Group no: - 8

Group Member:- **Shrut Shah – 19BCP125**

Shubham Kathiriya – 19BCP127

Vedant Patel – 19BCP138

Subject: - Cyber Security Lab

Division:-2

Lab 10:- Crawler

Introduction:-

With the advent of the era of big data, the need for network information has increased widely. Many different companies collect external data from the Internet for various reasons: analysing competition, summarizing news stories, tracking trends in specific markets, or collecting daily stock prices to build predictive models. Therefore, web crawlers are becoming more important. Web crawlers automatically browse or grab information from the Internet according to specified rules. According to the implemented technology and structure, web crawlers can be divided into general web crawlers, focused web crawlers, incremental web crawlers, and deep web crawlers. The basic workflow of a general web crawler is as follows:

- Get the initial URL. The initial URL is an entry point for the web crawler, which links to the web page that needs to be crawled
- While crawling the web page, we need to fetch the HTML content of the page, then parse it to get the URLs of all the pages linked to this page.
- Put these URLs into a queue;

- Loop through the queue, read the URLs from the queue one by one, for each URL, crawl the corresponding web page, then repeat the above crawling process;
- Check whether the stop condition is met. If the stop condition is not set, the crawler will keep crawling until it cannot get a new URL.

Procedure:-

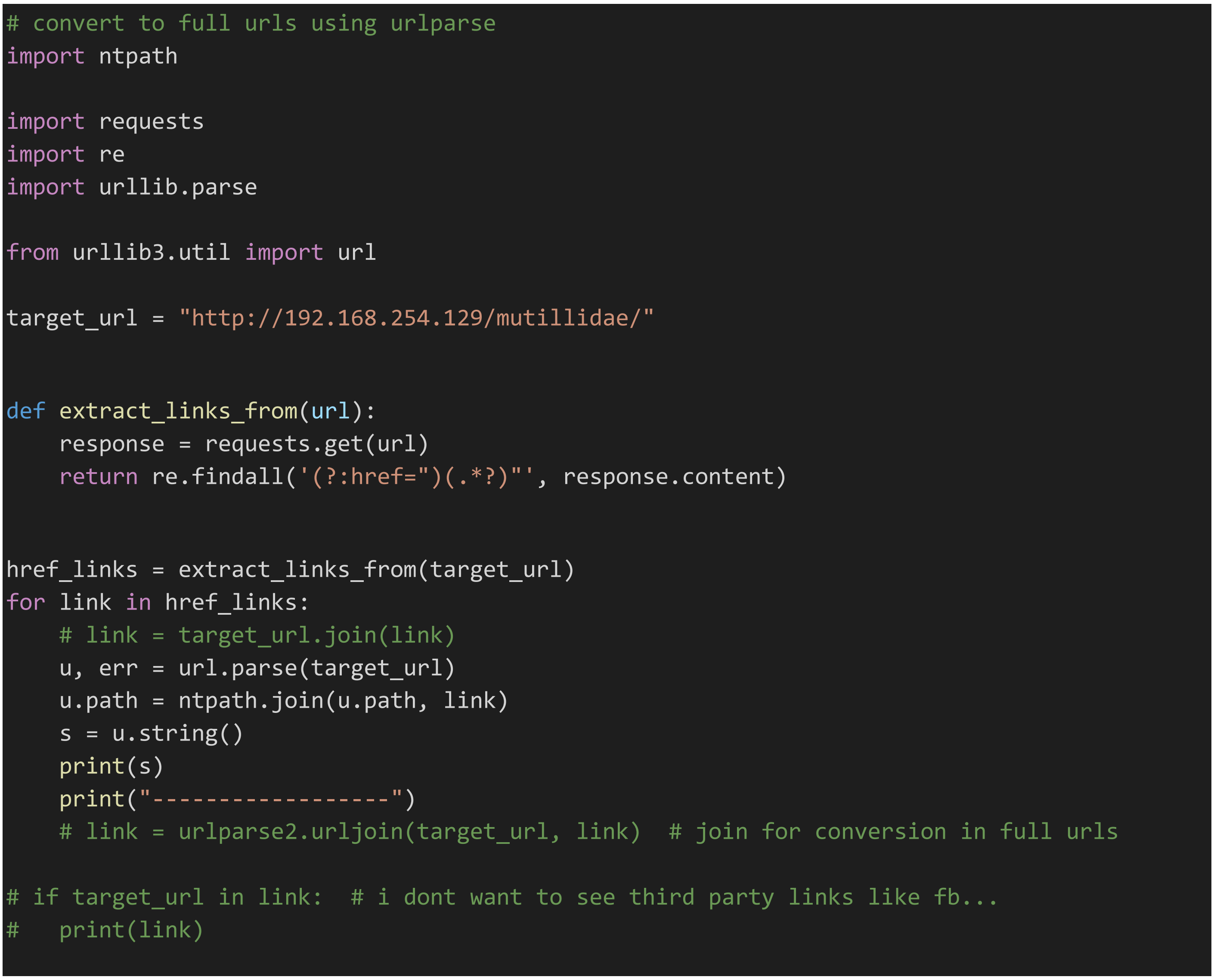
Steps to setup the environment.

1) Make sure that a browser such as Chrome, Edge or Firefox has been installed.
2) Download and install Python.
3) Download a suitable IDE like Atom, Spider, VS Code, PyCharm etc.
4) Install the required Python packages.
   a. Pip can be used to installed the required packages.
   b. Pip install beautifulsoup4
   c. Pip install requests
   d. Pip install lxml
5) Beautiful Soup is a library for easily parsing HTML and XML data.
6) Lxml is a library to improve the parsing speed of XML files.
7) Requests is a library to simulate HTTP requests.

Work:-

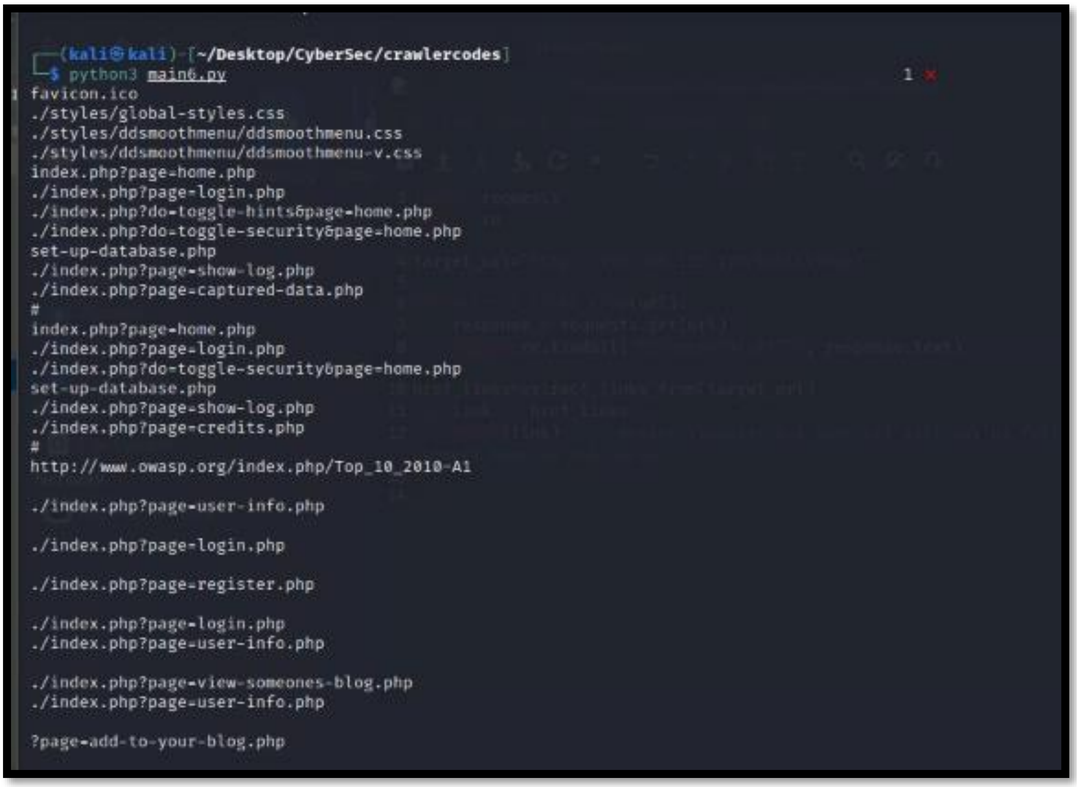
In this assignment, we are going to crawl the data from Wikipedia page. We are first going to open Tom Cruise Wikipedia page and extract all the links related to him. We will ignore other links and just keep links related to him. Those links will further be used to read more the data

Code:-

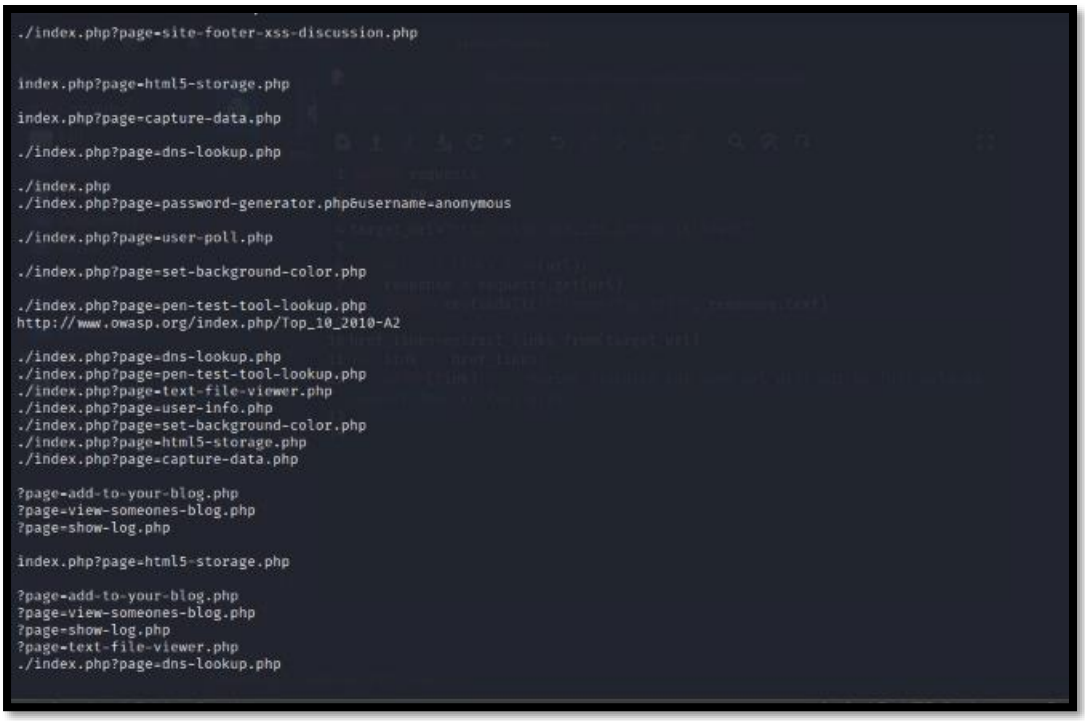


```python
# convert to full urls using urlparse
import ntpath

import requests
import re
import urllib.parse

from urllib3.util import url

target_url = "http://192.168.254.129/mutillidae/"


def extract_links_from(url):
    response = requests.get(url)
    return re.findall('(?:href=")(.*?)"', response.content)


href_links = extract_links_from(target_url)
for link in href_links:
    # link = target_url.join(link)
    u, err = url.parse(target_url)
    u.path = ntpath.join(u.path, link)
    s = u.string()
    print(s)
    print("------------------")
    # link = urlparse2.urljoin(target_url, link)  # join for conversion in full urls

# if target_url in link:  # i dont want to see third party links like fb...
#   print(link)
```

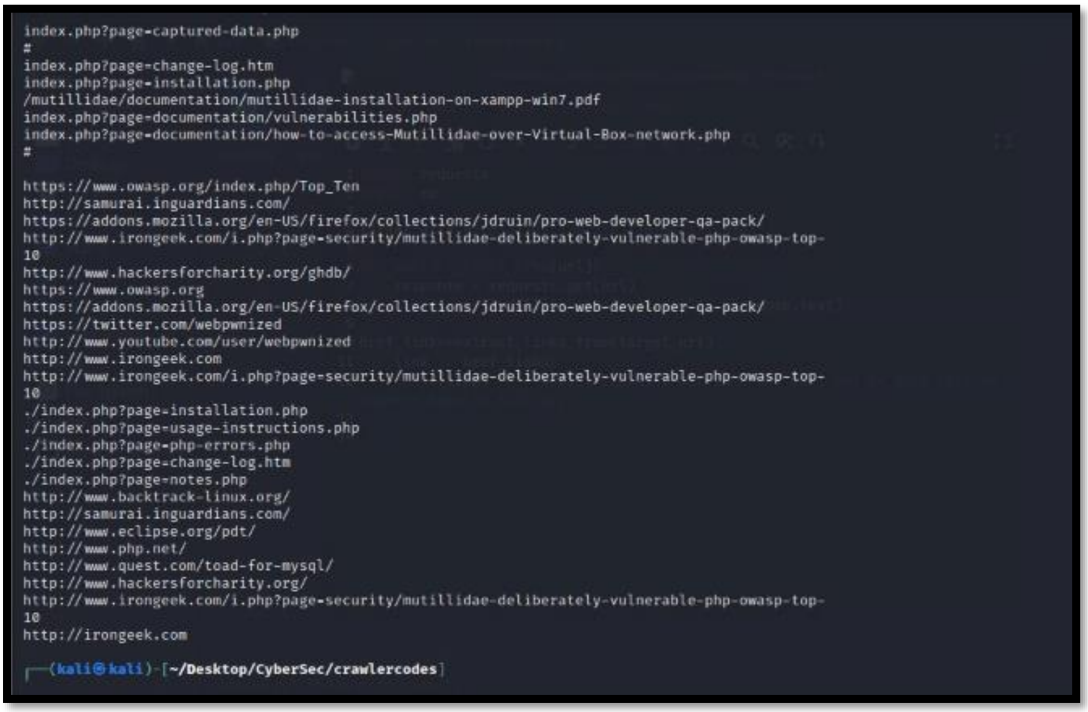
Output Screenshot:-



```
┌──(kali㉿kali)-[~/Desktop/CyberSec/crawlercodes]
└─$ python3 main6.py                                              1 ✗
favicon.ico
./styles/global-styles.css
./styles/ddsmoothmenu/ddsmoothmenu.css
./styles/ddsmoothmenu/ddsmoothmenu-v.css
index.php?page=home.php
./index.php?page=login.php
./index.php?do=toggle-hints&page=home.php
./index.php?do=toggle-security&page=home.php
set-up-database.php
./index.php?page=show-log.php
./index.php?page=captured-data.php
#
index.php?page=home.php
./index.php?page=login.php
./index.php?do=toggle-security&page=home.php
set-up-database.php
./index.php?page=show-log.php
./index.php?page=credits.php
#
http://www.owasp.org/index.php/Top_10_2010-A1

./index.php?page=user-info.php

./index.php?page=login.php

./index.php?page=register.php

./index.php?page=login.php
./index.php?page=user-info.php

./index.php?page=view-someones-blog.php
./index.php?page=user-info.php

?page=add-to-your-blog.php
```



```
./index.php?page=site-footer-xss-discussion.php


index.php?page=html5-storage.php

index.php?page=capture-data.php

./index.php?page=dns-lookup.php

./index.php
./index.php?page=password-generator.php&username=anonymous

./index.php?page=user-poll.php

./index.php?page=set-background-color.php

./index.php?page=pen-test-tool-lookup.php
http://www.owasp.org/index.php/Top_10_2010-A2

./index.php?page=dns-lookup.php
./index.php?page=pen-test-tool-lookup.php
./index.php?page=text-file-viewer.php
./index.php?page=user-info.php
./index.php?page=set-background-color.php
./index.php?page=html5-storage.php
./index.php?page=capture-data.php

?page=add-to-your-blog.php
?page=view-someones-blog.php
?page=show-log.php

index.php?page=html5-storage.php

?page=add-to-your-blog.php
?page=view-someones-blog.php
?page=show-log.php
?page=text-file-viewer.php
./index.php?page=dns-lookup.php
```



```
index.php?page=captured-data.php
#
index.php?page=change-log.htm
index.php?page=installation.php
/mutillidae/documentation/mutillidae-installation-on-xampp-win7.pdf
index.php?page=documentation/vulnerabilities.php
index.php?page=documentation/how-to-access-Mutillidae-over-Virtual-Box-network.php
#

https://www.owasp.org/index.php/Top_Ten
http://samurai.inguardians.com/
https://addons.mozilla.org/en-US/firefox/collections/jdruin/pro-web-developer-qa-pack/
http://www.irongeek.com/i.php?page=security/mutillidae-deliberately-vulnerable-php-owasp-top-
10
http://www.hackersforcharity.org/ghdb/
https://www.owasp.org
https://addons.mozilla.org/en-US/firefox/collections/jdruin/pro-web-developer-qa-pack/
https://twitter.com/webpwnized
http://www.youtube.com/user/webpwnized
http://www.irongeek.com
http://www.irongeek.com/i.php?page=security/mutillidae-deliberately-vulnerable-php-owasp-top-
10
./index.php?page=installation.php
./index.php?page=usage-instructions.php
./index.php?page=php-errors.php
./index.php?page=change-log.htm
./index.php?page=notes.php
http://www.backtrack-linux.org/
http://samurai.inguardians.com/
http://www.eclipse.org/pdt/
http://www.php.net/
http://www.quest.com/toad-for-mysql/
http://www.hackersforcharity.org/
http://www.irongeek.com/i.php?page=security/mutillidae-deliberately-vulnerable-php-owasp-top-
10
http://irongeek.com

┌──(kali㉿kali)-[~/Desktop/CyberSec/crawlercodes]
```

What are the limitations of a crawl?

❖ Rules you set for SEO crawlers

- ➢ Unlike for google bot, you can set many of the parameters of an SEO bot’s behaviour. This will determine how the crawl is carried out, and what pages the bot can discover.
- ➢ A very obvious example is a crawl limit in the form of a maximum number of URLs. If this parameter is set and the number is too low for your site, the bot won’t be able to crawl all of your pages–but not for a technical reason!

❖ Anti-bot prejudice

Sites that don’t play well with bots often create problems with crawls.

This includes sites that refuse access to bots. Sites like this might have legitimate reasons for excluding bots:

- ➢ The site owner does not want the site indexed.
- ➢ The site requires a login, a cookie, or another verification method that standard bots can’t provide.
- ➢ The site has restrictive authorizations after a bad bot experience, such as scraping, monitoring by competitors, or an attack meant to bring down a server.

- The site uses a third-party protection service that blocks all unknown bots or that blocks bots on the service’s blacklist, whether or not the site itself has had a bad experience.

How to overcome problems with crawls

Crawling may be an essential tool in the SEO toolbox, but when taken alone it is not an SEO panacea. As we have seen above, many of the shortcomings of a crawl can be mitigated or eliminated by two major strategies:

1. Addressing bot limitations:-

Limitations imposed on bots can be linked to the technology behind the crawler, rules set by the website, or options chosen by person setting up the crawl. Working with the website’s development team, using the right crawler with appropriate crawl settings can remove many of the obstacles to crawling your pages and obtaining useful analyses.

2. Providing access to additional data:-

By nature, a crawl’s view of the website cannot include business or website performance data–key indicators for the marketing decision-making process–since these are not contained on the web page itself. Cross-analysis between crawl data and behavioral, ranking, or even business data can turn a crawl analysis into a fine-tuned decision-making tool.

Once you’ve overcome the limitations that made your crawl results less than reliable, you’ll find a wealth of insights to drive your SEO strategy.