

# Darknet Traffic Analysis: Investigating the Impact of Modified Tor Traffic on Onion Service Traffic Classification

Dr.CHANDRA MOHAN

<sup>[1]</sup>Assistant Professor ,Malla Reddy Engineering College for Women (Autonomous Institution),  
Hyderabad,[imransk@gmail.com](mailto:imransk@gmail.com)

<sup>[2],[3],[4]</sup> Student , Malla Reddy Engineering College for Women (Autonomous Institution), Hyderabad,  
[jaanu10443@gmail.com](mailto:jaanu10443@gmail.com)<sup>[2]</sup>, [abhinaya2003sri@gmail.com](mailto:abhinaya2003sri@gmail.com)<sup>[3]</sup>, [sreemgav.52@gmail.com](mailto:sreemgav.52@gmail.com)<sup>[4]</sup>

## Abstract:

Classifying network traffic is important for traffic shaping and monitoring. In the last two decades, with the emergence of privacy concerns, the importance of privacy-preserving technologies has risen. The Tor network, which provides anonymity to its users and supports anonymous services known as Onion Services, is a popular way to achieve online anonymity. However, this anonymity (especially with Onion Services) is frequently misused, encouraging governments and law enforcement agencies to de-anonymize them. Therefore, in this paper, we try to identify the classifiability of Onion Service traffic, focusing on three main contributions. First, we try to identify Onion Service traffic from other Tor traffic. The techniques we have used can identify Onion Service traffic with >99% accuracy. However, there are several modifications that can be done to the Tor

traffic to obfuscate its information leakage. In our second contribution, we evaluate how our techniques perform when such modifications have been done to the Tor traffic. Our experimental results show that these conditions make the Onion Service traffic less distinguishable (in some cases, the accuracy drops by more than 15%.) In our final contribution, we identify the most influential feature combinations for our classification problem and evaluate their impact.

## INTRODUCTION:

Tor is an anonymity network that hides the identity of its users by routing the traffic through multiple intermediary nodes. Tor also supports the provision of anonymous services known as Onion Services (also known as hidden services) with onion as the top-level domain name. Tor's ability to act as a censorship circumvention tool has encouraged security experts, network defenders, and law enforcement agencies to identify Tor traffic from other encrypted

and non-encrypted traffic. For example, tried to classify Tor traffic from non-Tor Traffic, tried to classify the application types in Tor traffic, and tried to classify Tor traffic from other anonymity network traffic such as I2P traffic and Web-mix Traffic. However, in this work, we intend to explore the distinguishability of Onion Service traffic from standard Tor traffic using traffic analysis. We formulate three research questions to act as a foundation for our work. First, we try to answer the question A standard Tor circuit that is created to visit a web service on the Internet via Tor consists of three Tor nodes. An Onion Service circuit, which is the only way to access an Onion Service, consists of six Tor nodes. As the traffic in both these circuits (standard Tor and Onion Service) is encrypted, we assume that we can use the information leaked from the metadata (e.g. direction, timestamps, packet size) to identify unique patterns that can distinguish them. Onion Services have been used to host illegal websites, and more recently, they have been used as Command and Control (C&C) servers for botnets. Therefore, from the perspective of governments and law enforcement agencies, they want to track and shut down such services and regulate the Onion Service traffic. Even businesses might find it useful to restrict access to such websites in order to protect their

systems from potential bad actors (e.g. hackers) and attacks. As a result, having techniques for identifying Onion Service traffic can be useful for two main reasons;

1. Such techniques can act as a stepping stone for fingerprinting of Onion Services.
2. They can be useful to restrict Onion Service traffic in sensitive and confidential systems.

Second, we try to investigate the same problem under different settings. Specifically, we try. There are certain techniques that can be implemented in Tor to change its traffic patterns. Introducing padding [10], using dummy bursts and delays [11], and splitting the traffic [12] are a few examples of such techniques. These techniques<sup>1</sup> have been developed with the intention of obfuscating the information leakage of Tor traffic. The main importance of answering RQ2 is that we can confirm whether our findings from RQ1 will hold true as and when such modifications are introduced to the Tor traffic. If we are able to still distinguish Onion Service traffic, it is an indication that these modifications are not effective in masking Onion Service traffic, if they are realised in the future. If the modifications do affect the Onion service classifiably, it opens up questions about the validity of prior works, such as anon a setting with those modifications implemented. As outcomes of RQ2 can open up further research avenues on Tor

traffic classification, we argue RQ2 is worth evaluating.

## **Related works:**

### **“Tor: The second-generation onion router,”**

We present Tor, a circuit-based low-latency anonymous communication service. This second-generation Onion Routing system addresses limitations in the original design by adding perfect forward secrecy, congestion control, directory servers, integrity checking, configurable exit policies, and a practical design for location-hidden services via rendezvous points. Tor works on the real-world Internet, requires no special privileges or kernel modifications, requires little synchronization or coordination between nodes, and provides a reasonable trade-off between anonymity, usability, and efficiency. We briefly describe our experiences with an international network of more than 30 nodes. We close with a list of open problems in anonymous communication.

### **“Enhancing Tor’s performance using real-time traffic classification,”**

Tor is a low-latency anonymity-preserving network that enables its users to protect their privacy online. It consists of volunteer-operated routers from all around

the world that serve hundreds of thousands of users every day. Due to congestion and a low relay-to-client ratio, Tor suffers from performance issues that can potentially discourage its wider adoption, and result in an overall weaker anonymity to all users. We seek to improve the performance of Tor by defining different classes of service for its traffic. We recognize that although the majority of Tor traffic is interactive web browsing, a relatively small amount of bulk downloading consumes an unfair amount of Tor's scarce bandwidth. Furthermore, these traffic classes have different time and bandwidth constraints; therefore, they should not be given the same Quality of Service (QoS), which Tor offers them today. We propose and evaluate Dafter, a machine-learning-based approach that classifies Tor's encrypted circuits by application in real time and subsequently assigns distinct classes of service to each application. Our experiments confirm that we are able to classify circuits we generated on the live Tor network with an extremely high accuracy that exceeds 95%. We show that our real-time classification in combination with QoS can considerably improve the experience of Tor clients, as our simple techniques result in a 75% improvement in responsiveness and an 86% reduction in download times at the median for interactive users.

### **“Characterization of Tor traffic using time-based features,”**

Traffic classification has been the topic of many research efforts, but the quick evolution of Internet services and the pervasive use of encryption makes it an open challenge. Encryption is essential in protecting the privacy of Internet users, a key technology used in the different privacy enhancing tools that have appeared in the recent years. Tor is one of the most popular of them, it decouples the sender from the receiver by encrypting the traffic between them, and routing it through a distributed network of servers. In this paper, we present a time analysis on Tor traffic flows, captured between the client and the entry node. We define two scenarios, one to detect Tor traffic flows and the other to detect the application type: Browsing, Chat, Streaming, Mail, VoIP, P2P or File Transfer. In addition, with this paper we publish the Tor labelled dataset we generated and used to test our classifiers.

### **“Tor traffic classification from raw packet header using convolutional neural network,”**

As the amount of network traffic is growing exponentially, traffic analysis and classification are playing a significant role for efficient resource allocation and

network management. However, with emerging security technologies, this work is becoming more difficult by encrypted communication such as Tor, which is one of the most popular encryption techniques. This paper proposes an approach to classify Tor traffic using hexadecimal raw packet header and convolutional neural network model. Comparing with competitive machine learning algorithms, our approach shows a remarkable accuracy. To validate this method publicly, we use UNB-CIC Tor network traffic dataset. Based on the experiments, our approach shows 99.3% accuracy for the fractionized Tor/non-Tor traffic classification.

### **“Inferring application type information from Tor encrypted traffic,”**

Tor is a famous anonymity communication system for preserving users' online privacy. It supports TCP applications and packs application data into encrypted equal-sized cells to hide some private information of users, such as the running application type (Web, P2P, FTP, Others). The known of application types is harmful because they can be used to reduce the anonymity set and facilitate other attacks. However, unfortunately, the current Tor design cannot conceal certain application behaviours. For example, P2P applications usually upload and download files

simultaneously and this behavioural feature is also kept in Tor traffic. Motivated by this observation, we investigate a new attack against Tor, traffic classification attack, which can recognize application types from Tor traffic. An attacker first carefully selects some flow features, e.g., burst volumes and directions to represent the application behaviours' and takes advantage of some efficient machine learning algorithm to model different types of applications. Then these established models can be used to classify target's Tor traffic and infer its application type. We have implemented the traffic classification attack on Tor and our experiments validate the feasibility and effectiveness of the attack.

### **“Anonymity services tor, I2P, Mononym: Classifying in the dark (web),”**

Traffic classification, i.e. associating network traffic to the application that generated it, is an important tool for several tasks, spanning on different fields (security, management, traffic engineering, R&D). This process is challenged by applications that preserve Internet users' privacy by encrypting the communication content, and even more by anonymity tools, additionally hiding the source, the

destination, and the nature of the communication. In this paper, leveraging a public dataset released in 2017, we provide (repeatable) classification results with the aim of investigating to what degree the specific anonymity tool (and the traffic it hides) can be identified, when compared to the traffic of the other considered anonymity tools, using machine learning approaches based on the sole statistical features. To this end, four classifiers are trained and tested on the dataset: (i) Naïve Bayes, (ii) Bayesian Network, (iii) C4.5, and (iv) Random Forest. Results show that the three considered anonymity networks (Tor, I2P, Mononym) can be easily distinguished (with an accuracy of 99.99%), telling even the specific application generating the traffic (with an accuracy of 98.00%).

### **Methodology:**

Propose work consists of following modules

**Upload Dataset:** Using This Module, we will upload the dataset into the Application.

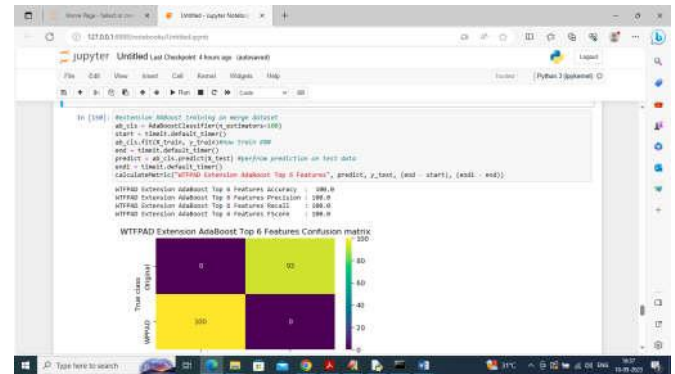
**Preprocess Dataset:** Using This Module We will dataset is pre-processed .

**Run Random Forest Algorithm:** Using This Module We will run the Random Forest Algorithm.

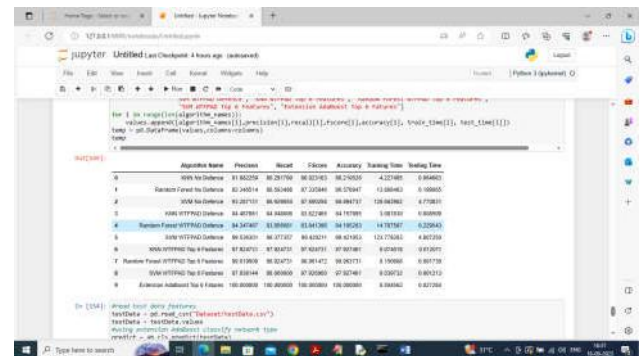
**Run SVM Algorithm:** Using This Module We will run the SVM Algorithm.

**Run KNN Algorithm:** Using This Module We will run the KNN Algorithm.

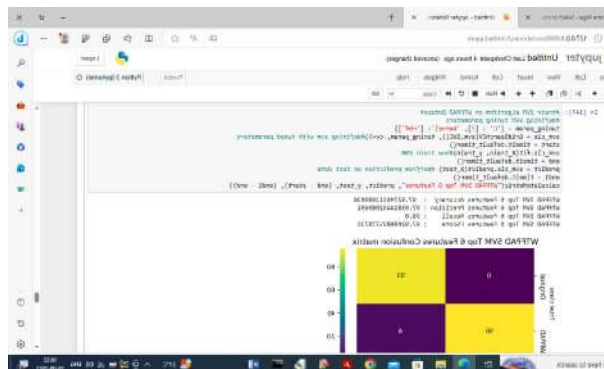
**RunBoost Algorithm:** Using This Module We will run the Boost Algorithm.



In above screen training extension ADABOOST algorithm and then after training extension got 100% accuracy

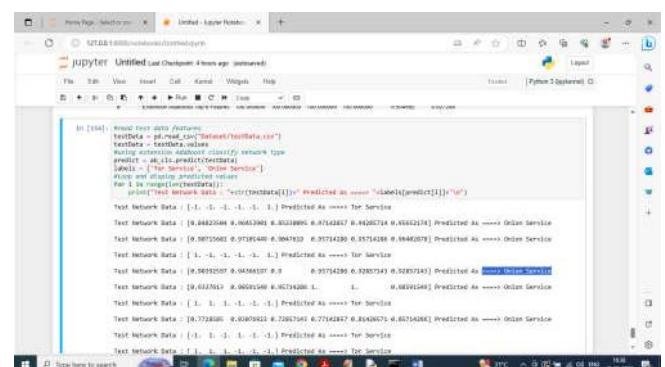


## RESULTS:



In above screen SVM got 97.92% accuracy

In above screen displaying all algorithm performance in tabular format and some algorithms can able to classify Tor and Onion Services with more than 95-99% accuracy



In above screen reading test Network data and then using extension ADABOOST algorithm

classifying network traffic as Tor or Onion services. in square bracket we can see test data and after arrow symbol can see predicted service type.

## CONCLUSION:

In this work, we answered three research questions focused on Onion Service traffic classification. We evaluated the applicability of supervised machine learning models to classify Onion Service traffic from other Tor traffic. We extracted fifty features from each traffic trace and used that feature set as input to the machine learning classifiers. Our results showed that KNN, RF, and SVM classifiers have the ability to distinguish Onion Service traffic from Tor traffic with a 99% accuracy. Then, we tried to identify whether state-of-the-art Website Fingerprinting defences affect the classifiability of Tor traffic. These defences introduce different modifications to try and obfuscate information leakage from traffic, and we evaluated how those changes affect the Onion Service traffic classification. Our experiments showed that the above classifiers, combined with our feature set, reduce the

performance for Onion Service traffic classification. However, we observed that the modified Tor traffic is still distinguishable. Moreover, we used three feature selection metrics, namely, information gain, Pearson's correlation, and Fisher Score, to identify the top features for this task. Those top features were able to provide >98% success for classifying Onion Service traffic from Tor traffic. However, they could not provide such good results when modified Tor traffic traces were used.

## REFERENCES:

- [1] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," in Proc. 13th USENIX Secure. Symp. (SSYM), San Diego, CA, USA, Aug. 2004, pp. 303–320.
- [2] M. Al Sabah, K. Bauer, and I. Goldberg, "Enhancing Tor's performance using real-time traffic classification," in Proc. ACM Conf. Compute. Common. Secure. (CCS), New York, NY, USA, Oct. 2012, pp. 73–84.
- [3] A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of Tor traffic using time-based features," in Proc. 3rd Int. Conf. Inf. Syst. Secure. Privacy (ICISSP), Porto, Portugal, Feb. 2017, pp. 253–262.

- [4] M. Kim and A. Anpalagan, "Tor traffic classification from raw packet header using convolutional neural network," in Proc. 1st IEEE Int. Conf. Know. Inno. Invention (ICKII), Jeju Island, South Korea, Jul. 2018, pp. 187–190.
- [5] G. He, M. Yang, J. Luo, and X. Gu, "Inferring application type information from Tor encrypted traffic," in Proc. 2nd Int. Conf. Adv. Cloud Big Data (CBD), Washington, DC, USA, Nov. 2014, pp. 220–227.
- [6] A. Monteiro, D. Cuonzo, G. Aceto, and A. Escape, "Anonymity services tor, I2P, Mononym: Classifying in the dark (web)," IEEE Trans. Dependable Secure Compute., vol. 17, no. 3, pp. 662–675, May 2020.
- [7] (May 2017). Wry Ransomware Analysis. Accessed: Apr. 26, 2023. [Online]. Available: <https://www.secureworks.com/research/wcryransomware-analysis>
- [8] (Jul. 2019). Keeping a Hidden Identity: Mirai C&Cs in Tor Network. Accessed: Apr. 26, 2023. [Online]. Available: <https://blog.trendmicro.com/trendlabs-security-intelligence/keeping-a-hidden-identity-mirai-ccsin-tor-network/>
- [9] (Nov. 2014). Global Action Against Dark Markets on Tor Network. Accessed: Aug. 4, 2020. [Online]. Available: <https://www.europol.europa.eu/newsroom/news/global-action-against-dark-markets-tor-network>
- [10] M. Juarez, M. Imani, M. Perry, C. Diaz, and M. Wright, "Toward an efficient website fingerprinting defence," in Proc. 21st Eur. Symp. Res. Compute. Secure. (ESORICS), Heraklion, Greece, Sep. 2016, pp. 27–46.
- [11] T. Wang and I. Goldberg, "Walkie-talkie: An efficient defence against passive website fingerprinting attacks," in Proc. 26th USENIX Secure. Symp. (SEC), Vancouver, BC, Canada, Aug. 2017, pp. 1375–1390.
- [12] W. De la Cadena, A. Mitzva, J. Hiller, J. Pennekamp, S. Reuter, J. Filter, T. Engel, K. Wehrle, and A. Panchenko, "Traffic Sliver: Fighting website fingerprinting attacks with traffic splitting," in Proc. ACM SIGSAC Conf. Compute. Common. Secure. (CCS), New York, NY, USA, Nov. 2020, pp. 1971–1985.
- [13] J. Hayes and G. Danezis, "k-fingerprinting: A robust scalable website fingerprinting technique," in Proc. 25th USENIX Conf. Secure. Symp. (SEC),



Austin, TX, USA, Aug. 2016, pp. 1187–1203.

[14] X. Bai, Y. Zhang, and X. Niu, “Traffic identification of Tor and web mix,” in Proc. 8th Int. Conf. Intel. Syst. Design Appl. (ISDA), Kaohsiung, Taiwan, vol. 1, Nov. 2008, pp. 548–551.

[15] O. Berthold, H. Federate, and S. Koppell, “Web Mixes: A system for anonymous and unobservable Internet access,” in Proc. Int. Workshop Design Issues Anonymity Unobservability, in Lecture Notes in Computer Science, vol. 2009, H. Federate, Ed., Berkeley, CA, USA, Jul. 2000, pp. 115–129.