Tor & Tor Onion Services Attacks

\*Nabhanyu Halgeri,\*

*Abstract*— The Onion Router, also known as TOR, is a free and open-source software program that routes internet traffic through a series of volunteer operated servers to provide anonymity and privacy to its users. The servers that are used are known as "nodes" or "relays." With the help of Tor, it becomes difficult for anyone to track a user’s online activities.

However, despite the security and privacy Tor provides, several attacks have taken place on Tor in recent years. In this paper, we discuss the different types of attacks that have been carried out, along with the strategy and motivation behind them. Further, an attack on Tor is implemented, and the vulnerable areas through which the Tor network can be compromised are analyzed.

# Introduction (*Heading 1*)

Smart Tor is a free and open source programmed that enables users to browse the internet anonymously and securely to give online anonymity. The acronym "Tor" stands for "The Onion Router. Tor uses a method of multi layered to masquerade the traffic and utilize its own encryption to encrypt the traffic as it travels through different nodes and servers to it final destination. The traffics is difficult to track since each node in the network only knows the previous node and the following node where it is going in the chain.

Tor also uses bridges which acts as nodes but works for access to the Tor network and get around censorship, users use relays called "Tor bridges." Relays known as bridges are not publicly mentioned in the Tor directory, making it more difficult to stop them. Without their internet service provider or government being able to tell that they are using Tor, users in blocked areas can utilise a bridge to connect to the Tor network. The hidden services of Tor helps in hosting of website and services with anonymity and have an ending of .onion, furthermore they are only available and reachable within tor network. Even though the Tor offers and has a high level of digital anonymity it is not impenetrable and invulnerable to attacks that could be exploited.

Onion Router, or Tor, is a free and open-source software project that promises to give its users internet anonymity and privacy. As part of a project to safeguard official communications, the United States Naval Research Laboratory (NRL) created it in the middle of the 1990s. Later, it was made publicly available in 2002. A decentralised network of computers that enables anonymous internet access, was formally introduced in 2004. Almost 7,000 volunteer-run machines, often referred to as nodes or relays, are included in the network and are dispersed across the globe. It is challenging to pinpoint the origin of the data back to the user because a user's data is routed through several of these nodes when they join the Tor network, it is difficult to trace the origin of the data back to the user.

Tor Onion Services, originally known as Hidden Services, were disclosed by the Tor project in 2004. It makes it difficult to pinpoint the location of the server or the owner by enabling the anonymous hosting of web services throughout the Tor network. Onion Services, a pseudo-top-level domain that can only be accessed through the Tor network, can be reached using an A.onion address.

# What is Tor Tor onion service?

This Internet users look for methods to anonymize their network data due to growing privacy and security concerns. The Tor system was created by the Pentagon as a research based. In the information age, it has evolved into an instrument for preserving freedom of speech and privacy [5]. The Tor Project has a negative reputation despite having good intentions. The expansion of Tor and the anonymity it offers have made the network a haven for illegal activity known as the "Dark Web," much like any big, growing city draws criminals [6]. A well-known illustration of a hidden service is Silk Road, a drug-selling website that the FBI took down in 2013 [5]. Ross Ulbricht, the site's administrator, was detained on suspicion of being "Dread Pirate Roberts," the site's fictitious founder, and he was given a life term.

A crucial resource for people who respect their online privacy and security is the Tor network. Users are looking for solutions to prevent their personal information from being used by hackers and other harmful actors considering the rising frequency of internet security breaches and data dumps. Users can hide their online identities and make it difficult for others to follow their digital footprints by using the Tor network.

Despite its acceptance and usefulness, the Tor network has come under fire and developed a bad reputation because of its links to illegal activity on the "Black Web." Like any large, developing metropolis that draws criminals, the Tor network's anonymity has spawned an underground market for illicit activity [6]. It is important to remember that the majority of people who use the Tor network do so for legal reasons, such as to access prohibited information and preserve their online privacy.

We shall give a thorough analysis of the Tor network in this essay, covering its development, history, and services. We'll look at the various network attacks that have been made as well as the tactics hackers have employed to infiltrate the Tor network. We'll also go through current studies on strengthening Tor's security and show how a traffic analysis attack may really be implemented on the network.

Users, developers, and researchers may improve the security and privacy of the Tor network by using the knowledge gathered from this study. We hope that by providing a greater knowledge of the challenges and opportunities that the Tor network presents, this article will assist users in making decisions that will protect their online privacy and security.

# onion routing / service

In this part we talk about the onion service, which is a network built on a low-latency onion-routing architecture, where traffic is forwarded through arbitrarily chosen Onion Routers (ORs), encrypting data with numerous onion skins to preserve unlikability [7].

In this situation, an OR can also be referred to as a relay, node, or simply a router. Each stream can be telescopically silently channelled through the network, which means that each router only knows the relays that come before and after it [8]. The source of the stream is only known to the first relay, the entry point. The only relay that is aware of the client's location is the last relay, the exit node. Only encrypted information is exchanged between the onion router(s) [9]. Symmetric cryptography is used to layer-encrypt data, which is then unwrapped by a relay before being forwarded to the following relay in the chain [104]. Typically, a circuit has three relays.

Diagram

Description automatically generated Figure 1: Onion Routing

The low-latency onion-routing architecture used by the Tor network allows traffic to be passed through arbitrary Onion Routers (ORs). To maintain unlikability, this routing method encrypts data using many onion skins.

A relay, node, or straightforward router can all be used to describe any one of the network's onion routers. Each router in the network is only aware of the relays that come before and behind it as the traffic stream is silently funnelled through the system [8]. Just the first relay, which serves as the entrance point, is aware of the stream's source. Only the final relay, the exit node, is aware of the location of the client. The onion routers only communicate using encrypted data [9].

Symmetric cryptography is used to layer-encrypt the data, ensuring its confidentiality and anonymity, which is then unwrapped by a relay before being passed to the next relay in the chain [10]. A circuit in the Tor network typically consists of three relays, ensuring that the data is sent through the network in an anonymous and safe manner.

In essence, onion routing is a fundamental component of the Tor network, which guarantees anonymity and privacy to its users. Data is securely and anonymously routed through the network because to the low-latency onion-routing architecture of the network, in which each relay only knows about the relays that come before and after it. Layer-encrypting data with symmetric cryptography adds an extra layer of protection by preventing data from being intercepted by attackers

# Attacks Seen on Tor

The biggest network for anonymous contact is called Tor. Recent papers debate the efficacy of Tor and discuss the flaws in the Onion Router design. De-anonymizing attacks are increasingly utilizing these weaknesses. The need for hybrid attacks that can be used at the network layer, protocol layer, or application layer has increased as attacks have become more complicated and potent over time. We'll talk about public Tor attacks and group them into categories for additional investigation. Some ethical flaws have also been created by Tor's freedom and privacy principles. The network's cover attracts criminal activity, which has damaged its image. Detection of Replay CCPAs

The majority of Tor attacks concentrate on figuring out which client and server are utilizing the Tor network for communication [10]. De-anonymization is the procedure in question [11]. A circuit between the client and an exit node has been established in the Tor network, and the exit node is in communication with the host. A hidden service is being provided under a pseudonym, and the attacker wants to connect that pseudonym to the operator's real identity, either directly or through an intermediary step (such as a physical location or IP address) [12] [13]. The attacker also wants to confirm that the client and the server are communicating.

??? Users using Tor may potentially be the victim of social engineering attacks like phishing emails or texts, which deceive recipients into disclosing their identity or jeopardising their security. For instance, a hacker could send a message that appears to be from a reliable source, like a friend or a Tor developer, and encourage the user to divulge personal information or download a malicious file [6]. This kind of assault might be challenging to recognise and has the potential to jeopardise the security of the entire Tor network.

The security and privacy of the Tor network are being constantly enhanced by researchers and developers in order to allay these worries. This entails creating fresh methods for encrypting and safeguarding data as well as finding and fixing Tor software flaws [7]. Also, initiatives are being made to inform Tor users of the dangers of the network and to offer advice on the best ways to be anonymous and private online.

Attacks on the Tor network continue to be a major worry despite these measures. Attackers will probably keep coming up with new and more advanced ways to compromise the network's security as its popularity continues to rise. To maintain the anonymity and privacy of Tor users, it is crucial to keep looking into and creating new methods, as well as to be always on the lookout for new threats to the network.

There are various categories of De-Anonymizing Techniques and Attacks namely –

Passive and active attacks

Single-end and end-to-end attacks

Attacks can be divided into seven categories according to their strategy and objective ->

Correlation Attacks End-to-end - Passive Attack

Congestion Attacks End-to-end - Active Attack

Timing Attacks End-to-end - Active Attack

Fingerprinting Attacks Single-end - Passive Attack [14]

Denial of Service Attacks Single-end - Active Attack

Supportive Attacks - Not classified.

Revealing Hidden Services Attacks Not classified.

## **Correlation attacks**

The correlation attack uses packet timings, packet flows and packet sizes to determine and correlate the network flows which in turn helps to conduct the attacks over TOR. The attacker finds a correlation between the entry and exit nodes to confirm that a communication is taking place between the client and the server. Continue the adversary to build a fresh FDI attack is thus only limited by independently checking state variables (in cyberspace), but the optimal CCPA still has a chance of evading detection.

Further examination of the improved CCPA reveals that while the cyber-attack may be able to negate the effects of the

# Topiccc

## Sub Topic

.

## Sub topic

# Implementation

## Tor client, server and client+ server Attacks

Attacks can still be made against Tor and Tor browsers. Majority of attacks on Tor is exploiting vulnerabilities to traffic analysis, confirmation attacks, and probable guard discovery these are the attacks that have always been under the limelight among the top researchers and users. However, several other lesser-known attacks exist and in recent years they are some of the most infectious and top-rated attacks. Since the beginning of Tor, research on onion networks has revealed that adversaries are capable of assaulting three different things:

**Client:** To identify it, a Tor network client is chosen.

**Server:** An attempt is made to identify or weaken the Tor onion (hidden) service.

**Network:** Multiple malicious Tor nodes are typically used to target the larger Tor network.

Additionally, there are generic assaults that target several Tor entities; often, both the client and server are attacked simultaneously.

## Induced Tor Guard Selection( client side Tor threat)

Tor clients can be persuaded to use a malicious Tor guard (entry) node by, among other things, changing the target's traffic capabilities, obstructing connections to trustworthy entry nodes on the network, and so on. End-to-end correlation and other assaults are considerably aided by this.

## P2P Information Leakage (client side Tor threat)

Peer-to-peer connections are taken advantage of to obtain the client's IP address. For instance, when clients use the BitTorrent protocol to connect over Tor and communicate with the torrent tracker, opponents can obtain the IP address of those clients. The IP address and listening port of peers who can share the requested resource are retrieved by torrent trackers, tracker lists can be retrieved anonymously over Tor, the P2P connection itself cannot, making it vulnerable to MitM attacks that could lead to a list containing the IP address of a malicious torrent peer. This indicates that it is possible to identify the client whose IP address sent the tracker request (through Tor).

## Raptor Attacks (client side Tor threat)

Tor traffic is carried by autonomous systems (ASes), which have advanced eavesdropping capabilities. To deanonymize clients, an AS or a group of cooperating ASes can do time analyses between the client and the first relay and between the last relay and the destination. Three methods are used to accomplish this: altering BGP routingarchive.org so that more ASes can analyse a user's traffic; modifying BGP announcements so that ASes are chosen on paths to and from relay nodes; and conducting timing analysis on unidirectional traffic at both communication ends.

## Cell Counting and Padding [15] (server side Tor threat)

An onion (hidden) service is compelled to connect to a malicious rendezvous point in this attack. A signature is added to the communication by a specifically created set of Tor padding cells (of a particular number) and a cookie or token produced by the client. It is possible to determine which guard node was selected by the onion service and thus its IP address if an entry node controlled by an attacker monitor recognizes these signatures.

## Off-path MitM Attacks [16] [15] (server side Tor threat)

Man-in-the-middle attacks can be launched against the targeted service by an adversary who has gained access to (or assumed ownership of) the onion service's private key. The only way to prevent this attack is to stop using them. Onion address and generate a new one because there is no revocation mechanism for onion services.

## Cell Traffic Analysis [17] (server + client side Tor threat)

Network traffic analysis-capable adversaries insert packets (particular, repeating traffic in the TCP connection) server-side and attempt to observe these packets client-side using statistical correlation. As a result, Tor client traffic can be recognized if a client is linked to a malicious server and the adversary has control over a lot of entry (guard) nodes, one of which is picked in a particular Tor circuit.

## **Abstract of our Attack Implementations**

We have homed on to two different types of attacks, first implementation being a DDOs attack which can be launched with publicly available hosts and DNS like cloud flair and another variant where we have a hidden services created with onion service and to launch a DDOS attack.

Furthermore, the second implementation attack that we try to demonstrate is which is being actively employed by threat actors and cyber criminals over Tor client distribution and injecting an obscure and specialised malware which changes the crypto wallet addresses.

## Implementation Attack – 1

The first one that we try to demonstrate is a DDOS attacks that can be launched through crating a hidden/ shared hosting service. The set up has been configured with Debian and Ubuntu installations. In addition, we are using public nameserver like 1.1.1.1 (from Cloudflare) or 8.8.8.8 (from Google) this will help us to create the shared hosting service through Cloudflare. For creating of the hidden services, we generate a skeleton configuration that will allow us to generate an address, we can name it accordingly to our mode of attack. In this case we are naming the address to be “coffeandpandora”.

To create a hidden service with our address name we use a docker container to initiate the process. Once the container is downloaded, we can then generate our site skeleton and it will yield the following output.

$docker run -it --rm -v $(pwd)/web:/web coffeandpandora /hiddentorser generate ^ coffeandpandora

[+] Generating the address with mask: ^coffeandpandora

[+] Found matching domain after 137072 tries: coffeandpandora.onion

[+] Generating nginx configuration for site coffeandpandora.onion

[+] Creating www folder.

[+] Generating index.html template.

Once we have our skeleton ready, we can deploy and run it by running the docker container.

With docker run – d – restart=always –name hide\ - v $(pwd)/web:/web coffeandpandor/hiddentorser

Graphical user interface, text

Description automatically generated

Screen shot 0 – Hosted website running on hidden service of onion locally.

To execute the DDOS attack we have customised the payload to come from TOR network.

The modus of attack that we use is by one VM generating and running a shared hosting service and another one running a hidden running service. These would be attacking the victim system (This is implementation is done on internal networks and have not fully exposed to the onion services for safety and security concerns)

A brief description on how the DDos works -

1. Initial timestamp and increment the counter.

2. Init a new Tor session, this can be our shared or hidden services. It is more feasible to use hidden services as these are more obscure and more difficult to render. The shared will be blocked by the original hosting services like google and Cloudflare, even though Cloudflare hosting services has been used many times for carrying out DDos attacks in the past.

3. Sending of random\_bytes = random.\_urandom(1490) to go back and increment even more to cause traffic overflow and lead to DDOS on the website or system that is targeted.

The execution – We can use cloud flare or have our own hidden service website that can be used to configure the DDOs attack. In this case we have used the cloud flair for demo purposes as there are more steps involved in setting up of the DDOs attack with an obscure hidden service. Ultimately, we have planned and provision to run the attack with hidden services which we can further work on developing. Below is the screen shot of running the attack. In the screen shot below shows the arrangements made for running DOS through hidden services [19].

Text

Description automatically generated

Screen Shot 1 – Test of DDos

Text

Description automatically generated

Screen Shot 2 – Code Snippet for DDos

## Implementation Attack – 2

Another Recent attack that has surfaced recently is a Tor browser combined with a malicious malware payload for distribution. The modus operandi of the use’s social media platforms such as YouTube and many others to distribute and propagate the malware. With the ongoing conflict of Ukraine and Russia, the tor nodes which are originated from Russia are being blocked resulting in a net loss of 150K user with the censorship on the Tor in Russia.

This has led to threat actors to capitalize on this and distributing out copies of rigged Tor browsers with claims of being secure and legit. The installers that are distributed carry names like 'torbrowser\_ru.exe,' and additionally contain the language packs allowing users to select their preferred language as it appears in a real Tor browser installer.

The archive extracts the malware in the background, executes it as a new process, and registers it for system auto starts while the default Tor browser is started in the front. Additionally, the malware conceals itself on the compromised system by using an uTorrent icon.

Diagram

Description automatically generated with medium confidence

Figure [20] – Tor client, source: Kaspersky

From 2022 to 2023 March 16,000 variants of these Tor installers have been detected in 52 countries, while the main target are Russia, China, and Eastern Europe, it has also been seen used in United States, Germany, France, the Netherlands, and UK.

Furthermore, our observations reveal that these attacks target people who use Tor for additionally anonymity for carrying payment in Bitcoin. This allows threat actors to have a malware with high obscurity and customizability, which they have successfully exploited by using a malware technique called clipboard hijacking. It is common and second nature for users to copy bitcoin addresses to the clipboard before pasting them into a webpage for payment because they are lengthy and difficult to enter. Using regular expressions, the malware scans the clipboard for recognisable crypto wallet addresses which it replaces it with a related cryptocurrency address controlled by the threat actors which is a huge list of 26K to 30k addresses. The threat actor's address will be pasted when the user pastes the cryptocurrency address, allowing the attackers to steal the sent transaction.

The following regular expression were found with our research inside the sample of the malware-

bc1[a-zA-HJ-NP-Z0-9]{35,99}($|\s) – Bitcoin

(^|\s)[3]{1}[a-km-zA-HJ-NP-Z1-9]{25,34}($|\s) – Litecoin/Bitcoin Legacy

(^|\s)D[5-9A-HJ-NP-U]{1}[1-9A-HJ-NP-Za-km-z]{32}($|\s) – Dogecoin

(^|\s)0x[A-Fa-f0-9]{40}($|\s) – ERC-20 (i.e. Ethereum, Tether, Ripple, etc)

(^|\s)[LM]{1}[a-km-zA-HJ-NP-Z1-9]{25,34}($|\s) – Litecoin Legacy

((^|\s)ltc1[a-zA-HJ-NP-Z0-9]{35,99}($|\s) – Litecoin

(^|\s)8[0-9A-B]{1}[1-9A-HJ-NP-Za-km-z]{93,117}($|\s) – Monero

(^|\s)4[0-9A-B]{1}[1-9A-HJ-NP-Za-km-z]{93,117}($|\s) – Monero

This is a basic version of the malware; the more advanced version involves the malware contacting a c&c server in real time and having it sent and generate a similar crypto address which matches the last few characters of the original address which reduces the chances of being caught.

Below is a test version of the malware that is created, here it can be seen that the regular expression of bitcoin wallets like bitcoin, etheruim, Litecoin and monero respectively. Below that we can add the attacker-controlled address, note the address that is highlighted is used only for demo purpose, it is a 25-digit address whereas bit coin address is 26. When we copy any bitcoin address it changes the actual address to the attacker’s address. This can be done not only for the crypto wallet but also for bank codes and name with e-transfer method.

Text

Description automatically generated

Screen Shot 3 – Code for Clip Malware

Text

Description automatically generated

Screen Shot 4 – Code for Clip Malware

A screenshot of a computer

Description automatically generated with medium confidence

Screen shot 5- Copying a Bitcoin address.

A screenshot of a computer

Description automatically generated with medium confidence Screen Shot 6- The Bit-coin address changed because of malware program.

The way to defend against this attack exists numerous ways.

1. Use caution when installing software or opening email attachments from sites you are unfamiliar with.

2. For all your accounts, use strong, one-of-a-kind passwords, and avoid copying and pasting them wherever feasible.

3.Whenever feasible, avoid copying and pasting confidential data, including passwords, credit card numbers, and other personal information.

4. Make use of a clipboard manager that will notify you when a program accesses the clipboard and will prevent unauthorized access.

5. Consider utilizing a virtual keyboard to protect yourself from viruses such as keyloggers and other threats that may be watching your keystrokes.

# Conclusion & Observation

### Although Tor claims to be a pinacle of anonimity abd security, there are security misconfogurations, vulnerabilities and scoial enginnering factors which always find a way to exploit the protocols and tenhiques.

# References

1. *https://github.com/Attacks-on-Tor/Attacks-on-Tor.git -*
2. *Tor. Who uses Tor? 2014. url: https://www.torproject.org/about/torusers.html.en*
3. *R. Snader and N. Borisov. "Improving Security and Performance in the Tor Network through Tunable Path Selection". In: IEEE Transactions on Dependable and Secure Computing (2010).*
4. *L. Overlier and P. Syverson. "Locating Hidden Servers". In: IEEE Symposium on Security and Privacy (2006).*
5. *W. Nicol. A Beginner's Guide to Tor: How to navigate through the underground Internet. 2016. url: http://www.digitaltrends.com/computing/a-beginners-guide-to-tor-how-to-navigate-through-the-underground-internet*
6. *Damon McCoy et al. "Shining light in dark places: Understanding the Tor network". In: Privacy Enhancing Technologies. Springer. 2008, pp. 63-76.*
7. *L. Overlier and P. Syverson. "Locating Hidden Servers". In: IEEE Symposium on Security and Privacy (2006).*
8. *M. Mulazzani, M. Huber, and E. Weippl. "Anonymity and Monitoring: How to Monitor the Infrastructure of an Anonymity System". In: IEEE Transactions on Systems, Man and Cybernetics.*
9. *Juha Salo. "Recent Attacks On Tor". In: Aalto University.*
10. *Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Tech. rep. DTIC Document.*
11. *Yixin Sun et al. "RAPTOR: routing attacks on privacy in tor". In: 24th USENIX Security Symposium (USENIX Security 15)*
12. *Ryan Pries et al. "A new replay attack against anonymous communication networks". In: Communications, 2008. ICC'08. IEEE International Conference on. IEEE. 2008.*
13. *S. Murdoch. "Hot or Not: Revealing Hidden Services by their Clock Skew". In: Proceedings of the 13th ACM Conference on Computer and Communication Security.*
14. *Sam DeFabbia-Kane. "Analyzing the effectiveness of passive correlation attacks on the tor anonymity network". PhD thesis. Wesleyan University,*Y. Zhang et al., “Home M2M networks: Architectures, standards, and QoS improvement,” IEEE Commun. Mag., vol. 49, no. 4, pp. 44–52, Apr. 2011.
15. Protocol-level Hidden Server Discovery Zhen Lingy, Junzhou Luo, Kui Wuy and Xinwen Fuz.
16. R Off-path Man-in-the-Middle Attack on Tor Hidden Services Amirali Sanatinia, Guevara Noubir Northeastern University.
17. Low-Cost Traffic Analysis of Tor Steven J. Murdoch and George Danezis University of Cambridge, Computer Laboratory 15 JJ Thomson Avenue, Cambridge CB3 0FD United Kingdom
18. Website://www.whonix.org/wiki/Speculative\_Tor\_Attacks#cite\_note-15.
19. bleepingcomputer.com/news/security/tor-and-i2p-networks-hit-by-wave-of-ongoing-ddos-attacks/
20. source Kasperkeyn