The author used shodan python API to search shodan query, so i actually check this API. it is straightforward.

Official website: <https://shodan.readthedocs.io/en/latest/index.html>

API download: <https://developer.shodan.io/api/clients>

Basic Shodan search program: <https://shodan.readthedocs.io/en/latest/examples/basic-search.html>

-> The API key is generated when you register an account. I created an account bc i'm not sure whether there would be any charge method or sth (weird thought but just in case).

So the API key i use : sy3UmXUvE16OjS49q9VmKO8b5gzZOPCl

Useful info about shudan api search result:

1. Search result return a dictionary
2. Keys() of this dictionary: [u'matches', u'total']
   1. matches : list with len(): len(result['matches'])
   2. total : total number of results
3. Each list of matches is a dic with keys(): [u'\_shodan', u'hash', u'os', u'ip', u'isp', u'port', u'hostnames', u'location', u'timestamp', u'domains', u'org', u'data', u'asn', u'transport', u'ip\_str']
4. Location field is a dic with keys(): result['matches'][0]['location'].keys() -> [u'city', u'region\_code', u'area\_code', u'longitude', u'country\_code3', u'country\_name', u'postal\_code', u'dma\_code', u'country\_code', u'latitude']
5. That’s all i know for now, we could use these facets to filter data we need.

How to use the program:

1. Open terminal and direct to the dis.py file directory
2. Type in (underlined part): python dis.py A850+Telemetry+Gateway /BroadWeb/ (could be any number of queries)
3. Return query name and connection on terminal, create csv file named ‘results.csv’ (queries~connections) and ‘total.csv’(queries~country name, latitude, longitude, os, data, ip).

-> So the followings are what i did and the result, i upload csv to this same folder (underline is the code you could copy and run on terminal):

Bruce$ python dis.py A850+Telemetry+Gateway ABB+Webmodule Allen-Bradley /BroadWeb/ Cimetrics+Eplus+Web+Server CIMPLICITY CitectSCADA EIG+Embedded+Web+Server eiPortal EnergyICT HMS+AnyBus-S+WebServer i.LON ioLogik Modbus+Bridge ModbusGW Modicon+M340+CPU Niagara+Web+Server NovaTech+HTTPD Powerlink Reliance+4+Control+Server RTS+Scada RTU560 Simatic+HMI SIMATIC+NET Simatic+S7 SoftPLC TAC/Xenta WAGO webSCADA-Modbus

Result:

query name: Simatic+HMI Connections: 108

query name: SoftPLC Connections: 1196

query name: NovaTech+HTTPD Connections: 0

query name: Modbus+Bridge Connections: 103

query name: Allen-Bradley Connections: 5471

query name: EIG+Embedded+Web+Server Connections: 144

query name: Cimetrics+Eplus+Web+Server Connections: 11

query name: /BroadWeb/ Connections: 8

query name: eiPortal Connections: 29

query name: SIMATIC+NET Connections: 223

query name: CIMPLICITY Connections: 39

query name: webSCADA-Modbus Connections: 0

query name: A850+Telemetry+Gateway Connections: 50

query name: EnergyICT Connections: 914

query name: TAC/Xenta Connections: 0

query name: Reliance+4+Control+Server Connections: 13

query name: Simatic+S7 Connections: 923

query name: i.LON Connections: 3314

query name: HMS+AnyBus-S+WebServer Connections: 49

query name: Niagara+Web+Server Connections: 27186

query name: Modicon+M340+CPU Connections: 70

query name: ABB+Webmodule Connections: 7

query name: ioLogik Connections: 290

query name: CitectSCADA Connections: 0

query name: RTU560 Connections: 3

query name: ModbusGW Connections: 71

query name: RTS+Scada Connections: 4

query name: WAGO Connections: 174

query name: Powerlink Connections: 775

The second time i run it:

python dis.py A850+Telemetry+Gateway ABB+Webmodule Allen-Bradley /BroadWeb/ Cimetrics+Eplus+Web+Server CIMPLICITY CitectSCADA EIG+Embedded+Web+Server eiPortal EnergyICT HMS+AnyBus-S+WebServer i.LON ioLogik Modbus+Bridge ModbusGW Modicon+M340+CPU Niagara+Web+Server NovaTech+HTTPD Powerlink Reliance+4+Control+Server RTS+Scada RTU560 Simatic+HMI SIMATIC+NET Simatic+S7 SoftPLC TAC/Xenta WAGO webSCADA-Modbus

Error: Please upgrade your API plan to use filters or paging.

Update april 15:

For the purpose of research it is better to show the complete results. There are more interesting information we could use for research purpose.

>>> sample['matches'][0]

{'\_shodan': {'crawler': 'c9b639b99e5410a46f656e1508a68f1e6e5d6f99',

'id': 'c60399a0-5ac3-4344-bca3-3b677de83e3a',

'module': 'http',

'options': {}},

'asn': 'AS12716',

'data': 'HTTP/1.1 200 OK\r\nServer: ABB RTU560\r\nConnection: close\r\nContent-Type: text/html\r\nContent-Length: 341\r\n\r\n',

'deprecated': {'html': {'eol': '2016-12-31', 'new': 'http.html'},

'title': {'eol': '2016-12-31', 'new': 'http.title'}},

'domains': [],

'hash': -851157544,

'hostnames': [],

'http': {'components': {},

'host': '213.226.45.163',

'html': '<html>\r\n <head>\r\n <title>RTU560</title>\r\n </head>\r\n <frameset cols="175,\*">\r\n <frame name="menu" src="/ABBMain/Menu" scrolling="auto">\r\n <frame name="main" src="/fs/ssi/start.htm" scrolling="auto">\r\n <noframes>\r\n Your browser does not support frames, please use another browser!\r\n </noframes>\r\n </frameset>\r\n</html>\r\n',

'html\_hash': 1981634668,

'location': '/',

'redirects': [],

'robots': None,

'server': 'ABB RTU560',

'sitemap': None,

'title': 'RTU560'},

'ip': 3588369827,

'ip\_str': '213.226.45.163',

'isp': 'Mobiltel Ead',

'location': {'area\_code': None,

'city': None,

'country\_code': 'BG',

'country\_code3': 'BGR',

'country\_name': 'Bulgaria',

'dma\_code': None,

'latitude': 42.69999999999999,

'longitude': 23.33330000000001,

'postal\_code': None,

'region\_code': None},

'org': 'Mobiltel Ead',

'os': None,

'port': 80,

'timestamp': '2017-04-15T13:34:16.449036',

'title': 'RTU560',

'transport': 'tcp'}