JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 39, 1371-1382 (2023)
DOI: 10.6688/JISE.202311_39(6).0008

Convenient Detection Method for Anonymous Networks "I2P vs Tor"

MEHDI MEROUANE
DIC Laboratory
Department of Electronics
Blida 1 University
Blida, 09100 Algeria
E-mail: mehdi_merouane@univ-blida.dz

Tor and I2P are the most popular low-latency anonymous communication networks, which use the onion routing technique to protect user anonymity. The use of Tor or I2P within a company can create network security risks. These two anonymous networks are different in a number of technical details, some of which are operated by hackers. In this paper a comparison of The Onion Router and the Invisible Internet Project "Tor vs I2P" is presented, in order to find a solution to detect the use of this kind of communication. Both networks will be contrasted and compared in their most important aspects namely node selection, performance and scalability. The detailed comparison of the two systems will allow us to develop a solution to detect the use of the anonymous "Tor & I2P" network by implementing their signatures in a Snort intrusion detection system with newly designed rules.

Keywords: ToR, I2P, anonymity, snort, signatures, IDS

1. INTRODUCTION

Several governments, organizations and other institutions have the capacity and resources to control Internet access. This power can be abused to filter content on the Internet, shutting down servers, revealing the identity of Internet users. So, there is a demand for anonymous internet access to escape censorship and surveillance and protect private life from the government or other powers. Anonymity allows you to use the Internet without the risk of reprisals. For this reason, anonymous communication attracted the attention of researchers and Internet users. As anonymous communication networks become develop to support more users, more and more anonymity tools become available free of charge. Some of these tools include proxy servers, virtual private network (VPN) services, the onion router (Tor) and the invisible internet project (I2P) [1].

Tor and I2P are the most popular common low latency anonymous communication networks, which use the onion routing technique to protect the anonymity of users through an open network of onion routers run by volunteers. Although Tor and I2P was designed to meet needs related to benevolent use, the latter present limits that you should be aware of so as not to overlook certain risks of security. The Tor network is often used for the distribution of goods and content illegally, while I2P could also be used by cybercriminals or terrorists to set up anonymous IRC networks or to develop file transfer applications. The use of Tor / I2P within a company can create network security risks such as the disclosure of confidential documents and the risk of infection by malwares. Therefore, it is recom-

Received May 20, 2022; revised September 9, 2022; accepted November 5, 2022.
Communicated by Meng Chang Chen.

mended that companies and administrations detect or even block communications that could be established to Tor / I2P nodes. In this paper, a detailed and comprehensive comparison of the two systems is presented. This will allow us to set up an architecture based on the signatures of these two anonymous networks in order to design our own detection rules in the Snort IDS and test this in a real architecture such as our university campus network.

2. RELATED WORD

Cyber security experts work on detecting any traces coming from Tor or I2P using different techniques and tools, but this is never an easy task. Previous works such as [10, 11] are based on analysis of Tor Traffic within a Tor node. Microsoft Cloud App Security (MCAS), for example, is able to provide alerts about the activity of a Tor IP address by detecting its traffic. However, this solution is specific to the Azure cloud environment. There are also publicly available lists of IP addresses and nodes used for monitoring by various applications and network protocols, including firewalls, IDS/IPS, NetFlow. In [13], the authors propose a system to detect and block Tor traffic in a network using deep packet inspection (DPI) and IDS-Bro for detection. In [9], the authors made a comparative study between the anonymous network Tor vs I2P but completely theoretical to differentiate between the functioning of the two networks. In the presented technique, the characteristics of Tor traffic are obtained not from deep packet inspection only of the three-way handshake. As a result, we now propose a practical solution on the investigation of anonymous (encrypted) traffic. To the best of the authors' knowledge, there is no study focusing on the method of implementing signatures taking into account encrypted traffic, which is the objective of this work.

3. I2P CONCEPT

The "Invisible Internet Project" (I2P), whose primary goal is anonymity, is a network that is isolated from other networks and functions as a network on top of the current internet infrastructure. The following anonymous web services can be developed using this network: blogs, forums, emails, SSH, outbound proxies, etc. The fact that sender and recipient only ever speak indirectly only through a number of routers known as tunnels ensures anonymity when using I2P. Users of this network can not only access or create content, but also develop online communities. Similar to the standard fundamental internet services like online browsing, email, and blogging, I2P offers a variety of services [2].

I2P operates at speeds comparable to those of the Internet. However, its structure and decentralization produce a setting that defies censorship and encourages the free exchange of ideas. In locations where information is restricted or forbidden, mirror sites hosted on the network give users access to news sources and other services.

3.1 Routing Mechanism

I2P Router A sends messages through a one-hop tunnel using I2P Router B and receives messages also using a one-hop tunnel using I2P F. The I2P D router also uses one-hop tunnels, where it sends data through the I2P E router, and receives data through the

I2P C router. The number of hops in an I2P tunnel varies between 1 and 7, the more hops in a tunnel the more anonymity increase, but reduces performance, since data must cross more intermediate nodes. Fig. 1 illustrates an example of routing mechanism of the I2P network.

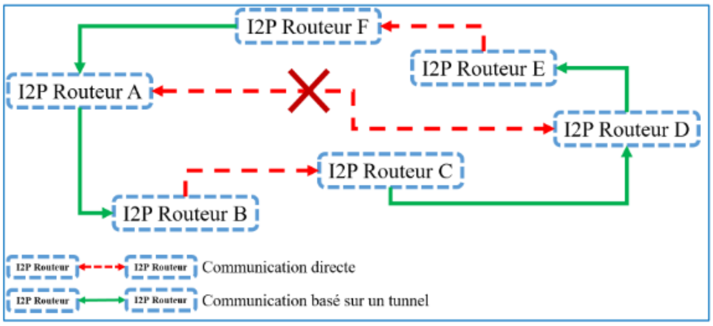


Fig. 1. Routing mechanism [1].

The I2P network is based on three key components: routers (or nodes), tunnels and the NetDB network database:

- Routers: are the users of the I2P software. All users pass communications through their machine.
- Tunnels: are unidirectional paths made up of several routers. Each router can be part of several inbound and outbound tunnels.
- The NetDB network database: contains information on routers and services available on the network. Specific routers, named Floodfill, are responsible for storing and keeping this database up to date.

Once its tunnels have been built, the node, to contact another node, will retrieve its «RouterInfo» in NetDB and therefore the list of gateways of the incoming tunnels from the remote node. Then, the node sends in one of its outgoing tunnels a message containing the information of one of the incoming tunnels of the remote node allowing the exit node of the outgoing tunnel to know to whom to retransmit the message [3].

3.2 I2P Tunnel

In order to establish anonymous and secure communication, I2P uses different layers of encryption. Communication security is therefore based on symmetric and asymmetric cryptography. I2P uses four ciphers:

1. For outgoing tunnels, the sender adds multiple layers of encryption (one per node traversed). Each layer is then removed (decrypted) by each node crossed.
2. I2P uses "garlic" encryption to protect the content of information passing between the various routers in the tunnel. The principle of garlic routing is to encrypt several messages together and to circulate them in the same packet, this allows on the one hand to accelerate the transfer speed but also to make it more difficult to analyze the traffic in Fig. 2.

3. The AES256 symmetric encryption algorithm is used for end-to-end transmitted message encryption (the key is encrypted using ElGamal).
4. The ElGamal system is used for asymmetric encryption which takes place end-to-end between sender and receiver. This is also used for NetDB registrations and requests sent to the "Floodfill" router.

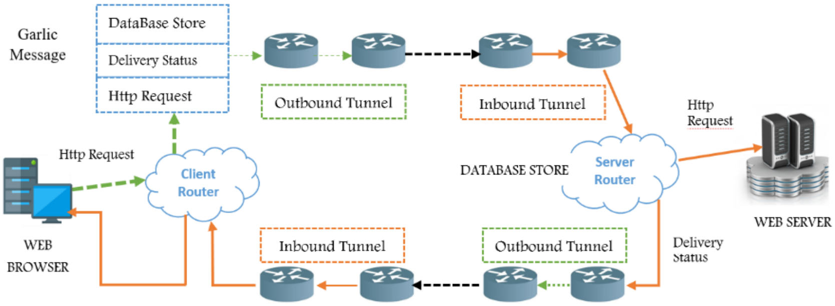


Fig. 2. I2P communication diagram.

4. TOR CONCEPT

Tor is one of the best-known and most popular tools for privacy and anonymity on the Internet. It is both free software and an overlay network distributed in nodes on a voluntary basis. Under the term decentralized are hidden thousands of servers are made available by volunteers called nodes and which act as relays to allow the anonymization of connections [4]. Tor was originally designed with the following considerations in mind:

- Deployability: Tor must be deployable and used in the real world. So, it doesn't have to be expensive and it doesn't have to be difficult to implement.
- Simple design: The protocol and security settings should be easy to understand. Additional features impose implementation and complexity costs.
- Usability: A hard-to-use system logically has fewer users, and because anonymization systems hide users from other users, usability is a real need for security.
- Flexibility: Tor's protocol must be flexible and well-specified, so that Tor can be used as a basis for research. Tor wants to establish itself as a standard.

4.1 Onion Routing

In Fig. 3, we can see a user establishing an onion router circuit, communication is layered and encrypted with each public key of the nodes in the circuit. Each node in the circuit knows only its predecessor and its successor [5]. Nodes within Tor can be of several types:

- The Onion Router (OR, also called relay): These are the nodes that make up the circuits used through the network. They are the ones that send packets through the Tor cloud.
- Client nodes (also called proxy onions, OPs, by abuse of language): These are the nodes that connect to the network, or more precisely the Tor software clients.

- Directory Servers (also called authority Servers): These are the servers that reference known ORs, they are the network directories.
- Tor can place a heavy demand on your corporate network bandwidth. This can permanently expose your organization to a Distributed Denial of Service (DDoS) attack, which can make your server, service, or infrastructure unavailable.

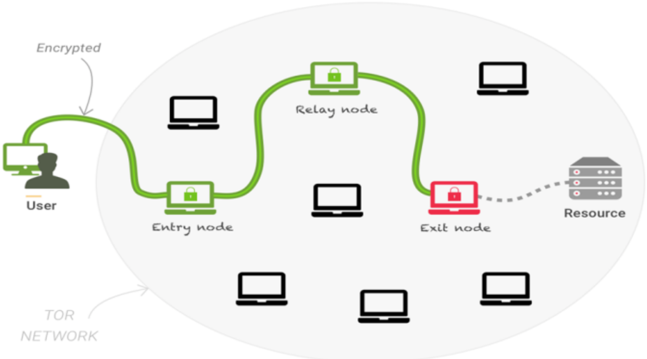


Fig. 3. Tor communication.

5. SUMMARY: I2P VS. TOR

The key philosophical difference between the well-known Tor network and I2P is that I2P tries to move existing Internet services into the I2P network and provide service implementations within the framework whereas Tor enables anonymous access to external Internet services implemented and operated separately. While Tor has hidden services and I2P has exit nodes, the canonical usage of Tor is accessing external services and the canonical usage of I2P is accessing integrated services. I2P and Tor also differ in a number of technical details, some of which are key to an attack.

The main technical differences between the two projects are summarized below:

5.1 The Onion Router (TOR)

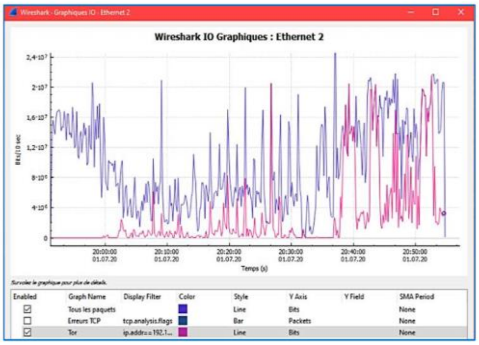
- Directory servers which are the central database of the network.
- Internal relays through which traffic flows from Tor to Tor.
- The exit nodes to the Internet.
- Tor protects every element of the message in three layers of encryption, a technique called "onion routing."
- To increase performance, Tor configures multiple users to follow the same path across the network.
- Tor uses two-way encrypted connections to relay data to egress nodes or hidden services, it always uses three relays to traverse its network. In addition, Tor does not support the UDP protocol.
- Since Tor always uses IP addresses, it uses DNS resolution for normal internet browsing.

- In TOR, only the ingress node knows the identity of the user and the egress node is the only one that knows the recipient.

5.2 Invisible Internet Project (I2P)

- Each node is a router and there is no distinction, unlike Tor. Each I2P user is a router in a tunnel relaying traffic.
- I2P has a network database (NetDB), distributed and maintained by Floodfill routers. No central server exists.
- I2P uses a variation of onion routing called garlic routing to create anonymous conec-ions.
- I2P uses one-way connections between each server in its tunnels. It supports TCP and UDP which provides better performance for some applications.
- I2P therefore does not use an IP address/ domain name match to access the various services.
- In I2P, a node cannot know whether the messages it transmits come directly from a user or from another node in the tunnel.

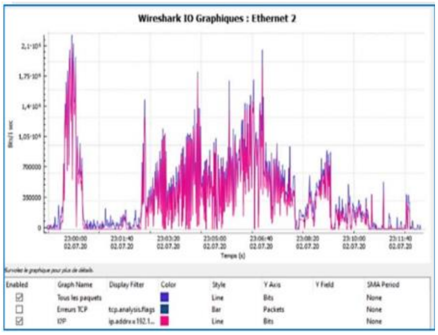
Bandwidth: We notice that the consumption of bandwidth when downloading via I2P is greater compared to downloading via ToR this comes down to the number of hops (node) used by I2P which are 8 nodes in total (4 inbound, 4 outbound), compared to those of Tor which are 3 nodes in total.



Survolez le graphique pour plus de détails.

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	Tous les paquets		🟦	Line	Bits		None
□	Erreurs TCP	tcp.analysis.flags	🟦	Bar	Packets		None
☑	Tor	ip.addr==192.1...	🟪	Line	Bits		None

Fig. 4. Bandwidth for I2P.



Survolez le graphique pour plus de détails.

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	Tous les paquets		🟦	Line	Bits		None
□	Erreurs TCP	tcp.analysis.flags	🟦	Bar	Packets		None
☑	I2P	ip.addr==192.1...	🟥	Line	Bits		None

Fig. 5. Bandwidth for I2P.

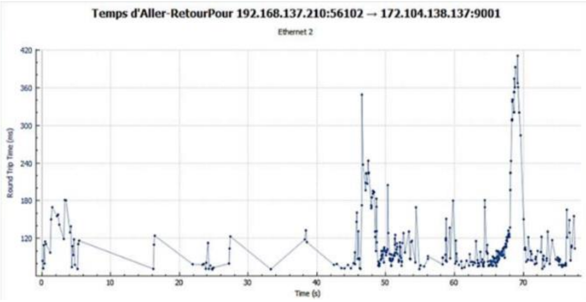


Fig. 6. Latency (RTT) for ToR.

Latency: This is the time required to convey a packet through a network. Latency can be measured in several ways: RTT “Round-Trip Time”, RTD round-trip delay time “Round-Trip Delay”. For Tor, The RTT reaches its maximum value of “380ms” and it has “20ms” as minimum value.

The graph represents only 2 large spikes of the RTT, and the rest is between “20ms-180ms”. For a given communication plus the RTT is greater the slower the requests.

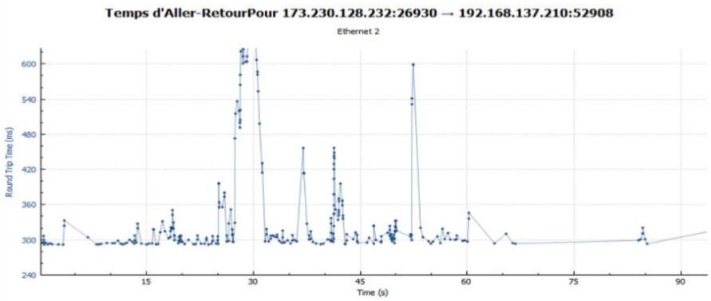


Fig. 7. Latency (RTT) for I2P.

On the other hand, for the I2P, the RTT reaches its maximum value of “600ms” and it has “290ms” as a minimum value, the graph presents several peaks of the RTT between “380ms-600ms”, and the rest is between “290ms-340ms”. RTT in Tor is about half the RTT in I2P, which means Tor performs better than I2P when it comes to downloading and Tor performs better than I2P when browsing the normal web anonymously.

6. EXPERIMENTATION

In this section, we go over the research that supports the comparison of the two anonymous networks. It should be highlighted right away that it is challenging to find and halt Tor and I2P in a business network. The administrator should compare the differences between traffic from the Tor / I2P network and regular web traffic in order to identify the Tor network and I2P. This essentially involves obtaining their identification through the Tor / I2P network’s digital signatures. To this end, we will first construct a client-server architecture whose objective is to mimic a corporate network (Fig. 8). Following that, we will compare the performance of I2P and the Tor network. Then, using an analysis of the varied

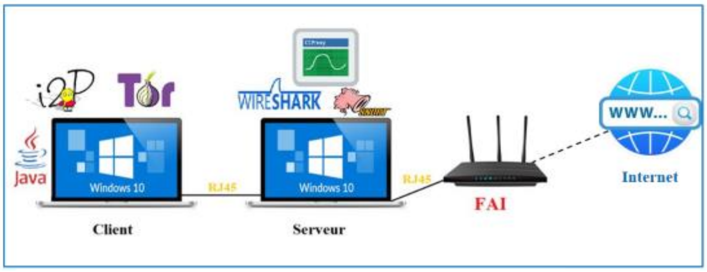
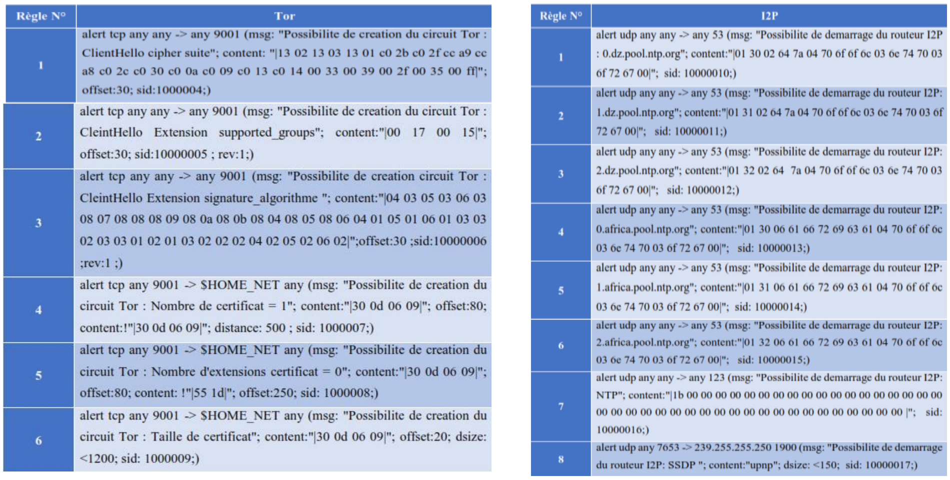


Fig. 8. Our network designs.

traffic from both regular browsers and the Tor/I2P browser, we will determine the presence of Tor/I2P on our network, revealing the hallmarks of these two anonymity networks (Wireshark analysis). A collection of IDS Snort rules was developed to detect "Tor Vs I2P" traffic after investigating and analyzing the behavior and characteristics of Tor/I2P (Fig. 9).



Règle N°	Tor
1	alert tcp any any -> any 9001 (msg: "Possibilite de creation du circuit Tor : ClientHello cipher suite"; content: "|13 02 13 03 13 01 c0 2b c0 2f cc a9 cc a8 c0 2c c0 30 c0 0a c0 09 c0 13 c0 14 00 33 00 39 00 2f 00 35 00 ff|"; offset:30; sid:1000004;)
2	alert tcp any any -> any 9001 (msg: "Possibilite de creation du circuit Tor : CleintHello Extension supported_groups"; content:"|00 17 00 15|"; offset:30; sid:10000005 ; rev:1;)
3	alert tcp any any -> any 9001 (msg: "Possibilite de creation circuit Tor : CleintHello Extension signature_algorithme "; content:"|04 03 05 03 06 03 08 07 08 08 08 09 08 0a 08 0b 08 04 08 05 08 06 04 01 05 01 06 01 03 03 02 03 03 01 02 01 03 02 02 02 04 02 05 02 06 02|";offset:30 ;sid:10000006 ;rev:1 ;)
4	alert tcp any 9001 -> $HOME_NET any (msg: "Possibilite de creation du circuit Tor : Nombre de certificat = 1"; content:"|30 0d 06 09|"; offset:80; content:!"|30 0d 06 09|"; distance: 500 ; sid: 1000007;)
5	alert tcp any 9001 -> $HOME_NET any (msg: "Possibilite de creation du circuit Tor : Nombre d'extensions certificat = 0"; content:"|30 0d 06 09|"; offset:80; content: !"|55 1d|"; offset:250; sid: 1000008;)
6	alert tcp any 9001 -> $HOME_NET any (msg: "Possibilite de creation du circuit Tor : Taille de certificat"; content:"|30 0d 06 09|"; offset:20; dsize: <1200; sid: 1000009;)

Règle N°	I2P
1	alert udp any any -> any 53 (msg: "Possibilite de demarrage du routeur I2P : 0.dz.pool.ntp.org"; content:"|01 30 02 64 7a 04 70 6f 6f 6c 03 6e 74 70 03 6f 72 67 00|"; sid: 10000010;)
2	alert udp any any -> any 53 (msg: "Possibilite de demarrage du routeur I2P: 1.dz.pool.ntp.org"; content:"|01 31 02 64 7a 04 70 6f 6f 6c 03 6e 74 70 03 6f 72 67 00|"; sid: 10000011;)
3	alert udp any any -> any 53 (msg: "Possibilite de demarrage du routeur I2P: 2.dz.pool.ntp.org"; content:"|01 32 02 64 7a 04 70 6f 6f 6c 03 6e 74 70 03 6f 72 67 00|"; sid: 10000012;)
4	alert udp any any -> any 53 (msg: "Possibilite de demarrage du routeur I2P: 0.africa.pool.ntp.org"; content:"|01 30 06 61 66 72 69 63 61 04 70 6f 6f 6c 03 6e 74 70 03 6f 72 67 00|"; sid: 10000013;)
5	alert udp any any -> any 53 (msg: "Possibilite de demarrage du routeur I2P: 1.africa.pool.ntp.org"; content:"|01 31 06 61 66 72 69 63 61 04 70 6f 6f 6c 03 6e 74 70 03 6f 72 67 00|"; sid: 10000014;)
6	alert udp any any -> any 53 (msg: "Possibilite de demarrage du routeur I2P: 2.africa.pool.ntp.org"; content:"|01 32 06 61 66 72 69 63 61 04 70 6f 6f 6c 03 6e 74 70 03 6f 72 67 00|"; sid: 10000015;)
7	alert udp any any -> any 123 (msg: "Possibilite de demarrage du routeur I2P: NTP"; content:"|1b 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |"; sid: 10000016;)
8	alert udp any 7653 -> 239.255.255.250 1900 (msg: "Possibilite de demarrage du routeur I2P: SSDP "; content:"upnp"; dsize: <150; sid: 10000017;)

Fig. 9. Snort Detection rules comparing ToR vs. I2P.

6.1 Tor Detection

After analyzing the packets from the "TCP Handshake" and "TLS Handshake" connections, we were able to find several Tor identifiers that can characterize Tor traffic compared to ordinary traffic, we provide the list of identifiers below,

1. The TCP port.
2. The "Cipher Suites" in the "ClientHello" message.
3. ''Supported groups'' in the "ClientHello" message
4. "Signature Algorithm" used in "ClientHello".
5. Type of extension in the "ClientHello" message.
6. The length of the "Certificate" message.
7. "Common Name" in the "Certificate" message.
8. The extension number of the "Certificate" message.

These credentials will be used as digital signatures to detect Tor traffic, detection is performed using a "Snort" network intrusion detection system. Tor uses TLS in an unusual way that allows us to indicate Tor usage by looking only at encrypted traffic.

6.2 I2P Detection

After analyzing the packets of the I2P network during its three different states we were able to find I2P identifiers that can characterize I2P traffic compared to ordinary traffic, we provide the list of identifiers below,

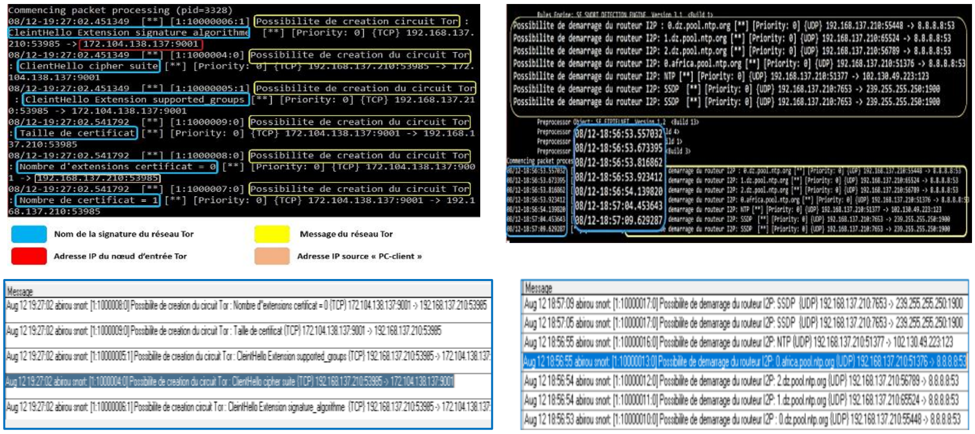
1. Sends “dz.pool.ntp.org” queries to the DNS server to find the NTP servers.
2. Sends “africa.pool.ntp.org” queries to the DNS server to find the NTP servers.
3. Synchronization request to the NTP server “africa.pool.ntp.org”.
4. Total length of SSDP packet and SSDP / UPnP source port used.
5. The number of UDP packets with a port greater than 9000.
6. The number of TCP packets with the PSH / ACK flag with a port greater than 9000.

These credentials will be used as digital signatures to detect I2P traffic, detection is performed using a “Snort” network intrusion detection system.

To test the reliability of our rules, we will perform two tests:

1. Browsing with the Firefox browser and the Chrome browser using the PC-client to see if Snort generates false alerts (false positives).
2. Browse with the Tor browser and start the I2P router alternately using the PC-client to see if Snort detects the use of these.

As soon as we launch Tor Browser, we noticed that Snort generates all six alerts at the same time, these correspond to the Tor Browser rules shown in the table above. This means detection of the use of the Tor browser. After we start the I2P router we noticed that Snort generates all six alerts at the same time, these correspond to the I2P network rules presented in Fig. 10 (the table in black).



```
Commencing packet processing (pid=3328)
08/12-19:27:02.451349  [**] [1:10000006:1] Possibilite de creation circuit Tor :
CleintHello Extension signature algorithme  [**] [Priority: 0] {TCP} 192.168.137.
210:53985 -> 172.104.138.137:9001
08/12-19:27:02.451349  [**] [1:1000004:0] Possibilite de creation du circuit Tor
: ClientHello cipher suite [**] [Priority: 0] {TCP} 192.168.137.210:53985 -> 172.
104.138.137:9001
08/12-19:27:02.451349  [**] [1:10000005:1] Possibilite de creation du circuit Tor
 : CleintHello Extension supported_groups [**] [Priority: 0] {TCP} 192.168.137.21
0:53985 -> 172.104.138.137:9001
08/12-19:27:02.541792  [**] [1:1000009:0] Possibilite de creation du circuit Tor
: Taille de certificat [**] [Priority: 0] {TCP} 172.104.138.137:9001 -> 192.168.1
37.210:53985
08/12-19:27:02.541792  [**] [1:1000008:0] Possibilite de creation du circuit Tor
: Nombre d'extensions certificat = 0 [**] [Priority: 0] {TCP} 172.104.138.137:900
1 -> 192.168.137.210:53985
08/12-19:27:02.541792  [**] [1:1000007:0] Possibilite de creation du circuit Tor
: Nombre de certificat = 1 [**] [Priority: 0] {TCP} 172.104.138.137:9001 -> 192.1
68.137.210:53985
```

Nom de la signature du réseau Tor
Message du réseau Tor
Adresse IP du nœud d’entrée Tor
Adresse IP source « PC-client »

```
Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.1  <Build 1>
Possibilite de demarrage du routeur I2P : 0.dz.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:55448 -> 8.8.8.8:53
Possibilite de demarrage du routeur I2P: 1.dz.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:65524 -> 8.8.8.8:53
Possibilite de demarrage du routeur I2P: 2.dz.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:56789 -> 8.8.8.8:53
Possibilite de demarrage du routeur I2P: 0.africa.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:51376 -> 8.8.8.8:53
Possibilite de demarrage du routeur I2P: NTP [**] [Priority: 0] {UDP} 192.168.137.210:51377 -> 102.130.49.223:123
Possibilite de demarrage du routeur I2P: SSDP  [**] [Priority: 0] {UDP} 192.168.137.210:7653 -> 239.255.255.250:1900
Possibilite de demarrage du routeur I2P: SSDP  [**] [Priority: 0] {UDP} 192.168.137.210:7653 -> 239.255.255.250:1900
        Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
        Preprocessor 08/12-18:56:53.557032 ld 4>
        Preprocessor 08/12-18:56:53.673395 ld 1>
        Preprocessor 08/12-18:56:53.816862 <Build 3>
Commencing packet proces 08/12-18:56:53.923412
08/12-18:56:53.557032 [ 08/12-18:56:54.139820 demarrage du routeur I2P : 0.dz.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:55448 -> 8.8.8.8:53
08/12-18:56:53.673395 [ 08/12-18:57:04.453643 demarrage du routeur I2P: 1.dz.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:65524 -> 8.8.8.8:53
08/12-18:56:53.816862 [ 08/12-18:57:09.629287 demarrage du routeur I2P: 2.dz.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:56789 -> 8.8.8.8:53
08/12-18:56:53.923412 [ demarrage du routeur I2P: 0.africa.pool.ntp.org [**] [Priority: 0] {UDP} 192.168.137.210:51376 -> 8.8.8.8:53
08/12-18:56:54.139820 [ demarrage du routeur I2P: NTP [**] [Priority: 0] {UDP} 192.168.137.210:51377 -> 102.130.49.223:123
08/12-18:57:04.453643 [ demarrage du routeur I2P: SSDP  [**] [Priority: 0] {UDP} 192.168.137.210:7653 -> 239.255.255.250:1900
08/12-18:57:09.629287 [ demarrage du routeur I2P: SSDP  [**] [Priority: 0] {UDP} 192.168.137.210:7653 -> 239.255.255.250:1900
```

```
Message
Aug 12 19:27:02 abirou snort: [1:1000008:0] Possibilite de creation du circuit Tor : Nombre d'extensions certificat = 0 {TCP} 172.104.138.137:9001 -> 192.168.137.210:53985
Aug 12 19:27:02 abirou snort: [1:1000009:0] Possibilite de creation du circuit Tor : Taille de certificat {TCP} 172.104.138.137:9001 -> 192.168.137.210:53985
Aug 12 19:27:02 abirou snort: [1:10000005:1] Possibilite de creation du circuit Tor : CleintHello Extension supported_groups {TCP} 192.168.137.210:53985 -> 172.104.138.137:
Aug 12 19:27:02 abirou snort: [1:1000004:0] Possibilite de creation du circuit Tor : ClientHello cipher suite {TCP} 192.168.137.210:53985 -> 172.104.138.137:9001
Aug 12 19:27:02 abirou snort: [1:10000006:1] Possibilite de creation circuit Tor : CleintHello Extension signature_algorithme {TCP} 192.168.137.210:53985 -> 172.104.138.137:
```

```
Message
Aug 12 18:57:09 abirou snort: [1:10000017:0] Possibilite de demarrage du routeur I2P: SSDP {UDP} 192.168.137.210:7653 -> 239.255.255.250:1900
Aug 12 18:57:05 abirou snort: [1:10000017:0] Possibilite de demarrage du routeur I2P: SSDP {UDP} 192.168.137.210:7653 -> 239.255.255.250:1900
Aug 12 18:56:55 abirou snort: [1:10000016:0] Possibilite de demarrage du routeur I2P: NTP {UDP} 192.168.137.210:51377 -> 102.130.49.223:123
Aug 12 18:56:55 abirou snort: [1:10000013:0] Possibilite de demarrage du routeur I2P: 0.africa.pool.ntp.org {UDP} 192.168.137.210:51376 -> 8.8.8.8:53
Aug 12 18:56:54 abirou snort: [1:10000012:0] Possibilite de demarrage du routeur I2P: 2.dz.pool.ntp.org {UDP} 192.168.137.210:56789 -> 8.8.8.8:53
Aug 12 18:56:54 abirou snort: [1:10000011:0] Possibilite de demarrage du routeur I2P: 1.dz.pool.ntp.org {UDP} 192.168.137.210:65524 -> 8.8.8.8:53
Aug 12 18:56:53 abirou snort: [1:10000010:0] Possibilite de demarrage du routeur I2P : 0.dz.pool.ntp.org {UDP} 192.168.137.210:55448 -> 8.8.8.8:53
```

Fig. 10. Detection of anonymous traffic (Tor vs. I2P).

7. DISCUSSION

From several tests done with Tor browser, we got the same IP address from the ingress node, this means that the latter is fixed for a while. In our case since we installed the Tor browser, our address IP of the entry node has not changed (two to three months) this confirms the analysis part.

The combination of the two conditions in our Tor rules allowed us to avoid false alerts, as these two conditions in the six alerts triggered at some time can only be generated by the Tor browser. It is also possible to identify I2P traffic during the initialization / bootstrap phase because we were able to quickly detect DNS requests to the internet. Starting the I2P router always begins by requesting the IP addresses of the NTP servers from dz.pool.ntp.org, this behavior will be distinguished quickly because generally PCs in an enterprise do not request an NTP request to synchronize the time.

In network management, firewall rules are often used to allow or filter traffic. Popular blocking techniques are often based on the port number, protocol signature, and IP address. However, anonymity networks, including Tor and I2P, are designed to withstand censorship. As a result, any attempt to permanently block these networks could cause collateral damage. For port-based censorship, blocking onion relay ports or directory information exchange ports is effective in blocking Tor relays momentarily, and blocking UDP port 123 would prevent I2P to work properly because the I2P router software needs NTP to work properly.

A more efficient approach is destination filtering. To implement this approach, a censor must compile a list of active I2P peer addresses and block access to all of them. This address-based blocking approach will have a severe impact on the process of forming new I2P tunnels, thus preventing users from accessing the I2P network momentarily. Additionally, a simpler but still effective way to prevent new users from accessing I2P is to block access to the I2P reboot servers, which are required for the boot process. Therefore, new users will not be able to access the I2P network if they cannot retrieve RouterInfo from other peers. IDS also saves detected TOR packets (Tor alerts) in a log file which will be used to create a list of destination IP addresses for Tor packets, this list will be implemented in a proxy server to block the connection. In addition, our detection method raises a number of interesting open questions such as,

- Relay mixes that are provided to the Tor client to establish circuits and connections need more research, a thorough evaluation of the allocation policy and its potential for location leakage or other identity information.
- Study the flows of the I2P network in detail in order to provide a more complete understanding of the network as a whole. This will contribute to an effective investigation of I2P activities.

In order to evaluate the proposed rules, background traffic was generated along with Tor and I2P connections after a realistic test with real traffic captured in a campus network for a period of three months. With these new rules we were able to identify all connections generated from a Tor browser or that use I2P services, the two following figures show the number of Tor and I2P detection alerts.

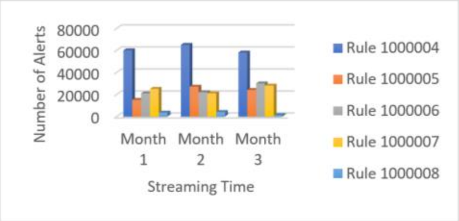


Fig. 11. Proportion of Snort rules triggered for Tor traffic in the campus.

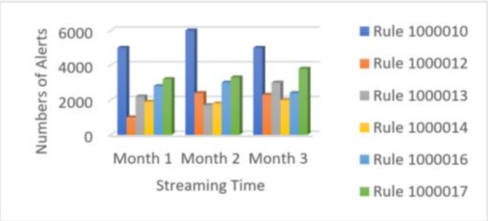


Fig. 12. Proportion of Snort rules triggered for I2P traffic in the campus.

8. CONCLUSION

In a business, an employee's internet connection is supposed to be strictly professional. The employer has the right to prohibit private use of the Internet at work if it considers that such use is detrimental to the performance of the employee and the company. However, this rule is not always observed. Some employees allow themselves to unblock sites to which access is prohibited by the employer. Several unblocking methods are available, including those using specific software such as TOR and I2P. These software's conceal the IP address allowing another path to be used to access blocked sites. However, using Tor / I2P within an enterprise can pose network security risks. Among these:

- ✓ Software (Tor / I2P) can be used to bypass controls put in place to prevent the disclosure of confidential documents.
- ✓ Tor / I2P can encrypt all traffic on your network and make it very difficult to monitor your activities.
- ✓ Any user who downloads content through Tor / I2P could therefore put the corporate network at risk of malware infection.
- ✓ Tor exit nodes can monitor traffic passing through employee devices and capture any unencrypted information such as login and password.

Through this study, we were able to understand that although Tor and I2P provide similar functionality, there are some major differences between them. Tor works at the TCP stream level, while I2P traffic can use both TCP and UDP. Tor has a centralized architecture, while no single entity has a complete view of the I2P network due to its decentralized nature. As a result, we performed Tor network discovery based on analyzing and comparing web traffic from regular browsers to Tor browser traffic. Tor disguises its connection to look like an ordinary HTTPS connection. We analyzed his TLS handshake process. Next, we performed I2P network discovery based on traffic analysis when starting the I2P router. During the initialization phase the I2P router needs the NTP protocol to function properly by sending DNS queries to obtain the IP addresses of the nearest NTP servers.

We have used these characteristics to identify I2P traffic. As a final step, we implemented the characteristics of the two networks as rules in a Snort intrusion detection system. This system then generated alerts that are triggered as soon as the two networks are launched. This allowed us to confirm that detection of Tor network usage and I2P network usage in an enterprise is possible with our approach.

REFERENCES

1. J.-P. Timpanaro, T. Cholez, I. Chrisment, and O. Festor, "Evaluation of the anonymous I2P network's design choices against performance and security," in *Proceedings of the 1st International Conference on Information Systems Security and Privacy*, 2015, pp. 46-55.
2. C. Egger, J. Schlumberger, C. Kruegel, and G. Vigna, "Practical attacks against the I2P network," in *Proceedings of International Workshop on Recent Advances in In-*

trusion Detection, *Research in Attacks*, *Intrusions*, *and Defenses*, 2013, pp. 432-451.

3. A. Alharbi *et al*., "Exploring the topological properties of the tor dark web," *IEEE Access*, Vol. 9, 2021, pp. 21746-21758.
4. Anonymity Services Tor, I2P, "JonDonym: Classifying in the dark (web)," *IEEE Transactions on Dependable and Secure Computing*, Vol. 17, 2020, pp. 662-675.
5. E. Jardine, "The dark web dilemma: Tor, anonymity and online policing," *Global Commission on Internet Governance Paper Series*, 2015, pp. 1-24.
6. M. Merouane, "Interception of P2P Traffic in a Campus Network," *Romanian Journal of Information Technology and Automatic Control*, Vol. 29, 2019, pp. 21-34.
7. A. Springall, C. DeVito, and S.-H. S. Huang, "Per connection server-side identification of connections via tor," in *Proceedings of IEEE 29th International Conference on Advanced Information Networking and Applications*, 2015, pp. 727-734.
8. R. Liggett, J. R. Lee, A. L. Roddy, and M. A. Wallin, "The dark web as a platform for crime: An exploration of illicit drug, firearm, CSAM, and cybercrime markets" in *The Palgrave Handbook of International Cybercrime*, 2020, pp. 91-116.
9. A. Ali and M. Khan, "ToR vs I2P: A comparative study," in *Proceedings of IEEE International Conference on Industrial Technology*, 2016, pp. 1748-1751.
10. Z. Ling, J. Luo, K. Wu, W. Yu, and X. Fu, "Towards discovery of malicious traffic over Tor," in *Proceedings of IEEE Conference on Computer Communications*, 2014, p. 14021410.
11. M. AlSabah, K. Bauer, and I. Goldberg. "Enhancing Tor's performance using realtime traffic classification," in *Proceedings of ACM Conference on Computer and Communications Security*, 2012, pp. 73-84.
12. F. A. Saputra, I. U. Nadhori, and B. F. Barry, "Detecting and blocking onion router traffic using deep packet inspection," in *Proceedings of International Electronics Symposium*, 2016, pp. 283-288
13. J.-X. Zhang and L.-Y. Zhang "Anonymous CP-ABE against side-channel attacks in cloud computing," *Journal of Information Science and Engineering*, Vol. 33, 2017, pp. 789-805.

Mehdi Merouane (马赫迪·马尔万) received his Ph.D. degree in Telecommunications from the Blinda 1 University of Electronics department. Currently, he is an Assistant Professor in the same University. His research focuses on computer network and security based on digital signatures, provable security, cryptographic protocols and cloud security. Dr. Mehdi has published over 20 papers at prestigious journals and conferences.