

## DOCUMENTATION

- **“url\_fetch.py”:**

This code retrieves internal links from a website, and stores them in a text file, namely, “url.txt”. For this, I have created a set named “links” to store all the unique links retrieved from a URL (here, “<https://wireshark.org/>”). Analyzing the initial results I noticed that there were links like “/”, “/news” etc., i.e, they were not complete. Hence, while writing these links into the text file, I added the website’s URL in front of the incomplete links, in order to complete them (Line 21-22). Also, our work demanded the use of only internal links, hence I wrote only those links into the text file, that had “wireshark.org” present in them (Line 30). After analyzing the results again, I found that there were links that didn’t actually lead to valid websites, and had extensions like “.pdf”, “.zip”. Therefore, lastly I filtered out these links (Line 23 and Line 31), to make sure they are not written in the text file.

- **“Sequence.py”:**

The pcap files created by “*profile.py*”, are processed and decrypted in this code. Each pcap file is decrypted and filtered using filters “*tcp.dstport==443 && dns*” (for DoH packets), “*tls.record.length*” (for message size) and “*frame.time\_delta\_displayed*” (for inter arrival time between packets). Using these the DNS sequence for each pcap file is created and printed. Later, the code was modified and all these DNS Sequences were written to files saved as their URL numbers, and put into a folder named “Output”. There were 1000 files in this folder, and each had 10 DNS Sequences in it.

- **“automation.sh”:**

This bash script is written for the automation of the entire process. Firstly, the url.txt file is created for which “*url\_retrieve.py*” is called first. Next, “*firefox\_collect\_samples.sh*” is made executable and “*key.log*” is created and made executable to store the key for the pcap files and then “*profile.py*” is called since, that would capture traffic and create the pcap files for each url in “url.txt”. Lastly, “*Sequence.py*” is called which would create the DNS sequences for each pcap file.

### **Instructions to run the code**

- Firstly, all the files should be saved in the same directory.
- Open terminal and change directory to the one where all files are saved.
- Make “automation.sh” executable by, entering the following command:  
`$ chmod a+x automation.sh`
- Run the script:  
`$ ./automation.sh`

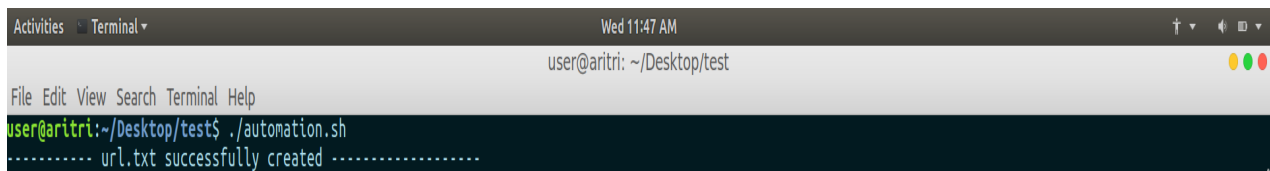
### **Links to the Dataset which was created after the execution of the following files:**

- Decryption Key:  
[https://drive.google.com/file/d/11gD9GH5rjLADLn5DuR\\_SNqt9I0-PIyzc/view?usp=sharing](https://drive.google.com/file/d/11gD9GH5rjLADLn5DuR_SNqt9I0-PIyzc/view?usp=sharing)
- Dataset 1 (pcaps of URLs 1-300):  
<https://drive.google.com/file/d/1rHCEYKsb31sRF6CTDXTOYF4nqWqHZsCd/view?usp=sharing>

- Dataset 2 (pcaps of URLs 301-700):  
<https://drive.google.com/file/d/17qogOXiNc6WGV3kYDXI0F5JOyY8DfFSd/view?usp=sharing>
- Dataset 3 (pcaps of URLs 701-1000):  
<https://drive.google.com/file/d/15coZnVXf61BfiqASQXP0-4SAVBkWLQbV/view?usp=sharing>

## Output

- After execution of the first command in the script, the following message would be displayed: “url.txt successfully created”.

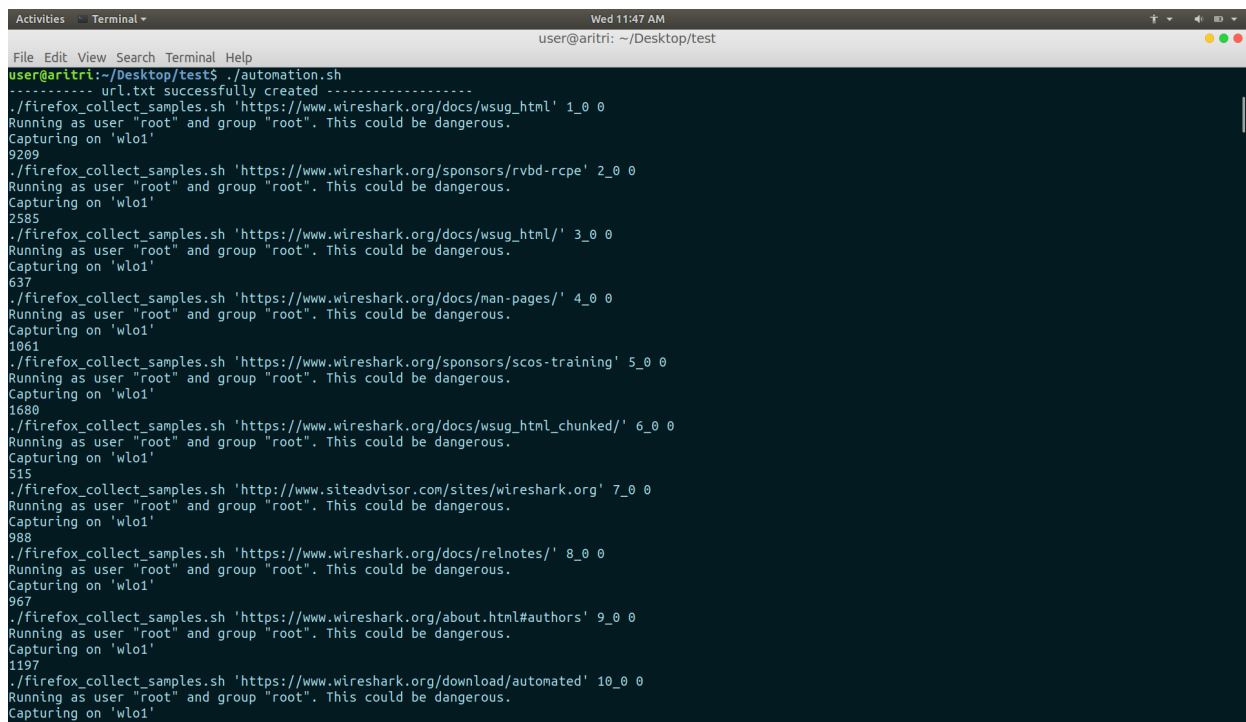


```

Activities Terminal Wed 11:47 AM
user@aritri: ~/Desktop/test

File Edit View Search Terminal Help
user@aritri:~/Desktop/test$ ./automation.sh
..... url.txt successfully created .....
```

- Next, “profile.py” would start execution and following commands would be displayed:



```

Activities Terminal Wed 11:47 AM
user@aritri: ~/Desktop/test

File Edit View Search Terminal Help
user@aritri:~/Desktop/test$ ./automation.sh
..... url.txt successfully created .....
./firefox_collect_samples.sh 'https://www.wireshark.org/docs/wsug_html/' 1_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
9209
./firefox_collect_samples.sh 'https://www.wireshark.org/sponsors/rvbd-rcpe' 2_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
2585
./firefox_collect_samples.sh 'https://www.wireshark.org/docs/wsug_html/' 3_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
637
./firefox_collect_samples.sh 'https://www.wireshark.org/docs/man-pages/' 4_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
1061
./firefox_collect_samples.sh 'https://www.wireshark.org/sponsors/scos-training' 5_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
1680
./firefox_collect_samples.sh 'https://www.wireshark.org/docs/wsug_html_chunked/' 6_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
515
./firefox_collect_samples.sh 'http://www.siteadvisor.com/sites/wireshark.org' 7_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
988
./firefox_collect_samples.sh 'https://www.wireshark.org/docs/relnotes/' 8_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
967
./firefox_collect_samples.sh 'https://www.wireshark.org/about.html#authors' 9_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
1197
./firefox_collect_samples.sh 'https://www.wireshark.org/download/automated' 10_0 0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
620
```

- After the capture of all the urls in url.txt are over, “sequence.py” would be executed, for which a url and its respective DNS sequence would be printed as follows:

```

Activities  Terminal  Wed 11:53 AM
user@aritri: ~/Desktop/test

File Edit View Search Terminal Help
Capturing on 'wlo1'
652
./firefox_collect_samples.sh 'https://code.wireshark.org/review/gitweb?p=wireshark.git;a=tree' 50_0_0
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlo1'
801

----- URL -----
https://www.wireshark.org/docs/wsug_html

----- DNS Sequence -----
['98', '0.000000000', '87', '0.000204792', '83', '0.000104904', '86', '0.000084540', '55,82,55,80,51,87,51,80,51,80,51,98', '0.077784803', '100', '0.174361494', '88', '0.091896301', '88', '0.128507703', '76', '0.109172738', '88', '0.273145741', '87', '0.040567251', '80', '0.335484573', '90', '0.484709836', '89', '1.037415379', '84', '1.193567752', '78', '0.000485515', '78', '0.000591613', '71', '0.000433081', '79', '0.000365570', '51,78,55,77,51,80,55,85,51,79,51,76,51,82', '0.079020643']

----- URL -----
https://www.wireshark.org/sponsors/rvbd-rcpe

----- DNS Sequence -----
['80', '0.000000000', '80', '0.000901249', '98', '0.002194074', '98', '0.011596961', '55,82,55,87,55,83,55,86,51,80,51,87,55,90', '0.064651187', '100', '0.195787828', '87', '0.070625928', '88', '0.077411011', '88', '0.145339945', '88', '0.337264245', '80', '0.725648200', '84', '2.273498963', '78', '0.000392363', '78', '0.000390030', '71', '0.000277333', '79', '0.000282558', '51,78,55,77,51,80,55,85,51,79,55,76,51,82', '0.082426989']

----- URL -----
https://www.wireshark.org/docs/wsug_html/

----- DNS Sequence -----
['81', '0.000000000', '81', '0.004273920', '98', '0.014783451', '98', '0.068108144', '83', '0.043404576', '80', '0.000787815', '80', '0.001034396', '87', '0.000709290', '100', '0.211158490', '87', '0.090925772', '88', '0.059904195', '88', '0.100669937', '88', '0.334850010', '80', '0.312790913', '76', '0.582180581', '84', '2.765055749', '78', '0.000097045', '78', '0.000084595', '71', '0.000068631', '79', '0.000064676', '51,78,55,77,51,80,55,85,51,79,51,76,55,82', '0.081139892']

----- URL -----
https://www.wireshark.org/docs/man-pages/

----- DNS Sequence -----
['93', '0.000000000', '93', '0.025914731', '98', '0.000438273', '98', '0.016731169', '100', '0.222023951', '87', '0.103379070', '88', '0.065870082', '88', '0.083014919', '88', '0.340088770', '80', '0.340035892', '83', '0.332299127', '87', '0.000460372', '80', '0.036779423', '87', '0.179864549', '76', '0.134988813', '84', '2.986727169', '78', '0.000378326', '78', '0.000308186', '71', '0.000261064', '79', '0.000300216', '51,78,55,77,51,80,55,85,51,79,51,76,55,82', '0.079846699']

```