



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH


IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 5, May 2024

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.423

 9940 572 462

 6381 907 438

 ijirset@gmail.com

 www.ijirset.com

Efficient Enumeration of URLs of active Hidden Servers over Anonymous Channel (TOR)

Chethan Kumar N, Shambhulinga, Surya M, Yashaswini K N, Dr. Nagesh R

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur,
Karnataka, India

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur,
Karnataka, India

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur,
Karnataka, India

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur,
Karnataka, India

Associate Professor, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur,
Karnataka, India

ABSTRACT: Efficient enumeration of URLs of active hidden servers typically refers to the process of discovering and listing the web addresses (URLs) of websites or servers that are hosted on the dark web or other anonymous networks. The dark web is a part of the internet that is intentionally hidden and requires specific software, such as Tor, to access. It is often associated with anonymity and privacy, making it a space where various types of content and services, legal and illegal, can be hosted. In the proposed approach we are developing a novel framework model using web crawling and DNS enumeration & subdomain enumeration.

The Onion Router , which is more commonly known as TOR is the most popular low latency anonymity network to date in search engine. Tor provides anonymity to users and supports the deployment of anonymous services network, known as hidden services. However, as the anonymity provided by Tor was available to everyone, it quickly became an accessory to cybercrime and other criminal activities , as well as a tool for terrorists to anonymously spread their propaganda everywhere. This forced Law Enforcement Agencies (LEA) and governments to find ways to break its anonymity. Tor being an anonymity network, the most common objective of a Tor attack is to de-anonymising its users and services through de-anonymisation attacks.

In response to de-anonymisation, pro-anonymity researchers attempted to strengthen users' expected anonymity by improving security and fixing known bugs. The Tor network also grew in size over the years. This growth, combined with the security improvements, played a vital role in securing the Tor network against most legacy de-anonymisation of attacks. We conduct an updated survey with a comprehensive overview of de-anonymisation attacks on the Tor network. Although there are some past surveys of Tor attacks, most consider all types of Tor attacks, while we have narrowed the scope to only include de-anonymisation attacks. Our survey covers about 30 more de-anonymisation attacks than most past surveys.

KEYWORDS: Enumeration, Hidden server, dark web, novel framework model, web crawling , DNS enumeration , subdomain enumeration, anonymous , cybercrime, de-anonymise, pro-anonymity.

I. INTRODUCTION

Over the past few decades, many online services have impacted the daily lives of Internet users. With that, a genuine concern has emerged as to how to browse the Internet while maintaining privacy. Privacy-preserving mechanisms over the Internet are all the more important for whistle-blowers and citizens of totalitarian governments, who are usually in dire need to protect their online identity. Other use cases of anonymous networks include sensitive communications of military and business organizations over the public Internet.

The above reasons have led to the research and development of anonymous communication systems. The early anonymity systems such as Mix-Net [3], Babel, and Mix minion were not widely adopted as they suffered from high latency issues and are now superseded by low-latency systems, as we now discuss.

The Onion Routing (TOR) services are designed to anonymised clients who wants to perform transaction or any other activity without disclosing their activities or in simple terms we can say TOR allows anonymous web browsing. Some malicious users are using this network to perform malicious activities which may harm normal or government entities.

To overcome from above issue we are developing a crawler which can crawl all TOR URLs running on specific terms or queries and by using this services normal or government peoples can know what all TOR URLs running actively.

The goal of this problem statement is to develop Proof of Concept (PoC) to enumerate URLs (.onion) of active hidden servers hosted over TOR. Teams are supposed to examine the cryptographic security controls and survey existing vulnerabilities in underlying security architecture of TOR network to develop PoC for efficient enumeration of URLs of active hidden services hosted over TOR.

Moreover, we discuss more than 15 de-anonymisation attacks published after 2016 (not reported in most of the previous works), including attacks that use advanced techniques such as deep learning. We also note that some survey do not include details on website fingerprinting attacks and hidden service attacks, which are important types of deanonymisation attacks. It is useful to have a well-defined taxonomy focusing on de-anonymisation attacks. In this regard, we present a new multi-level taxonomy to categorise de-anonymisation attacks on Tor. We use different discriminating factors at each level of the taxonomy to deliver a systematic categorisation.

We classify and discuss each attack, focusing on the Tor circuit component(s) used and the method of execution. We also try to draw conclusions on the practicality of those attacks. Finally, we provide insights into how Tor's research has impacted its development since its initial deployment. We highlight several significant milestones over the years that are relevant to de-anonymisation attacks and discuss how security improvements have made some of the previously possible attacks unfeasible. We hope that this work will be a valuable resource for anyone who wants to gain knowledge or engage in research in this area.

II. LITERATURE REVIEW

Efficient enumeration of active hidden server URLs over anonymous channels like TOR is a critical task with various implications, ranging from security to privacy and beyond. Here's a literature survey highlighting key works and approaches in this area:

1. "Crawling the Dark Web a survey" by Xiangwen Zhao et al. This survey provides an overview of various techniques and challenges involved in crawling the Dark Web, including strategies for efficient enumeration of active hidden services over TOR.
2. "OnionScan Practical Deanonymization of Hidden Services" by Sarah Jamie Lewis et al. This work presents OnionScan, a tool for scanning and analyzing the security posture of TOR hidden services. It sheds light on the methods used for discovering and enumerating active hidden services.
3. "Characterizing TOR Hidden Services" by Alex Biryukov et al. This paper focuses on characterizing the structure and behavior of TOR hidden services. It discusses methods for efficiently enumerating hidden services and analyzes their prevalence and distribution.
4. "Identifying Dark Web Resources" by Keyword Search by Tan, J. et al. This paper proposes a method for identifying Dark Web resources using keyword search techniques. It addresses the challenge of efficiently enumerating relevant hidden services based on user queries.
5. "Distributed Crawling of Darknet Markets" by Tomaž Pečar et al. This study explores distributed crawling techniques for Darknet markets, providing insights into the strategies for efficiently discovering and enumerating active hidden services within these markets.
6. "Towards a secure and efficient Darknet Search Engine" by Tomaž Pečar et al. This work presents a framework for building a secure and efficient Darknet search engine. It discusses approaches for indexing and querying TOR hidden services while preserving anonymity and privacy.
7. "A Study of Web Security on the Dark Web" by Kurokawa, Y. et al. This study investigates the security landscape of the Dark Web, including the enumeration and categorization of TOR hidden services. It discusses challenges and opportunities for improving security in this context.

8. “Analyzing the TOR Network: A Quantitative Analysis of Hidden Services and Their Content” by Simon P. Ahrens et al. This research provides a quantitative analysis of TOR hidden services and their content, offering insights into the distribution and characteristics of active hidden services.
9. “Crawling the Dark Web: A Case Study of Jihadist Forums” by Weimann, G. This study focuses on crawling and analyzing jihadist forums on the Dark Web, highlighting the challenges and methodologies involved in enumerating and monitoring such hidden services.
10. “Understanding the Dark Web: A Survey on DEEP Web Analysis” by Khattak, S. M. et al. This survey provides a comprehensive overview of techniques and tools for analyzing the Dark Web, including approaches for efficiently enumerating hidden services over TOR.

These works collectively contribute to our understanding of efficient enumeration techniques for active hidden server URLs over anonymous channels like TOR, addressing challenges related to security, privacy, and information retrieval on the Dark Web.

III. METHODOLOGY

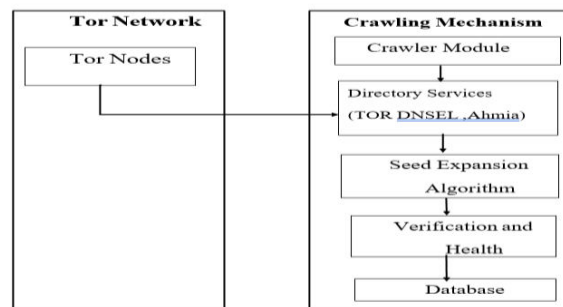


Figure 1: Flowchart

In this project, we are using an experiment based, quantitative research method to address the research problem, since proper analytics of large quantities of data requires efficient and effective machine learning applications.

The project was based on data collected from a malware behaviour analysis system, which was later transformed into a data set and subjected to machine learning algorithms to make the classification whether the malware was using the Tor network for its operations or not. To assess the performance of the algorithms, multiple evaluation procedures and metrics were used. The exact figures are presented in objectives section.

The malware behavior report database that contains "washed down" malware reports that were fetched from the Cuckoo sandbox machine via sync. A script of our project toolset was used to extract the features from the Cuckoo report files into the database. The database itself was used as training data for the classifiers.

The machine learning based classifier - a program that used the malware behaviour reports as training data to make classifications and identify .onion URLs from them. Writes the result in an output file. The Tor network consists of "onion routers" (OR) - servers that act as traffic relay nodes through the network.

We recognize three different types of relays: entry, exit, and middle ones. When connecting to the Tor network, these relays form virtual circuits (or paths), through which the traffic is routed.

The entry nodes designated as trusted (which entails they are stable and reliable for connectivity and conformity to the protocol) initiate the circuit, the middle relays (at least one for each triple) extend the circuit, and dedicated exit relays provide a connection to a destination server.

In addition to the ORs, there are nine central Directory Authority (DA) servers controlled by a number of selected trusted parties around the world .

3.1 METHODOLOGY OVERVIEW

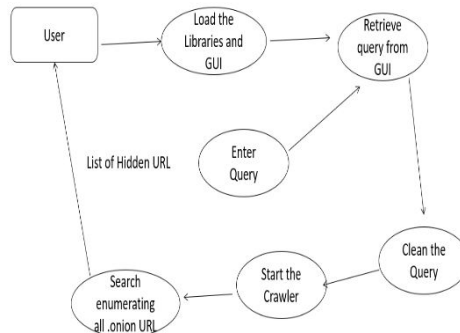


Fig 2:Data flow Diagram

ENTRY QUERY:

The "Enter Query" module serves as the entry point for users to input their query of interest, which can be any topic they wish to search for on the TOR network. Users may enter queries such as "Online Betting," "Credit Cards," "Online Banking Services," or any other topic. This module acts as the interface through which users interact with the system, initiating the process of TOR URL enumeration based on their input. It provides flexibility and accessibility for users to explore various topics anonymously on the TOR network, contributing to the overall usability and functionality of the system.

CLEAN QUERY:

The "Clean Query" module is responsible for processing and refining the user-entered query to ensure compatibility with TOR URL enumeration. This involves removing special characters, converting spaces to underscores, and formatting the query in a manner that can be effectively used for searching TOR URLs. By standardizing the format of the query, this module facilitates seamless integration with subsequent stages of the enumeration process. Additionally, it helps to enhance the accuracy and efficiency of TOR URL discovery by ensuring that the query is properly formatted and optimized for search operations within the TOR network.

ENUMERATE TOR HIDDEN SERVER URL's:

The "Clean Query" module is responsible for processing and refining the user-entered query to ensure compatibility with TOR URL enumeration. This involves removing special characters, converting spaces to underscores, and formatting the query in a manner that can be effectively used for searching TOR URLs. By standardizing the format of the query, this module facilitates seamless integration with subsequent stages of the enumeration process. Additionally, it helps to enhance the accuracy and efficiency of TOR URL discovery by ensuring that the query is properly formatted and optimized for search operations within the TOR network.

TOR URL GRAPH:

The "TOR URL Graph" module offers a visual representation of the relationship between the user-entered query and the number of TOR URLs enumerated. It generates a graph that depicts the query as the central node, with edges connecting to nodes representing the enumerated TOR URLs. This graph provides insights into the distribution and density of TOR URLs related to the query, allowing users to better understand the scope and prevalence of relevant content within the TOR network. By presenting information in a graphical format, this module enhances the interpretability and accessibility of the enumeration results, enabling users to navigate the TOR ecosystem more effectively.

User Interaction development with tkinter: tkinter a standard GUI library in Python, is used to develop the user interface. It provides widgets like buttons, labels, entry fields, and text boxes to create an interactive interface for users to input queries, view results, and interact with the application seamlessly.

Query Cleaning : The "Clean Query" function removes unnecessary characters and formats the user-entered query to make it compatible with TOR URL enumeration. It replaces spaces with the appropriate format (e.g., replacing spaces with plus signs) to ensure the query can be effectively used for searching TOR URLs.

TOR URL Enumeration : The "Enumerate TOR Hidden Server URLs" function sends a request to the AHMIA search engine (<https://ahmia.fi/search/>) with the user-entered query. It retrieves the HTML content of the search results page and extracts TOR URLs using regular expressions. This approach leverages an existing TOR search engine to efficiently enumerate active TOR hidden server URLs related to the user's query.

Visualization with matplotlib : Matplotlib, a popular plotting library in Python, is utilized to create a bar graph visualizing the relationship between user queries and the number of enumerated TOR URLs. The "TOR URL Graph" function generates a bar graph where each bar represents a query, and the height of the bar corresponds to the number of TOR URLs enumerated for that query.

Error Handling and User Feedback: The application incorporates error handling mechanisms to handle cases where the TOR URL enumeration request fails or returns unexpected results. It provides informative messages to the user through the text box to communicate the status of the enumeration process and any encountered errors.

Functionality and Usability : The application offers essential functionalities such as cleaning queries, enumerating TOR URLs, visualizing enumeration results, and exiting the application. These functionalities are presented in a user-friendly manner, allowing users to interact with the application intuitively and efficiently explore TOR hidden services related to their queries.

Overall, the project combines GUI development, web scraping, data processing, and visualization techniques to create a comprehensive tool for enumerating active hidden server URLs over the TOR network and presenting the results in a visually appealing and accessible manner.

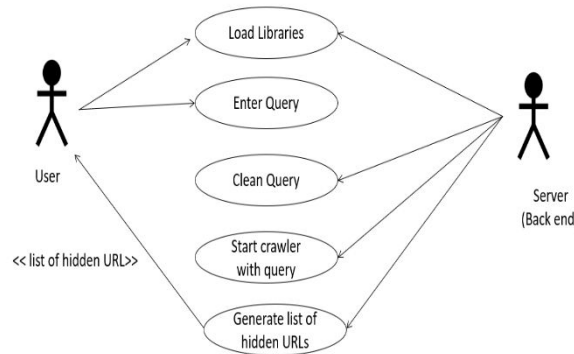


Fig 3 : Use case Diagram

Efficient enumeration starts with querying the TOR network or using TOR-specific search engines to discover hidden services relevant to the desired topic. This may involve crawling TOR websites, accessing TOR directories, or utilizing TOR-specific search APIs. Filtering the search results based on keywords or topics of interest helps narrow down the focus to relevant hidden services. This step reduces noise and ensures that only pertinent URLs are considered for enumeration. Once the search results are obtained, parsing the HTML content and extracting .onion URLs is essential. Regular expressions or HTML parsing libraries can be used to identify and extract the URLs from the retrieved content efficiently. Deduplicating the extracted URLs and validating their existence are crucial steps to ensure the accuracy of the enumeration process. Duplicate URLs should be removed, and validation checks should be performed to verify the availability and legitimacy of the hidden services. As the TOR network comprises a vast and dynamic ecosystem, scalability and performance optimization are essential for efficient enumeration. Utilizing parallel processing, asynchronous requests, and caching mechanisms can help improve the speed and scalability of the enumeration process. Throughout the enumeration process, maintaining user anonymity and privacy is paramount. Utilizing TOR proxies or routing requests through the TOR network ensures that enumeration activities remain anonymous and untraceable. Continuous monitoring and maintenance of the enumeration system are necessary to adapt to changes in the TOR network and ensure the ongoing accuracy and relevance of the enumerated URLs. Regular updates and adjustments to the enumeration algorithms and techniques may be required to keep pace with evolving trends and developments in the TOR ecosystem. Adhering to legal and ethical guidelines, such as respecting the privacy and security of TOR users and

adhering to TOR network policies, is essential. Ensuring compliance with relevant laws and regulations governing internet activities and data privacy is crucial for conducting enumeration activities responsibly.

3.2 Problem Domain, Problem Statement and Detailed Analysis

In this section, we provide some background information to explain our taxonomy and the attacks discussed in this paper. The Tor network, which is one of the most widely used anonymity networks today (along with other popular networks such as I2P and Freenet), has been using the concept of onion routing. Tor is an overlay network based on Transmission Control Protocol (TCP) that builds circuits from a user to the destination server, which generally consists of three voluntary relays. Figures 4 show the components of a Tor network for a standard circuit, and hidden services respectively

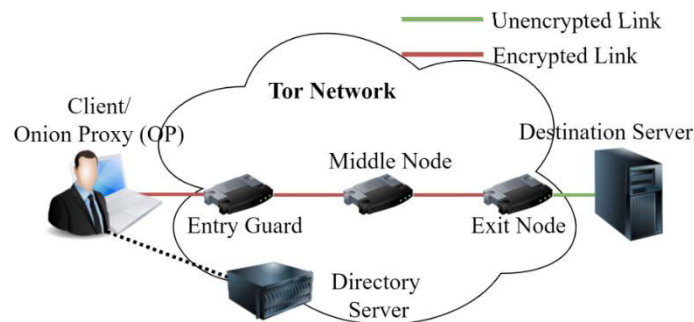


Fig 4: Component of TOR Network(standard tor circuit)

Problem Statement: In this project we first classified Tor attacks into four main categories based on the objective of the attack and explained those categories with examples. Following this, we elaborated on de-anonymization attacks with a taxonomy based on the components used for attack execution. We also provided insights into the evolution of these attacks over the years, and attempting to develop model which can prevent the attacks in enumeration of URLs.

Types of Attacks

Passive Attacks: In 2007 described one of the earliest attacks to de-anonymise Tor circuits. Their attack is carried out in two phases. In the 1st phase, the attacker needs to control a large number of Tor relays. Either introducing new malicious nodes into the network or hijacking existing nodes in the network can achieve that. In the earlier versions of the Tor network, the router could advertise an incorrect bandwidth and uptimes to the Directory Server (DS). These values were not verified by the DS or the OP when selecting that particular router for a circuit.

Required for this attack could be reduced by advertising false bandwidths for low bandwidth connections, thus increasing the chances of the adversary-controlled routers being selected as the entry or exit nodes of a circuit. Moreover, the malicious routers could have fewer restrictions on their exit policies to further increase the chances of being selected as exit nodes. If a circuit selects only one compromised relay, that relay can stop the traffic flow and force the circuit to rebuild with a different set of relays. This path disruption can be repeated until a target circuit selects two compromised routers as its entry and exit nodes. The next phase of the attack requires traffic correlation. In this phase, each malicious router in the network has to log information for each cell received, including its position in the circuit, local timestamp, previous connection's IP address and port, and next hop's IP address and port. A centralized authority that receives the above details from all malicious routers can execute a correlating algorithm to associate the sender with the receiver.

Active Attacks: In their attack, a malicious exit router modified the HTTP traffic to the client by inserting an invisible frame that contained a JavaScript code. The Tor user's browser executed this JavaScript code, which sent regular distinctive signals to a malicious web server. As the Tor client selects a new circuit at regular intervals to increase its anonymity, if one of the malicious entry guards is selected at a certain time, the attacker can use timing analysis to de-anonymise the user.

3.3 Objectives of the Proposed Work

- The project model checking was based on data collected from a malware behaviour analysis system,
- The analysis results which was later transformed into a data set and subjected to machine learning algorithms to make the classification whether the malware was using the Tor network for its operations or not.
- Analysis of URL enumeration process and patterns of processing.
- To assess the performance of the algorithms, multiple evaluation procedures and metrics were used. The decision tree (DT) and the logistic regression (LR) classifiers were used for the evaluation

IV. CONCLUSION

Efficient enumeration of URLs of active hidden servers typically refers to the process of discovering and listing the web addresses (URLs) of websites or servers that are hosted on the dark web or other anonymous networks. The dark web is a part of the internet that is intentionally hidden and requires specific software, such as Tor, to access. It is often associated with anonymity and privacy, making it a space where various types of content and services, legal and illegal, can be hosted. In the proposed approach we are developing a novel framework model using web crawling and DNS enumeration & subdomain enumeration.

Utilizing web scraping techniques, the project efficiently enumerates TOR hidden server URLs based on user queries. By querying the AHMIA search engine and parsing the HTML content, the project extracts .onion URLs related to the user's query, providing comprehensive results for exploration. The visualization of enumeration results through a bar graph offers insights into the distribution of TOR URLs across different queries, enabling users to identify trends and patterns within the TOR network. The graph provides a visual representation of the relationship between user queries and the number of enumerated TOR URLs, enhancing data interpretation and analysis. In conclusion, the project on efficient enumeration of URLs of active hidden servers over the TOR network provides a comprehensive solution for discovering and exploring hidden services while prioritizing user anonymity, privacy, and usability. Through the integration of various methodologies and considerations, the project offers a valuable tool for navigating the TOR ecosystem and accessing hidden content securely and efficiently.

Abbreviation

TOR – Tor Onion Router
 DNS – Domain Name System
 URL – Uniform Resource Locator
 OP – Onion Proxy
 GUI _ Graphical User Interface

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). <https://doi.org/10.1016/j.jnca.2020.102753>



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details