IJARASEM

ISSN: 2395-7852

International Journal of Advanced Research in Arts, Science, Engineering & Management

Volume 11, Issue 6, November - December 2024

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.583

☎ +91 9940572462 ✆ +91 9940572462 ✉ ijarasem@gmail.com @ www.ijarasem.com

Efficient Enumeration of URLs of Active Hidden Servers over Anonymous Channel

Ch.Vasavi[1], S.Cherishma Sree[2], B.Chandu Ajay [3], P.Ram Vishal[4]

Assistant Professor, Department of Computer Science and Engineering, Anurag University, Hyderabad, Telangana, India[*1]

UG Student, Department of Computer Science and Engineering, Anurag University, Hyderabad, Telangana, India[*2,3,4]

ABSTRACT: This project presents an innovative approach to the efficient enumeration of hidden service URLs (.onion) on the TOR network using a custom-built graphical user interface (GUI) application. The tool leverages web scraping techniques, targeting the Ahmia search engine, to retrieve and analyze active hidden server URLs. Through the integration of Python's Tkinter library for GUI development and the use of requests and regular expressions for content mining, the application simplifies the process of discovering and visualizing .onion URLs. The tool allows users to input search queries, clean them for web compatibility, and display the results in real time. Additionally, the application provides graphical insights into the search results by generating visual data representations using matplotlib. This paper discusses the architecture, development process, and functionality of the tool, providing a new method for researchers to explore the TOR network.

KEYWORDS: TOR network, .onion URLs, hidden services, web scraping, Ahmia search engine, Tkinter, data visualization, network security, anonymous browsing

I.INTRODUCTION

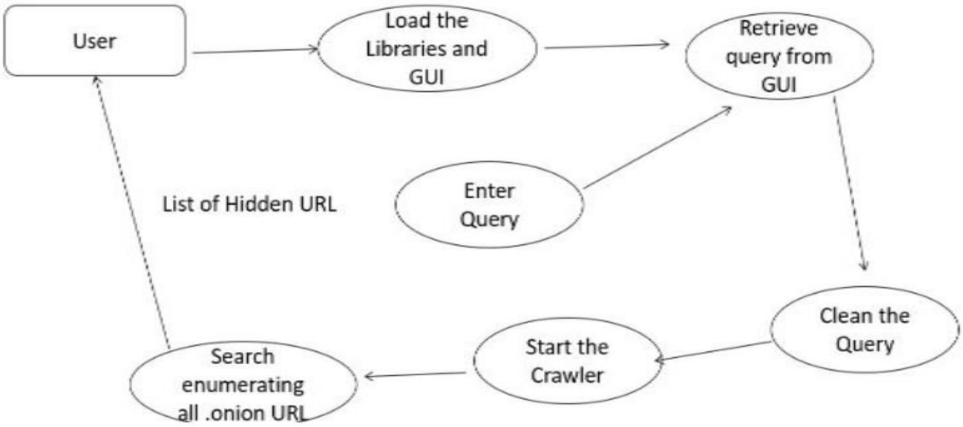
The TOR (The Onion Router) network is known for allowing anonymous communication and hosting hidden services, which can only be accessed through .onion URLs. These hidden services play an important role in protecting privacy, but they also present unique challenges for researchers, especially in the field of cybersecurity. Finding and listing these hidden services is crucial for understanding how the TOR network operates, but traditional search engines can't index .onion sites, making this process more difficult.

Ahmia is a search engine that focuses on indexing .onion websites, but manually searching through it can be time-consuming. This project aims to make that process easier by creating a simple graphical user interface (GUI) application that automatically searches for hidden service URLs. Using Python and web scraping techniques, the application interacts with Ahmia's search engine to retrieve results based on user input. The tool also includes data visualization features that allow users to see the results of their searches in a graph.

This project provides a straightforward way for researchers and cybersecurity professionals to explore the TOR network more efficiently, helping them discover active hidden services without the hassle of manual searching.

II.RESEARCH METHODOLOGY

The main focus of this project was to create a simple GUI-based application that can automatically find .onion URLs (hidden services) on the TOR network. The process involved several key steps: understanding the requirements, building the application, scraping the web for data, and visualizing the results.



1. **Understanding Requirements:** The first step was to identify the need for this project. Researchers and cybersecurity professionals often need a more efficient way to find .onion URLs, which led to the idea of developing a tool that could automate this process. The Ahmia search engine, which is specifically designed for searching .onion sites, was chosen as the best platform to scrape data from.
2. **Building the Application:** The application was built using Python's Tkinter library to create a simple and easy-to-use graphical interface. The interface includes:

- An input field for users to type their search queries.
- Buttons for cleaning the query, starting the search, generating graphs, and closing the app.
- A text area that displays the results (the .onion URLs) after a search is completed.

The idea was to keep the design straightforward so that anyone, regardless of technical background, could use it without difficulty.

3. **Scraping the Web:** The core of the project involved web scraping, where I used the requests library to send search queries to the Ahmia search engine. I added a User-Agent header to make the request look like it was coming from a web browser, which helped prevent the search engine from blocking the request. After getting the search results, I used regular expressions to extract the .onion URLs from the webpage. These URLs were then displayed in the text area of the app, and any duplicate URLs were removed.
4. **Visualizing the Results:** To make the results more meaningful, I added a feature that shows the number of .onion URLs found for each query in a bar graph. This was done using the matplotlib library. The graph helps users see how effective their search queries were at finding hidden TOR services, making it easier to compare results from different searches.
5. **Testing the Tool:** Once the application was built, I tested it with a variety of search queries to ensure it worked as expected. This included checking if the tool could properly find and display .onion URLs, handle any errors when a request failed, and create accurate graphs from the data

III.RESULTS AND DISCUSSION

After developing the application, I ran several tests to check how well it could find .onion URLs on the TOR network. I tried different search queries to see how effective the tool was, and the results gave some useful insights.

Search Results: The application worked as expected, successfully retrieving .onion URLs for every query I entered. In most cases, it was able to find multiple hidden service URLs from the Ahmia search engine and display them in the app. For example, when I used a query like "cybersecurity," the tool returned several active .onion sites related to that topic. Each search produced a unique list of URLs, which was also stored for later use in generating graphs.

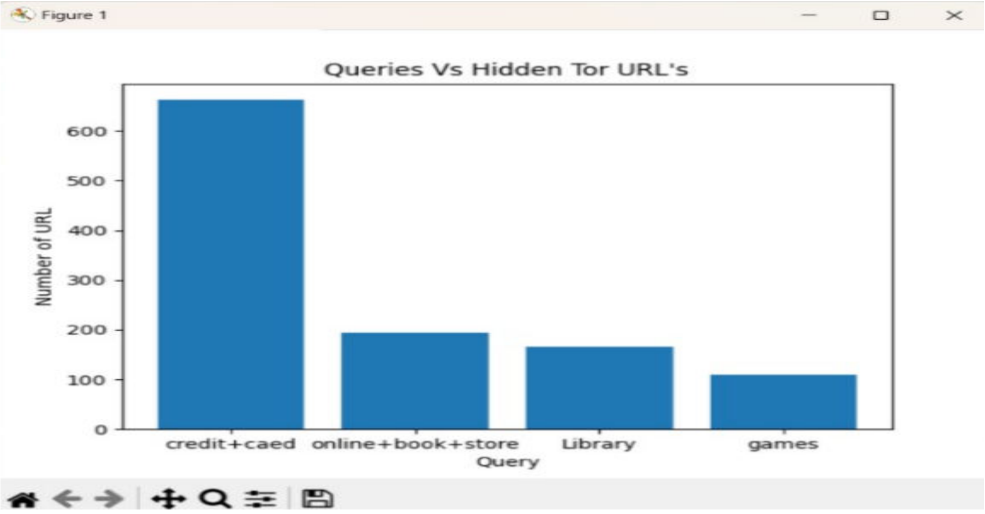
Graph Results: The graph feature turned out to be quite helpful in visualizing the search results. For every query, the tool created a bar graph that showed the number of .onion URLs found. This made it easy to compare how different search queries performed. For instance, more general queries like "marketplace" or "forum" brought back a larger number of URLs, while more specific queries such as "privacy tools" resulted in fewer. The graph provided a quick way to see which search terms were more effective at discovering hidden services.

Ease of Use: One of the goals of this project was to make the tool easy to use, and the tests confirmed that it achieved this. The simple design of the interface, combined with automatic search and graph creation, made it user-friendly even for people without a technical background. Users didn't need to manually visit websites or look through raw HTML, which saved a lot of time and effort when searching for hidden services on the TOR network.

Challenges and Limitations: While the tool worked well, there were a few challenges. The main issue is that the success of web scraping depends on the structure of the Ahmia website. If Ahmia changes how it displays search results, the tool may not be able to extract URLs properly. Some queries also returned fewer results because of the limited number of hidden services in certain categories. Additionally, since the tool relies on internet requests, occasional network issues or request failures could happen.

Future Improvements: There are a few ways the tool could be improved. One idea is to expand the search capability to include other TOR search engines beyond Ahmia, which could help find even more .onion URLs. Another improvement would be to make the tool more robust against errors, so it can handle failed requests or changes in the website's structure better.

IV.FORMATTING FIGURE



V.CONCLUSION

This project successfully created a simple and efficient tool for finding hidden service URLs on the TOR network. By using web scraping and a user-friendly interface, the tool allows users to easily search for .onion URLs through the Ahmia search engine. The ability to generate graphs also makes it easier to compare the effectiveness of different search queries.

Though the tool performs well, there are still some challenges, like its dependence on the structure of the Ahmia website and potential network issues. However, these can be addressed in future versions by adding support for more TOR search engines and improving how the tool handles errors.

In summary, this tool provides a practical solution for anyone looking to explore the TOR network, especially researchers and cybersecurity professionals. It cuts down on the time and effort needed for manual searches and makes the process of discovering hidden services more straightforward

DECLARATION

Study Limitations

This tool has a few limitations that affect its performance. It relies entirely on the Ahmia search engine, so any changes to Ahmia's structure or availability could disrupt the tool's functionality. Additionally, the tool's search results are limited to what Ahmia indexes, meaning it might miss other active .onion sites. It also depends on a stable internet connection and successful web requests, which can be unreliable. The effectiveness of the tool is also influenced by the specificity of the search queries, with more specific queries returning fewer results. Finally, users must consider the legal and ethical aspects of web scraping and working with anonymous networks like TOR.

Acknowledgements

We would like to extend our heartfelt thanks to everyone who supported us throughout the development of this project. Our sincere gratitude goes to our mentors and advisors, whose valuable guidance and constructive feedback were crucial in helping us overcome challenges and refine our approach. We are also grateful to the open-source communities, particularly Ahmia, for providing the resources that made this project possible. Their contributions to web security and the TOR network were integral to the success of our work. Lastly, we would like to thank our friends, families, and teammates for their constant encouragement, collaboration, and support during this journey.

Funding source

This project was developed independently and did not receive any external funding.

Competing Interests

We confirm that there are no competing interests involved in this project. The work was done purely for academic and research purposes, and there were no financial, personal, or professional factors that influenced the results or conclusions.

HUMAN AND ANIMAL RELATED STUDY

Ethical Approval

This project did not require formal ethical approval since it does not involve human or animal subjects. However, we acknowledge the importance of ethical considerations, especially regarding privacy and anonymity on the TOR network. The tool is intended for educational and research purposes, aimed at assisting cybersecurity professionals. We are committed to ensuring responsible use of the tool, emphasizing the need to respect the rights and anonymity of individuals using the TOR network for legitimate reasons.

Informed Consent

This project did not require formal ethical approval since it does not involve human or animal subjects. However, we acknowledge the importance of ethical considerations, especially regarding privacy and anonymity on the TOR network. The tool is intended for educational and research purposes, aimed at assisting cybersecurity professionals. We are committed to ensuring responsible use of the tool, emphasizing the need to respect the rights and anonymity of individuals using the TOR network for legitimate reasons.

REFERENCES

1.Antoine Amarilli, Pierre Bourhis, Louis Jachiet, Stefan Mengel “A Circuit-Based Approach to Efficient Enumeration” July 2017 ,44th International Colloquium on Automata, Languages, and Programming
2. Masashi Kiyomi, Yoshio Okamoto & Toshiki Saitoh “Efficient Enumeration of the Directed Binary Perfect Phylogenies from Incomplete Data” June 2012 , International Symposium on Experimental Algorithms
3. J. Salo, “Recent attacks on Tor.” 2010 B/11/papers/salo.pdf. Accessed:2020-05-10. [Online].
4. Esra Erdin, Chris Zachor, Mehmet Hadi Gunes “How to find hidden users: A survey of attacks on anonymity networks,” July 2015, IEEE Communications Surveys & Tutorials (Volume: 17, Issue: 4, Fourthquarter 2015)
5. Baby Shamini P,Sandhiya V Vibilleshnee U; Yamini S “Identification of URL Fuzzing and Subdomain Enumeration Using Raccoon Tool” June 2021, 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)
6. Alex Biryukov;,Ivan Pustogarov, Ralf-Philipp Weinmann “Trawling for Tor Hidden Services: Detection, Measurement, Deanonymization” June 2013 , 2013 IEEE Symposium on Security and Privacy
7. Zhen Ling, Junzhou Luo, Kui Wu, Xinwen Fu “Protocol-level hidden server discovery”July 2013 , 2013 Proceedings IEEE INFOCOM
8. Ahmed, Y.A., Kocer, B., Huda, S., et al.: A system call refinement based enhanced minimum redundancy maximum relevance method for ransomware early detection. J. Network Computer Appl. 167(102), 753 (2020)

International Journal of Advanced Research in Arts, Science, Engineering & Management (IJARASEM)

| Mobile No: +91-9940572462 | Whatsapp: +91-9940572462 | ijarasem@gmail.com |

www.ijarasem.com