De-anonymizing Web Traffic: A Survey

Shane O'Neill
University of Delaware
Newark, Delaware 19716
shaneo@udel.edu

ABSTRACT

Anonymous communication has been a goal of many since the birth of the modern web. There have been many proposed solutions over the past decade, and a very successful project known as Tor. Since its introduction to the academic work in the seminal paper *Tor: The second-generation onion router* [8], Tor has been the subject of an immense amount of scrutiny and research. Governments continue to try to de-anonymize Tor in an attempt to censor information. In this paper, I summarize the different attacks and solutions for obtaining privacy online. This survey primarily covers three fields of research into internet anonymity: traditional internet architecture, Tor, and newly proposed software solutions and computer networks.

KEYWORDS

Computer Networks, Privacy, Tor, Anonymity, De-anonymization

ACM Reference format:
Shane O'Neill. 2017. De-anonymizing Web Traffic: A Survey. In *Proceedings of Rui Zhang's CISC859: Advanced Topics in Communications, Distributed Computing Networks: Network Security, University of Delaware, Spring'17,* 7 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

At its core, the subject of web traffic anonymity is a response to internet censorship. It is often not enough to simply encrypt the contents of messages, but necessary to keep communicating parties secret as well. Internet censorship is a very real problem. Whether for political or commercial gain, governments around the world restrict the flow of information in and out of their respective countries — usually by way of an all encompassing firewall, access to certain websites like Google or YouTube is highly restricted. Competing news websites can be blocked, creating a dangerous political environment. Information about world events can be selectively cherry picked in an Orwellian fashion to control the narrative of good and bad.

Rui Zhang's CISC859: Advanced Topics in Communications, Distributed Computing Networks: Network Security, Spring'17, University of Delaware

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

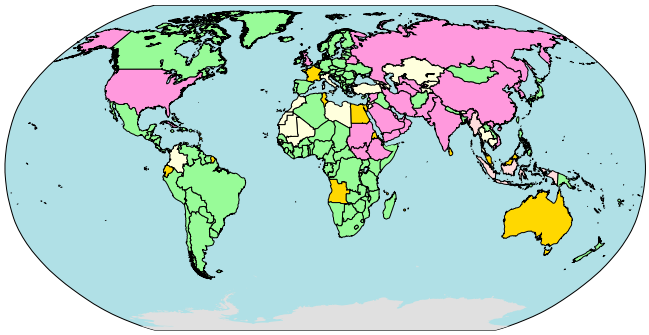

Figure 1: Image showing the levels of internet censorship in the world. Pink areas are subject to pervasive levels of censorship. White areas have substantial levels. Yellow and Green have little to none [7].

Several key papers such as *Fingerprinting websites using traffic analysis* [10] showed the vulnerable nature of the current internet architecture. The bottom line is that the internet was simply not designed with privacy in mind.

In an effort to combat this, users have created special protocols and applications to hide what websites and other users they communicate with. The most popular piece of software that is a direct response to internet censorship is Tor. Tor is a volunteer-driven computer network that conceals the identities of communicating parties [21]. With over two million daily users as of the time of this writing, Tor is by far the most popular solution for obtaining browsing anonymity [22]. Its popularity has inspired whistle-blowers to act, new research to grow, and information to spread freely. This popularity is not without a price, however. Since its launch in 2002, Tor has been the subject of malicious attackers wanting to de-anonymize and control its network, as well as governments attempting to subjugate its volunteer-based information relays.

Lastly, not only is it important to protect user privacy, it also important to protect service privacy. The location of services like website and IRC channels should be able to remain secret in response to potential information censorship. This is an equally challenging goal to which the only real solution currently available is through the use of Tor. However, there have been studies conducted that argue that the current Tor configurations are not enough [4] [3] [13] [19].

2 TERMINOLOGY

Terminology for this area of research is quite varied. There are many words that are used interchangeably even though they refer to the same concept.

Confidentiality The notion that the information being exchanged is kept secret from unauthorized parties.

Privacy Often confused with confidentiality, Privacy is the notion that the information (whether or not it is kept secret) cannot be traced back to a particular user.

Traffic Any exchange of packets over a computer network.

Fingerprinting The act of mapping web traffic to a particular user.

Virtual Private Network (VPN) A common service offered that acts as a proxy for internet requests. The idea behind VPNs is that anyone looking to snoop on a user's traffic will simply see them sending and receiving requests to the proxy (*e.g.* vpnservice.com), instead of the actual service (*e.g.* google.com).

Service A physical server somewhere that is connected to the interface and offering a service to users. This can be a website, SSH server, FTP server, IRC server, etc [19].

Hidden Service A server whose IP Address, and subsequent physical location is kept secret from those using the service.

The Tor Project The company founded in 2006 that develops and maintains Tor software [8].

The Tor Network The Tor Network is a group of volunteers that communicate with one another using a special Tor protocol (Onion Routing) and the *Tor Client*. Information is passed along *Relays* in order to keep information and the identities of communicating parties a secret [8].

The Dark Net A term that refers to sites that are not indexable by search engines. Sites whose IP Addresses are kept hidden (See: *Hidden Services*). This term has been widely exaggerated to mean illegal computer hacking networks or websites. It has since fallen out of popularity.

Tor Client Also known as the Tor Browser. This is the official Tor software client that one can use to browse the internet using the Tor protocol.

Relay Another piece of software that volunteers can run to help anonymize communicating parties. The Tor Network is dependant on volunteers running relays. Entry Nodes and Exit Nodes are relays that are the last in the chain of communication. See section 3.2.1.

Node Another name for a relay.

Entry Node The first node that a Tor Client connects to when a request has been made. Important node because it will know the identity of the request. This is sometimes known as a *Guard Node*.

Exit Node The last node before the true destination. The node that actually will make the request to the server (*i.e.*, GET google.com). Important node because it can be used in timing attacks to identify the true requester. This is called *Exit traffic tampering*.

Circuit A sequence of relays that a message will travel through inside the Tor network. Through clever usage of Private/Public keys, each relay in the circuit only ever knows where data is coming from, and the next relay to send it to.

Next hop In a Tor relay circuit $[1, 2, 3, ..., k, ...n]$, from any particular relay k that is in the chain, the next hop is simply relay $k + 1$.

Previous hop In a Tor relay circuit $[1, 2, 3, ..., k, ...n]$, from any particular relay k that is in the chain, the next hop is simply relay $k - 1$.

Imposter Relay Usually used in the same context of a Sybil relay. These relays are malicious and usually are duplicated relays (Multiple Tor Relays running on the same physical computer). See *Sybil*.

Sybil Attack An attacker obtains a large amount of influence over the network and uses this influence to gain some advantage, or exploit some assumption. In the context of Tor, an attacker can spawn multiple Tor relays on a single address. This allows the attacker to have a higher chance of being a selected relay for any circuit, including entry and exit nodes [25].

3 RESEARCH CHALLENGES

Generally, research in this field attempts to answer three questions:

(1) Is there a current vulnerability in privacy preserving techniques? And how is it performed?
(2) To what extent is the accuracy involved in mapping a user to his web activity, including:
   - True positive
   - False positive
   - True negative
   - False negative
(3) How can this vulnerability be solved?

Answering these questions is extremely challenging. If a vulnerability is found, it is hard to test the impact. Often times, the vulnerability can be studied only in large, real systems (The internet, Tor, etc), and can not be realistically simulated. Testing for accuracy is even harder, as that requires setting up an experiment that is comparable to that of a large network. Often times, it is easy to show promising experimental results in simulated computer networks, but testing against actual systems is unreliable at best.

3.1 "Traditional" Ways

Research that falls under this category is research that does not include drastic changes to the current internet architecture. Typically, this means that the author is only concerned with existing network stacks, protocols (HTTPS, etc), and techniques such as VPNs. And seeks to show the security, or lack thereof, surrounding them.

The most common technique for internet privacy is the use of a VPN service. The market for internet privacy continues to grow [11]. Dozens of companies offer high quality VPN software and services for users to hide their internet usage. There is a fair amount of research suggesting that the use of VPNs is often not enough for privacy. With clever analysis of the amount of data being sent and when, an attacker can match a user to a website with great accuracy [10].

3.2 Investigating and Securing Tor

Tor is largely a response to website fingerprinting. The goal of it to conceal what websites a user is viewing and of course the data being sent and received. Its popularity is evidence for the strength of Tor. And it is the best example of a volunteer based computer

network for the goal of privacy that really works. As such, it is the subject of a lot of research and scrutiny.

3.2.1 The Tor Protocol. Tor was the subject of landmark paper *Tor: The Second-Generation Onion Router*. This paper launched Tor into the academic spotlight and currently has over 3400 citations. In the paper, Roger Dingledine, Nick Mathewson, and Paul Syverson describe the goals of Tor, how it works, and its defense against attacks [8]. For the sake of this paper, I will briefly summarize how Tor works.

Tor works by way of whats called Onion routing. Encrypted data is sent through a series of relays until it reaches its destination, to which it is then sent back through a series of relays which may or may not be the same as before until it reaches the original sender. The sender starts by packaging their request by encrypting them with public key cryptography of every relay along the chain. When data arrives at a given relay, a "layer" of the packet is unencrypted by the relay and the next hop is revealed. The relay then sends the packet to the next hop and the process continues. In essence, this type of routing creates a system where no relay can simultaneously know both (1) the final destination of the relay chain, and (2) the original sender of the request. Entry nodes can know (1), and exit nodes will know (2), but these cannot be the same node.

Another core feature of Tor are hidden services. Briefly, hidden services are servers that do not reveal their IP Address and other information to users. Their mechanism is complicated but there are a few key components that are more important for attacks then the rest of the protocol:

Hidden service The service in question. Usually a website, IRC channel, file server, etc.
Client The client connecting and using the hidden service.
Introduction Points (IP) Relays chosen by the hidden service that act as âĂIJgatewaysâĂİ from the Rendezvous point.
Hidden service directories (HSDir) Relays that hold information about Introduction points so that clients can find them.
Rendezvous Point (RP) A relay chosen by the Client which forwards all data between the Client and hidden service.

The majority of attacks on hidden services involve compromising an Introduction Point, Hidden service directory, or Rendezvous Point.

3.2.2 Sybil Attacks. Sybil attacks are very strong attacks in any reputation based system. Tor relays are run by volunteers. Anyone can host a real relay and be part of relay chains in transferring data. Ideally, each relay will correspond to exactly one user. This means that every user has an equal chance of being selected to be a member of a relay chain. This will have the strongest anonymity because no one person will have ownership of both an entry node and exit node for any given relay chain. On the opposite hand, let us imagine a hypothetical where one user controls every relay. This would certainly remove all privacy because the attacker could control every relay chain and be able to map a senders request through a chain, to their exit node, and to the destination. Intuitively, the more relays any one user has, the higher probability that their relays will be chosen for more relay chains. And potentially many of their own relays in a single chain. Generally, in fingerprinting, Sybil attacks are not enough on their own, but the use of Sybils makes almost every attack stronger. This is important because lots of the attacks outlined in section 4 depend on having control of one or more relays. Having many Sybils in the network will make this more likely to happen.

3.3 New Protocols and Software

Lastly, research into traffic fingerprinting can be involved with new software, protocols, and networks. These papers are less common than others, and generally receive less attention. This is a much more difficult area to conduct research in because any newly proposed solution to internet privacy must come with a strong argument on why it is better than our current solutions. This is not to discourage research in this area, as landmark research often takes the form of something completely new and creative [8].

4 SELECT PAPER SUMMMARIES

Fingerprinting websites using traffic analysis, Andrew Hintz 2002.[10]. This paper was one of the first papers to explore proxy services that claim to provide internet privacy. SafeWeb is the service analyzed in this paper, the author proposes the concept of fingerprinting websites. Typically, when a user visits a website, they download an HTML file, usually some javascript files, and images. For example, visiting www.cnn.com might have you download 40 files, each of these with a unique size. TCP connections will be opened and the files will be fragmented and sent back to the user. This is a deterministic event. Any user visiting www.cnn.com will download those 40 files, open a TCP connection, and chunk the data back to the user. The deterministic nature of this event allows one to construct a *fingerprint* for www.cnn.com. As in, if this amount of data, in this fragment configuration, and in this time frame is sent to a user, it is likely that the user had just visited www.cnn.com. If we wanted to know if a user was visiting www.cnn.com, even if this user was behind a web proxy like SafeWeb, all we would need to do is monitor the data being sent from SafeWeb to the users computer for the www.cnn.com *fingerprint*. This paper is important because it shows that using a proxy service alone is often not enough for privacy.

Peek-a-boo, i still see you: Why efficient traffic analysis countermeasures fail, Dyer, Kevin P and Coull, Scott E and Ristenpart, Thomas and Shrimpton, Thomas 2012.[9]. A great expansion on the previous paper, *Peek-a-boo* shows that not only do proxy services fail to provide anonymity, any traffic tunneling technique is not enough to protect a user's privacy. Every tunneling technique, like SSH tunneling, is vulnerable to *traffic analysis*. Countermeasures such as purposely altering the length of packets was shown to still not be enough in the face of statistical traffic analyses techniques. In fact, this paper covers 9 separate "traditional" countermeasures in the face of traffic analysis and shows that they all are incomplete defenses.

Tor: The second-generation onion router, Dingledine, Roger and Mathewson, Nick and Syverson, Paul 2004.[8]. This was a landmark paper and the introduction of Tor to the academic world. It covers design goals, mechanisms, modifications of previous routing protocols, real world examples, and even potential attacks and defenses.

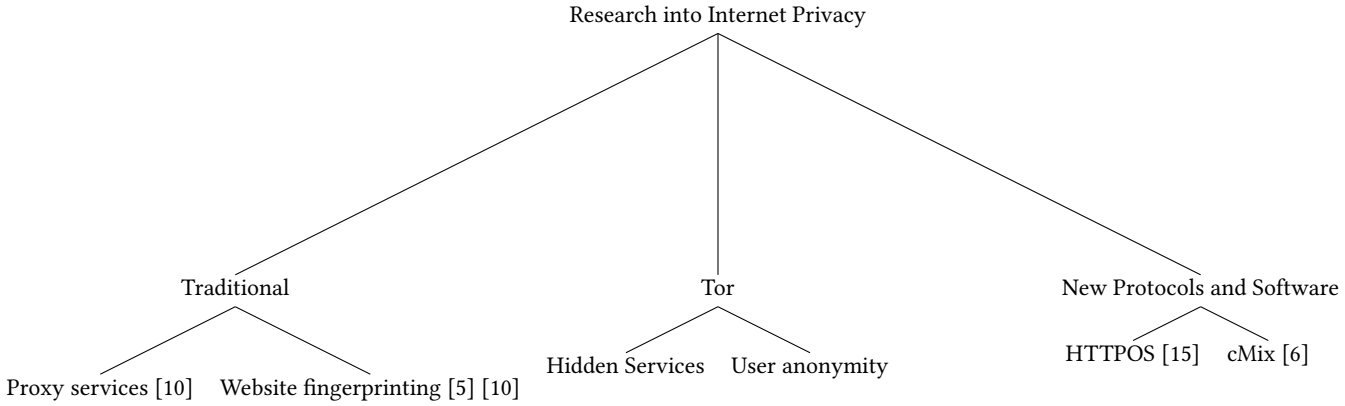

Figure 2: Taxonomy: The most common areas of research in internet privacy

It has been cited thousands of times and sparked academic research into Tor. This is an important paper in terms of internet anonymity as every other paper concerned with Tor most likely references this paper.

Shining light in dark places: Understanding the Tor network, McCoy, Damon and Bauer, Kevin and Grunwald, Dirk and Kohno, Tadayoshi and Sicker, Douglas 2008. [16]. This was the first paper to really investigate the use of Tor. It answers questions like "What websites are users visiting?", "How many users does a typical exit relay serve?", "Is Tor being used more for large data transfers or normal web browsing?" This is largely a statistical paper, and although it does not concern itself with user fingerprinting, it showed the power that exit nodes hold inside the Tor network. This fact became inspiration for a variety of attacks on users privacy.

Low-cost traffic analysis of Tor, Murdoch, Steven J and Danezis, George 2005.[18]. One of the first attacks against Tor was outlined in this seminal paper. It is a fact that for any relay, there must be multiple different users sending data through the node for it to provide privacy at all. In turn, this means that multiple users consume resources on any one machine. The more users that are using the relay, the most resources will be consumed. Resources consumed affects the latency of the connection. It is possible to observe this latency and apply any statistical methods to determine a variety of information about the connection. Along with this simple fact, an attack controlling an exit node can (1) measure the latency of the tor network, and (2) use that information to map users to requests. For example, if malicious exit node A knows that the latency of its circuit is 500ms, but a normal circuit is only 100ms, the attacker could know if someone is using their particular exit node based on measuring how much time passes from when a request first enters the Tor network and when it reaches the compromised exit node. The attacker would then know that the request belongs to that particular user.

Low-resource routing attacks against tor, Bauer, Kevin and McCoy, Damon and Grunwald, Dirk and Kohno, Tadayoshi and Sicker, Douglas 2007.[2]. 2 years after the previous paper, and several updates to Tor, another timing attack was outlined that exploits Tor's relay chain optimization mechanisms. It works in a similar fashion to the 2005 attack: Gain information about the latency in the Tor network. Compare the latency to exit node statistics. This allows you to map users to exit nodes, and subsequent requests to the real destination.

Locating hidden servers, Overlier, Lasse and Syverson, Paul 2006.[19]. Another seminal paper that is one of the first to investigate hidden services. A malicious attacker can connect to a hidden service as a Client, while controlling the relay whose next hop is the hidden service. It is possible to perform certain traffic patterns from the client node, and observe them from the compromised relay. If this is the case, the attacker knows that his/her client is using his/her relay in the circuit connecting to the hidden service. Thus, the attacker can determine lots of information of hidden service directories, rendezvous points, and even the hidden services IP Address in some cases.

Hot or not: Revealing hidden services by their clock skew, Murdoch, Steven J 2012.[17]. A similar paper to the previous one. However, this one uses a creative technique of measuring a computers clock skew. Often hard to measure, clock skew is the slight timing difference between an electronic signal and the different components of a computer. By measuring the clock skew and temperature of the hidden service, we can correlate a user to requests even though they travel through the Tor network.

Identifying and characterizing Sybils in the Tor network 2016.[25]. One of the first papers to elaborate on Sybils in the Tor network. Sybils are an extremely strong adversary, and their presence can be used to make every attack stronger by making it more likely that an attacker is the owner of any particular relay. This paper is as much a survey paper as it is a research paper. The authors created a program they call Sybilhunter to parse past datasets and new datasets they collect to try to identify malicious relays. Relays are deemed malicious if they tamper with response data (HTML tampering), or forward the request to an unrecognized IP Address (Man-in-the-middle attack). This paper is one of the most interesting papers in Sybil research as it categorizes the different outstanding attacks currently unfolding in the Tor network.

Effective Attacks and Provable Defenses for Website Fingerprinting. 2014.[23] and Toward an efficient website fingerprinting defense,Juarez, Marc and Imani, Mohsen and Perry, Mike and Diaz, Claudia and Wright, Matthew 2016.[12]. Similar to the paper *Peek-a-boo, i still see you: Why efficient traffic analysis countermeasures fail*, these paper show that SSH-tunneling, or Tor, is often not enough for anonymity. Here, however, the authors present convincing defenses. The outlined attacks are similar to previously defined ones: statistical analysis of the timings of requests and responses allow users to be mapped to requests to websites. By obfuscating our requests with other "noise" packets, we can effectively hide our real requests. Though, a new problem is introduced with this solution: Obtaining a desired level of anonymity, while maintaining the lowest possible bandwidth overhead becomes a hard problem.

(nothing else) MATor (s): Monitoring the Anonymity of Tor's Path Selection (2014) by Backes, Michael and Kate, Aniket and Meiser, Sebastian and Mohammadi, Esfandiar [1]. Along with having a creative name, MATor is system to derive sender, recipient, and their related anonymity based on Tor's path selection algorithm. An extremely complex paper, MATor is unique because it shows that three relays for a tor circuit is not enough to provide anonymity. The paper offers *DistrbuTOR*, a new path selection algorithm that improves privacy, but at the cost of additional bandwidth in the Tor network.

Trawling for tor hidden services: Detection, measurement, deanonymization (2013) by Biryukov, Alex and Pustogarov, Ivan and Weinmann, Ralf-Philipp [4]. This paper is extremely intuitive in its hidden service attack. It capitalizes on the deterministic nature of the Client-Hidden Service handshake protocol. To recap, a client introduces itself to an Introduction Point for the hidden service. Next, The Introduction Point contacts the hidden service with the client's information and a 3 node circuit is established between the hidden service and the client. All communication between the client and hidden service is through this circuit with the Rendezvous Point being the first node after the service. During this protocol, a fixed number of inter-relay organizational packets are sent to coordinate and establish the circuit. The deterministic nature of this protocol allows an attacker to know definitely if he or she controls node in the circuit. If an attacker determines that his node happens to be the Rendezvous Point (by counting the fixed number of circuit-establishing packets), then he/she knows that the previous IP address must be the IP address of the hidden service. To combat this, the authors suggest the addition of trusted guard nodes that can as proxies to the hidden service. They would have increased vetting for their selection. However, even this is suspected to not be enough. To truly protect against this attack, the number of communication packets sent while establishing a circuit must be non-deterministic. This would prevent the attacker from being able to determine what node they possess in the circuit.

Content and popularity analysis of Tor hidden services (2014) by Biryukov, Alex and Pustogarov, Ivan and Thill, Fabrice and Weinmann, Ralf-Philipp [3]. A follow-up paper with two of the authors of the previous paper. This paper uses the technique for the previous to gather information about onion addresses and hidden services. It is proof that there is a serious vulnerability concerning hidden services in Tor. Not only that, the authors show that you can identify clients of a given hidden service if you own the node that gets selected to be a hidden services directory. This paper also is the most revealing paper so far on the actual uses of hidden services. Tor has gotten a reputation as being a haven to drug dealers and other criminals, but advocates of Tor claim that the majority of services are responses to internet censorship. This paper definitively shows that both types of services exist. They capture real Silk Road (Drug marketplace) addresses, but also real DuckDuckGo (Anonymous search engine) addresses. Their collection of hidden service addresses is far from exhaustive, but if someone were so inclined, they could employ these techniques to form a most complete hidden service address pool. Shortly after this paper was published, the Silk Road was compromised and shutdown by the United States government.

Circuit fingerprinting attacks: Passive deanonymization of tor hidden services (2015) by Kwon, Albert [13]. A continuation of previous papers on hidden service deanonymization. However, this paper assumes a different attack model than the previous papers examined. To recap, other attacks have one commonality between their attack models: An attacker has to continuously attempt to connect to the hidden service in order to monitor the information of the relays that are under their control. This paper is different. Similar to *Trawling for tor hidden services...*, the author takes advantage of the deterministic nature Tor exhibits when a connection is established between a client and service. The process of connecting to a hidden service is different than normal services. It's *fingerprint* (see: [10]) is recognizable. With this in mind, we can determine if a particular user is using a hidden service or not. Next, we can apply normal website fingerprinting techniques to de-anonymize the service/client relationship. The author offers a short defense in the form of added "noise" into Introductory Points. This is a similar response to website fingerprinting (HTTPOS, etc). However, this would add additional bandwidth to the network.

Website fingerprinting at internet scale (2016) by Panchenko, Andriy and Lanze, Fabian and Zinnen, Andreas and Henze, Martin and Pennekamp, Jan and Wehrle, Klaus and Engel, Thomas [20]. This paper is an extension of other website fingerprinting research. The authors of this paper provide a training set of over 300,000 different fingerprints that include pages other than indexes, a common assumption of the previous website fingerprinting papers. The authors use a combination of tcpdump and python to automate the collection of data. The fact that this repository of data exists now makes all previous attacks that depend on website fingerprinting training sets much stronger.

HTTPOS: Sealing Information Leaks with Browser-side Obfuscation of Encrypted Flows (2011) by Luo, Xiapu and Zhou, Peng and Chan, Edmond WW and Lee, Wenke and Chang, Rocky KC and Perdisci, Roberto [15]. This is newly proposed protocol that attempts to use obfuscation to prevent website fingerprinting. HTTPOS (aka "HTTP or HTTPS with Obfuscation) employs many different techniques to prevent traffic analysis. To start, one of the strongest types of website fingerprinting is comparing the size of outgoing (client-side) packets to that of a pre-determined training set to classify what site a particular user is likely visiting. To combat this, HTTPOS will

simply modify packet lengths. There are few ways that allow one to modify packet lengths without sacrificing functionality:

(1) Appending text to the `Referer` field
(2) Adding additional fields to the packet
(3) Replacing asterisks content-type descriptions with the actual content names

This will add a few bytes to the HTTP header, causing many of the most popular website fingerprinting training sets to fail. Another popular technique in website fingerprinting is flow timings. As in, when a sequence of packets is sent to the server and how long until another sequence is sent. Often times, websites will request information from the browser, then sends information to the browser, to which the browser responds with additional information. This is a very easily recognized pattern with a measurable average time. To combat this, HTTPOS will delay sending information to the server when the server requests. The delay will not be long enough to cause any loss of functionality, but enough to cause a misclassification in traffic analysis.

In general, HTTPOS works well. But it has obvious downsides. HTTPOS is a response to common website fingerprint training sets. Hypothetically, if there were large enough training sets available for fingerprinting HTTPOS, it would be a less effective solution. Also, HTTPOS purposely slows down the speed (ever so slightly) at which you browse the internet, in an effort to confuse monitors.

cMix: Anonymization by high-performance scalable mixing (2016) by Chaum, David and Javani, Farid and Kate, Aniket and Krasnova, Anna and de Ruiter, Joeri and Sherman, Alan T and Das, Debajyoti [6]. One of the most ambitious papers of the field, cMix is a newly proposed cryptographic protocol suite that claims to provide payload secrecy, sender-recipient unlinkability, and sender authentication. Similar to Tor, cMix employs *Onion routing* to achieve sender/receiver anonymity. It uses a variety of different techniques such as symmetric encryption, but avoids expensive public-key operations. cMix attempts to anonymize messages between nodes by requiring a fixed length for each packet. The core cMix protocol is complex, and the authors are extremely verbose in their description. The new protocol claims to be resistant to the fingerprinting attacks that plague Tor.

5 FUTURE RESEARCH DIRECTIONS

There is a lot of unsolved problems in this field. The three largest challenges in this field of research are:

(1) Website fingerprinting
(2) Hidden service Introduction Point and Rendezvous Point selection in a volunteer based system
(3) The Large presence of Sybils in the Tor network

Each of these problems have been around for quite some time, and they are very difficult to solve. Website fingerprinting has shown to be an effective technique for de-anonymization for almost two decades. Standard internet architecture is simply not enough, and I think that a new internet architecture of communication must be designed, or at least an overlay network such as Tor or cMix. I believe that the Tor protocol has been so exhaustively explored, it might need serious re-factoring in order to satisfy its goals against the large amount of proven attacks.

Hidden service selection is also difficult by nature of a volunteer-based system. Intuitively, there **must** be at least one node that will definitely know the IP address of the hidden service. If this node is not trusted, then there is always a possibility that the IP address of the hidden service will be leaked to an adversary. I think the solution to this problem lies in the area of much higher vetting of the nodes that are selected to be part of a session with a hidden service. That includes hidden service directories, Introduction Points, Rendezvous Points, and guard nodes. All of these nodes are points of weaknesses if compromised, and there is at least one academic paper that outlines an attack involving them. Each of these nodes must somehow be selected based on a reputation system. Perhaps a history of good deeds. The short 24 hour qualification period is not strong enough to ward off attackers.

Lastly, Sybils in the Tor network are extremely worrying because they potentially undermine all of the goals of Tor: Data encryption, sender/receiver anonymity, service anonymity, availability. The presence of Sybils subsequently makes many other attacks much stronger due to the highly increased probability of an attacker obtaining a critical node. Sybils are extremely hard to detect and are common in more than just the Tor network like social media networks Facebook and Instagram.

6 CONCLUSIONS

Unlike other fields, research into internet anonymity is unique in its difficulty to test. Experiments are often performed on real systems with real users. There are also variety of ethical implications in trying to de-anonymize systems while conducting research. Several studies have unknowingly stumbled upon censored material, or had to throw away data that contained sensitive information about users. Another facet of this field, especially concerning Tor, is that it changes frequently. Tor is still being developed and improved upon. It can often be discouraging performing an exhaustive study into Tor as there is always a chance that your contributions are quickly going to be forgotten once Tor receives updates.

The internet has been the single largest exchange of new ideas, pictures, videos, and discussion in human history. It has propelled us into an age of conversation and collaboration. As oppressive regimes attempt to censor information, its our duty to continuously seek new and better solutions. I believe this field will not reach maturity anytime soon, and the goal of perfect internet anonymity will be fought against by governments and explored by academics for many years to come.

REFERENCES

[1] Michael Backes, Aniket Kate, Sebastian Meiser, and Esfandiar Mohammadi. 2014. (nothing else) MATor (s): Monitoring the Anonymity of Tor's Path Selection. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 513–524.
[2] Kevin Bauer, Damon McCoy, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. 2007. Low-resource routing attacks against tor. In *Proceedings of the 2007 ACM workshop on Privacy in electronic society*. ACM, 11–20.
[3] Alex Biryukov, Ivan Pustogarov, Fabrice Thill, and Ralf-Philipp Weinmann. 2014. Content and popularity analysis of Tor hidden services. In *Distributed Computing Systems Workshops (ICDCSW), 2014 IEEE 34th International Conference on*. IEEE, 188–193.
[4] Alex Biryukov, Ivan Pustogarov, and Ralf-Philipp Weinmann. 2013. Trawling for tor hidden services: Detection, measurement, deanonymization. In *Security and Privacy (SP), 2013 IEEE Symposium on*. IEEE, 80–94.
[5] Xiang Cai, Xin Cheng Zhang, Brijesh Joshi, and Rob Johnson. 2012. Touching from a distance: Website fingerprinting attacks and defenses. In *Proceedings of the*

2012 ACM conference on Computer and communications security. ACM, 605–616.

[6] David Chaum, Farid Javani, Aniket Kate, Anna Krasnova, Joeri de Ruiter, Alan T Sherman, and Debajyoti Das. 2016. cMix: Anonymization by high-performance scalable mixing. In *Proceedings of ACM CCS*, Vol. 2016.

[7] Wikimedia Commons. 2011. Plaques of Lambda Phages on E. coli XL1-Blue MRF. (2011). https://en.wikipedia.org/wiki/File:Internet_Censorship_and_Surveillance_World_Map.svg File: LambdaPlaques.jpg.

[8] Roger Dingledine, Nick Mathewson, and Paul Syverson. 2004. Tor: The second-generation onion router. Technical Report. DTIC Document.

[9] Kevin P Dyer, Scott E Coull, Thomas Ristenpart, and Thomas Shrimpton. 2012. Peek-a-boo, i still see you: Why efficient traffic analysis countermeasures fail. In Security and Privacy (SP), 2012 IEEE Symposium on. IEEE, 332--346.

[10] Andrew Hintz. 2002. Fingerprinting websites using traffic analysis. In International Workshop on Privacy Enhancing Technologies. Springer, 171--178.

[11] Ariel Hochstadt. VPN Use and Data Privacy Stats for 2017. (????). https://www.vpnmentor.com/blog/vpn-use-data-privacy-stats/

[12] Marc Juarez, Mohsen Imani, Mike Perry, Claudia Diaz, and Matthew Wright. 2016. Toward an efficient website fingerprinting defense. In European Symposium on Research in Computer Security. Springer, 27--46.

[13] Albert Kwon. 2015. Circuit fingerprinting attacks: Passive deanonymization of tor hidden services.

[14] David Lazar and Nickolai Zeldovich. 2016. Alpenhorn: Bootstrapping secure communication without leaking metadata. In Proceedings of the 12th Symposium on Operating Systems Design and Implementation (OSDI), Savannah, GA.

[15] Xiapu Luo, Peng Zhou, Edmond WW Chan, Wenke Lee, Rocky KC Chang, and Roberto Perdisci. 2011. HTTPOS: Sealing Information Leaks with Browser-side Obfuscation of Encrypted Flows.. In NDSS, Vol. 11.

[16] Damon McCoy, Kevin Bauer, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. 2008. Shining light in dark places: Understanding the Tor network. In International Symposium on Privacy Enhancing Technologies Symposium. Springer, 63--76.

[17] Steven J Murdoch. 2006. Hot or not: Revealing hidden services by their clock skew. In Proceedings of the 13th ACM conference on Computer and communications security. ACM, 27--36.

[18] Steven J Murdoch and George Danezis. 2005. Low-cost traffic analysis of Tor. In Security and Privacy, 2005 IEEE Symposium on. IEEE, 183--195.

[19] Lasse Overlier and Paul Syverson. 2006. Locating hidden servers. In Security and Privacy, 2006 IEEE Symposium on. IEEE, 15--pp.

[20] Andriy Panchenko, Fabian Lanze, Andreas Zinnen, Martin Henze, Jan Pennekamp, Klaus Wehrle, and Thomas Engel. 2016. Website fingerprinting at internet scale. In Network & Distributed System Security Symposium (NDSS). IEEE Computer Society.

[21] The Tor Project. Tor: Overview. (????). https://www.torproject.org/about/overview.html.en

[22] The Tor Project. 2017. Tor Metrics. (2017). https://metrics.torproject.org/

[23] Tao Wang, Xiang Cai, Rishab Nithyanand, Rob Johnson, and Ian Goldberg. 2014. Effective Attacks and Provable Defenses for Website Fingerprinting.. In USENIX Security. 143--157.

[24] Tao Wang and Ian Goldberg. 2016. On realistically attacking Tor with website fingerprinting. Proceedings on Privacy Enhancing Technologies 2016, 4 (2016), 21--36.

[25] Philipp Winter, Roya Ensafi, Karsten Loesing, and Nick Feamster. 2016. Identifying and characterizing Sybils in the Tor network. arXiv preprint (2016).