Article

Not peer-reviewed version

Cryptographically Upgrading TOR Network to Enforce Anonymity by Enhancing Security and Improving Performances

mohamed chahine ghanem *

Posted Date: 14 January 2025

doi: 10.20944/preprints202307.0982.v2

Keywords: TOR; Online Anonymity; Onion Routing; Multi-layer encryption; authenticated- encryption; ExperimenTOR; JAVA Crypto; Circuit Construction; Routing protocols; Cell Encapsulation; AES; OCB

Article

Cryptographically Upgrading TOR Network to Enforce Anonymity by Enhancing Security and Improving Performances

Mohamed Chahine Ghanem

Affiliation: mohamed.chahine.ghanem@liverpool.ac.uk

Abstract: The Onion Route Network (also called TOR) is by far the most efficient and widely used anonymity platform with millions of users daily and an expanding size and capacities. Since its public deployment in 2002, the Onion Routing network (also known as TOR) has maintained its leading position and dozens of propositions aiming to improve its performance and enhance the security (anonymity and privacy) have been made. Given the significance of this research area, this work seek to contribute into the improvement of TOR by investigating and testing revolutionary cryptographic and routing mechanism. This work is justified by the current TOR vulnerability and observed weaknesses, and set the challenging aim of covering these security flaws by proposing the relevant security and performances improvement such as the authenticated-encryption for onion construction, the lightened onion encapsulation approach and the secured circuit selection and cell routing mechanisms. The obtained results from implementing the proposed improvements and testing them into a TOR-like simulation platform permit to validate not only, the performances and security contributions brought by the improvement, but also the suitability of their potential implementation into the real TOR network.

Keywords: TOR; online anonymity; onion routing; multi-layer encryption; authenticated-encryption; ExperimenTOR; JAVA crypto; circuit construction; routing protocols; cell encapsulation; AES; OCB

1. Introduction

The implementation of the second generation of TOR didn't cover appropriately all the security issues, in fact, some TOR protocols remained vulnerable and especially TOR authentication protocol (TAP) and some others improvements were introduced just after that Zhang proved that the security of TAP does not imply the security of the overall system especially against concurrent execution attack (Zhang, 2009). In fact, as the TOR protocol security will not be guarantee by only the sequential execution of multiple TAP but also by securing the paths construction mechanism and most important the multilayers encryption and processing algorithms (Benmeziane et al, 2011).

The most important current challenge which TOR is facing are resisting to internet traffic analysis which could lead to linking several intercepted communications to the corresponding parties or linking multiple communications to a user (Nia et al., 2014). To deal with these emerging challenges some improvements should be introduced to TOR. Securing OR modules consists mainly of two parts: firstly, securing the onion construction algorithm, and secondly, implementing a secure way for key exchange such as "the one-way authenticated key exchange protocol" (1WAKE) which was introduced for the first time, in 2011, by Goldberg et al. (Backes et al., 2012).

Previous works had already tackled the integrity issue and concluded that TOR can guarantee an end-to-end rather than hop-to-hop integrity (Backes et al., 2012). This work will focus on Onion construction algorithms which typically use several layers of symmetric encryptions proceeded and followed by operations and keys exchange protected by Public Key encryption along with the use of integrity and authenticity check mechanisms, such as Hashing and MAC.

The aims behind this work is to study formally and academically the existing onion wrapping and unwrapping algorithms over them four core properties: correctness, security of statefulness, synchronicity and cipher-text in order to identify the current weakness reasons and try to introduce

 doi:10.20944/preprints202307.0982.v2

some improvement to the existing encryption policy in order to enforce which will be the main topic in of this work (Kate & Goldberg, 2010).

2. Research Background and Rational

As TOR imposed itself as the universally uncontested online anonymous communication tool and attract more and more user across the world for both professional and personal use. Therefore, the security of this network should be formally reviewed and some components adopted by TOR to have to be improved to guarantee security and enhance performances.

The aims targeted by this work are to, firstly, study cryptographically the existing TOR's cryptosystem and improve the existing solution in order to cover all the existing vulnerabilities that are making TOR a target for different kind of attacks. Secondly, reduce the current delays that TOR is experiencing by balancing and optimizing the use of resources and especially those responsible of performing encryption/decryption and routing activities. Finally, improving the network overall security and traffic illusion over the internet by introducing a new way of defining routing circuit and also introduce the sessional multi-path routing for TOR cells to enhance security and eliminate the risk of successfully de-anonymise or link TOR traffic to one or more user.

The primary objectives of this work are related to the improvement of the current TOR cryptosystem by enforcing the existing end-to-end integrity with a node-to-node authentication checking mechanism. This approach will help to resolve the current weaknesses of data authenticity within TOR without using the heavyweight control methods proposed by Backes et al. (2012). The multi-layers encryption called "Onion Encryption" will be improved by introducing AES-OCB (Off-Set Code Book mode) which is a block cipher encryption allowing the achievement of both confidentiality (privacy) and Authenticity simultaneously, it is also a very efficient algorithm especially by allowing a perfect parallelism in operations and working in online mode. The TOR crypto-system should also be improved by introducing a new way of constructing the Onion (multi-layered TOR cells encapsulation and encryption) and exchange keys between different parties (OP, ORs, DA).

The second set of objectives targets the performance improvement and the reduction of delays caused by the high level and useless redundancy of encryptions implemented especially on Onion construction (encapsulation). Despite the fact that TOR network is considered as a "low latency", the heavy-weight encryption and the routing an untrusted mean of communication. Thus, Anonymity system became crucial tools for a variety of targeted people aiming to protect themselves when using internet.

2.1. Online Privacy

Privacy is a major concern or all Internet users and is becoming more difficult to set a fixed perimeter of privacy online. Internet users' privacy is controversial aspect which is still being debated by the internet community, government and NGOs. users assumptions about the control level they have control over their private information is usually wrong and the best example is when they engage in online activities such as online social networking which is essentially based upon sharing of private information but they consent of disclosing such information to known or unknown parts. Moreover, over years and the importance that constitute such information, entire organisation are devoted to compromise the user privacy for different purposes varying from legal to criminal and from business to national security.

2.2. Online Anonymity

Online anonymity is occur when an internet user is not identifiable (distinguishable) within a set of other users, Online Anonymity is righteous and necessary in many scenarios, such as protecting Internet user privacy, improving system security, bypassing Internet censorship, satisfying some antivirus requirement, and protecting Internet users' computer from hackers' attacks. With the wider use of the Internet and more improving computer hacker technologies, the Internet's users are anxious

 doi:10.20944/preprints202307.0982.v2

to gain more powerful ability to keep their privacy and security. They want to request a better network tool or technology to protect their private information from being monitored by the Internet sniffers and hackers. With this requirement, anonymous communication has become more and more popular on the Internet (Kate & Goldberg, 2010).

Anonymity is righteous and necessary in many scenarios, such as protecting Internet user privacy, improving system security, bypassing Internet censorship, satisfying some antivirus requirement, and protecting Internet users' computer from hackers' attacks. With the wider use of the Internet and more improving computer hacker technologies, the Internet's users are anxious to gain more powerful ability to keep their privacy and security. They want to request a better network tool or technology to protect their private information from being monitored by the Internet sniffers and hackers. With this requirement, anonymous communication has become more and more popular on the Internet (Kate & Goldberg, 2010).

2.3. Anonymous Communication Networks

With more and more sensitive and private communications transiting over the Internet, there was a need to develop platform and solution which can guarantee and maintain the privacy and security of these communications. (Dingledine et al., 2014). Anonymity communication networks (ACNs) emerged as a the perfect solution allowing people to conceal their identities online by guaranteeing the un-linkability of the users' IP addresses, their digital fingerprint, and their online activities (Pfitzmann & Hansen, 2008). Anonymous communication networks became an essential component in the nowadays challenging cyber security world. Overall, there two main categories of ACN; high-latency systems and low-latency systems.

High-latency systems such as Babel and Mixmaster tend to sacrifice performance to ensure a perfect security especially against type of threads called correlation attacks in which Eve monitor Alice's packets timing and patterns to perform a deep and complex statistical analysis aiming to match both part of each communication, these systems use advanced techniques such as delaying packets and injecting invalid packets in order to gather as much information they can get to produce an accurate time or pattern matching. From their part, the developer of this solution are constantly introducing heavyweight protection technologies, thus the transmission delay (latency) on this solution is high making this solution unsuitable for sensitive transmission activities such as HTTPS in which the time accuracy is crucial. High-latency anonymity networks can only guarantee the anonymity for applications which can tolerate an intentional delays such as e-mails and blogs (Danezis et al. 2003).

Nevertheless, as the major part of nowadays online activities is interactive such as browsing the web, instant messaging and social media, the high-latency system were useless and didn't offer the intended flexibility of use. Therefore, low-latency anonymity systems have been developed to cover this issue. The low-latency systems attempt to balance the security (confidentiality, integrity, availability, anonymity and privacy) and usability

Later in 2012, a third-generation of TOR was deployed and took the place of the first one, several new functionalities were added in order to maintain the efficiency of the solution which faced hard time especially with some weaknesses revealed due to the exploits against MD5 hashing which was replaced by SHA256, DES symmetric encryption which was replaced by AES128 (Danezis et al., 2010). The sudden increase of the computing power and parallel processing was also a big issue for TOR's cryptanalysis resistance. The new version was improved by adding a revolutionary protocols and functions such as: the new Forward Secrecy protocol, new shipping protocol in which several TCP packets can share the same circuit, introduce Leaky-pipe circuit topology and a new Congestion control, new dynamic exit policies, source-to-destination integrity checking and hiding the used services. Some existing function were removed such as mixing, padding, and traffic shaping where some others were modified like the Separation of "protocol cleaning" from anonymity protocol (Dingledine et al., 2014). In UK, it is estimated that a couple of thousands of users are using TOR on

 doi:10.20944/preprints202307.0982.v2

regular basis (Figure 2) for different purposes varying from legitimate protection (activist, politics and journalist) to illegal activities (cyber-criminals, terrorists, hackers).

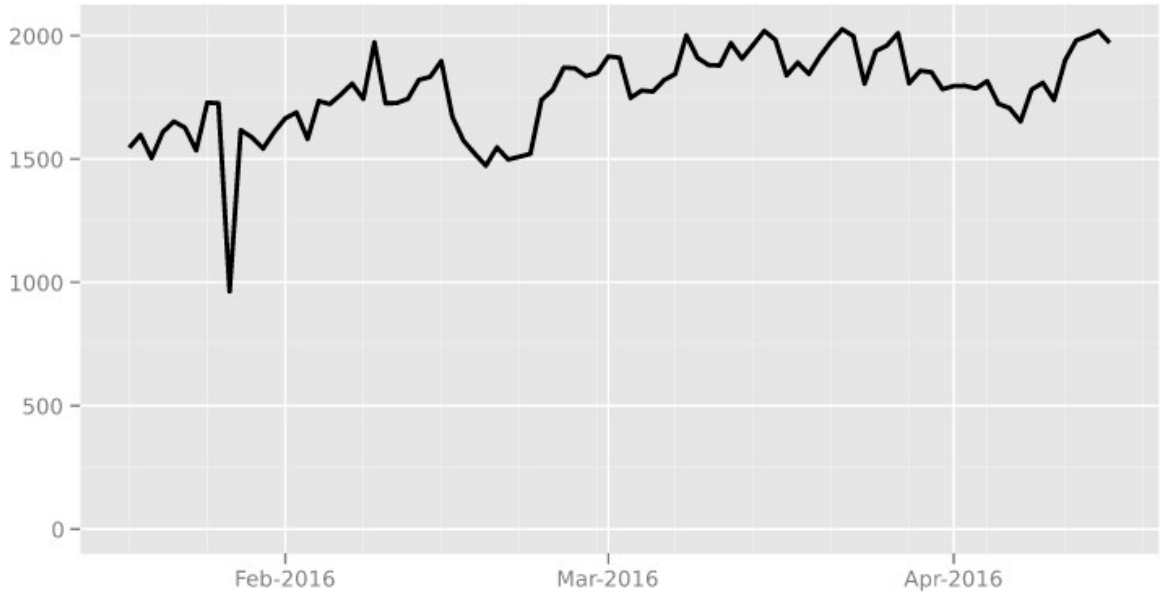

Figure 2. the number of regular TOR user in the UK for the first quarter of 2016 (TOR Project, 2016).

2.3.1. About TOR

TOR network is the most famous and widely used low-latency anonymity platform, it relies on multi-layer encryption in form of onion and special routing concepts (Reed et al., 1998) to achieve anonymity. Today, the network consists of about 6000 routers operated (run) by volunteers across the world (Tor Project, 2016) which are called Onion Routers. Each OR is a part of one or more routing path called Circuits and is identified by its contact information (IP address, ID, public keys, geographic location and other functional parameters). All these information is stored into an independent directory authorities (DA) which is responsible of the network consensus and control the network with a minimum information requirement. Onion Proxy (OP) downloads from the DA the consensus documents and the descriptors and use it to establish communication circuits through TOR network before to reach their Internet destinations. Currently, each TOR circuit is composed from three nodes (ORs): Entry guard OR, Middle OR, and Exit OR and traffic is transiting throughout the network in form of fixed-size unites called Cells. Pproposals covered different areas such as congestion, scalability, routing, and security. Nevertheless, the most urgent challenges which TOR is facing could be divided into three major domains:

- As TOR is relying on an aging (componential breakable) encryption and integrity mechanisms for which global adversaries are or will soon be able and capable of compromising. Thus, the existing pretended cryptology guarantee such as onion construction (wrapping ciphers), keys exchange protocols and the integrity and authentication checks mechanisms should be reviewed.
- The delays caused by the congestion on the network and also by the heavyweight encryption and decryption.
- The identification of TOR user through internet and traffic analysis which leads usually to linking several intercepted communication to involved parties or linking multiple communications to a single user (Nia et al., 2014).

Several academic research works are currently carried out aiming to improving the TOR network. This formal approach was initiate by Backes et al. (2012) tackling for the first time in a formal way the security of TOR and proposing some solution for onion construction (wrapping) and key exchange protocols such as the 1-WAKE.

 doi:10.20944/preprints202307.0982.v2

Regarding TOR circuits selection algorithm, research have tackled the vulnerabilities associated to the first generation of TOR circuit construction and selection algorithms, they ended by proposing (have been now implemented and applied into TOR) the concept of controlled selection Guard ORs and bandwidth/uptime caps (AlSabah & Goldberg, 2013b). Meanwhile, a particularly interesting research carried by Snader and Borisov (2012) in which they proposed two major enhancements; Firstly, they recommended after experimenting that bandwidth value which will be used for selecting the circuit should be measured centrally by the directory server using opportunistically sampling function rather than getting this value from ORs by allowing them to report their own capacity which could be misleading. Secondly, they propose a routing function to be integrated into the circuit selection algorithm which weights faster ORs more heavily but leave the user the choice between secure anonymity and performance.

Nia et al. (2014) proposed a novel anti-detection mechanism which used a function inside the crypto-system for data pattern generation and timely propagate the streams of data. Despite the effectiveness of the proposed system, it present some difficulties in its implementation on the real world TOR and thus an eventual implementation could lead to radical changes on TOR which is risky. On the same year, Haraty and Zantout (2014) had presented TOR cryptosystem and detailed the operating principles and features. They had presented some thread that are facing the Second Generation Onion Routing without giving any solution or proposing an improvement. Also the ultimate aims were the resistance to traffic analysis, and better hiding the users' identities. However, since first and second generation TOR system relied on slow encryption mechanism and vulnerable routing and circuit establishment functions for which global adversaries possessing adequate resources was able to compromise and thus confidentiality, integrity and authenticity of data might be threaten.

Recently, a work carried out by Lee et al. (2015) tackled the existing TOR weaknesses especially those related to selecting the routing path for cells' shipping and the risk of fake intermediate and exit nodes and proposed a mitigation policy which was not implemented. AlSabah and Goldberg (2015) in their work entitled "Performance and Security Improvements for Tor" presented a state of current research directions focusing on the Tor network. They gathered in one paper almost all the TOR known design weaknesses and security flaws and improvement challenges facing TOR and enumerates the currently unresolved issues. However, the paper neglected two key points related to TOR: the multi- layers encryption challenges and the delays which is facing the real world TOR user.

On the mathematic and cryptographic side, only few research tackled the issue of proving the security of TOR's protocols and mechanism. A notable work carried out by Backes et al. (2012) entitled "Provably Secure and Practical Onion Routing" demonstrated the security and efficiency of TOR cryptosystem by elaborating formal framework for TOR. The work was considerably rich in content and divided into two parts: in the first part they introduced a cryptographic definition of global function behind TOR called the forward secrecy which was described as the optimal and perfect "in theory" to be implemented in the real-world TOR along with the One Way Key Exchange (1WAKE) Protocol used for the anonym authentication and Onion construction algorithm. The second part of their work was about proposing some improvements in the existing TOR integrity mechanism and also introducing new way of performing onion encryption into the CTR mode. However, this work neglected some primordial aspects related to TOR real world implementation flexibility and the overall efficiency. Backes et al. (2012) research remain to date the only formal and theoretical TOR analysis and the algorithms and functions introduced were

Time correlation is a advanced type of passive attacks which attempt by using information about connection time, and flow duration to correlate them to TOR users or connection via an Entry or Exit OR connection to an external internet address (Johnson et al., 2010). This type of attack can also be active when attackers inject their own timing patterns into the traffic. Several research studied and emonstrated that this attack can be extremely effective when the adversary possess the adequate computing power and ressources which remains crucial to bypass TOE's stream multiplexing strategy (Fu &Ling, 2009).

 doi:10.20944/preprints202307.0982.v2

2.3.2. Path Selection Attacks

Several previous works have proved that it is possible to take advantage from the current TOR path selection algorithm flaws to develop efficient attacks. In fact, (Overlier & Syverson, 2006) proved mathematically and statistically that client who repeatedly generate fresh paths can becomes more vulnerable as attackers gain a higher probability of controlling both the Guard and Exist ORs into the selected circuit. Moreover, to compromise the anonymity of TOR users an attacker can inject enough compromised (fake) ORs which increase the probability that a connection start and end within compromised ORs, this attack is known as Sybil attack (Dahal et al., 2015). Moreover, a much more efficient attack variant was proposed by Ling et al., (2011) which exploited the traffic load-balancing algorithm based on ORs which assess and report on a real time base the bandwidth and transmit to the TOR directory server (Perry, 2007). The researcher demonstrated that, by artificially changing the advertised bandwidth of ORs, an attacker is able to compromise almost half of TOR traffic while controlling only 10% of the TOR network ORs. This attack was carried out recently by US researcher working with the FBI to de-anonymise successfully several TOR users accessing illegal drug selling website "silk road 2" and since that known as FBI attack.

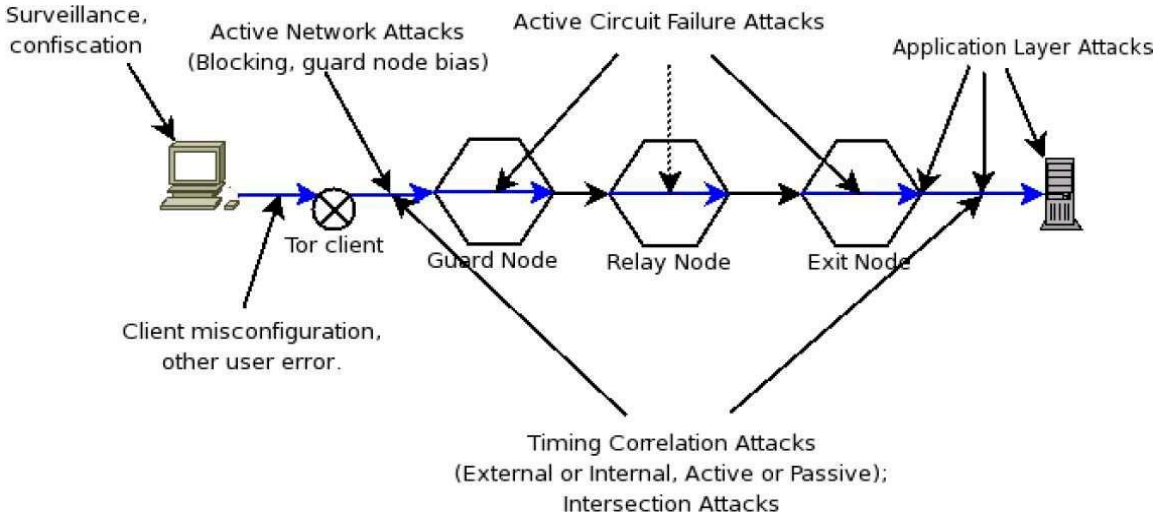

Figure 1. TOR points of attacks (Perry, 2007).

TOR still present several vulnerabilities and potential points of attack, it is obvious that the system have no protection against a global adversary who can control both end of the circuit which was later in 2015 known as FBI attack. A researcher had implemented in early 2015 an attack targeting initially the illegal drug market "silk road 2" in which the managed to introduce a number of fake ORs inside the TOR network taking advantage of their status of academic and research organisation and the available digital asset and computational power, they implemented then an attack which diverted a part of TOR legitimate and real traffic through a compromised entry and exit OR. Once having data transiting via 2 controlled OR out of 3. They launched a tinning and traffic correlation attack to determine the last OR of the circuit. At the end, researchers were able to de-anonymise several thousand of TOR users including a user suspected to be the manager of the drug market. This attack was mainly due to the unfair TOR circuit choosing and construction which leave the pseudo-random function the possibility of choosing a compromised Exit OR (Johnson et al., 2010). Moreover, the current TOR cryptosystem enable adversary who possess the computational power and technical deployment over internet to observe and intervene actively on some fraction of network traffic by deleting, replaying, modifying or forging fake network traffic over TOR could compromise some fraction of the ORs and therefore compromise some directory servers. In fact, this flaw is mainly due to the lack of node-to-node integrity check between TOR ORs which was initially scarified in favour of better multi-layer encryption performance which allow a more efficient TOR by reducing the delays caused by heavy-weight encryption.

 doi:10.20944/preprints202307.0982.v2

3. Methodology and Proposed Improvements and Enhancements

3.1. Related Improving Research

Previously, researchers had proposed plenty of improvement and enhancement techniques and methods. Where the majority were not realistic or unsuitable to be implemented on the real life situation, some on the other hand were implemented and produced the aimed security and performance goals. The perfect example of the academic research contribution into TOR is the work entitled "Anonymity and one-way authentication in key exchange protocols" (Goldberg et al., 2012) in which the TOR authentication protocol TAP was deeply reviewed to guarantee a Perfect forward secrecy. Furthermore, some others research outcomes were also implemented such as the separation of protocol cleaning from anonymity, the reviewing of mixing, padding in encryption algorithms, the use of multiple TCP streams in one circuit known as multiplexing, the local congestion control function and the enhanced end-to- end integrity checking. The following tree sketched by (AlSabah & Goldberg, 2015) illustrates and summarises the research activity on TOR.

3.2. Problem Tackled During This Works

This work look for improving TOR anonymity by enforcing some security mechanism and enhancing the performance of others. The proposed improvements will focus on improving TOR overall security and performances along with preserving and enforcing the current features which showed some weaknesses against the recent advanced attacks. This work state a clear statement in which an adversary external to TOR at any location with any capabilities should be unable to link (at large scale or locally) to link or identify any TOR user. This property is known as end-to-end unlinkability which is defined as guaranteeing the anonymity of the source regardless the destination's location. Note that the anonymity targeted is for both TOR sender anonymity and sender-receiver anonymity.

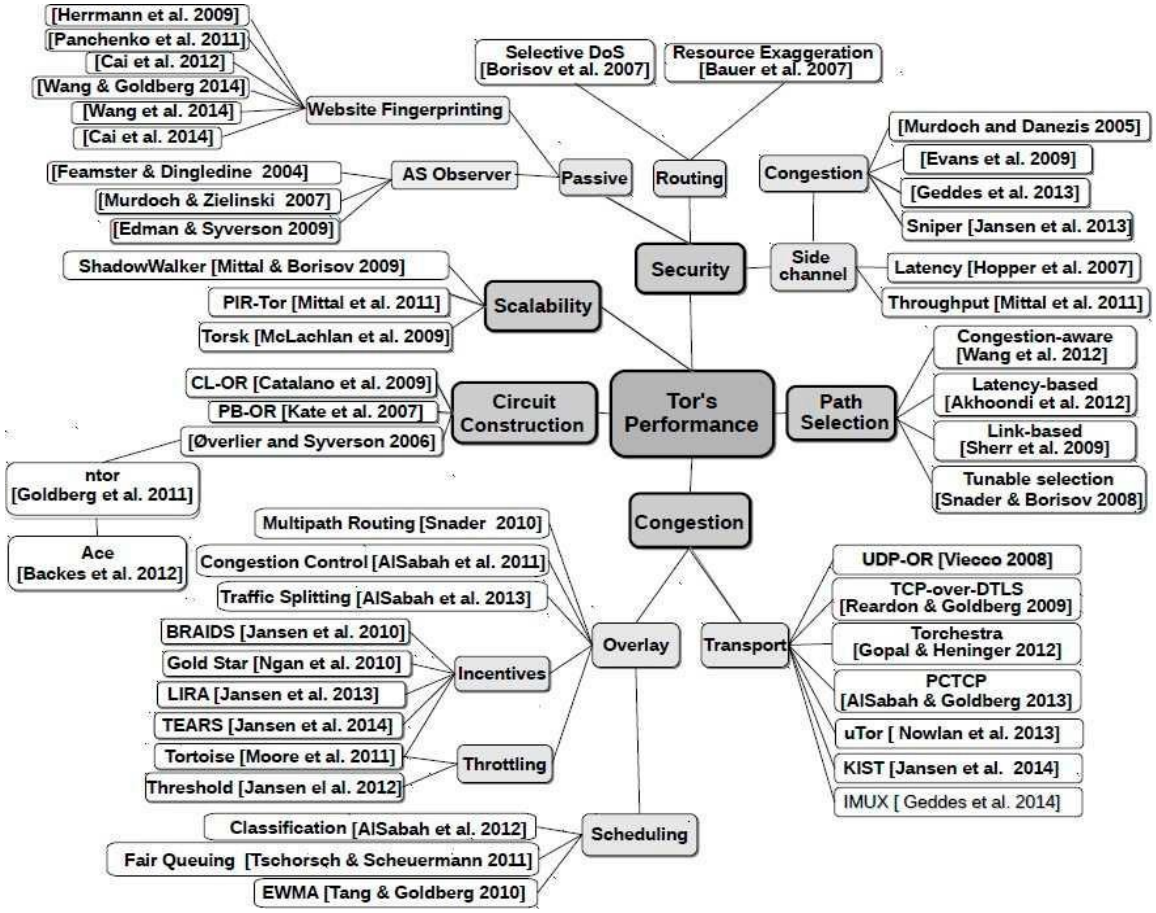

Figure 2. TOR research and related works (AlSabah et al., 2015).

 doi:10.20944/preprints202307.0982.v2

Meanwhile, to preserve TOR reputation in the future, the security design should be reviewed and adapted to emerging threat and achieve better security and performances. In fact, several TOR flaws and vulnerabilities including the newly emerged ones remains untreated, the following is part of the desired properties and capabilities to be included in TOR:

a) ***Cryptographic and performance enhancement***

The current TOR deployment, despite the fact that it is considered as a low-latency anonymity solution, is facing performance trouble which impact both the quality of the service and the platform usability and flexibility to be used in as much possible situation. Reducing delays is a priority and is closely related to the cryptography used over TOR.

b) ***Circuit selection security***

An attacker, whatever his capabilities, should not be able to redirect any TOR client to choose a compromised circuit (controlled Guard and Exit ORs) or even modify the packet header to change path without being detected. The adversary should not learn forwarding information of uncompromised OR, OR's geographic positions, or the total number of hops on a circuit (length).

c) ***Sessions' and Users' Un-linkability***

An adversary should not be able to perform timing attack to link Cells (packets) from different users or sessions, even between the same set of sources and destinations.

d) ***Network Confusion and Diffusion***

An adversary eavesdropping on multiple links in the network should not be able to correlate two or more packets and determine that are from the same user by observing the bit patterns in the packet headers or data.

e) ***Node-to-Node cells' authentication***

In addition to the existing Data **secrecy and end-to-end integrity,** inter ORs TLS authentication and Directory server authentication, TOR should include selective authentication check mechanism to identify Rogue Cells or any forged cells injected by attacker into the circuit for any intention (misusing TOR computing power and causing DoS, inject fake Cells).

3.3. Proposed Improvements and Enhancements

During this work, several improvements will be proposed covering different research directions which all aims to address the current design weaknesses, security and performances issues described in Section 3. The security is a wide term which in TOR context refer to three thing; information security (Confidentiality, Integrity and Authenticity), Anonymity and un-linkability. The information security part is tackled in this work by proposing an improved AES mode guaranteeing the Confidentiality and the Authenticity simultaneously, where the anonymity and un-linkability are tackled by reviewing the TOR wrapping mechanism, circuit construction and routing within TOR which appear to be not relevant for the security but in reality it's the most important factors for anonymity and un-link-ability. Moreover, this work tackle the performance issue in TOR and the related security flaws and vulnerabilities, as several attacks against TOR exploited the delays and the poor performances (timing attack, users' clustering, and path selection attack).

3.3.1. Multi-Layer Encryption Improvement

In this part, we assume that the AES (Advanced Encryption Standard) functioning principle is well known. Nevertheless, we will describe it in details at the appendix. The purpose of this part is to investigate the suitability of replacing the current AES implementation in CBC mode used for cells' encryption by the authenticated- encryption OCB mode (Bogdanov et al, 2014).

 doi:10.20944/preprints202307.0982.v2

OCB (offset code-book) is a new and revolutionary implementation of AES block cipher guaranteeing authentication along with the traditional confidentiality (privacy) of the user data, this type of ciphering id called *authenticated-encryption scheme.* Moreover, OCB mode is surprisingly and remarkably fast as it achieves authenticated encryption quicker and consuming almost the same duration as oAES encryption only CTR mode. Therefore, by adopting OCB the user can achieve in the cheapest way two out of three information security goals in an optimised and secure manner as OCB is considered as a simple cipher and resources efficient cipher when it comes to implementation in either hardware or software (Krovetz & Rogaway, 2014). In nowadays cryptography OCB solve three majors cipher issues:

- OCB eliminate the problem of authenticated-encryption with associated-data (AEAD),
- The OCB nonce required to encrypt and decrypt should not be necessary random as it utelise a counter,
- OCB can encrypt data of any size without padding it to any convenient-length and therefore save some precious computing power.

TOR like all others Cryptosystem present cryptographic vulnerabilities related to its corposants, and one of these is the unauthenticated Cells traveling throughout the network leaving the possibility of forging fake Cells an inject them inside the network for malicious purposes. The ORs authentication mechanism remains insufficient. To introduce a node-to-node authenticity check on TOR, two options are available; the first option is an AES mode providing both privacy and authenticity separately (CBC, CTR and others) which perform separately encrypting and then computing the associate authentication using two different keys, the cost of having authenticated-encryption is thus the cumulated cost of encrypting with the cost of MAC.

a) ***Working principle of the OCB mode***

AES-OCB is a block-cipher with a block length and a key (K) of 128 bits each. It also uses a nonce (N) of 96 bits and an associated incremented counter value (Δ). The OCB detailed working principle is as follow (algorithm in Appendix 2):

- First the plaintext M is divided into blocks of 128 bits each $M = M_1 \ldots M_m$, Here there is two cases; the data size in bit is a multiple of 128, or there is a remainder and therefore the algorithm require a padding.
- Secondly, a Checksum of 128 bits is calculated Checksum = $M_1 \oplus \ldots \oplus M_m$ and will be used later during the authentication process.
- Thirdly, an initialisation function "Init" take place and using the nonce N which is concatenated with a 32 bits constant value to produce a 128 bits value called "Top". Later, the Ktop = E_K (Top) is computed and Stretched to produce the 256 bits value Stretch = Ktop || (Ktop⊕(Ktop<<8)) (left shift by 8 positions KTop and replace the empty by zeros). The value Init(N) which is the initial value for Δ.
- Fourthly, for each block "i" the increment is called to increment the Δ, XOR with the Mi, and encrypted using the key K and the algorithm AES-OCB as showed in the scheme. Later, the output of the encryption in stage 4 is XOR again with the Δ to produce C_i... The authenticated ciphertext is $CT = C_1 C_2 \ldots C_m \; T$.

Afterward, the authentication value which is 128 bits length is computed by processing the associated data A which is XOR with the value of Δ for each block in the same way as the encryption part and later encrypted using the same key K. the result of all the blocks is XOR together to produce a 128 bits length authentication value Auth. Finally the checksum of the initial data is XOR again with the Δ, encrypted using the key K and then XOR with the authentication value Auth to produce a final authentication Tag "T" for the whole data (Krovetz & Rogaway, 2014).

 doi:10.20944/preprints202307.0982.v2

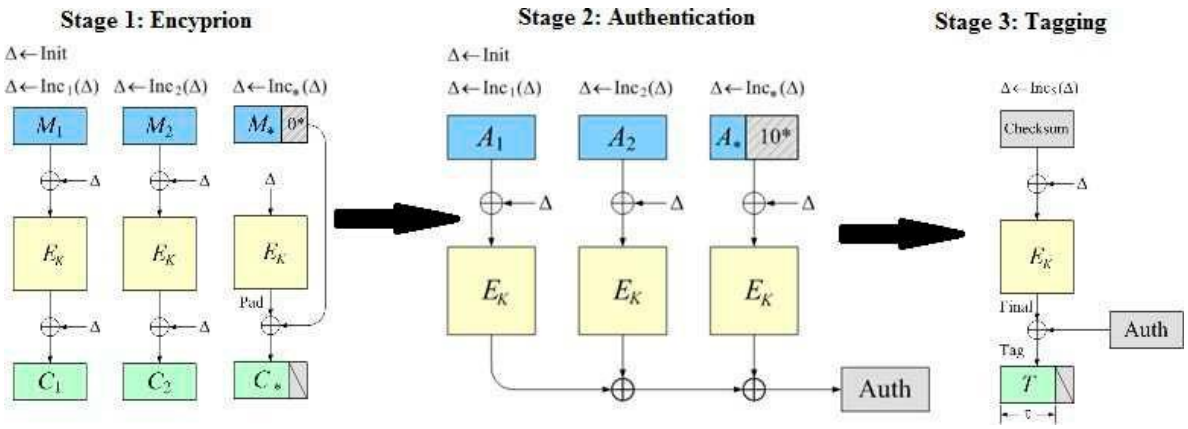

Figure 3. OCB mode functioning schemes (Bogdanov et al, 2014).

The Decryption under OCB mode is faster and simpler. By having given K, N, and CT, the receiver recover the initial message M following the normal decrypting way. Then, the authentication tag T is re-computed and compared with the received one to determine the authenticity of the received message (Krovetz & Rogaway, 2014).

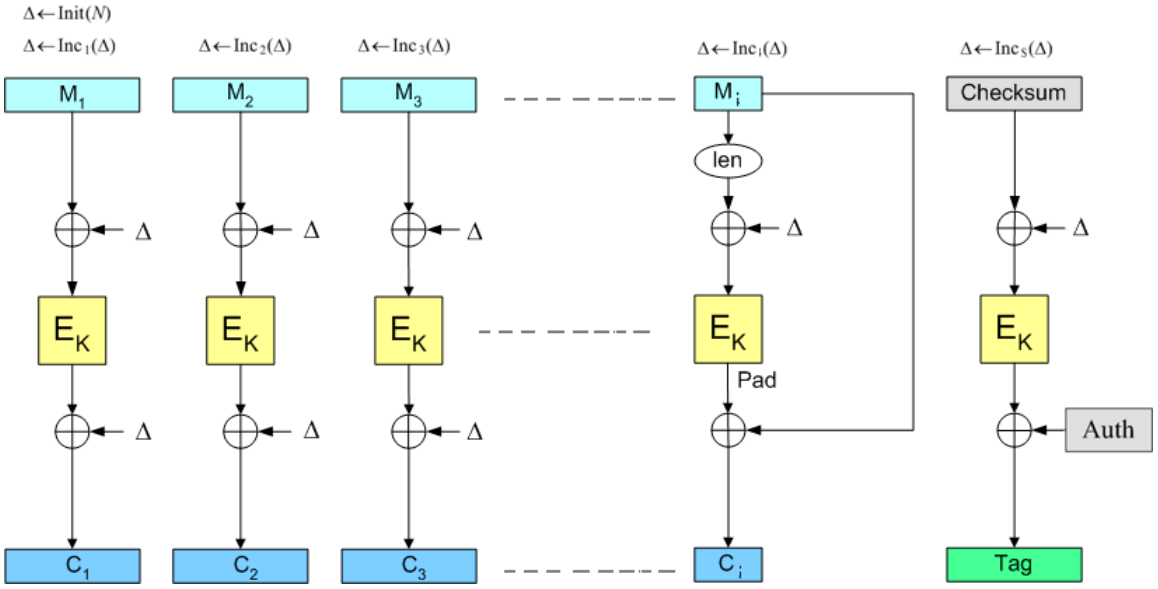

Figure 4. AES-OCB version 3 encryption and authentication operations scheme (Krovetz & Rogaway, 2014).

There is three different implementation (versions) of OCB mode. The adopted version for this work is version 3 which is the most optimised version in term of operation number and computational power. Although, the performance improvement will be relatively small, it remain crucial for TOR to adopt an authenticated-encryption for onion construction. In fact, in addition to the performance enhancement, implementing AES in OCB mode will bring the following properties:

- Fully parallelizable operations of the block ciphering can be performed simultaneously. Thus, OCB is very efficient and suitable for hardware encrypting at high network speeds.
- Block-ciphering scheme make it strong and resist better to the new timing attacks which the other mode like CBC would be vulnerable.
- OCB is a single key scheme as it use the same key for encryption and authentication which make it more efficient in term of memory use.
- OCB can process any data size without requiring it to be a multiple of the block length. Moreover, no external padding function is used and thus it economise time as there is no bits-waste in the ciphertext due to padding.
- The main computational function used beyond the block-ciphering is XOR which is very

 doi:10.20944/preprints202307.0982.v2

time and power efficient function (three 128 bits XOR per block).

- OCB can be perfectly used into memory-limited systems as the main memory cost the amount needed to hold the AES sub-keys.

b) ***Choice justification***

An authenticated-encryption scheme enable two parties sharing a secret symmetric key to communicate in a manner that ensures both privacy and authenticity. AES implementation in OCB mode is designed to be time (time consumed to perform encryption) and resources (processor and memory) efficient in both software and hardware. In fact, the algorithm is perfectly adapted to restricted environments requiring accuracy and pseudo-synchronization along with providing provable security and authenticity (Krovetz & Rogaway, 2014). During this work we will start by assessing OCB Security and performance versus others competitor integrated authenticated-encryption modes. The use of an incremented nonce for each encryption and the decryption process by OCB is one of the major strength. In fact, It is required that the nonce should be unique (not necessary random, secret or unpredictable) for each message but OCB rely on a counter value which ensure that each nonce is different. Thus, the importance of the unicity of the nonce is crucial to maintaining perfect authenticity and privacy.

On the other hand, OCB competitor scheme and particularly CCM and GCM which offer an integrated authentication along with encryption will be assessed in similar testing environments during this work. Moreover, the traditional approach for achieving authenticated-encryption which rely on composite functions (encryption following by MAC or MAC followed by encryption) will be assessed alongside with OCB competitors evaluation, two implementations of both CBC and CTR mode followed by MAC computing will be performed to serve as a reference on the performance evaluation. In this work the implementation of CBC and CTR mode will not use separate keys, in fact we will use the same 128 bits key for encrypting the plaintext and then to calculate the associated MAC of the resulting ciphertext. Nevertheless, for security measure the CBC IV (Initialisation Vector) will not be derived from the key but instead it will be generated using a different function. The following table, summarize the different mode of implementation and the picked candidates for potential adoption instead of the existing CBC/CTR and are: OCB CCM, GCM, and EAX:

c) ***Cryptographic features comparison against competitors***

To evaluate the features of each candidate in addition to the practical performances, this work rely on the following points to determine the suitability of the authenticated encryption mode, the features are summarised in the following table and divided into three major part:

Table 1. Authenticated encryption features comparison (Krovetz & Rogaway, 2014).

Feature	CCM	GCM	OCB
Security Proved	Yes	Yes	Yes
Online ability	No	Yes	Yes
Key requirement	128 bits block size	128 or 64 bits block size	128 or 64 bits block size

- **Provably secure**: all the three modes are proved to be mathematically secure by assuming that the used with block cipher (AES) is pseudorandom permutation. As far as the cryptography permit, AES is proved secure and thus both three modes of implementation are absolutely secure.
- **Online message processing**: this feature is crucial for the suitability of the mode as the modes should be able to process data without knowing the whole length in advance as the TOR have no pre-set or pre-defined data length. Moreover, this feature is highly desired for a memory restricted environment which is the case of ORs in this part, CCM mode fail to achieve the set baselines.
- **Cipher requirements:** CCM mode is developed to only work with ciphers using block size of 128 bits, while GCM and OCB can work with cipher using different block size (64/128 bits).

 doi:10.20944/preprints202307.0982.v2

Nevertheless, this feature will not affect the CCM mode as the block size in TOR is 128 bits which is anyway more efficient and better for performances.

3.3.2. The Encapsulation Approach (Onion Wrapping Method)

TOR use Cells as mean of transporting TCP/IP data throughout the network to the exit OR which will be in charge of transmitting it to the destination under TCP/IP protocol. Currently, TOR perform a multi-layer encryption- encapsulation which mean that the initial data is placed into fixed size cells of 512 bytes each (509 bytes for data and 3 bytes for header) and then encrypted three times using the three ORs constituting the routing circuit keys into the inverse order. In other words, the whole data along with the next OR address or the final destination is encrypted three times (figure).

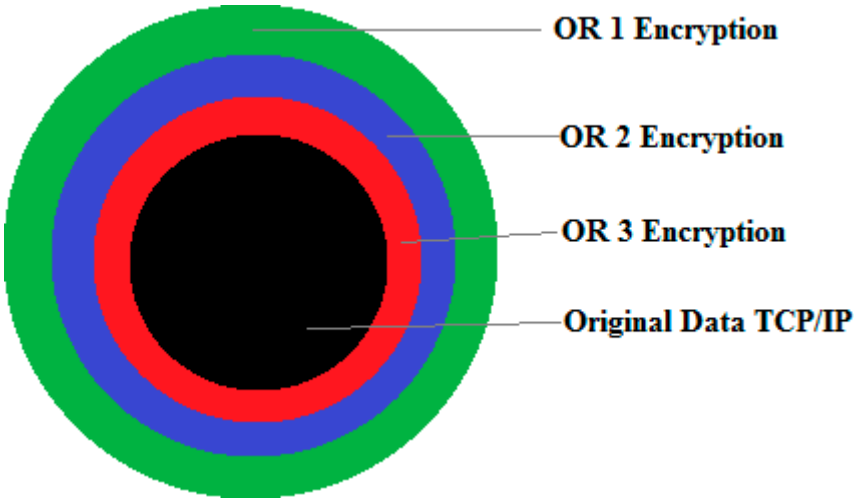

Figure 5. the onion multi-layers encryption approach.

This approach of multi-layer encryption is the hearth of the TOR system as it allow only the ORs part of the Circuit and In Ordered way to have access to the information related to the next OR in the circuit or the final destination of data and thus achieving the anonymity of the sender (figure). However, giving the delays caused by heavyweight encryption of relatively big data this mechanism of encapsulation and wrapping became problematic as it, in one hand slowdown the performances and in the other hand was proved mathematically that this approach does not bring additional security to the system. In fact, in cryptography encrypting the same data using the same function (algorithm) several time using different keys will give the same security of encrypting it once using a composite key which is the aggregation of all the keys (not the addition but it is mathematically determined).

 doi:10.20944/preprints202307.0982.v2

The current TOR Onion Construction Algorithm

Get the data and the final destination

Construct the circuit and get keys ready

Divide data into TOR cell size

First layer:

Data= Original Data+ Final Destination

Address Data= Encrypt (K-OR3,Data);

Second Layer:

Middle-Add= Encrypt (K-OR2, Data + OR3 Address);

Third Layer:

To summarize, improving the encapsulation and wrapping on TOR will not only improve security (privacy and anonymity) but also enhance the network overall security and resilience as the current relays on TOR are being exploited by several attacks (correlation and timing attacks).

d) ***Proposed Improvement:***

Instead of multi-encrypting TOR Cells several time to produce an onion wrapping which only circuit ORs will be able to unwrap, we proposed a much efficient and time saving approach which perform a full encryption of the whole original data including both Exit OR address (Cell Header) and TCP/IP (Header and Data) in the first phase using the Exit node AES shared-secret Key. Then, for the remaining ORs (Entry OR and Middles ORs) only the cell header will be encrypted. In cryptography, encrypting several time data using different keys (k1, k2,…kn) is equivalent to ONE encryption using a composite key K. Thus, the current TOR encryption of the whole Cells several times is useless and cause performance slowing down only as one layer of strong encryption n is enough.

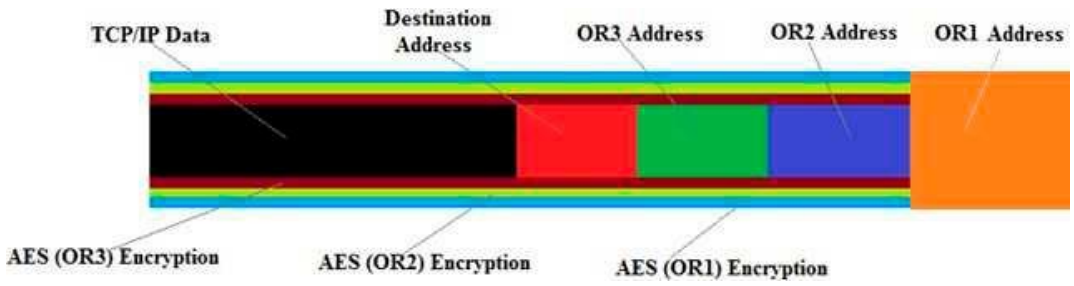

Figure 6. TOR current cell multi-layers and encapsulation approach.

Moreover, the current TOR cells structure is vulnerable and should be reviewed, we propose that only the internal Cell Header contain the Circuit ID, where the external ones (OR1 and OR2) should contain only the address of the next OR and Command. Giving the fact that TOR is managed locally, including this kind of information cause redundancy causing the slowdown of the operation and also leave the circuit ID exposed to threats especially when a fully compromised OR is a part of the circuit.

The proposed approach of encapsulation and onion construction work as follow:

 doi:10.20944/preprints202307.0982.v2

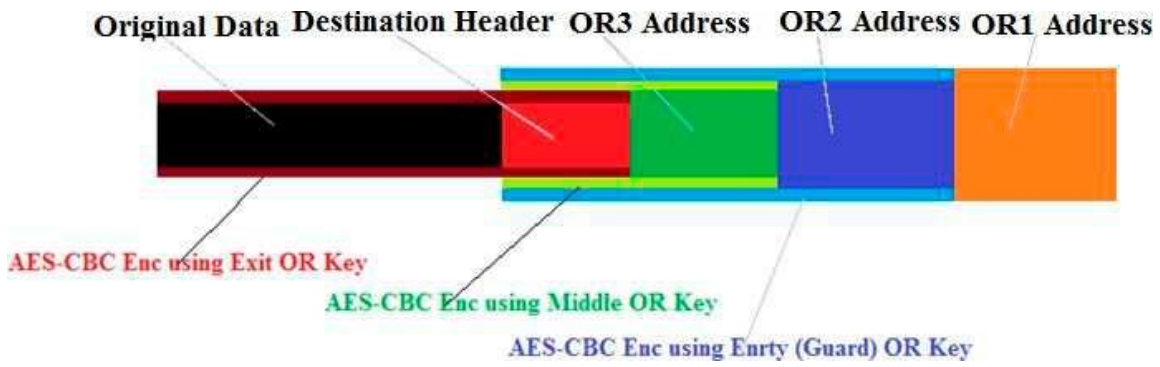

Figure 7. TOR proposed cell multi-layers and encapsulation approach.

The Proposed TOR Onion Construction Algorithm

Get the data and the final destination;

Construct the circuit and get keys ready;

Split data into TOR cell size;
First layer:

Data= Original Data+ Final Destination Address

Data= Encrypt (K-OR3, Original Data);

Exit-data= Encrypt (K-OR3, Final Destination Address);

Second Layer:

Middle-data= Encrypt (K-OR2, Exit-data + OR3 Address);

Third Layer:

Entry-Data= Encrypt (K-OR1, Middle-data + OR2 Address);

Basically, the proposed wrapping algorithm will save a crucial time and componential power. In theory, two (02) round of encrypting data of 509 bytes will be saved along with preserving the same security of the existing TOR. In cryptography encrypting the same text several layers using different (but same size) keys will have the same security of encrypting only one time using the composite key. As AES is provably secure and the key used is also assumed secure enough to suffice on one layer of encryption.

3.3.3. The Number of Intermediate ORs

TOR users anonymous online activities is mainly due to two computing technology Cryptography and Routing, TOR utilizes a series of ORs and makes users' data traveling through a number of hops before it reaches the final destination. By ensuring that each OR have not more information than its predecessor and its successor in the circuit, TOR hide the origin or the destination of the cells containing data and therefore guarantee users' anonymity. Given the aforementioned principle, it is obvious that the more is the number of ORs into a circuit, the better is the source of data (client) is hidden and thus anonym, tracking back the communications will become very complex and the majority of times just impossible to perform. Meanwhile, the number of ORs influence the

 doi:10.20944/preprints202307.0982.v2

connection performance as the long circuits cause more delays (latency) and running interactive applications requiring time precision connection becomes impossible. Hence TOR developers, were seeking the best trade-off between a secure connections that enables perfect anonymity while keeping connections latency bearable. Following several research and testing, TOR developers finally adopted the three (03) ORs circuits length which the current TOR deployment use. When a client is communicating with a server, the data is routed through three intermediate ORs before leaving the TOR network and reach its destination. This choice is defined as the optimal balance between security and usability of TOR.

A continuous debate has been raised regarding the appropriate circuit length especially after the 2015 FBI attack which with the help of researchers were able to control, at several occasion, both the Guard and the Exit ORs and therefore performed an advanced attack to de-anonymise several TOR users including "Drug website Silk-Road-2" owner. As consequence, current TOR short three (03) ORs circuit length will be critically reviewed in this work and several improvement strategies will be considered including increasing the length of the circuit and adopting new routing strategies such as controlled exit OR which will be discussed later in this work. TOR developers' intention behind the choice of a default three ORs circuit is to provide the best balance between the security and performance. In fact, this choice include an Entry OR, an Exit OR and an additional OR aiming to obfuscating the link between the entry and the exit ORs in such way that even if an attacker is able to compromise either of these ORs, the middle OR will constitute the last layer of defence as the attacker can only observe encrypted traffic and it cannot directly deduce the identity of the user. Nevertheless, with the rapid increase and development of the computational power and cryptanalysis attacks, this defence layer become meaningless and can be compromised by performing timing correlation attack.

a) Proposed Improvement

A systematic thought is that increasing the circuit length further would produce an increase into the TOR security. Unfortunately, this operation will incur a significant impact on TOR performance and penalise further the network as the more ORs are involved in transporting one cell, the more times the same cells are relayed in the network before reaching its destination. To determine the impact of increasing circuit length onto TOR anonymity, we will experience different cases in which the variables will be either the number of OR into the circuit or the routing methodology itself such as controlled exit OR selection and network link-status depending selection.

In this work, we implemented a dynamic TOR circuit construction function which have as input the natural number P which is the length of the circuit, then we uses this function to measure the impact of a longer or shorter circuit on the performances. In the proposed function we proposed different circuits building approach for evaluation purposes, also rebuilding function was modified along with the initial function following two main criteria: time interval and circuit performance. Furthermore, as TOR connections terminating in the public internet, the weakest points for attack in the circuit is obviously the Exit ORs (last OR in the circuit). We introduced the notion of "Controlled Exit OR" in the circuit construction in which the algorithm responsible of defining the ORs which will take part in the circuit is adapted to choose the exit ORs from a pre-defined list which reflect in the real TOR the list of trusted ORs. This proposed solution is expected to reduce considerably the FBI attack success against TOR despite the number of rogue (fake) OR inserted into TOR network (Steven et al., 2011).

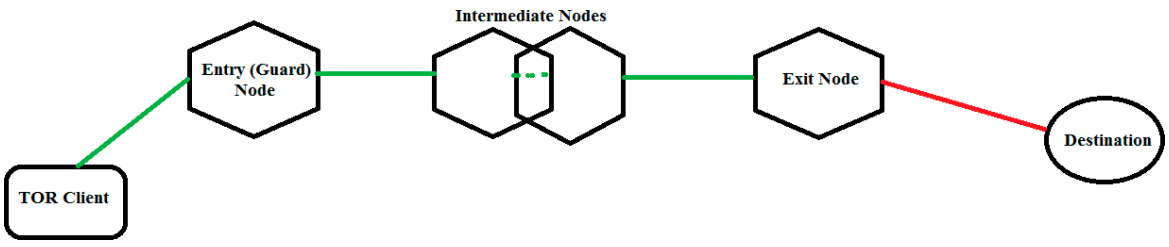

Figure 8. Dynamic TOR circuit length scheme.

 doi:10.20944/preprints202307.0982.v2

3.3.4. Circuit ORs' Selection Approach

In this work we will improve the path selection process by proposing a novel path selection hybrid-algorithm relying on varying path length, controlled exit OR and real-time performance assessment functions. We also employ ORs parameters from a simulation of TOR to compare the proposed algorithm efficiency against.

In the current TOR implementation, ORs for circuit construction are selected at a uniformly random basis aiming to guarantee that ORs are selected uniformly and thus increases the probability and the uncertainty of an attacker trying to de-anonymise users by guessing the ORs used in a particular circuit. Although, because of the heterogeneity in resources caused by ORs of different capacity (computing power, bandwidth) and the emergence of a new classes of attacks, the selection algorithm was edited by TOR development team in the second generation of the network, the following changes were introduced in form of exception in the algorithm:

- No OR should be used in the same circuit more than once,
- ORs in the same circuit should belong to different class of TOR network,
- A special treatment for co-administered ORs is introduced by marking them as the same family,
- Directory Authorities will assign flags to ORs basing on the following parameters: performances, status, position and role.

Moreover, some important features were added following the attacks on TOR circuit selection in 2014. In fact, ORs selection algorithm was again changed in such way that entry OR (guard) is only selected from a subset of ORs classified as "entry guards" and which are particularly "trusted" and continuously authenticated to check the status. The entry guard subset is a group of ORs which are constantly active, having a bandwidth of at least 250 KB/s (Dingledine et al. 2014). In reality, the implemented selection algorithm allowed 3 ORs guard to be assigned for a user for a period of 30 to 60 days and used in combination for all circuits. However, due to the limit probability of attacks and the impact on the TOR overall traffic homogeneity, this algorithm was abandoned to allow client the use of only one entry OR for the same period of time (Dingledine & Mathewson, 2015).

On the other hand, current TOR circuit selection algorithm states that selecting the remaining ORs on the circuit (Middle and Exit) is proportional to the available bandwidth. This choice aims to ensure that powerful ORs are chosen more often. Nevertheless, the bandwidth information which TOR base on for making decision is being advertised by the ORs themselves which leave the opportunity for rogues ORs to advertise false data in order to acquire more traffic. Logically thinking, if a capable attacker with significant resources is able to inject an important number of rogue ORs having the best performance, it will be able to re-direct a significant amount of TOR traffic via these ORs (which will be middle ORs or exit OR of the selected circuits) and therefore being able to perform a time- analysis attack to determine the User as the Entry OR information will be already disclosed for the Middle OR (Steven et al., 2011).

a) ***Proposed Improvement***

 doi:10.20944/preprints202307.0982.v2

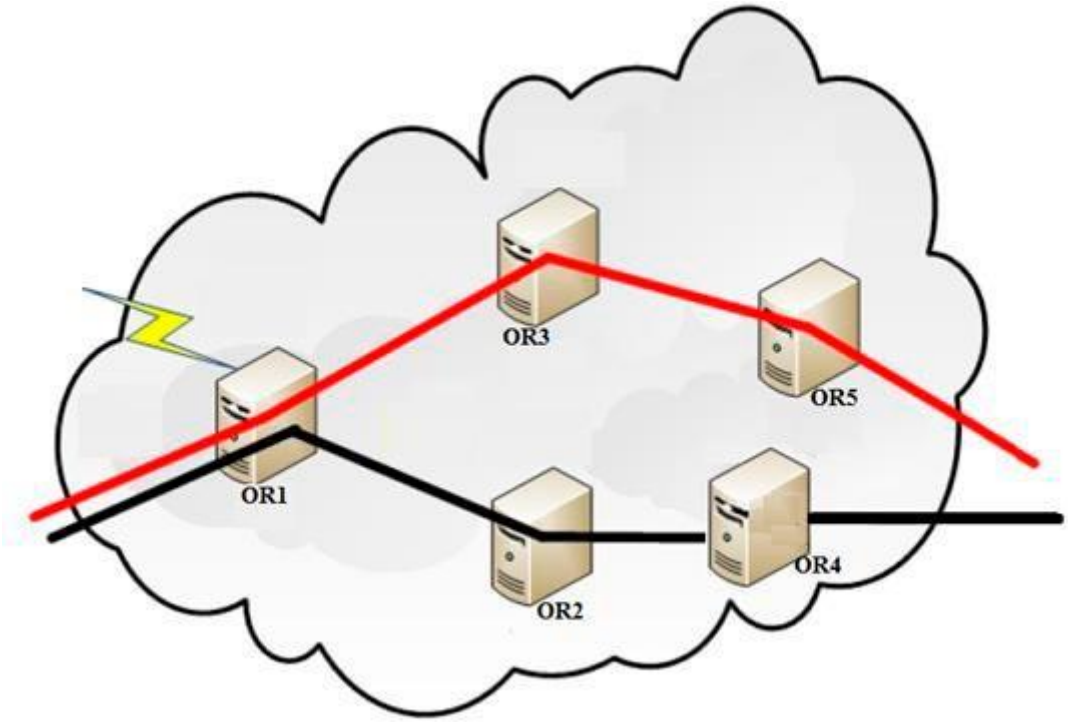

Figure 9. TOR circuit selection scheme.

The proposed improvement for the circuit selection algorithm is twofold:

First, we introduce a new sub-function in the section algorithm imposing the selection of the Exit OR from a pre-defined subset only. This subset will be completely different from the entry guard subset (no common ORs) and contain more ORs . This method is called during this work "Controlled Exit" which will be implemented into the emulation platform for test purpose. It is evident that such method will have an impact on traffic homogeneity, but the security enhancement will be greater than the overall performance. Moreover, the proposed method will help into protecting the TOR hidden- services from being disclosed during the attacks.

The Modification introduced to the current Circuit ORs Selection Algorithm

Proposition 1: Controlled Exit OR:

- Defined a list of Exit ORs (5 ORs out of the 20 ORs constituting the simulation network,
- When choosing and Exit OR the DA should assign an exit ORs from the pre-defined list,

Proposition 2: Best Possible Circuit

- This algorithm bases on the real time data provided by the DA regarding the links status assuming that there is other traffic and not all the link have the same capacity.

The second proposed improvement is related to the choice of the ORs basing on the advertised capacity (bandwidth). The misleading information that rogue (fake) ORs can provide during the

 doi:10.20944/preprints202307.0982.v2

selection process could be fatal for user security. Thus, the directory authority responsible of the collection of such information and processing should not trust this information and rather evaluate or estimate itself the capacity of each router and therefore faster ORs will not have any more a higher probability. In this work, the concept of prudential-processing is introduced in which the calculation of ORs' capacities is done in a real-time basis relying on both provided information by the ORs, historical status and credibility. This method will be implemented and tested into the emulation platform.

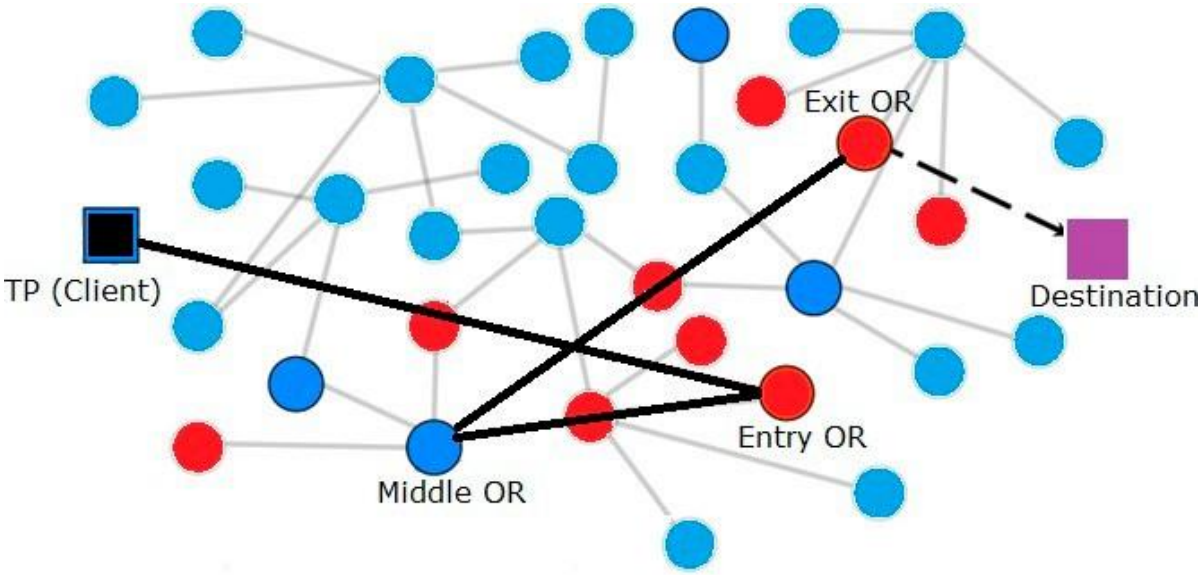

Figure 10. the proposed controlled exit approach for TOR circuit selection.

b) ***Dynamic circuit construction with traffic management***

To tackle the delays and security issues related to the choice of circuit ORs, we introduced an enhanced circuit selection algorithm named Controlled exit with congestion-aware ORs Selection at Client side. In this work we tried to adopt and implement a circuit construction method proposed by (Wang et al.,2012) in which the multi-criteria circuit selection algorithm rely on both the status and performance data and real time indicator. First, TOR's default bandwidth-weighted OR selection algorithm is used to construct circuits. Then, the proposed algorithm will use an opportunistic and active-probing function to calculate the circuit's latency value.

3.3.5. Cells' Multi-Circuit Routing (Limited to Three Circuits)

Multipath routing approach on TOR has been previously tackled by Snader et al. (2010) in context of improving TOR networking performances. The research simulated a case of downloading data fragment of 1MB each over a dedicated TOR simulation network. The files were divided into blocks of 512 Bytes and routed from the sources to destination over multiple circuits. The proposed mechanism was working as follow. An algorithm assign each Client (OP) two different Entry and Middle OR, the mechanism used the same Exit OR for both Circuit (Figure 29).

 doi:10.20944/preprints202307.0982.v2

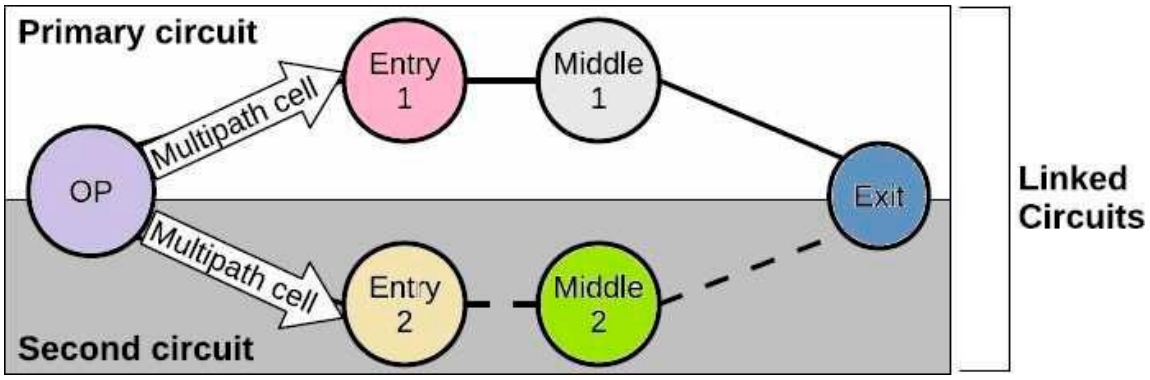

Figure 11. TOR existing multi-circuit routing mechanism.

This research observed that the security and throughput of the routed traffic were significantly enhanced. However, two circuits performance was less well than single circuit as the chances of choosing a slow OR as part of the circuits double and thus the median transfer time increased for the two circuits. Nevertheless, the research highlighted the fact that the security enhancement of such proposition is also considerable and also that the risk of including a compromised OR will be certainly be affected if this function is used.

a) ***Proposed Improvement***

The current TOR deployment each Client (OP) is assigned by default three different Entry (guard) OR which will be the only Entry guard that this client can have (use) for a certain duration. This research will rely on this feature and implement an algorithm which will generate for each linked Traffic (having the same destination server) three different circuits. This Multipath routing has introduced to enhance security for TOR users and also the Entry and Exit OR themselves. TOR client starts by building a three different circuits using the following algorithm:

Multi-Path circuit construction
For each TOR new Connection

// from the assigned 3 Entry OR for
1- Select a different Entry (Guard) OR from the list of OP three possible Entry ORs;
// a random sub-function will be used
2- Select Randomly a Middle OR;
// from the assigned 3 Entry OR for
3- Select a different Exit OR from the list of Exit OR;
// complete the circuits construction
4- Perform hand-shake, authentication and keys exchanges
// for each data being sent to the same destination
5- Divide the Data equally to 3 parts and send each consecutive 3 cells throughout the 3 different circuits.

 doi:10.20944/preprints202307.0982.v2

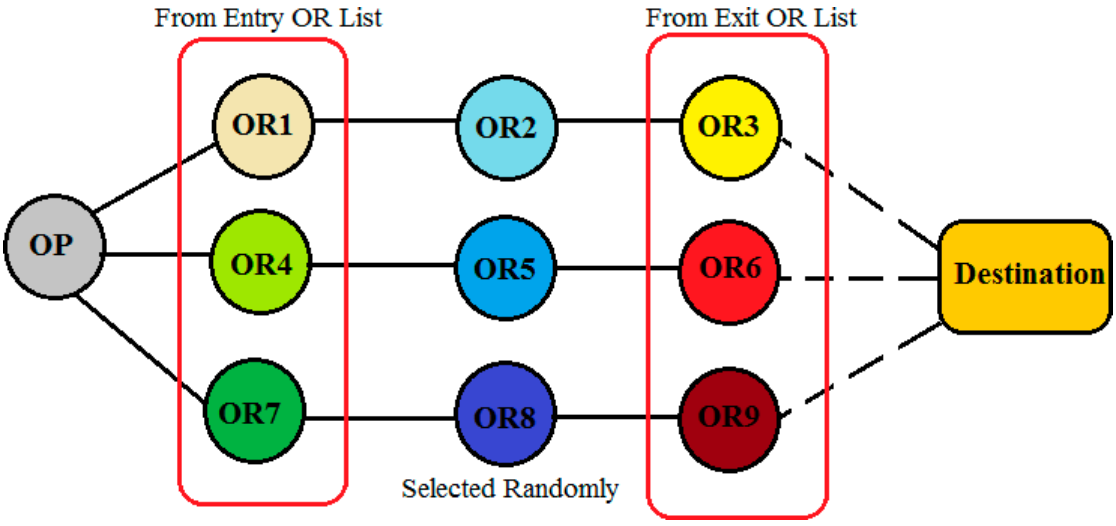

Figure 12. the proposed TOR multi-circuit routing mechanism.

5. Implementation, Testing and Results

Testing is a crucial phase of any system development or improvement and require a particular intention as the results of this phase will determine if the system should or not goes to life or the improvement are valid and thus could be implemented in the real world system. The proposed improvements and enhancements should be measured and evaluated only by adopting a scientific approach for comparison which can determine whether or not the obtained results are satisfying and can bring the intended contribution along with preserving the system main features and resources. On the other hand, Cryptographically Improving TOR design decisions, such as adopting new ciphers, integrity checks mechanisms, routing method and encapsulation technique, is a complex process which require a reliable test-bed.

An obvious testing approach is to use the real TOR network to conduct the different testing and measurements as TOR is open-source software, hence easy for researchers to implement and run experiments. Nevertheless, this option is ethically not suitable, as during the testing and experimentation other TOR user's. Therefore, to test the efficiency of the proposed improvements and enhancement for a potential incorporation into TOR (cryptosystem or routing system), it is essential to simulate the TOR on a virtual environment along with preserving the same behaviour and accuracy of the real TOR network. To achieve this, we adopted an existing TOR simulation platform and introduced several modification and code changes which reflect the proposed improvement.

This choice is justified by the fact that testing on the real network will incur serious privacy concerns to TOR users' security and adopting a non-standard test-bed will negatively impacting the confidence on the obtained results. Therefore, after having carefully evaluated the scientific risk and the feasibility along with the required componential power, testing efficiency and accurate which the test-bed should fulfil. This work adopt a two phase testing approach starting by evaluating the efficiency of the candidates alone (with just program implementation and testing into JAVA compiler) and then implement the selected (accepted) improvement into the simulation TOR along with the native function to produce a comparison and eventually validating the results obtained.

5.1. Simulation Test-Beds Design and Adoption

To practically demonstrate the performance enhancement brought by the proposed improvements, two isolated test-bed platform were designed/adopted. The first emulation platform is adopted to run both current and improved TOR simulation as the first simulation will run a native TOR with the current deployment, parameters and configurations in a simulation network of twenty (20) ORs (Nodes) running virtually along with AD (directories server), three OP (clients) and Destination Server. Among the ORs, three are considered as Guard (entry) ORs. The testing will starts

 doi:10.20944/preprints202307.0982.v2

by assessing and recording the performances of the current TOR version which will serve as a reference to evaluate the efficiency of the proposed solutions, later several modifications into OP code, DA and ORs will occurs to reflect the proposed solution.

On the other hand, and before implementing the improvement into the TOR simulation platform, a cryptographic implementation and testing of the candidate cipher modes to replace the current CBC mode is performed, this testing will include a simulation of the TOR cryptosystem (key exchange, Cells Encryption/Decryption and Onion Wrapping-Encapsulation) into a JAVA platform. All the algorithms and mechanisms will be implemented in JAVA and tested. After confirming the elected candidate to be integrated into TOR for real-world simulation and testing, the candidates (Cipher Modes, New onion Construction approach) will be incorporated into the simulation platform TOR as new version of TOR running on the same simulation network to produce comparable results.

The proposed improvements will be included gradually and tested into the test-bed in the following planned order:

- Implementing, testing and comparing the obtained results of AES-OCB, AES-GCM and AES-CCM,
- Implementing the proposed Wrapping approach (multi-layered encapsulation), testing and comparing the obtained results,
- Implementing the two variant of Circuit construction algorithms, testing and comparing the results,
- Implementing the variable circuit length algorithm, testing and comparing the obtained results,
- Implementing the Multi-path Cell routing algorithm, testing and comparing the obtained results,
- Validation of the results and discussion.

In this section, a genuine and scientific assessment of the proposed improvement impact on TOR will be performed by implementing the proposed improvement into a testing platform and run the tests and comparing performances of both current and improved TOR.

The conception and design of a specific test-bed allowing the implementation and the test of the proposed improvement in a native, similar and accurate TOR environment along with avoiding any unethical activity by testing these improvement in the real world TOR. Therefore, this research will adopt a Model Net network emulation platform which will lodge a TOR specific simulation system called ExperimenTOR. ModelNet allows to implement realistic network structure, topologies, routing approach, delay and other features. Furthermore, it enables running real or modified C++ code (TOR) on the emulated platform to measure performances. The sought goals behind adopting this platform is to test the proposed improvement in a truly consistent, significant and reliable test-bed guaranteeing the same features as the real world TOR network in one hand, and on the other hand follow the ethical code and procedures which is crucial in this kind of research as adopting the real TOR with its users as testing platform will be considerate as unethical in the best-case scenario or illegal in the worst. Moreover, the measurement issue is also a preponderant factor on adopting this test-bed. In fact the implementation and testing phase seek to transit from theory to implementation and evaluation in practice, the sought measurements behind are to accurately measure the timing, the resilience and the efficiency along with the compatibly and behaviour of the proposed improvement.

5.2. ExperimenTOR Platform

To perform real and accurate experimentation on TOR without affecting other users, this work will implement the proposed improvements and enhancements into a TOR simulation dedicated platform called "ExperimenTOR" developed by Bauer et al. (2011) for TOR research purposes. It is based on the Model Net network emulation test- bed and can be used locally or virtually but requires several FreeBSD machines to work as ORs along with emulator core server responsible for networking the TOR architecture (Bauer & Sherr, 2012). Client (OP) can be run on several emulators. ExperimenTOR run the original TOR network deployment and code and can simulate different size network varying from 2 to several hundred depending on the componential power and memory res resources. ExperimenTOR is used to generate a downscaled network of TOR (small TOR prototype)

 doi:10.20944/preprints202307.0982.v2

and incorporate module for measuring and comparing TOR performances. Several reputed research on TOR used ExperimenTOR for emulation and simulation, such as (Wacek et al., 2013) research which relied on ExperimenTOR to perform an analysis of the OR selection technique and impact on anonymity and performance properties (Bauer & Sherr, 2011).

Moreover, (AlSabah et al., 2015) research used ExperimenTOR for evaluating their research on improving and surveying security in TOR. ExperimenTOR prototypes were deployed at many research centres in the United States and Canada. The current version consist of several FreeBSD machine with the ModelNet emulator kernel module and other machines running Linux working as ORs, OPs (clients), and application processes within the emulated topology. ExperimenTOR is a combination of python and C++ codes and include experiment and performances measurement toolkits such as "torperf" written in C++ for performance measurement purpose (Bauer & Sherr, 2011). For our work, we adopted this virtual test-bed which will be running on VIRTUAL BOX machine with the following architecture:

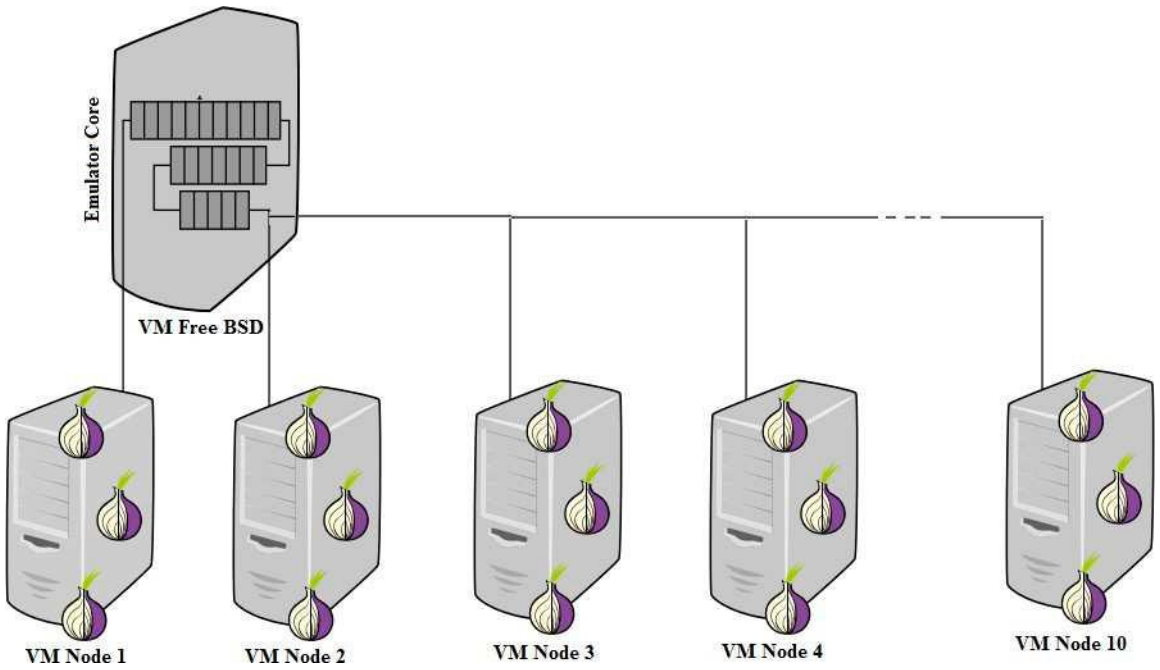

Figure 13. Experiment TOR emulation platform architecture.

5.3. TOR Test-Bed Topology and Deployment

The first step into implementing the proposed solution and before running experiments in the test-bed is to design a TOR network realistic topology to be diploid later into the emulator. This work uses information from a real TOR Directory Server (DA) and applied the deployment into a small size simulation network at the corresponding componential power and bandwidth capacity of the ORs, Host and Server within the virtual topology by assigning a real bandwidth value and realistic network latencies to each end-host. In this work, the test-bed topology is as follow:

5.3.1. Simulation Network Size

The number of ORs into the simulation network used for implementation and testing is a crucial, because a small network might not reflect the same deployment or capture the same network features and effects that the real TOR network can experience. Moreover, the Network parameters should be adjustable to reflect the real-world situation and being flexible to allow the gradual incorporation of the proposed improvements along with the performance change testing and measurement. Moreover the virtualization option which this research adopt is tricky, despite the fact that this option will help to avoid compromising legitimate used and unethical, but also to run experimentations is a

 doi:10.20944/preprints202307.0982.v2

controllable and scalable environment, but the virtualisation will affect the performance and therefore the results of the experimentation. Thus, for each improvement incorporation test during this work, a benchmarking test will precede it using the current TOR deployment and feature which will serve as a comparison reference. The testing platform include the following:

- Twenty (20) FreeBSD Onion Routers including 3 dedicated Guard (Entry) ORs,
- One Directory Server (DA),
- Three (03) Onion Proxy (Clients),
- One service server (destination),
- Different size testing data

5.3.2. Test-Bed Preparations and Configuration

Before the actual tests could be carried out, the ExperimenTOR platform require reconfigured and benchmarking. Then, the proposed improvement were implemented in C++ and incorporated into the different modules of the test-bed. Moreover, some modifications were introduced to the modules: **torperf:** the performance measurement module in which several code addition and modification will occur in this module to include new functions required for testing purposes.

code: several modification and new implementation will be incorporated especially into the "crypto function", "test", "config" and "or" source code. The proposed cryptographic and onion construction improvement were implemented in this module.

log: new logging and processing function will be included in this module.

lighttpd: This is the source code for a very light-weight web server. A customised web servers to host web objects for clients to download will be included.

routers: Each router is assigned new routing, capabilities and type information along with numbered data directory. Routers 1-5 are configured as Entry (guard), router 18-20 are configured as Exit and all other routers are middle. The proposed circuit construction algorithm (ORs selection and routing approach, multi path routing and dynamic circuit length).

tcpping: use ICMP pings to measure round-trip times between virtual ORs. We introduce a new TCP- based ping timing measurement along with TCP's SYN and RST packets to calculate RTT.

tools: several script modification are introduced especially the configuring, running, and stopping functions.

torperf: the TOR performance assessor module which will be modified to measure several new variable related to encryption, circuit construction and routing approach performances.

5.3.3. Logging and Measurements

Native TOR implementation provides several measuring and logging features varying from a high-level error logs to a detailed performances measurement. However, these log still present crucial lack as the information does not include performance logging. In this work, the lgging for tests and experiences is ensured by the embedded ExperimenTOR logging module called "TORPerf". Though, several modification were introduced to deal with the edited module and function. Logging is performed at the network level (directory server), ORs level and at OP level (TOR software).

5.4. Testing and Results

In this section, the testing of the implemented proposed improvements and enhancements will be performed several times and results will be logged in details

5.4.1. Testing the Proposed Ciphers and Encapsulation Approach

The NetBeans Platform is a reliable and flexible application architecture allowing a time-tested architecture and the componential power usage. In this work, the implementation (in java) of the proposed improvement (cipher modes and encapsulation) along with the TOR-like crypto-system is run on the NetBeans platform first. The aim of this step is to select the appropriate candidate to be

 doi:10.20944/preprints202307.0982.v2

implement in the TOR test-bed and also measure the cryptographic and power performance of each implementation. As benchmarking, this work will use different combinations of data with sizes varying from 16 to 5120 bytes in different scenarios. All measurements were taken on Platform running on Intel Core i7-2820QM CPU at 3.4 GHz and a restricted RAM memory of 2048 MB.

The second step, is to encode (implement) the selected algorithm mode (OCB) in C++ to be incorporated with ExperimenTOR module. In fact, we implemented the proposed improvement and enhancement function and algorithm along with preserving the current mode (original TOR) in order to produce a comparison. For each combination of parameters, the performance is measured several times depending on the experiment length. For the encryption algorithm performance assessment, the implementation and testing into the simulation platform will be preceded by testing all candidates AES modes (CBC+MAC, CTR+MAC, CCM, GCM and OCB) on JAVA NetBeans and we ran the experimentations using java debug and project profile mode to determine and measure the time spent on processing for encryption and decryption along with the resources usage.

- AES-OCB-128,
- AES-CCM-128,
- AES-GCM-128,
- AES-EAX-128,
- Add a separate MAC calculation function to the existing CBC and CTR code.

During the third step, the proposed encapsulation and onion construction approach was implemented within a TOR-like light-weight cryptosystem for performance measurements purpose. The proposed onion construction approach (with encryption and structuration) is implemented on beside the current (Existing) TOR encapsulation approach and several testing scenarios of various Cells Format wrapping, padding and encrypting were performed to assess the efficiency and the improvement before incorporate the proposed improvement on the ExperimenTOR platform for validation.

Finally, after testing the proposed wrapping (onion construction) approach. The algorithm is implemented in Python (plus the Browser C++) and integrated in the TOR simulation platform (TOR browser and the ORs) to be tested in TOR condition, measure the performance enhancement and assess the suitability.

a) ***Experimentation Results and Discussion***

The results obtained initially Java implementation of AES encryption algorithm in three modes; CBC (first generation TOR), CTR (second generation) and OCB (with disabling the authentication TAG processing and only performing encryption) in order to pre-assess the efficiency of each mode for different data size varying from 16 to 5120 bytes. The results obtained were predicted and showed that CTR (without authentication) perform better than OCB and CBC. Thus, without the need of node-to- node authentication the current TOR adopted mode seems to be the perfect choice. The figure summarize the performance of each mode.

Table 2. the AES modes testing results on JAVA.

Input size (Bytes)	**16**	**64**	**128**	**256**	**512**	**1024**	**2048**	**5120**
Time Consumed (ms) per mode								
CBC	64.11	102.31	198.34	344.76	634.21	1123.44	2011.54	4329.33
OCB	10.54	34.25	56.23	98.50	176.32	322.90	603.74	1488.60
CTR	5.82	20.58	36.77	65.88	129.98	245.34	468.13	1198.86

 doi:10.20944/preprints202307.0982.v2

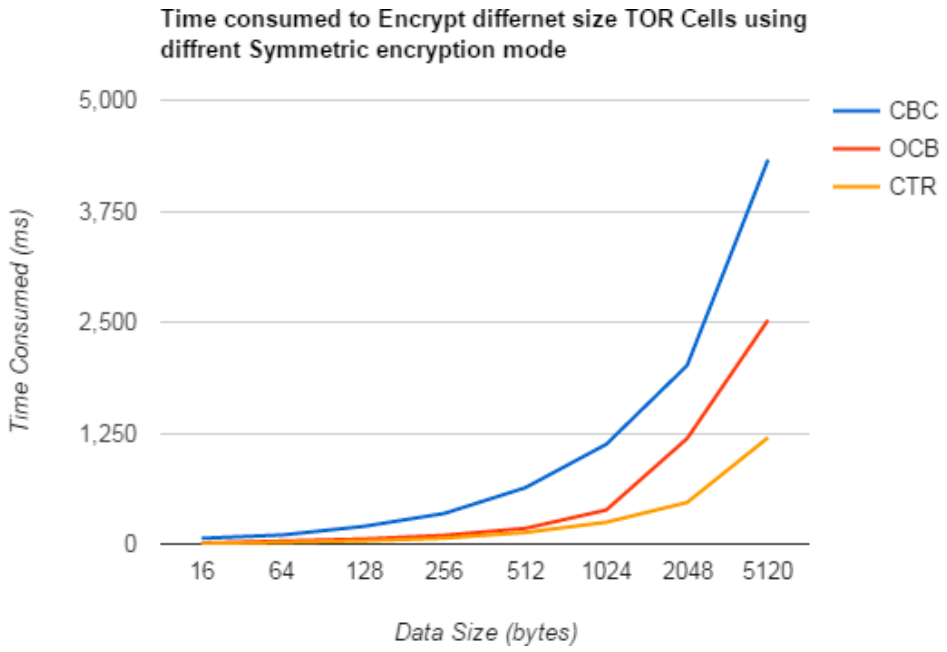

Figure 14. The comparative graph of the AES encryption-only modes performances.

Then, we implemented (coded) an authentication processing function to CBC and CTR mode with preserving the same encryption implementation, where the OCB was implemented entirely by activating the authentication function. We tested the three implementation in the same way. Here, the results obtained (figure) were different. Regardless the slightly slowdown in OCB performance, it perform better then CBC+MAC and CTR+MAC. Therefore, to introduce the node-to-node authentication in TOR it is obvious that the classic implementation of encryption followed by authentication is a wrong decision.

Table 3. the performances of AES modes encryption plus authentication.

Data Size / **Time Consumed (ms) per mode**	**16**	**64**	**128**	**256**	**512**	**1024**	**2048**	**5120**
CBC+MAC	84.11	122.31	218.34	364.76	674.21	1323.44	2191.54	4629.33
CTR+MAC	35.82	80.58	96.77	165.88	329.98	645.34	1068.13	2398.86
OCB	29.54	54.25	76.23	138.50	246.32	472.90	793.74	1578.60

 doi:10.20944/preprints202307.0982.v2

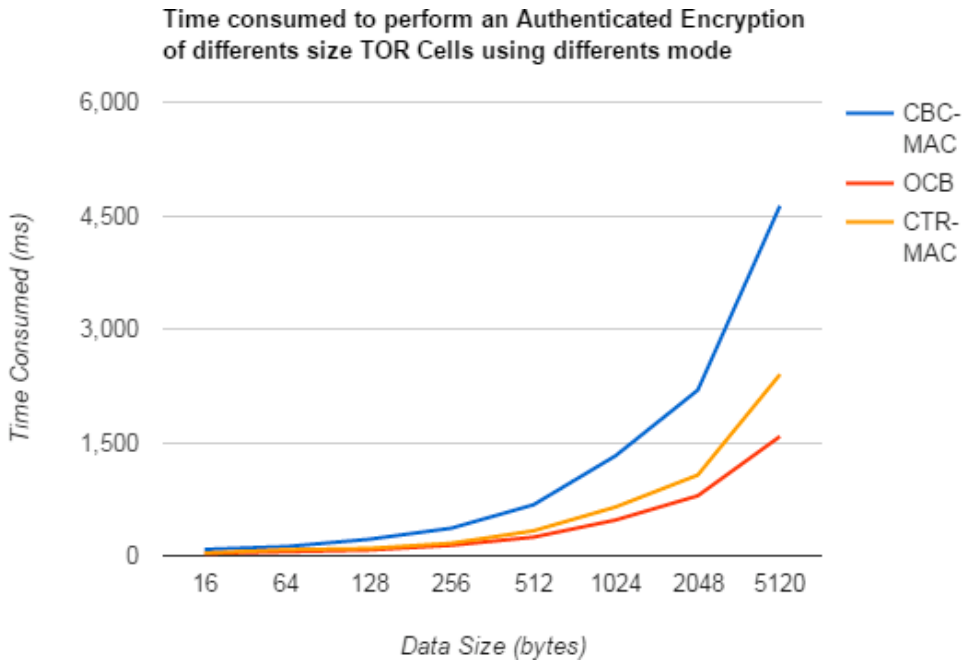

Figure 15. The comparative graph of the AES authenticated encryption modes performances.

Nevertheless, OCB is not the only AES implementation mode guaranteeing an integrated authentication. Three other modes; CCM, GCM and EAX should also be considered. Therefore, we implemented the six (06) mode together using the test-bed resources (javax.crypto.Cipher) in which the implementation is optimised and implemented natively. And the results obtained (figure) showed that three candidate could be replacements of the current TOR mode; CCM, GCM and OCB.

Table 4. The performance of all AE candidate in java compiler.

Data size (Bytes)	**16**	**20**	**32**	**64**	**128**	**256**	**512**	**768**	**1024**	**1536**	**2048**
Time consumed (ms) per Mode											
CBC	136	167	207	232	248	259	269	278	286	299	312
CTR	89	109	140	162	193	225	246	261	269	275	282
EAX	91	98	123	133	162	177	193	216	228	257	268
CCM	52	59	79	83	102	129	148	157	160	167	174
GCM	45	51	73	75	88	104	116	127	131	139	153
OCB	36	48	65	73	89	101	113	120	127	135	148

 doi:10.20944/preprints202307.0982.v2

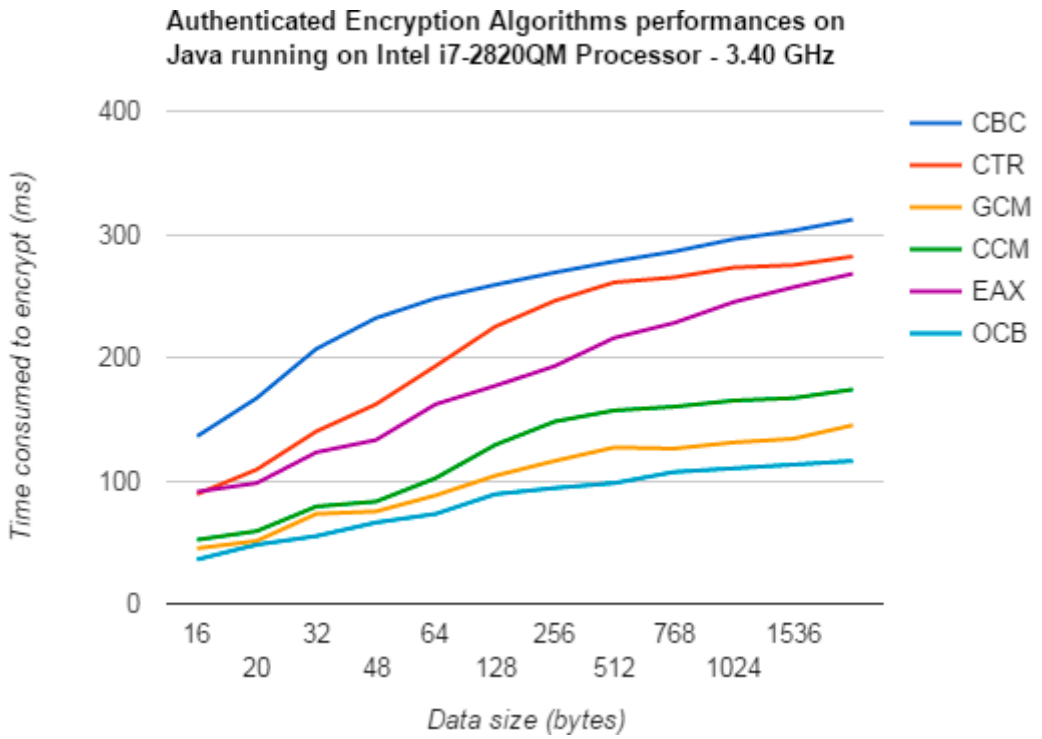

Figure 15. graphic representation of the performances of all candidates inn JAVA.

b) ***Encryption testing results***

From the obtained results and the research carried on the authenticated-encryption, OCB is by far the best integrated mode as it perform both encryption and authentication in one round (single pass). OCB is patented in US in case of commercial use, however TOR is an open-source and free software/service and thus this restriction doesn't apply. Moreover, OCB accomplishes the authentication processing without using the heavy-weight universal hashing computing which make it not only the best mode in term of performance but also in term of resources use and the implementation in both hardware and software. Nevertheless, the Nonce generation choice and use in online system can be tricky as Nonce require to be unique for every encryption (no need to be random as it is the case of IV in CTR, GMC and CCM modes).

c) ***Resources use analysis***

The Performance comparison is not enough to constitute a criteria for selecting an encryption algorithm or mode, in fact the componential power required/consumed by a cipher is also an important factor. In TOR context this power is limited and therefore precious. In term of Memory requirement, OCB has the smallest memory usage as it mainly relies on the block cipher operation and data with no use of other function. The memory required for GCM is the largest due to the Galois function and it uses more extra pre-computation table for speed acceleration. On the other hand, the computation cost of CCM is the highest compared with the OCB and GCM as it requires two cipher invocations for each block, while OCB mode require merely one cipher call for each block along with the special function which have a low computation complexity.

d) ***Choice and Discussion***

In the real world situation, encryption modes present various pitfalls due to the combined privacy and authenticity processing such as the failure of guarantee a proper key separation, the misuse of the MAC which open security breach into the cipher allowing cryptanalysis or brute-force attacks, mismanaging the IVs or Nonces. OCB and GCM are provably secure meet these required standards but remain dependent on the correct implementation especially on software which mean more vulnerabilities threats and enemies. On this work a high importance is accorded to mode

 doi:10.20944/preprints202307.0982.v2

selection, thus we tested all the potential authenticated-encryption schemes which are OCB, CCM and GCM.

6. Conclusions

6.1. Work Output and Conclusion

This work proposes several cryptographic and routing improvements for the second generation (current) TOR network. The proposed solution covers two main area; enforcing the security and enhancing performances. The authenticated-encryption AES mode called OCB and the new approach of onion construction (encapsulation) aim respectively to enforce TOR security by providing a crucial Node-to-Node authenticity and lightening the multilayer encryption approach along with preserving the same security level. In fact, TOR cryptosystem present several useless redundancy which cause more delays with no security clear or proved enforcement. This work adopted a scientific approach into the proposition, implementation and testing of the proposed cryptographic improvements as it first started by investigating the different candidates and solution, then testing these candidates into a purely cryptographic environment, and finally implementing the selected solutions into a specific simulation and platform to assess the performances and compare them to the existing implementation. The adopted research, implementation and testing methodology consolidate the trust into the obtained results which showed the sought efficiency and the suitability.

On the other hand, this work targeted to improve the TOR security and performance by enhancing some noncryptographic mechanisms (routing and functioning) which are closely related to the TOR cryptography, in fact the existing TOR circuit (routing path) selection and construction mechanism which seems to be a pure routing matter is very related to TOR security. Hence, this work proposed some improvement regarding the circuit selection approach (controlled exit) which could contribute into the TOR resistance especially against path selection attack (known as FBI attack) which was very effective into de-anonymise TOR users. Moreover, a multi-circuit routing approach was proposed which aims to balance the security and performance over TOR. Nevertheless, the proposed TOR dynamic circuit length solution testing showed its limits as it impact significantly the performance without a clear performance improvement.

The implemented improvements are based on pervious academic works and research outcomes especially regarding the implementation of the third version of the OCB mode appeared in 2014. The emulation platform ExperimenTOR was slightly modified to include more measurement functions on one hand, and on the other hand accommodate the proposed improvement. Nevertheless the core of the platform remains the same.

6.2. Further Works

Regarding possible future work in this topic, several possible research directions could be investigated. These are mainly related to improving and advancing the current TOR protocol, mechanism and defences. In fact, TOR enemies list is getting expanding every day and so is the need of online privacy and anonymity for people. First of all, a research project could work on the next generation of TOR in the post-quantum computing. Secondly, the IPv6 migration is a new challenge for TOR future and should be addressed properly. Thirdly, the current TOR overall design should be deeply reviewed to incorporate so revolutionary solution and move to virtualisation or cloud computing.

6.3. Work Evaluation

Many methods, techniques and approaches were used during the development of this work. Some of them new but mostly the author of this Dissertation has applied the knowledge and skills gained in the Taught and developments modules of the two previous semesters of the MSc. course. In approaching project management, together with the techniques and tools used the author

 doi:10.20944/preprints202307.0982.v2

demonstrated his skills in this field. Giving the nature of the research carried out in this work, the application or adoption of the classical research and development methodologies. The [practical implementation and testing of the proposed improvement and enhancement was preceded by a scientific evaluation (theory) in which the proposed solution were studied in deep. During this work several detail were deliberately dropped as the amount of technical details could impact the meaning. Nevertheless, all details are presented in appendixes; AES algorithm details, TOR detailed specification and cryptography use, detailed functioning of some features. This work constitutes a scientific contribution into the research of enhancing and improving TOR network security and performance. A deep investigation of the weaknesses and flaws causing the security issues and slow performance on current TOR implementation was carried out at the beginning of the work. Later, a scientific approach was adopted to validate the selection of the most suitable solution among the candidates. Finally, a specific test-beds were designed in which the selected solutions were implemented and the required modification were performed in order to assess in TOR- like environment the performances improvement and security enhancement brought by the proposed solution and therefore validate the findings.

References

AlSabah, M. and Goldberg, I. (2015). Performance and Security Improvements for Tor: A Survey. http://eprint.iacr.org/2015/235.

AlSabah, M., Bauer, K., Elahi, T. and Goldberg. (2013a). The Path Less Travelled: Overcoming Tor's Bottlenecks with Traffic Splitting. In Privacy Enhancing Technologies - 13th International Symposium, PETS 2013, Bloomington, IN, USA, July 10-12, 2013. Proceedings. Springer, 143–163.

AlSabah, M. and Goldberg, I. (2013b). PCTCP: Per-Circuit TCP-over-IPsec Transport for Anonymous Communication Overlay Networks. In Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13, Berlin, Germany, 349–360.

AlSabah, M., Bauer, K., and Goldberg, I. (2012). Enhancing Tor's Performance Using Real-Time Traffic Classification. In Proceedings of the 19th ACM Conference on Computer and Communications Security (CCS '12). ACM, New York, NY, USA, 73–84.

AlSabah, M., Bauer, K., Goldberg, I., Grunwald, D., McCoy, D., Savage, S. and Voelker, G. (2011). DefenestraTor: Throwing Out Windows in Tor. In Privacy Enhancing Technologies. 11th International Symposium, PETS 2011, Waterloo, ON, Canada, July 27-29, 2011. Proceedings. Springer Berlin Heidelberg, 134–154.

Antonakakis, M., Edman, M. and Syverson, P. (2009). As-Awareness in TOR Path Selection. In Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS 2009, Chicago, Illinois, USA, ACM 380–389.

Backes, M., Goldberg, I., Kate, A., & Mohammadi, E. (2012). Provably Secure and Practical Onion Routing. Computer Security Foundations Symposium (CSF), 25, 369-385.

Bauer, K. and Sherr, M. (2011). ExperimenTor: A Testbed for Safe and Realistic Tor Experimentation. USENIX 2011, http://www.usenix.org/events/cset11.

Bauer, K., Mccoy, D., Sherr, M. and Grunwald, D. (2011). ExperimenTOR: A test-bed for safe and realistic tor experimentation. In Proceedings of the USENIX Workshop on Cyber Security Experimentation and Test (CSET). Bogdanov, A., Lauridsen, M. and Tischhauser, E. (2014). AES-Based Authenticated Encryption Modes in Parallel High-Performance Software. IEEE library.

Fu, X. and Ling, Z. (2009). One Cell is enough to break Tor's Anonymity. White Paper for Black Hat DC 2009.

Boyd, W. (2011). A Simulation of Circuit Creation in Tor. Master thesis submitted at Wesleyan University, Connecticut April, 2011.

Benmeziane. S., Badache. N. & Bensimessaoud. S. (2011). Tor Network Limits. International Conference on Network Computing and Information Security, 1, 200-205.

Burstein, A. J. (2008). Conducting cyber security research legally and ethically. 1st USENIX Workshop on Large-Scale Exploits and Emergent Threats, Berkeley, CA, USA, pages 1-8. USENIX Association.

Camenisch, J., Lysyanskaya, A. (2005). A formal treatment of onion routing. 25th Annual International Conference in Advances in Cryptology CRYPTO 2005, 169-187.

Carnielli, A. and Aiash, M. (2015). Will TOR Achieve its Goals in the Future Internet? An Empirical Study of using TOR with Cloud Computing. 2015 29th International Conference on Advanced Information Networking and Applications Workshops.

Casenove, M., Miraglia, A. (2014). Botnet over Tor: The Illusion of Hiding. 6th International Conference on Cyber Conflict P.Brangetto, M.Maybaum, J.Stinissen (Eds.), NATO CCD COE Publications, Tallinn.

Castelluccia, C., De Cristofaro, E. and Perito, D. (2010). Private information disclosure from web searches. In Mikhail J. Atallah and Nicholas J. Hopper, editors, Privacy Enhancing Technologies, 6205 of Lecture Notes in

Computer Science, 38-55.
Dahal, S., Lee, J., Kang, J. and Shin, S. (2015). Analysis on End-to-End Node Selection Probability in TOR Networking, IEEE ICOIN 2015 ISBN: 978-1-4799-8342-1/15.
Danezis, G. Diaz, C. and Syverson, P. (2010). Systems for Anonymous Communication. In CRC Handbook of Financial Cryptography and Security, CRC Cryptography and Network Security Series, B. Rosenberg, and D. Stinson (Eds.), 341-390.
Darcie, W., Boggs, R., Sammons, J. and Fenger, T. (2013). Online Anonymity: Forensic Analysis of the Tor Browser Bundle. ICDFSC 2013.
Dingledine, R. and Mathewson, N. (2016a). TOR Directory Specification.
https://gitweb.etorproject.org/torspec.git/tree/dir-spec.txt. (2016). Accessed March 2016.
Dingledine, R. and Mathewson, N. (2016b). TOR Protocol Specification.
https://gitweb.torproject.org/torspec.git/tree/tor-spec.txt. (2016). Accessed March 2016.
Dingledine, R., Mathewson, N. & Syverson, P. (2004). Tor: The Second-Generation Onion Router. 13th Security Symposium (USENIX), 303–320.
Dingledine, R., Mathewson, N., Murdoch, S. & Syverson, P. (2014). Tor: The Second-Generation Onion Router Draft 2014. http://www.cl.cam.ac.uk/, Accessed on 11-02-2014.
Douceur, J. (2002). The Sybil Attack. In: Proceedings of the 1st International Peer To Peer Systems Workshop (IPTPS 2002). Volume 2429 of LNCS, Springer.
Feigenbaum, J., Johnson, A. and Syverson, P. F. (2007). Probabilistic analysis of onion routing in a black-box model. 6th ACM Workshop on Privacy in the Electronic Society (WPES), 1–10.
Goldberg, I., Stebila, D. and Ustaoglu, B. (2012). Anonymity and one-way authentication in key exchange protocols. IEEE ICPA 2012.
Haraty, R.A. & Zantout, B. (2014). The TOR Data Communication System: A Survey. Journal of Communications and Networks, 16, 415-420.
Huhta, O. (2014). Linking Tor Circuits. MSc Information Security dissertation submitted to University College London.
Ghanem, M.C., Chen, T.M., Ferrag, M.A. and Kettouche, M.E., 2023. ESASCF: expertise extraction, generalization and reply framework for optimized automation of network security compliance. IEEE Access, 11, pp.129840-129853.
Jansen, R., Geddes, J., Wacek, C., Sherr, M. and Syverson, P. (2014). US Never Been KIST: Tor's Congestion Management Blossoms with Kernel-Informed Socket Transport. Proceedings of the 23rd USENIX Security Symposium, San Diego, CA ISBN 978-1-931971-15-7.
Johnson, A., Wacek, C., Jansen, R., Sherr, M. and Syverson, P. (2010). Users Get Routed: Traffic Correlation on Tor by Realistic Adversaries, Association for Computing Machinery, ACM, US.
Kate, A. and Goldberg, I. (2010). Distributed Private-Key Generators for Identity-Based Cryptography. 7th Conference on Security and Cryptography for Networks (SCN), 436–453.
Krovetz, T. and Rogaway, P. (2014).OCB implementation and performance analysis. IETF RFC publications.
Lazzari, M. (2014). Systematic Testing of Tor. Submitted as Master Thesis, ETH Zurich.
Ling, Z., Luo, J., Yu, W. and Fu, X. (2011). Equal-sized Cells Mean Equal-sized Packets in TOR. IEEE Communications Society subject matter experts for publication in the IEEE ICC 2011 proceedings.
Marquis, M. (2013). For their eyes only. The Commercialization of Digital Spying citizen lab Canada global security research.
Mccoy, D., Bauer, K., Grunwald, D., Kohno, T. and Sicker, D. (2008). Shining light in dark places: Understanding the tor network. 8th international symposium on Privacy Enhancing Technologies, PETS '08, 63-76, Berlin.
Murdoch, S. and Watson, R. (2007). Metrics for Security and Performance in Low-Latency Anonymity Systems. University of Cambridge, UK.
Ghanem, M.C. and Ratnayake, D.N., 2016, June. Enhancing WPA2-PSK four-way handshaking after re-authentication to deal with de-authentication followed by brute-force attack a novel re-authentication protocol. In 2016 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (CyberSA) (pp. 1-7). IEEE.
Nia, M.A., Karbasi, A.H. & Atani, R.E. (2014). Stop Tracking Me: An anti-detection type solution for anonymous data. 4th International eConference on Computer and Knowledge Engineering (ICCKE), 14, 685-690.
Øverlier, L. and Syverson, P. (2006). Locating hidden servers. In Proceedings of the 2006 IEEE Symposium on Security and Privacy, Oakland, CA, US, IEEE Computer Society.
Perry, M. (2007). Securing the Tor Network, Black Hat USA 2007 Supplementary Handout.
Reardon, J. and Goldberg, I. (2010). Improving TOR using a TCP-over-DTLS Tunnel. TOR project research papers, https://gitweb.etorproject.org.
Schanck, J., Whyte, W. and Zhang, Z. (2015). A quantum-safe circuit-extension handshake for Tor. Security innovation white paper.
Singh, S. (2015). Large-Scale Emulation of Anonymous Communication Networks. Matser thesis presented to the University of Waterloo.

 doi:10.20944/preprints202307.0982.v2

Ghanem, M., Mouloudi, A. and Mourchid, M., 2015. Towards a scientific research based on semantic web. Procedia Computer Science, 73, pp.328-335.

Ghanem, M., Dawoud, F., Gamal, H., Soliman, E., El-Batt, T. and Sharara, H., 2022, September. FLoBC: A decentralized blockchain-based federated learning framework. In 2022 Fourth International Conference on Blockchain Computing and Applications (BCCA) (pp. 85-92). IEEE.

Snader, R. and Borisov, N. (2008). A tune-up for Tor: Improving security and performance in the Tor network. Network & Distributed System Security Symposium, Interne Society.

Soghoian, C. (2011). Enforced Community Standards for Research on Users of the Tor Anonymity Network. Second Workshop on Ethics in Computer Security Research WECSR, 02, St. Lucia.

Stupples, D. (2013). Security Challenge of TOR and the Deep Web. The 8th International Conference for Internet Technology and Secured Transactions ICITST 2013.

Svenda, P. (2012). Basic comparison of Modes for Authenticated-Encryption (IAPM, XCBC, OCB, CCM, EAX, CWC, GCM, PCFB, CS). Masaryk University in Brno.

Farzaan, M.A.M., Ghanem, M.C., El-Hajjar, A. and Ratnayake, D.N., 2024. Ai-enabled system for efficient and effective cyber incident detection and response in cloud environments. arXiv preprint arXiv:2404.05602.

Syverson, S., Goldschlog, D. and Reeds, M. (1997). Anonymous connections and onion routing. Proceedings of the
IEEE Symposium on Security and Privacy, USA, 482-494.

TOR Deployment. (2016). TOR network detailed deployment. https://abouttor.tor.org, Acceded on April 2016.

TOR Flow. (2016). TOR flux across the world, https://torflow.uncharted.software 2016-1-13, accessed on April 2016.

TOR Project. (2016). TOR active users number in UK. https://metrics.torproject.org, accessed on April 2016. TOR Metrics. (2016). TOR Network overall bandwidth, https://metrics.torproject.org/bandwidth.html, accessed on April 2016.

Wacek, C., Tan, H., Bauer, K. and Sherr, M. (2013). An Empirical Evaluation of Relay Selection in TOR. In Proceedings of the Network and Distributed System Security Symposium - NDSS'13, The Internet Society.

Yenuguvanilanka, J. and Elkeelany, O. (2007). Performance Evaluation of Hardware Models of Advanced Encryption Standard (AES) Algorithm. Tennessee Tech University.

Dunsin, D., Ghanem, M.C., Ouazzane, K. and Vassilev, V., 2024. Reinforcement learning for an efficient and effective malware investigation during cyber Incident response. arXiv preprint arXiv:2408.01999.

Zhang, Y. (2009). Effective attacks in the tor authentication protocol. 3th International Conference on Network and