# Cracking the Anonymous IoT Routing Networks: A Deep Learning Approach

Gaurang Bansal, Vinay Chamola, *Senior Member, IEEE*, Amir Hussain *Senior Member, IEEE* and Muhammad Khurram Khan, *Senior Member, IEEE*

***Abstract*—In recent years, IoT technology has been one of the most rapidly expanding fields, connecting over 27 billion connected devices worldwide. Increasing security concerns, such as software flaws and cyberattacks, limit the use of IoT devices. Tor, also known as ”The Onion Router,” is one of the most popular, secure, and widely deployed anonymous routing systems in IoT networks. Tor is based on a worldwide network of relays operated by volunteers worldwide. Tor continues to be one of the most popular and secure tools against network surveillance, traffic analysis, and information censorship due to its robust use of encryption, authentication, and routing protocols. However, ToR is not anticipated to be entirely safe. The increasing computational capabilities of adversaries threaten Tor's ability to withstand adversarial attacks and maintain anonymity. This paper describes the foundation of the Tor network, how it operates, potential attacks against Tor, and the network's defense strategies. In addition, the authors present a framework for deep learning that uses bandwidth performance to identify the server's location in Tor, thereby compromising anonymity. This paper examines Tor's network's current and projected future in the Internet of Things.**

***Index Terms*—Tor network, Onion routing, Dark web, Privacy, Anonymity.**

## I. Introduction & Motivation

E-commerce, social media, cryptocurrency, cloud computing, and big data are examples of how the digital era has disrupted the traditional way of doing things in every social and economic sector. Data breaches and cyber thefts have become increasingly common because of digital products' rapid development and innovation. Customers are increasingly opting for products that promise data privacy and cybersecurity as a result of this development [1]. In digital marketplaces (e.g., digital payment systems and community forums), users want their online communications and transactions to be handled more anonymously. Dark wallets and anonymous networks, which provide data anonymization platforms, are meeting these demands. Tor is one such unknown network.

In a digital world plagued by cybersecurity concerns, the Tor network is one of many emerging technologies which attempt to fill the data privacy void. Anonymity can be achieved using Tor, which stands for 'The Onion Router,' an open-source privacy network. US Navy initially used Tor to censor government communications before it was made public [2].

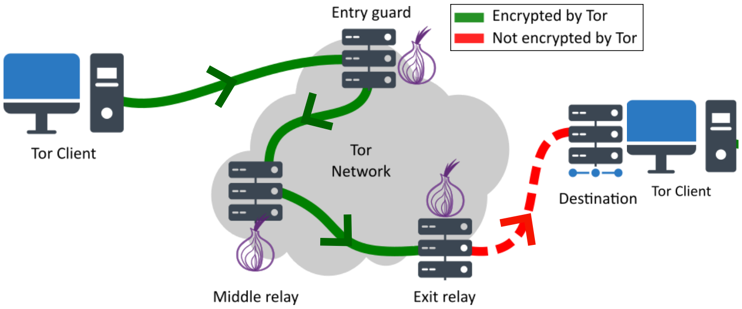


Fig. 1: Tor Network Model

The client and the server have always communicated directly in standard internet connections. If an eavesdropper were to utilise this technique, they might easily learn the user's identify and track their movements. The IP headers of direct encrypted connections are not concealed, thus the sender's and receiver's addresses as well as the amount of the data being sent are still visible. A user's private data may be exposed when confronted by attackers that use advanced traffic analysis techniques (to maintain anonymity).

All TCP communication from the end user is routed via a number of relays on the Tor network, which offers an extra degree of anonymity and privacy. Usually, this route is a dynamic circuit consisting of many hops. Figure 1 depicts a circuit with three relays, which are referred to as the ”entry relay,” ”middle relay,” and ”exit relay,” respectively. The Tor network's entrance relay is the only node that can trace TCP communication back to its original source, and the exit relay is the only node that can inspect the message's content and destination. Indecisive middle relays can't choose between the two options. The Tor network assures that the source, content, and destination of every online traffic are hidden from any one relay.

The authors next describe Tor's routing architecture, which is meant to make it impossible for a well-resourced attacker to learn the end-identity users' and the network's behaviour, even if relays are compromised [3].

The main contributions of this paper are as follows:

1) This paper presents the routing of messages in a Tor network. It discusses how Tor routes end-user traffic

Gaurang Bansal is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077, Singapore (email: gaurang@u.nus.edu)

Vinay Chamola is with the Department of Electrical and Electronics Engineering & APPCAIR, BITS-Pilani, Pilani Campus, India 333031 (e-mail: vinay.chamola@pilani.bits-pilani.ac.in)

Amir Hussain is with School of Computing, Edinburgh Napier University, Scotland, UK (email: A.Hussain@napier.ac.uk)

Muhammad Khurram Khan is with Center of Excellence in Information Assurance, King Saud University, Riyadh, Saudi Arabia (email: mkhurram@ksu.edu.sa)

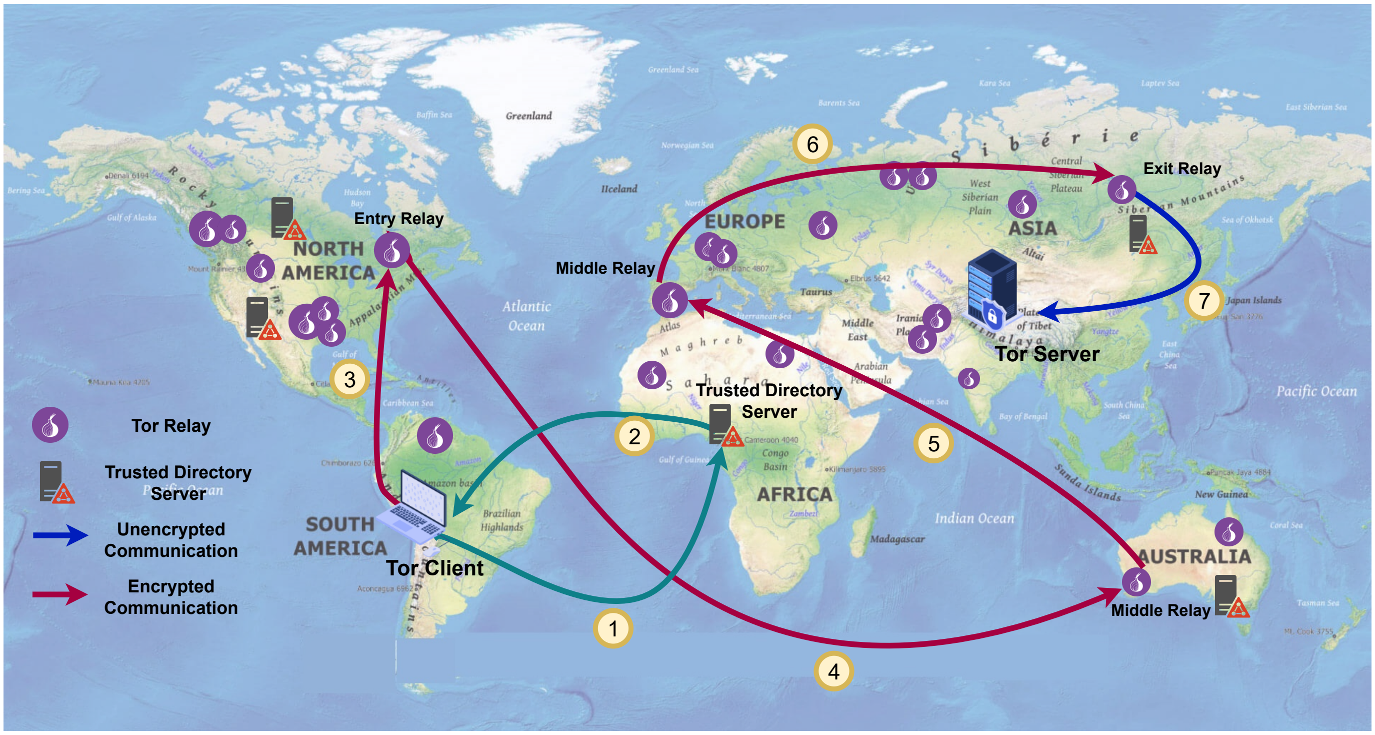


Fig. 2: Routing in Tor networks

through a randomized circuit by forming a network of relays.

2) Tor tries to give people anonymity and privacy on the Internet, but people who don't like these goals can attack Tor in many ways. This work highlights the various attacks and possible defense strategies in the Tor network.
3) We provide an unique networked attack that may reveal the location of an anonymous server and anonymous proxies by analysing "traffic variations" into a certain anonymity-preserving channel.

## II. Routing

In this section, the authors discuss the working of Tor networking on how Tor routes traffic. The steps of communicating the messages are described as follows.

1) When the TOR client $A$ wants to access the server $B$ via the Tor network, it queries its trusted directory server. It is noteworthy that the trusted directory server stores the IP address, port details, and public onion keys of all the TOR relays operational in its network [4].
2) The trusted directory server sends the client these details when the client queries the trusted directory server, as shown in Fig. 2.
3) Next, the client node $A$ randomly chooses one of the TOR relays from the list of the TOR relays sent by the trusted directory server (TDS). This is called the first relay or entry guard, which authors denote by $R_1$. Then the client $A$ establishes a Transport Layer Security (TLS) connection using the relay's public key. Both $A$ and the router $R_1$ establish a secure communication circuit $C_1$ between themselves by negotiating a shared secret key, as shown in Fig. 3. The client and $R_1$ use a single Diffie-Hellman-Merkle key exchange which authors denote as $K_1$ to negotiate messages as shown in the red box in Fig. 3. During the connection establishment with the relay, rather than establishing the connection again and again for communication, symmetric key $K_1$ generates two symmetric keys, one forward and one backward. $K_{1,F}$ encrypts all client communication to $R_1$, and $K_{1,B}$ encrypts all answers from $R$ to $A$ (shown in the green box in Fig. 3).
4) Once this one-hop circuit has been created, the client $A$ randomly chooses another relay $R_2$ with the help of a trusted directory server. The client $A$ sends the address of the router $R_2$ to the router $R_1$, along with its half of the Diffie-Hellman-Merkle protocol [5] (discussed in next section) using $K_{1,F}$. Then, $R_1$ performs a TLS (Transport Layer Security) handshake and circuit creation with $R_2$ as relay $R_1$ performed with client $A$. $R_1$ uses $R_2$'s public key, which replies with his half of the handshake and a hash of $K_2$. Finally, $R_1$ forwards this to the client $A$ under $R_{1,B}$
5) Next, the client generates $K_{2,F}$ and $K_{2,B}$ from $K_2$, and repeats the process for $R_3$ [6] as shown in Figure 3.
6) In communication among Middle relay relays, there is no need for a separate key generation as the middle relay relays forward the packets till they reach the exit relay. Since the TLS/IP connections are still active, the data that is sent back might make its way to the original sender.
7) The final relay, referred to as exit relay, sends the packet to the destination depending on whether the connection is encrypted or not. If the connection to the server is

encrypted, the exit relay will not be able to read the data in plaintext. A potential intruder would have to break through TLS encryption at each of four points: the client $A$, the entrance relay, the middle relay, and the exit relay. Because of this, analysing network data and launching cryptographic attacks is very challenging.

The Secure Sockets (SOCKS) interface provided by the Tor client software multiplexes TCP traffic across Tor after the whole circuit has been established ([7]). Since each relay only sees one hop in the circuit, it should be impossible for an eavesdropper or a compromised relay to determine the origin, destination, or data being sent via the connection. Tor's circuit route is randomised every 10 minutes, greatly concealing users' online movements ([3]).

Because of its focus on privacy and security, the Tor Browser Bundle (TBB) is the version of Mozilla Firefox preferred by users of the Tor network. TBB's HTTPS Everywhere plugin, which uses regular expressions to convert HTTP web requests into HTTPS whenever practical, and its unique handling of client-side scripting like Javascript are two examples of this. As a result, HTTP conversations will be encrypted if the webserver supports SSL or TLS connections. In this situation, the TBB will initiate a TLS handshake with the web server, but the conversation will take place inside the Tor network. In the next section, the authors discuss the encryption behind Tor that prevents adversaries from compromising the system.

## III. Message Encryption

Privacy both inside and outside of the Tor network requires the use of encryption. Within the network, communication between relays takes place using the Transport Layer Security (TLS) protocol [8]. A symmetric cipher and encryption key are mutually agreed upon after the client conducts a handshake with each relay during the circuit's building phase. The most common way is to set up a symmetric encryption key. The symmetric key exchange occurs through Diffie-Hellman-Merkle (DH) protocol, as described in the previous section. The By creating a fresh session key with each handshake, DH guarantees forward secrecy. Defending against attacks that may decode previously encrypted traffic even if the relay was compromised requires not storing the secret DH keys.

### *A. Symmetric Encryption (AES/DES/RC4)*

Following a successful TLS handshake and DH key exchange, the communicating parties may encrypt their data using a symmetric key and the session key that was just produced. The most popular symmetric-key algorithms are Data Encryption Standard, Advanced Encryption Standard, and RC4.

AES and DES are block ciphers, whereas RC4 is a stream cipher. While brute-force assaults were successful in cracking DES in 1998, the much more secure Advanced Encryption Standard (AES) cipher has subsequently rendered triple DES (3DES) obsolete. Despite this, 3DES is still frequently used on the web, especially in older versions of Microsoft's products [9–11]. Since it was designed to be implemented quickly in software, RC4 ciphers are both easy to use and quick. These days, protocols like Transport Layer Security (TLS) and Wired Equivalent Privacy (WEP) employ it more than any other software stream cipher.

Apart from symmetric encryption, some of the current Tor relays use an asymmetric protocol such as Rivest-Shamir-Adleman Algorithm (RSA) protocol or Elliptic-curve Cryptography (ECC) protocol described as follows:

### *B. Rivest-Shamir-Adleman Algorithm (RSA)*

**Background:** Rivest-Shamir-Adleman In the field of public-key cryptography, one may encounter the RSA Algorithm. The fact that the product of two big primes cannot be factored provides the basis for its security. RSA, like other public-key algorithms, requires both a public and private key in order to function. The public key is disseminated to the public and is used in cryptography and the validation of digital signatures. Decryption and digital signature creation are both tasks that need the private key. Therefore, only the owner of the private key may decode communications sent to them and digitally sign messages sent from them.

**Analysis:** However, RSA authentication and privacy are one-way and RSA procedures are computationally costly. To engage in secure, mutually authenticated two-way communication, each participant must possess an RSA key. A scenario like this is very improbable and usually impossible to realise. Thus, TLS employs RSA together with other, more efficient symmetric-key methods.

**Application:** The Tor network's key exchange mechanism is TLS. In this protocol, the receiver signs his reply to the sender with his private RSA key and then sends back a dynamically created (temporary) DH key to the sender. Once the sender verifies the digital signature, the recipient may be certain that they are receiving a genuine message. Consequently, the client may validate the trustworthiness of responses received from Tor relays in the circuit or the final web server.

### *C. Elliptic-curve Cryptography (ECC)*

**Background:** One alternative to RSA for public-key cryptography is elliptic curve cryptography (ECC). The ECC algorithm is based on the fact that it is impossible to determine the discrete logarithm of an element of a random elliptical curve with respect to a publicly known base point, in contrast to the RSA algorithm. The issue may be categorised as a discrete logarithm on an elliptic curve (ECDLP).

**Analysis:** ECC's introduction of a smaller key size is a major advantage. According to the most up-to-date NIST guidelines, the security provided by ECC with a critical size of 160 bits is on par with that provided by RSA and Diffie-Hellman with a critical size of 1024 bits. Also, RSA/2048 DH's bits is equivalent to ECC's 224 bits. It's important to remember that ECC keys grow at a far more leisurely pace than RSA/DH keys do. ECC provides a higher level of protection for a given key size than RSA. [12].

**Application:** Presently, Tor is switching from a TLS implementation based on RSA to one based on elliptic curves. The existing relays support TLS authentication through RSA and

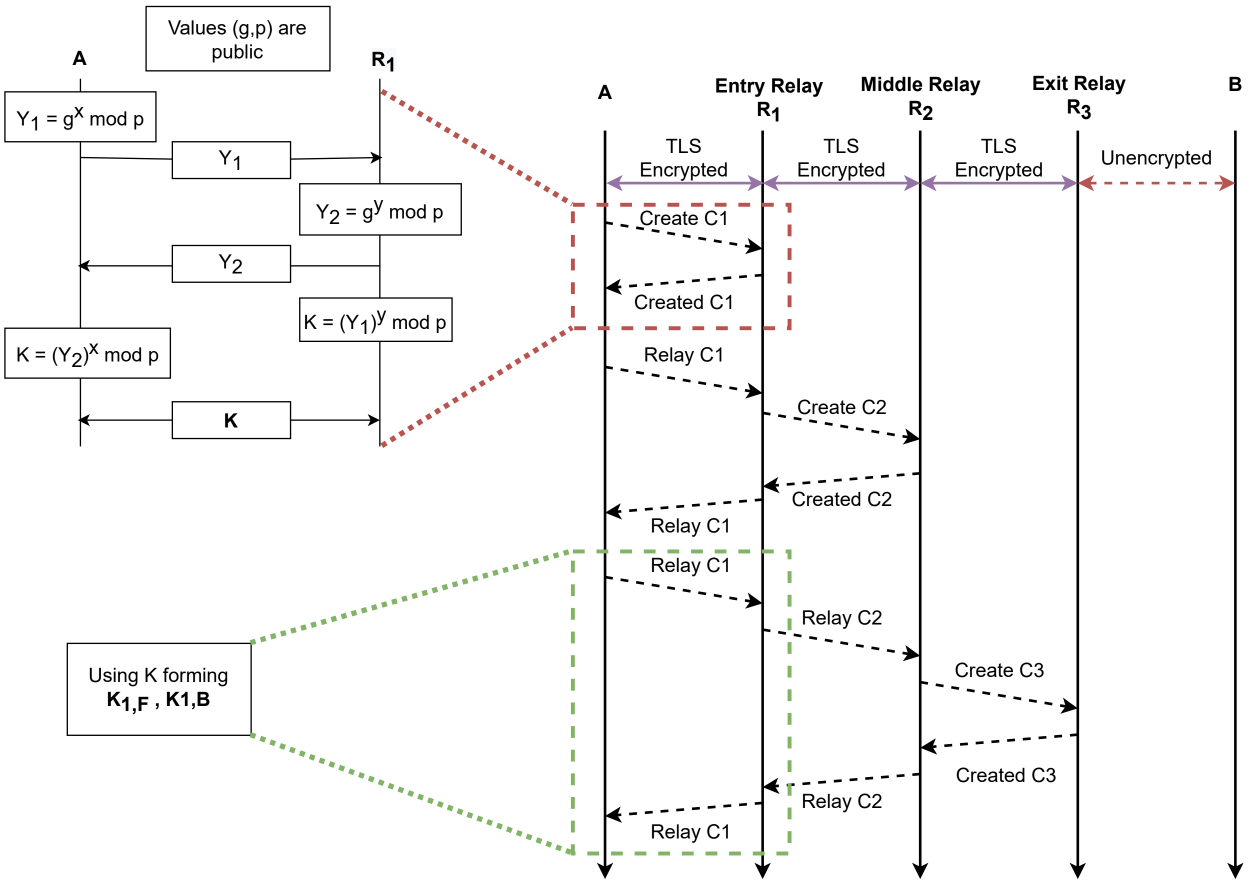


Fig. 3: Flow of the complete establishment of a circuit between A (Tor client) and B (server) using a Secure Sockets (SOCKS) interface, which multiplexes TCP traffic. Here C1, C2, and C3 are different connections. While X & Y is a secret message shared among parties.

elliptic curve public keys. Using ECC instead is expected to lessen the burden on computers and give more reliable safety.

## IV. Attacks against Tor Networks

Tor's online privacy and anonymity objectives are vulnerable to assault from a number of directions. The writers here describe the many threats to the Tor network and the countermeasures that may be taken. Figure 4 provides a catalogue of these assaults.

### A. *Social Engineering Attack*

**Threat:** When it comes to deanonymization, human error is a major issue. However, if a user's identity and location are exposed online, the anonymity and security that Tor provides against traffic analysis becomes ineffective. Users' digital footprints may be exploited to deanonymize them and expose their actions.

**Defense Strategy**: Although the Tor Project advises against it, proxies, VPNs, and Tor cannot protect you from having to enter your name or other identifying information into a website's form.

### B. *Malware Attack*

**Threat:** Prior to sending their traffic via Tor, Tor users' anonymity and privacy may be compromised by spyware, backdoors, and other malicious software. Imagine an attacker has the ability to covertly install or run the virus on the user's PC by using Tor. This might lead to the user's IP address being exposed or their machine being compromised.

**Defense Strategy:** On several fronts, Tor is able to counter this danger. To begin, the pre-installed NoScript plugin in the Tor Browser Bundle will only load scripts from trusted sites. For another, the TBB is always on the most recent version of Firefox and is accompanied by the developers' own security upgrades. Tor provides users with the Tails operating system as a last option. Tails uses Debian GNU/Linux as its basis, reroutes all network traffic over the anonymity network Tor by default, and boots from a rewritable RAM drive. The constant updates to Tor, Firefox, and Tails make it incredibly difficult for enemies to conduct widespread, successful assaults.

### C. *Predecessor Attack*

**Threat:** The first-hop relay has the user's IP address and may reveal their true identity. However, if an attacker controls the first relay in the circuit, they will know the user's IP address and might potentially identify the user's destination

Social Engineering Attack
Cryptographic Attacks
Malware Attack
Types of Attacks
Wiretapping Attack
Predecessor Attack
Timing Attacks

Fig. 4: Attacks in Tor Network

outside of Tor. An adversary might use this information to track users or prevent them from accessing the Tor network.

**Defense Strategy:** Tor tries to prevent this using entry guards. Guard relays are identified by directory servers as having superior speed and reliability. Tor client randomly selects guards from the pool and then uses only those guard-selected relays as the first node in each given circuit [4]. The goal of this selection is to shield users against the ”predecessor attack,” in which an adversary may achieve end-to-end correlation and deanonymize a user if compromised relays are selected for the first phase of the assault. A forerunner to the current circuit [13]. This strategy makes it more difficult for an attacker to connect relays to the Tor network and quickly begin monitoring users, since relays are originally simply the middle hop.

### D. *Timing Attacks*

**Threat:** Tor does not reorder or delay packets inside the network deliberately to accomplish its low-latency goal [3]. Imagine that a malicious actor has control over the Tor circuit's first hop as well as its final hop. If that's the case, the attacker could be able to launch timing assaults. The attacker would monitor traffic leaving the exit relay by listening for packets sent from a certain IP address. After collecting enough data, they could draw a plausible conclusion about the relationship between incoming and exiting traffic. If the user's connection to the web server is not secured, this might expose their online activity and deanonymize them.

**Defense Strategy:** When Tor was launched, it was vulnerable to a timing attack, but as the network grew in size and popularity, the chances of a successful traffic analysis assault became more remote. [4].

### E. *Wiretapping Attack*

**Threat:** Due to their scarcity and the fact that they are required for the last hop of the circuit, exit relays are highly prized on the Tor network. If an opponent were to take control of an exit relay, they would be able to see every communication exiting Tor. Even if the communication is encrypted, the exit relay may be able to read the DNS search and the HTTP headers. Without encryption, it would be possible for anybody to monitor all communications.

**Defense Strategy:** Encryption is Tor's main defence against this. HTTPS Everywhere, included in the Tor Browser Bundle, gives preference to HTTPS connections when interacting with web servers. Tor is unable to enforce this extra layer of encryption, but it does regularly update its directory servers.

### F. *Cryptographic Attacks*

**Attack:** Modern cryptography is the foundation of Tor's safety. Tor communication may be decrypted if an attacker compromised a widely used method, such as AES.

**Defense Strategy:** None of the known attacks are computationally possible with the available resources, despite the fact that they are quicker than a brute-force attack on AES. If the parameters (a finite cyclic group G and a generating element g in G) are selected correctly, then the Diffie-Hellman-Merkle key exchange is safe. Assuming they are, the best publicly available method for breaking DH would be to solve the discrete logarithm issue, for which there is currently no effective technique. To understand RSA, you must first understand the challenge of factoring the product of two huge prime integers. For integers, there is presently no known public method that factors in polynomial time. These cryptographic methods have a good reputation for safety.

## V. Performance Based Location Disclosure Attack

In this article, the authors present a unique remote-mounted attack that may reveal the identity of an anonymous server and anonymizing proxies. To break a user's anonymity, the authors deploy an adversary who can cause ”traffic oscillations” in a certain channel and then ”trickle” towards the user, as shown in Fig. 5. The authors consider five servers deployed across the globe in Germany, Hong Kong, China, Netherlands, USA. The authors introduce our tor relay (shown in black, deployed in India) to access the packets going through it. The compromised tor relay acts primarily as the middle relay. So it does not have any information on the information transmitted from tor client to the server. Our tor relay employs a single-end device that can monitor the total download time using LinkWidth [14]. The authors observed bandwidth and time taken to download the files from the server for three months, from June 2021 to August 2021, using Tor metrics available on Tor Project [15]. The authors analyzed our attack for three file sizes (50 KB, 1 MB, and 5 MB) from a server. Download times

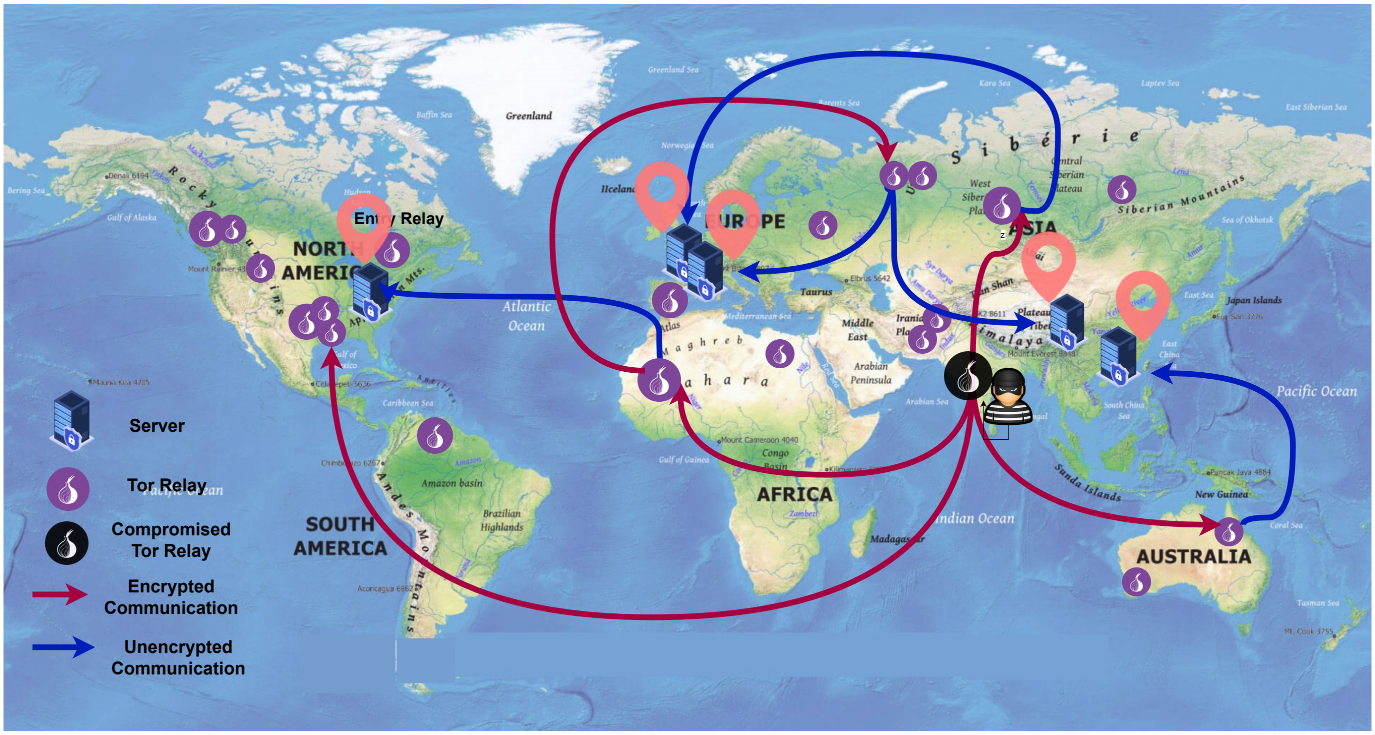


Fig. 5: Proposed Performance Based Location Disclosure Attack

include complete downloads of the shown file size and partial downloads of larger file sizes. The graph in Fig. 6 shows the range of measurements from the first to the third quartile and highlights the median.

The authors used 80% of the data for training a random forest classifier and 20% for exploratory analysis. As a meta-estimator, a random forest averages the results of many decision tree classifiers applied to different subsamples of the dataset in order to boost prediction accuracy and limit over-fitting. At the same time, 20% of the data was used to validate the classifier. The program was written in Python. The experimentation was run on MacBook Air (16 Gb, Mac OS M1 Processor).

To validate our attack, authors performed a series of experiments using different network conditions. The authors plot the Confusion Matrix to predict the server's location to the actual server location in Fig. 6. The exact server location was made available by Tor Project at [15]. Figure 6 (a) compares the predicted server location to the actual server location when the download file size is 50 KB. It can be observed from the figure that more than 80% sites were correctly identified, whereas 20% wrong predictions belonged to geographically closer areas, such as Germany and Netherlands or China and Hong Kong. At the same time, all requests from US servers were correctly classified. Figures 6 (b) and 6 (c) compare the predicted server location to the actual server location when the download file size is 1 MB and 5 MB, respectively. In comparison to Fig. 6(a), the prediction accuracy in Fig. 6(b) and Fig 6 (c) drops. Still, our model performs reasonably well, achieving an accuracy of 77% and 73%. Among Figures 6 (a), (b), and (c), the authors also observe that as the file size increases, it is much harder to identify the location of the server that is communicating the files. This is primarily due to the Tor network's ability to change the circuits with increased time duration.

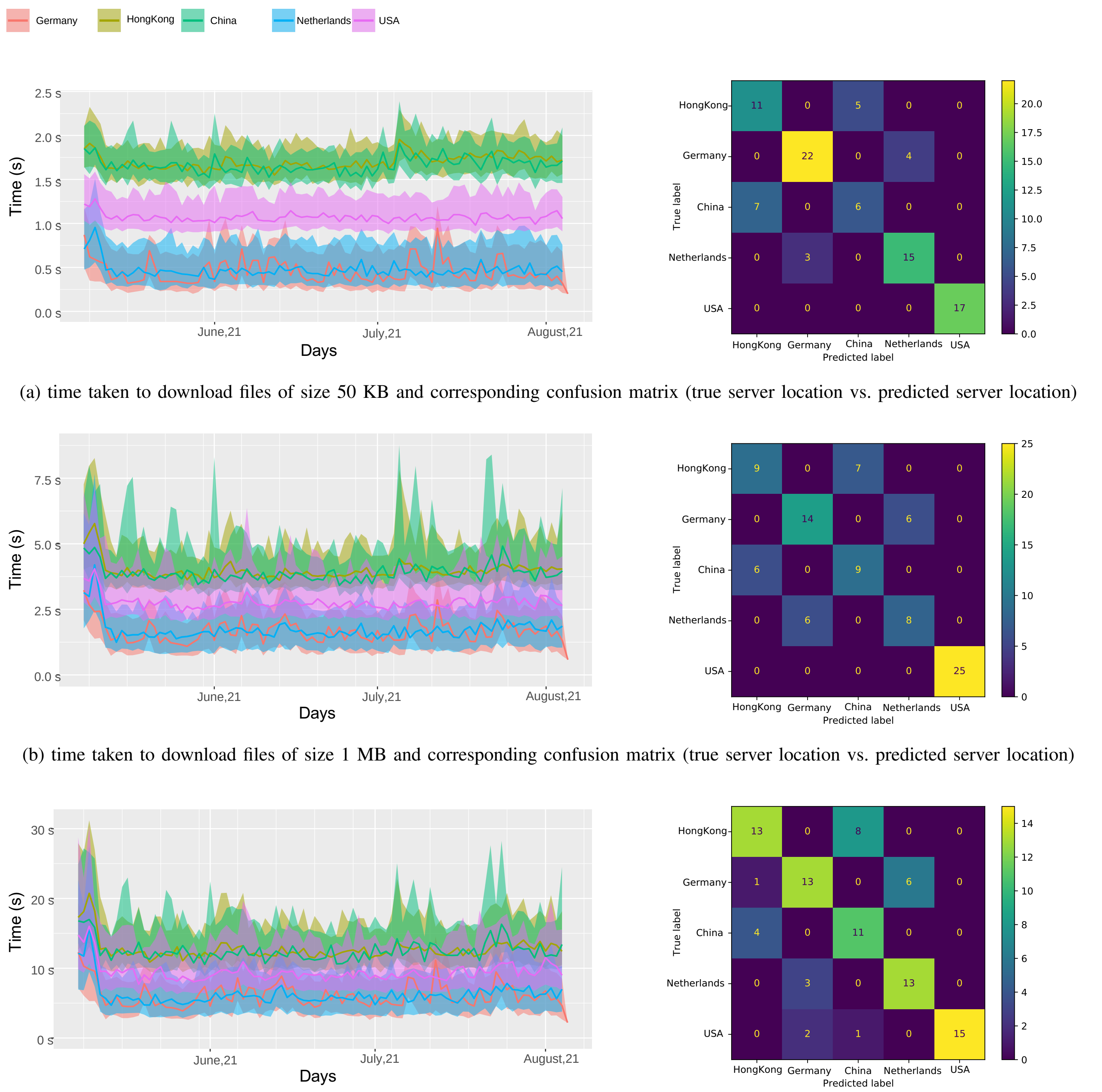
Our method of attack may be used against Onion Routing-based anonymity systems that prioritise low latency. Onion routing anonymizing technologies like Tor are exemplary.

## VI. Future Work

Tor allows communication over the Internet that enables online anonymity. Tor transmits internet traffic via a network of thousands of relays. Tor, at its heart, only provides anonymity at the network level. It will not assist in situations of identity revelation caused by computer programs. "For instance, when a user connects to Gmail, the computer or device they are using remembers their identity, so they do not have to log in again in the future." This will be stored on Tor, minimizing privacy. Second, surfing using Tor may be quite sluggish, so it is doubtful that many people would choose to switch to it. To reach their destinations, data packets traverse a roundabout path via Tor, bouncing between the machines of many volunteers. Since these technologies may be exploited to identify the machine being used, Tor removes most of the display and customization material of websites, making them seem like they were created decades ago. Therefore, there are several areas in which the Tor-based network demands improvement and innovation.

## VII. Acknowledgement

This work is partially supported by UK Engineering, and Physical Sciences Research Council (EPSRC) Grants Ref.



(a) time taken to download files of size 50 KB and corresponding confusion matrix (true server location vs. predicted server location)

(b) time taken to download files of size 1 MB and corresponding confusion matrix (true server location vs. predicted server location)

(c) time taken to download files of size 5 MB and corresponding confusion matrix (true server location vs. predicted server location)

Fig. 6: The graph shows the overall performance of time taken (in a sec) [shown in y axis] to download static files of different sizes (50 KB (a), 1 MB (b), and 5 MB (c)) over Tor from a server located as locations in Germany, HongKong, China, Netherlands, USA on the public Internet-connected via tor network. The right side of the figure shows the prediction results using a confusion matrix corresponding to the prediction of the server's location.

EP/M026981/1, EP/T021063/1, EP/T024917/1, and partially supported by 2022 Google Ph.D. Fellowship.

## VIII. Conclusion

Tor routes its traffic through a network of randomized relays worldwide to protect end-user privacy and anonymity. All data is encrypted using the session keys of each relay and decoded in an onion structure before being sent. So even a single compromised relay cannot correlate end-users with their activities. Compared to proxies or VPNs, Tor is much more resistant to adversaries because of this. It is one of the most trusted and commonly used defence mechanisms against network monitoring, data mining, and censorship.

## References

[1] G. Perrone, M. Vecchio, R. Pecori, R. Giaffreda *et al.*, "The day after mirai: A survey on mqtt security solutions after the largest cyber-attack carried out through an army of iot devices." in *IoTBDS*, 2017, pp. 246–253.

[2] A. Ahmed, A. R. Javed, Z. Jalil, G. Srivastava, and T. R. Gadekallu, "Privacy of web browsers: a challenge in digital forensics," in *International Conference on Genetic and Evolutionary Computing*. Springer, 2021, pp. 493–504.

[3] D. McCoy, K. Bauer, D. Grunwald, T. Kohno, and D. Sicker, "Shining light in dark places: Understanding the tor network," in *International symposium on privacy enhancing technologies symposium*. Springer, 2008, pp. 63–76.

[4] L. Xin and W. Neng, "Design improvement for tor against low-cost traffic attack and low-resource routing attack," in *2009 WRI International Conference on Communications and Mobile Computing*, vol. 3. IEEE, 2009, pp. 549–554.

[5] U. M. Maurer and S. Wolf, "The diffie–hellman protocol," *Designs, Codes and Cryptography*, vol. 19, no. 2, pp. 147–171, 2000.

[6] Z. Ling, J. Luo, W. Yu, X. Fu, W. Jia, and W. Zhao, "Protocol-level attacks against tor," *Computer Networks*, vol. 57, no. 4, pp. 869–886, 2013.

[7] "Tor security," https://github.com/Jesse-V/tor-security-paper.html, accessed: 2021-08-09.

[8] E. Rescorla and T. Dierks, "The transport layer security (tls) protocol version 1.3," 2018.

[9] K. Bauer, D. McCoy, D. Grunwald, T. Kohno, and D. Sicker, "Low-resource routing attacks against tor," in *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, 2007, pp. 11–20.

[10] A. Bogdanov, D. Khovratovich, and C. Rechberger, "Biclique cryptanalysis of the full aes," in *International conference on the theory and application of cryptology and information security*. Springer, 2011, pp. 344–371.

[11] A. Biryukov, D. Khovratovich, and I. Nikolić, "Distinguisher and related-key attack on the full aes-256," in *Annual International Cryptology Conference*. Springer, 2009, pp. 231–249.

[12] M. S. Farash, S. A. Chaudhry, M. Heydari, S. M. Sajad Sadough, S. Kumari, and M. K. Khan, "A lightweight anonymous authentication scheme for consumer roaming in ubiquitous networks with provable security," *International Journal of Communication Systems*, vol. 30, no. 4, p. e3019, 2017.

[13] M. K. Wright, M. Adler, B. N. Levine, and C. Shields, "The predecessor attack: An analysis of a threat to anonymous communications systems," *ACM Transactions on Information and System Security (TISSEC)*, vol. 7, no. 4, pp. 489–522, 2004.

[14] S. Chakravarty, A. Stavrou, and A. D. Keromytis, "Linkwidth: a method to measure link capacity and available bandwidth using single-end probes," 2008.

[15] "Tor metrics," https://metrics.torproject.org/torperf.html, accessed: 2021-08-09.

Gaurang Bansal is a doctoral researcher at the National University of Singapore (NUS) under Prof. Biplab Sikdar at Department of Electrical and Computer Engineering. He is a recipient of prestigious Google PhD Fellowship and NUS President Graduate Fellowship. Previously, he had completed his Master's and Bachelor's from BITS Pilani in 2020 2018, respectively. His research interests include cryptography, security, algorithms, blockchain, and IoT. He has extensive number of publications published in top tier conferences and journals such as IEEE Network Magazine, IEEE Transactions on Vehicular Technology, IEEE INFOCOM and more. Previously, he has organised and co-chaired various reputed workshops like IEEE Globecom IEEE PERCOM, and IEEE INFOCOM. He is as serving as Editor for ACM XRDS Magazine and member of Internet Engineering Task Force (IETF).

Vinay Chamola is an Assistant Professor in the Electrical and Electronics Department, Birla Institute of Technology & Science (BITS), Pilani, India, and is also a part of APPCAIR, BITS-Pilani. He received his B.E. (2010) and M.E. (2013) degrees from BITS, Pilani and PhD (2016) from National University of Singapore (NUS), Singapore. His research interests include Internet of Things, 5G network provisioning, Blockchain and Security. He has over 100 publications in high ranked SCI Journals including more than 60 IEEE Transaction, Journal and Magazine articles. He is an Area Editor of Ad Hoc Networks, Elsevier and the IEEE Internet of Things Magazine. He also serves as Associate editor in various journals like IEEE Networking Letters, IET Networks, IET Quantum Communications etc.

Muhammad Khurram Khan (Senior Member, IEEE) received the Ph.D. degree with focus on security and privacy from Southwest Jiaotong University, Chengdu, China, in 2007.,He is currently working as a Full Professor with the Center of Excellence in Information Assurance, King Saud University, Riyadh, Saudi Arabia. He is the Founder and CEO of the Global Foundation for Cyber Studies and Research, Washington, DC, USA. He has edited seven books and proceedings published by Springer-Verlag and IEEE. He has authored or coauthored more than 450 papers in international journals and conferences, and he is an inventor of ten U.S./PCT patents. His current research interests include cybersecurity, biometrics, multimedia security, and digital authentication.,Prof. Khan is the Editor-in-Chief of Telecommunication Systems (Springer-Nature). He is a Full-Time Editor/Associate Editor for several international journals/magazines, including IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, IEEE Communications Magazine, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, Journal of Network and Computer Applications (Elsevier), IEEE ACCESS, and IEEE Consumer Electronics Magazine. He is a Fellow of the Institution of Engineering and Technology, U.K., the British Computer Society, U.K., and the Future Technology Research Association International, South Korea, and a member of the IEEE Technical Committee on Security and Privacy and IEEE Cybersecurity Community.

Amir Hussain received the B.Eng. and Ph.D. degrees in electronic and electrical engineering from the University of Strathclyde, Glasgow, U.K., in 1992 and 1997, respectively.,Following post-doctoral and senior academic positions at the University of the West of Scotland, Paisley, U.K., from 1996 to 1998, the University of Dundee, Dundee, U.K., from 1998 to 2000, and the University of Stirling, Stirling, U.K., from 2000 to 2018, respectively, he joined Edinburgh Napier University, Edinburgh, U.K., as the Founding Head of the Cognitive Big Data and Cybersecurity (CogBiD) Research Laboratory and the Centre for AI and Data Science. His research interests include cognitive computation, machine learning, and computer vision.