify output directory: --screenshot output_directory
-v, --virtual-host         Verify host name via DNS resolution and search for virtual hosts.
-e DNS_SERVER, --dns-server DNS_SERVER                 DNS server to use for lookup.
-t, --take-over            Check for takeovers.
-r [DNS_RESOLVE], --dns-resolve [DNS_RESOLVE]          Perform DNS resolution on subdomains with a resolver list or passed in resolvers, default False.
-n, --dns-lookup           Enable DNS server lookup, default False.
-c, --dns-brute            Perform a DNS brute force on the domain.
-f FILENAME, --filename FILENAME                      Save the results to an XML and JSON file.
-b SOURCE, --source SOURCE                               anubis, baidu, bevigil, binaryedge, Bing, Bingapi, bufferoverun, brave, census, certspotter, criminalip, crtsh, dnsdumpster, duckduckgo, fullhunt, github-code, hackertarget, hunter,
```

输出如图所示，安装成功

## 2. 命令详解

---

- h, --help 显示帮助信息
- d, --domain 要搜索的目标
- l, --limit 输出数量，默认500
- S, --start 从采集到的信息编号“X”处开始执行采集，默认为0
- p, --proxies 使用代理，调用proxies.yaml
- s, --shodan 使用shodan查询
- screenshot 截图
- screenshot output\_directory 截图保存路径
- v, --virtual-host 使用DNS解析 (?)
- e, --dns-server 指定DNS解析服务器
- t, --take-over 在Host found处显示ip
- n, --dns-lookup 启用DNS服务器查找，默认为False  
状态 (开启 Starting active queries for  
DNSLookup. 块)

- c, --dns-brute 进行DNS域解析暴力破解
- f, --filename 指定输出文件名，格式支持JSON和XML
- b SOURCE,--source SOURCE 指定采集信息的源（如百度等）

## 3.使用例

---

### 3.1 普通调用

我们使用 -d指定搜索的目标，使用-b参数指定搜索引擎 baidu,命令如下

```
./theHarvester.py -d baidu.com -b baidu
```

[\*] Target: baidu.com

[\*] Searching Baidu.

[\*] No IPs found.

**[\*] Emails found: 37**

---

[air-info@baidu.com](mailto:air-info@baidu.com)

[apihelp@baidu.com](mailto:apihelp@baidu.com)

...

[zpkf@baidu.com](mailto:zpkf@baidu.com)

# [\*] Hosts found: 124

ai.baidu.com

aib.baidu.com

...

zhidao.baidu.com

## 3.2 子域名接管检查(?)

```
./theHarvester.py -d baidu.com -b baidu -t
```

加入 -r 参数后激活子域名接管检查(?)

[\*] Performing subdomain takeover check

[\*] Subdomain Takeover checking IS ACTIVE RECON

(调用Fly.io激活主动探查)

```
Takeover detected: http://gimg3.baidu.com
```

```
Type of takeover is: Fly.io
```

```
`Takeover detected: http://gimg3.baidu.com
```

```
Type of takeover is: Fly.io
```

...

```
Takeover detected: http://vse.baidu.com
```

Type of takeover is: Fly.io

## 3.3 DNS 查询

```
./theHarvester.py -d baidu.com -b baidu -e  
8.8.8.8 -n
```

新增DNS 查询块（没有回显不知到为什么）

[\*] Starting active queries for DNSLookup.

[\*] Hosts found after reverse lookup (in target domain):

## 3.4 DNS爆破（?）

```
./theHarvester.py -d baidu.com -b baidu -c
```

新增DNS爆破区块

根据自带字典进行DNS爆破

[\*] Starting DNS brute force.

Starting DNS brute forcing with 4989 words

[\*] Hosts found after DNS brute force:

100.baidu.com:103.211.221.225

123.baidu.com:110.242.68.3, 110.242.68.4

2012.baidu.com:157.255.77.215, 153.3.236.50,

157.255.77.214

911.baidu.com:10.92.154.58

...

z.baidu.com:111.206.209.78, 111.206.209.79

zt.baidu.com:111.206.209.18

发现输出的结果后面跟着i(?)

## 3.5 文件输出

```
./theHarvester.py -d baidu.com -b baidu -f
```

test

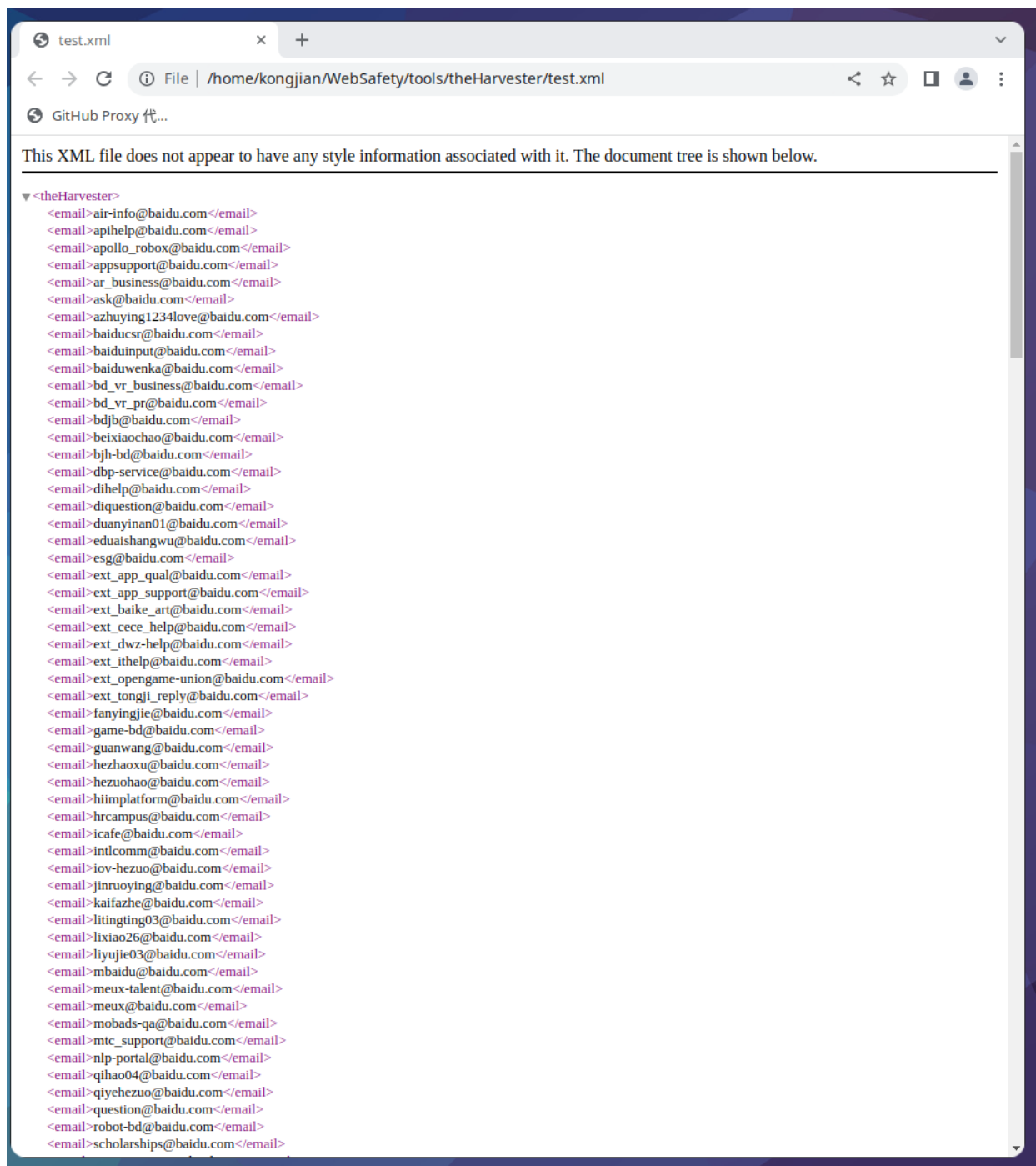
```
[*] Reporting started.
```

```
[*] XML File saved.
```

```
[*] JSON File saved.
```

会在当前目录下生成text.json 和 text.xml

用chrome打开xml



bat 打开 test.json

```
~/WebSecurity/tools/theHarvester (22:59:54 on master) → bat test_json
File: test_json
1 {
  "emails": [
    "air-info@baidu.com", "apihelp@baidu.com", "apollo_robox@baidu.com", "appsupport@baidu.com", "ar_business@baidu.com", "ask@baidu.com", "azhuying1234love@baidu.com", "baiducsr@baidu.com", "baiduinput@baidu.com", "baiduwenka@baidu.com", "bd_vr_business@baidu.com", "bd_vr_pr@baidu.com", "bdjb@baidu.com", "beixiaochao@baidu.com", "bjh-bd@baidu.com", "dbp-service@baidu.com", "dihelp@baidu.com", "diquestion@baidu.com", "duanyinan01@baidu.com", "eduaishangwu@baidu.com", "esg@baidu.com", "ext_app_qual@baidu.com", "ext_app_support@baidu.com", "ext_baike_article@baidu.com", "ext_cece_help@baidu.com", "ext_dwz_help@baidu.com", "ext_ithelp@baidu.com", "ext_opengame-union@baidu.com", "ext_tongji_reply@baidu.com", "fanyingjie@baidu.com", "game-bd@baidu.com", "guanwang@baidu.com", "hezhaosu@baidu.com", "hezhaosu@baidu.com", "hiimplatform@baidu.com", "hrcampus@baidu.com", "icsf@baidu.com", "intlcomm@baidu.com", "ioy-hezu@baidu.com", "jinruoying@baidu.com", "kaifazhe@baidu.com", "litingting03@baidu.com", "lixiao26@baidu.com", "liyujie03@baidu.com", "mbaidu@baidu.com", "meux-talent@baidu.com", "meux@baidu.com", "mobads-qa@baidu.com", "mtc.support@baidu.com", "nlp-portal@baidu.com", "qihao04@baidu.com", "qiyehezu@baidu.com", "question@baidu.com", "robot-bd@baidu.com", "scholarships@baidu.com", "service-tuijian@baidu.com", "shamingxiao@baidu.com", "softunion@baidu.com", "tanfuquan@baidu.com", "translate_api@baidu.com", "ueditor@baidu.com", "union1@baidu.com", "uxc@baidu.com", "v_lyuyi1in01@baidu.com", "v_xuchongchong@baidu.com", "wangdan40@baidu.com", "wangjingwei@baidu.com", "wangxinlei08@baidu.com", "webmaster@baidu.com", "wenkuhezu@baidu.com", "wurongqian@baidu.com", "xiaoduot-bd@baidu.com", "yangbin@baidu.com", "yangtianhang01@baidu.com", "ybb-support@baidu.com", "yuqing-help@baidu.com", "zengxing@baidu.com", "zhangtai01@baidu.com", "zhangxiaoyu16@baidu.com", "zhaorui16@baidu.com", "zpkf@baidu.com"],
  "hosts": [
    "01.baidu.com", "123.baidu.com", "3fmail.baidu.com", "abc.baidu.com", "access.mobads.baidu.com", "ai.baidu.com", "aib.baidu.com", "aidu.baidu.com", "als.baidu.com", "api.fanyi.baidu.com", "api.ime.baidu.com", "api.m.baidu.com", "api.map.baidu.com", "api.open.baidu.com", "app.baidu.com", "ar.baidu.com", "author.baidu.com", "b2b.baidu.com", "baidu.com", "baijiahao.baidu.com", "baike.baidu.com", "bama.baidu.com", "baozhang.baidu.com", "bar.baidu.com", "bdimg.share.baidu.com", "bos.nj.bpc.baidu.com", "bs.baidu.com", "c.baidu.com", "cang.baidu.com", "cece.baidu.com", "che.baidu.com", "chelianwang.baidu.com", "d.baidu.com", "data.zz.baidu.com", "dr.dh.baidu.com", "dss0.baidu.com", "dssl.baidu.com", "dunxin.baidu.com", "dueros-h2.baidu.com", "dumix.baidu.com", "duzpassport.baidu.com", "e.baidu.com", "echarts.baidu.com", "eclink.baidu.com", "ecloud.baidu.com", "email.baidu.com", "esg.baidu.com", "eux.baidu.com", "f3.baidu.com", "fanyi-api.baidu.com", "fav.baidu.com", "fc-ccimage.baidu.com", "fc.baidu.com", "fclink.baidu.com", "fox.baidu.com", "firefox.baidu.com", "gat.e.baidu.com", "gimg3.baidu.com", "gimg4.baidu.com", "gips0.baidu.com", "gips1.baidu.com", "gips2.baidu.com", "gips3.baidu.com", "gongyi.baidu.com", "graph.baidu.com", "haokan.baidu.com", "hdpreload.baidu.com", "hectorstatic.baidu.com", "help.baidu.com", "hezuo.baidu.com", "home.baidu.com", "houtai.baidu.com", "i.baidu.com", "i7.baidu.com", "i8.baidu.com", "i9.baidu.com", "idl1.baidu.com", "ife.baidu.com", "ihuisheng.baidu.com", "image.baidu.com", "ime.baidu.com", "implus.baidu.com", "index.baidu.com", "ipv6.baidu.com", "jianyi.baidu.com", "job.baidu.com", "jaccess.baidu.com", "jubao.baidu.com", "kankan.baidu.com", "lbsyun.baidu.com", "lcr.open.baidu.com", "leads.baidu.com", "learning.baidu.com", "list.baidu.com", "m.baidu.com", "m.help.baidu.com", "m11.baidu.com", "m2.baidu.com", "m3.baidu.com", "m6.baidu.com", "m8.baidu.com", "mail.baidu.com", "map.baidu.com", "mapp.baidu.com", "mc.baidu.com", "minibus.baidu.com", "mo.baidu.com", "moo.baidu.com", "mtc.baidu.com", "mtj.baidu.com", "nbaidu.com", "news.baidu.com", "nlp.baidu.com", "nsclick.baidu.com", "oline.baidu.com", "open.duer.baidu.com", "opendata.baidu.com", "p.qiao.baidu.com", "pan.baidu.com", "passport.baidu.com", "passportso.baidu.com", "pccrec.baidu.com", "pd.baidu.com", "post.baidu.com", "privacy.baidu.com", "publish.baidu.com", "qingpai.baidu.com", "qingting.baidu.com", "qlb.baidu.com", "quzheng.baidu.com", "research.baidu.com", "robot.baidu.com", "rules.baidu.com", "s.share.baidu.com", "sbaidu.com", "scholar.baidu.com", "scholarship.baidu.com", "sclink.baidu.com", "sec.baidu.com", "sensearch.baidu.com", "service.baidu.com", "sestat.baidu.com", "sh.baidu.com", "shurufa.baidu.com", "sifecenter.baidu.com", "spi.baidu.com", "sp2.baidu.com", "sp3.baidu.com", "ss0.baidu.com", "s1.baidu.com", "sti.baidu.com", "sux.baidu.com", "t1.baidu.com", "t19.baidu.com", "t11.baidu.com", "t12.baidu.com", "t13.baidu.com", "t14.baidu.com", "t15.baidu.com", "t16.baidu.com", "t2.baidu.com", "t3.baidu.com", "t7.baidu.com", "t8.baidu.com", "t9.baidu.com", "tag.baidu.com", "talent.baidu.com", "tieba.baidu.com", "tn.baidu.com", "tongji.baidu.com", "top.baidu.com", "trust.baidu.com", "trustrcv.baidu.com", "tuijian.baidu.com", "unditor.baidu.com", "union.baidu.com", "usa.baidu.com", "v.baidu.com", "voice.baidu.com", "vr.baidu.com", "vse.baidu.com", "wappass.baidu.com", "waidu.com", "web.im.baidu.com", "weishi.baidu.com", "wenka.baidu.com", "wenku.baidu.com", "wq.copyright.baidu.com", "ww.baidu.com", "www1.baidu.com", "www7.baidu.com", "xapp.baidu.com", "xiaodu.baidu.com", "xueshu.baidu.com", "yingxiatong.baidu.com", "yunce.baidu.com", "yuqing.baidu.com", "z.baidu.com", "zhaopin.baidu.com", "zhidao.baidu.com"],
  "shodan": []
}
```

## 4. 常见问题

1. 当使用 `./theHarvester.py -d baidu.com -b all` 时，会提示缺少API key，可以自行去各大网站查询相应的API key（大多数需要付费），把得到的API key写入到 `api-keys.yaml` 中的 `key:` 后，程序会直接读取API key进行调用引擎

## 个人问题

1. 在上文中标记（？）的