الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique

Mémoire de Master

Filière Télécommunication
Spécialité Réseaux & Télécoms

présenté par

LAICHI Hafsa

&

TAIEB BENABBES Abir

Le réseau anonyme Tor VS I2P " comparaison et détection"

Proposé par : Dr. MEHDI Merouane.

Année Universitaire 2019-2020.

Remerciement

Nos remerciements, avant tout, à ALLAH tout puissant pour la volonté, la santé et la patience qu'il nous a données durant toutes ces longues années d'études afin que nous puissions arriver à ce stade.

Nous remercions nos très chers parents qui ont toujours été là pour nous, nous les remercions pour leur soutien constant et leurs encouragements.

Nous tenons à exprimer notre reconnaissance à notre encadreur Monsieur Mehdi Merouane. Nous le remercions pour la confiance qu'il nous a accordée. Nous aimerons aussi le remercier pour ses conseils qui nous ont permis de mener bien ce travail. Qu'il trouve en ce mémoire l'expression de notre respect infini.

Nous adressons nos sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé nos réflexions et ont accepté à nous rencontrer et répondre à nos questions durant notre recherches.

Nous adressons nos sincères remerciements et notre reconnaissance également à tous les enseignants du département électronique pour la formation qui nous ont assurée et leur soutien tout au long de notre parcours universitaire.

Nous remercions le membre de jury pour nous avoir fait l'honneur de juger notre travail.

Afin de n'oublier personne, nos vifs remerciements s'adressent à tous ceux qui nous ont aidées à la réalisation de ce modeste mémoire.

Dédicace

Je dédie ce modeste travail comme signe de respect, reconnaissance et de remerciement :

À mes chers parents, qui ne cessent de me donner avec amour le nécessaire pour que je puisse arriver à ce que je suis aujourd'hui « Vous avez tout sacrifié pour vos enfants n'épargnant ni santé ni efforts. Vous m'avez donné un magnifique modèle de labeur et de persévérance. Je suis redevable d'une éducation dont je suis fier ».Que dieux vous protège et que la réussite soit toujours à ma portée pour que je puisse vous combler de bonheur.

À ma chère soeur Nesrine ainsi qu'à mon frère Abdoullah, en reconnaissance de leur affection toujours constante, ainsi que toute ma famille et mes proches.

À mes chères amies Manel, Fatma-Zohra, Sonya, Asma et Hadjer.

À Toutes mes amies, mes collègues.

À tous ceux qui m'estiment.

Abir

Dédicace

Avec un énorme plaisir, un cœur ouvert et une immense joie, que je dédie ce modeste travail :

À mes très chers parents, le symbole de tendresse et l'école de mon enfance, pour leurs patiences, leurs sacrifices, et leurs soutiens durant mes études, aucun hommage ne pourra être à la hauteur de l'amour dont ils ne cessent de me combler. Que dieu leurs procure bonne santé et longue vie.

À mes chères sœurs Houria et Hiba, pour leurs amours, ainsi que toute ma famille

À mes chers grands-pères que dieu les accueillent dans son vaste paradis

À mes chères amies Sara, Hala, Manel, Fatma-Zohra, Asma et Hadjer.

Toutes mes amies, mes collègues.

À tous ceux qui m'aiment.

À tous ceux que j'aime.

Hafsa

ملخص:

إن طور و ايدوبي هما أكثر شبكات الاتصال المجهولة شيوعا ذات زمن انتقال منخفض، والتي تستخدم تقنية توجيه البصل لحماية هوية المستخدم، فإن استخدام طور و ايدوبي داخل المؤسسة يمكن أن يؤدي إلى مخاطر في أمن الشبكة. والغرض من هذه الذاكرة هو مقارنة واكتشاف استخدام شبكة طور و ايدوبي.

ولكي نفعل هذا، ناقشنا مفهوم عدم الكشف عن الهوية والخصوصية في شبكة الانترنيت، ثم المقارنة بين الشبكتين المجهولتين طور و ايدوبي ، فقد سمح لنا هذا الأخير بالفهم الأفضل لعملهما. وأخيراً اكتشفنا استخدام شبكة طور و ايدوبي من خلال تنفيذ توقيعاتهم في نظام كشف التسلل سنورت.

كلمات المفاتيح : شبكة الطور؛ شبكة ايدوبي؛ سنورت؛ بصمات رقمية؛ وايرشارك؛ نظام كشف التسلل.

Résumé :

Tor et I2P sont les réseaux de communication anonyme à faible latence les plus populaires, qui utilisent la technique de routage à l'oignon pour protéger l'anonymat des utilisateurs, L'utilisation du Tor/I2P au sein d'entreprise peut engendrer des risques de sécurités de réseau. L'objectif de ce mémoire est la comparaison et la détection de l'utilisation du réseau Tor et I2P.

Pour ce faire, nous avons abordé le concept de l'anonymat et la vie privée, puis la comparaison entre les deux réseaux anonymes Tor et I2P, cette derniere nous a permis de comprendre mieux leurs fonctionnements. Enfin, nous avons détecté l'utilisation du réseau Tor et I2P par l'implémentation de leurs signatures dans un système de détection d'intrusions Snort.

Mots clés : Réseau Tor ; réseau I2P ; Snort ; IDS ; anonymat ; Wireshark ; signatures.

Abstract :

Tor and I2P are the most popular low-latency anonymous communication networks, which use onion routing technique to protect user anonymity, the use of Tor/I2P within an enterprise can create network security risks. The purpose of this memory is to compare and detect the use of the Tor and I2P network.

To do this, we discussed the concept of anonymity and privacy, then the comparison between the two anonymous networks Tor and I2P, the latter has allowed us to better understand their functioning. Finally, we detected the use of the Tor and I2P network by implementing their signatures in a Snort intrusion detection system.

Keywords : Tor network; I2P network; Snort; IDS; anonymity; Wireshark; signatures.

Liste des abréviations

A

AES : Advanced Encryptions Standard.

ADSL : Asymmetric Digital Subscriber Line.

ACK : Acknowledgement.

ARP : Address Resolution Protocol.

D

DNS : Domain Name System.

DHCP : Dynamic Host Configuration Protocol.

DSA : Digital Signature Algorithm.

DMZ : Demilitarized zone.

DOS : Denial of Service.

DDoS : Distributed Denial of Service.

E

ECC : Elliptic Curve Cryptography.

ECDHE : Elliptic Curve Diffie–Hellman Ephemeral.

F

FDDI : Fiber Distributed Data Interface.

FAI : Un Fournisseur d'Accès à Internet.

FTP : File Transfer Protocol.

FIN : Finish.

G

GCM : Galois Counter Mode.

H

HTTP : HyperText Transfer Protocol.

HIDS : Host Intrusion Detection System.

I

IP : Internet Protocol.

ISN : Initial Sequence Number.

ICMP : Internet Control Message Protocol.

I2P : Invisible Internet Project.

IPSec : Internet Protocol Security.

IDS : Intrusion Detection System.

IPS : Intrusion Prevention System.

IRC : Internet Relay Chat.

L

LAN : Local Area Network.

M

MAC : Message Authentication Code.

MAC : Media Access Control.

MSS : Maximum Segment Size.

N

NAT : Network Address Translation.

NIDS : Network Intrusion Detection System.

NTP : Network Time Protocol.

O

OSI : Open Systems Interconnection.

P

PAN : Personal Area Network.

PSH : Push.

PCap : packet capture.

PIN : Personal Identification Number.

R

RSA : Au nom de Ronald Rivest, Adi Shamir et Leonard Adelman.

RTT : Round Trip Time.

RTD : Round Trip Delay.

S

SHA : Secure Hash Algorithm.

SSH : Secure Shell.

SSL : Secure Socket Layer.

SSDP : Simple Service Discovery Protocol.

SYN : Synchronisation.

T

TTL : Time-to-live.

TCP : Transmission Control Protocol.

TOR : The Onion Router.

TLS : Transport Layer Security.

U

UDP : User Datagram Protocol.

URI : Uniform Resource Identifier.

UPnP : Universal Plug and Play.

V

VPN : Virtual Private Network.

W

WAN : Wide Area Network.

Table des matières

CHAPITRE II : Anonymat et vie privée.

CHAPITRE IV : Détection du réseau Tor et I2P.

Liste des figures

CHAPITRE I : Généralités sur les réseaux et la sécurité informatique.

CHAPITRE II : Anonymat et vie privée.

CHAPITRE III : Extraction des signatures numérique du réseau TOR & I2P.

CHAPITRE IV : Détection du réseau Tor et I2P.

Liste des tableaux

CHAPITRE II : Anonymat et vie privée.

CHAPITRE III : Extraction des signatures numérique du réseau TOR & I2P.

CHAPITRE IV : Détection du réseau Tor et I2P.

Introduction générale

1. Motivation

Internet a été développé à évolué et également été utilisé par tant d'activités positives, par exemple, en innovant à notre façon dans le partage d'informations et l'éducation, elle nous aide dans le domaine de la santé et de la médecine. Mais d'un autre côté, Internet peut également être utilisé à mauvais escient pour mener des activités négatives ou illégales, par exemple, le piratage et les attaques illégales et bien d'autres.

Plusieurs gouvernements, organisations et autres institutions ont la capacité et les ressources de contrôler l'accès à Internet. Ce pouvoir peut être abusé pour filtrer le contenu sur Internet, fermer les serveurs, révéler l'identité des internautes. Il existe donc une demande d'accès anonyme à Internet pour échapper à la censure et à la surveillance et protéger la vie privée du gouvernement ou d'autres pouvoirs. L'anonymat permet d'utiliser Internet sans risque de représailles. Pour cette raison, la communication anonyme a attiré l'attention des chercheurs et des internautes. À mesure que les réseaux de communication anonymes se développent pour prendre en charge plus d'utilisateurs, de plus en plus d'outils d'anonymat deviennent disponibles gratuitement. Certains de ces outils incluent des serveurs proxy, des services de réseau privé virtuel (VPN), le routeur oignon (Tor) et le projet Internet invisible (I2P).

Tor et I2P sont les réseaux de communication anonyme à faible latence les plus populaires, qui utilisent la technique de routage à l'oignon pour protéger l'anonymat des utilisateurs via un réseau ouvert de routeurs d'oignon géré par des bénévoles. Bien que Tor et I2P était conçu pour répondre à des besoins liés à une utilisation bienveillante, ces derniers présentent des limites dont il faut être conscient pour ne pas négliger certains risques de sécurité. Le réseau Tor est souvent utilisé pour la distribution de marchandises et contenus illégaux, allons jusqu'à la contrefaçon de cartes de crédit. Alors qu'I2P pourrait être aussi utilisé par des cybercriminels ou encore des terroristes pour mettre en place des réseaux IRC anonymes ou pour développer des applications de transfert de fichiers.

L'utilisation du Tor/I2P au sein d'entreprise peut engendrer des risques de sécurités de réseau comme la divulgation de documents confidentiels et le risque d'infection par des malwares. Donc Il est recommandé aux entreprises et aux administrations de détecter voire bloquer les communications qui pourraient être établies vers des noeuds Tor/I2P, par

un système de détection d'intrusion en analysant le flux de données en temps réel en se basant sur les signatures d'une base de données.

2. Problématique et objectifs

Vu les risques de sécurités qui peuvent être provoqués par l'utilisation du réseau TOR et I2P dans une entreprise. La question fondamentale suivante s'impose :

La détection de l'utilisation du réseau Tor et du réseau I2P dans une entreprise, est-elle possible ? Et quelle est la difference entre le réseau Tor et le réseau I2P ?

De cette problématique s'enchainent les questions suivantes :

- Qu'est-ce que l'anonymat et la vie privée sur internet ?
- Quels sont les outils permettant l'anonymat ?
- Qu'est-ce qu'un réseau Tor et un réseau I2P et quel est leurs fonctionnement ?
- Y-a-t-il une difference entre le trafic Tor/I2P et le trafic du web ordinaire ?
- L'implémentation des signatures Tor et I2P dans un système de détection d'intrusion est-elle possible ?

3. Structure du mémoire

Pour mener à bien notre travail, nous avons structuré notre mémoire en quatre chapitres répartis comme suit :

- Dans le chapitre 1, nous donnons un aperçu général sur les réseaux et la sécurité informatique.
- Dans le chapitre 2, nous définissons l'anonymat et vie privée sur internet. Par la suite nous présentons les outils d'anonymat ainsi la comparaison théorique entre le réseau Tor et le réseau I2P.
- Dans le chapitre 3, nous Extrayons les signatures numériques du réseau TOR et I2P.
- Dans le chapitre 4, nous détectons l'utilisation du réseau Tor et I2P en implémentant les signatures de ces derniers dans un système de détection d'intrusion.

En conclusion générale, nous synthétisons les principales parties du mémoire, en faisant ressortir les apports de notre travail ainsi que les perspectives prévues pour l'améliorer.

CHAPITRE I

Généralités sur les réseaux et la sécurité informatique

I.1 Introduction

Un réseau informatique est un ensemble d'équipements informatiques qui sont interconnectés entre eux à travers des protocoles de communication normalisés. Il sert à l'échange de données numériques et le partage des ressources entre systèmes et applications informatiques.

En effet, nous allons aborder dans ce premier chapitre, quelques notions sur les réseaux informatiques en général en présentant les types de ces derniers, ensuite nous donnons un aperçu sur les différentes couches du modèle TCP, ainsi que les différentes attaques, le concept de la sécurité informatique et la sécurité dans les réseaux.

I.2 Réseaux informatiques

I.2.1 Définition d'un réseau informatique

Un réseau informatique est un ensemble de moyens matériels qui rassemblent les périphériques et les supports de transmission et logiciels qui représentent l'ensemble de séquences d'instructions interprétables par la machine, mis en œuvre pour assurer l'échange de données et le partage des services entre ordinateur, station de travail et terminaux informatiques.

I.2.2 Classification des réseaux

Les réseaux informatiques se classent selon les critères suivants :

I.2.2.A Topologie

- **Topologie physique :** décrit la manière avec laquelle les différents nœuds sont reliés entre eux. (Bus, Étoile, Anneau…etc.).
  - Topologie en bus : est l'organisation la plus simple d'un réseau. En effet, dans une topologie en bus tous les ordinateurs sont reliés à une même ligne de transmission par l'intermédiaire de câble, généralement coaxial. Le mot « bus » désigne la ligne physique qui relie les machines du réseau.
  - Topologie en étoile : les ordinateurs du réseau sont reliés à un système matériel central appelé concentrateur. Il s'agit d'une boîte comprenant un certain nombre de jonctions auxquelles il est possible de raccorder les câbles réseau en provenance des ordinateurs. Celui-ci a pour rôle d'assurer la communication entre les différentes jonctions.
  - Topologie en anneau : dans un réseau possédant une topologie en anneau, les ordinateurs sont situés sur une boucle et communiquent chacun à son tour.

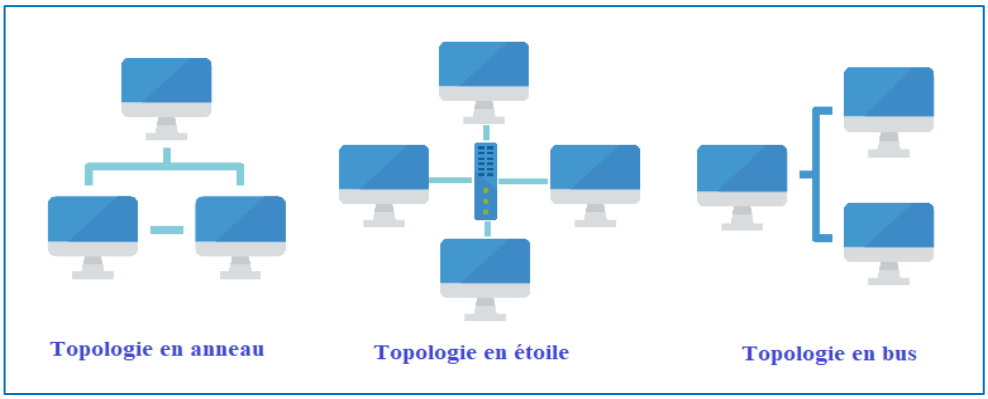

Figure I.1 : *Les topologies physiques.*

❖ **Topologie logique :** désigne la façon avec laquelle l'information est transmise d'un nœud à l'autre, les plus courantes sont Ethernet, Token Ring et FDDI.

- Ethernet est désormais la technologie du réseau local prédominante dans le monde, il fonctionne au niveau de la couche 1 et 2, les spécifications Ethernet prennent en charge différents support, bande passante, codage du signal et format de la trame.
- Anneau à jeton (*Token Ring*) : La méthode du passage du jeton est une méthode propre au réseau en anneau, c'est une technologie d'accès au réseau basé sur le principe de la communication au tour à tour.
- La technologie FDDI : (*fiber Distributed data interface*) est une technologie d'accès au réseau sur des lignes de type fibre optique. Il s'agit en fait d'une paire d'anneaux, l'un primaire et l'autre permettant de rattraper les erreurs du premier. FDDI est un anneau à jeton à détection et correction d'erreurs.

I.2.2.B Étendue géographique

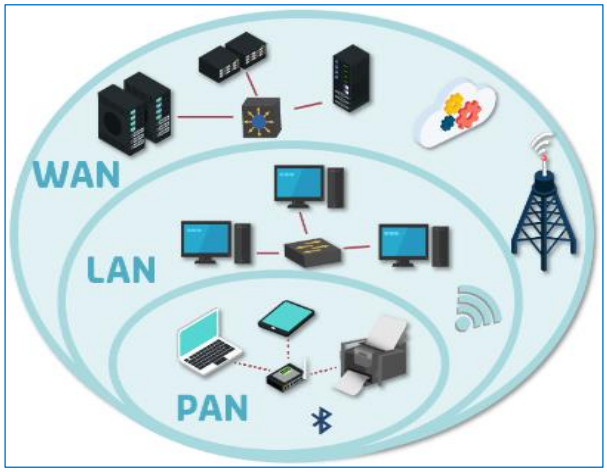

Figure I.2 : *Classification des réseaux selon l'étendue géographique [1].*

❖ **Réseau personnel PAN**

Un réseau PAN acronyme (*Personal Area Network)*, est un réseau domestique assurant l'interconnexion d'équipements ou périphériques (Laptop, Smartphone, PC, etc.), dans un espace d'une dizaine de mètres basés sur la technologie Bluetooth, Wi-Fi, etc… .

❖ **Réseau local LAN**

Un réseau local LAN acronyme (*Local Area Network*), est une infrastructure constituée d'équipements interconnectés entre eux, permettant l'échange et le partage des ressources communes limitées à une zone géographique restreinte. Il représente le cœur de la majeure partie de l'activité informatique dans une entreprise.

❖ **Réseau étendu WAN**

Un réseau étendu appelé WAN acronyme de (*Wide Area Network*), permet d'interconnecter des réseaux LANs sur des grandes distances géographiques (plus que 100 km), Les WANs fonctionnent grâce à des routeurs qui permettent de choisir le trajet le plus approprié pour atteindre un nœud du réseau.

I.2.2.C Architecture

On distingue généralement deux types de réseaux locaux :

❖ **Réseau poste à poste (*Peer to Peer*) :**

Les ordinateurs sont reliés par un support physique, chaque poste connecté peut mettre ses données et ses ressources à disposition du réseau, il peut être à la fois client et serveur ; il est plus adapté aux petites structures.

❖ **Réseau client / serveur :**

Dans une architecture client/serveur, parmi les machines du réseau, il y a une qui est considérée comme un serveur, elle est généralement très puissante et c'est elle qui délivre les informations (tels que la connexion) aux autres ordinateurs qui sont considérés comme des postes clients.

I.3 Equipements réseaux

L'infrastructure d'un réseau définit d'une part les équipements qui le composent et d'autre part, les connexions entre ces éléments. Elle établit donc les liaisons possibles entre les équipements et elle assure l'interconnexion des moyens physiques grâce à des protocoles de communication. Les principaux équipements d'une infrastructure sont :

I.3.1 Carte réseau (*Network Interface Card*)

La carte réseau est la composante la plus importante, elle est indispensable, c'est par elle que transitent les données envoyées et reçues dans un ordinateur du réseau. Chaque carte dispose d'une adresse MAC[1] : c'est l'adresse physique de la carte et permet d'identifier la machine dans un réseau [2].

I.3.2 Concentrateur ou hub

Il diffuse la trame reçue vers tous les équipements connectés. Cet équipement est un simple répéteur de données, il est utilisé pour créer un réseau local de type Ethernet [2].

I.3.3 Commutateur ou switch

Un commutateur réseau est un équipement qui relie plusieurs câbles ou fibres dans un réseau informatique ou un réseau de télécommunication, il fonctionne au niveau de la couche 2. Il transmet les bons paquets de manière sélective au port correct uniquement [2].

I.3.4 Routeur

Un routeur est un équipement assurant le routage des paquets, implémenté au niveau de la couche 3 du modèle OSI (couche réseau), permettant la configuration de l'adresse IP, du routage statique et dynamique, de la traduction d'adresses réseau (NAT[2]) statique et dynamique, et d'autres options [2].

I.4 Support de transmission

Il est nécessaire de relier les différentes unités de communication pour circuler les informations au sein d'un réseau à l'aide d'un support de transmission [2].

Un support de transmission est un canal physique qui permet de relier des périphériques, parmi ceux-ci on distingue :

- Câble à paire torsadée.
- Câble coaxial.
- Fibre optique.

I.5 Modèle TCP/IP

TCP/IP désigne communément une architecture réseau, cet acronyme désigne en fait deux protocoles étroitement liés : un protocole de transport, TCP (*Transmission Control Protocol*) qu'on utilise « par-dessus » un protocole réseau IP (*Internet Protocol*). Ce qu'on

[1] **MAC :** Media Access Control.
[2] **NAT:** Network Address Translation.

entend par « modèle TCP/IP », c'est en fait une architecture réseau en 4 couches contrairement au modèle OSI[3] qui est en 7 couches, dans cette architecture les protocoles TCP et IP jouent un rôle prédominant, car ils en constituent l'implémentation la plus courante. TCP/IP ne se préoccupe pas du contenu (les propos tenus par les utilisateurs dans les messages) ; il se contente d'assurer des fonctions qui facilitent les communications, le partage et la diffusion des informations [3].

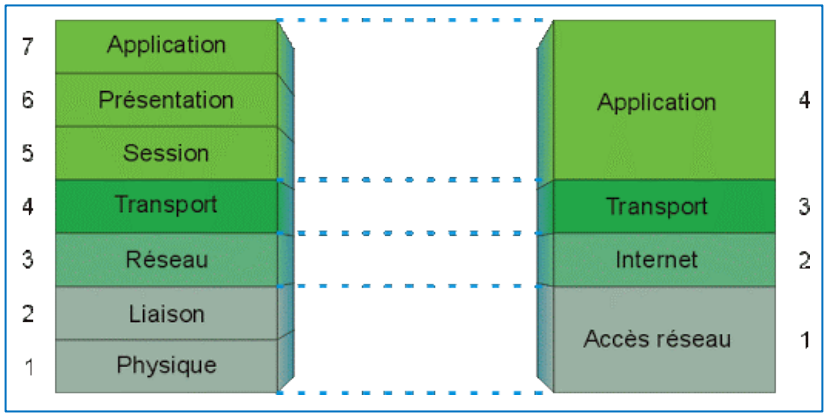

Figure I.3 : *Modèle OSI et TCP/IP.*

❖ **Couche Application**

- Elle est la couche de communication qui s'interface avec les utilisateurs.
- Exemples de protocoles applicatifs : HTTP[4], DNS[5], DHCP[6], FTP, …etc.
- Elle s'exécute sur les machines hôtes.

❖ **Couche Transport : TCP**

- Elle est responsable du dialogue entre les hôtes terminaux d'une communication.
- Les applications utiliseront TCP[7] pour un transport fiable et UDP[8] sans ce service.
- Les routeurs NAT et les pare-feu opèrent un filtrage au niveau de la couche transport.

❖ **Couche Internet : IP**

- Elle permet de déterminer les meilleurs chemins à travers les réseaux en fonction des adresses IPv4 ou IPv6.
- Les routeurs transfèrent le trafic IP qui ne leur est pas destiné.

❖ **Couche Accès au réseau : LAN/WAN**

- Elle organise le flux binaire et identifie physiquement les hôtes.
- Les commutateurs, cartes réseau, câbles, etc…. Font partie de cette couche.

[3] **OSI :** Open System Interconnection.
[4] **HTTP**: Hypertext Transfer Protocol.
[5] **DNS :** Domain Name System.
[6] **DHCP :** Dynamic Host Configuration Protocol.
[7] **TCP :** Transmission Control Protocol.
[8] **UDP :** User Datagram Protocol.

I.6 Communications et protocoles réseau

Un protocole de communication est un ensemble de règles et de procédures permettant de définir un type de communication particulier. Les protocoles sont hiérarchisés en couches, pour décomposer et ordonner les différentes tâches.

Le protocole définis les informations que la machine source doit fournir pour se faire comprendre par le réseau [4].

Pour correctement transmettre le message, les machines utilisent non pas un seul protocole mais une suite de protocole ajoutant chacun d'eux de nouvelles informations permettant de transmettre le message. On appelle cette organisation la suite de protocoles. Ainsi le message à envoyer est encapsulé par chacun des protocoles permettant sa transmission [4].

I.6.1 Encapsulation

Le message brut, même codé, ne peut être envoyé tel quel sur le réseau, il faut lui fournir d'autre informations comme le destinataire de la lettre ou l'expéditeur. Ces informations sont ajoutées au début ou à la fin du message à envoyer. L'encapsulation consiste alors a ajouté des informations au message brut en fonction des informations requises par le réseau. L'ensemble du message encapsulé est appelé « trame » et peut être envoyé sur le réseau. Il peut y avoir plusieurs phases d'encapsulation d'un message [4].

Pour transmettre du contenu d'un ordinateur à un autre, l'utilisateur va utiliser un programme qui construit un message enveloppé par un en-tête applicatif, HTTP par exemple. Le message subit une première encapsulation [4].

Le logiciel va utiliser un protocole de couche transport correspondant pour établir la communication avec l'hôte distant en ajoutant un en-tête TCP ou UDP [4].

Ensuite, l'ordinateur va ajouter un en-tête de couche Internet, IPv4 ou IPv6 qui servira à la livraison des informations auprès de l'hôte destinataire [4].

Enfin, ces informations seront encapsulées au niveau de la couche Accès qui s'occupera de livrer physiquement le message [4].

I.6.2 Ports

Lorsqu'un message est transmis à l'aide du protocole TCP ou UDP, les protocoles et services demandés sont identifiés par un numéro de port. Un port est un identifiant numérique, présent dans chaque segment, qui est utilisé pour conserver la trace de certaines conversations et de certains services de destination demandés [4].

Il existe différents types de numéros de port, comme :

- Ports réservés (numéros 0 à 1023) : Ces numéros sont réservés à des services et applications comme HTTP port 80 et DNS port 53.
- Ports dynamiques (numéros 1024 à 65535) : Également appelés ports éphémères, ces ports sont généralement affectés de façon dynamique à des applications clientes lorsqu'une connexion à un service est initiée par un client.

I.6.3 Protocoles

Les protocoles qui régissent les réseaux sont des protocoles définis, précis et acceptés par toutes les machines qui l'utilisent. Il existe de nombreuses suites des protocoles mais en voici quelques exemples

La suite de protocoles TCP/IP se compose aujourd'hui de beaucoup de protocoles donnés dans la figure suivante [4] :

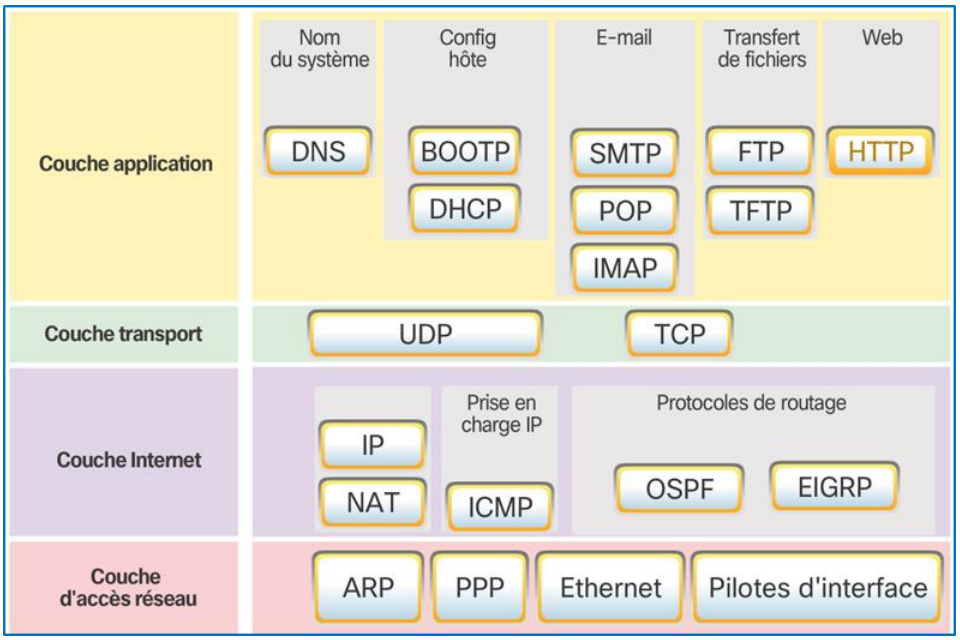

Figure I.4 : *Suite de protocoles TCP/IP [4].*

Les protocoles sont empilés en couches. Pour utiliser un protocole application, on a alors besoin d'utiliser un ou plusieurs protocoles par couche inferieur et d'effectuer l'encapsulation.

I.6.3.A Protocoles application

- **DNS :** Pour « *Domaine Name System* », permet de traduire le nom de domaine en adresses IP.

- **DHCP :** Pour « *Dynamic Host Configuration* », attribue dynamiquement des adresses IP aux clients et permet de réutiliser les adresses IP non utilisés.
- **FTP :** Pour « *File Transfert Protocol* », définis les règles qui permettent à l'utilisateur d'un hôte d'accéder à des fichiers sur un autre hôte du réseau et de transférer des fichiers vers cet hôte distant
- **HTTP :** Pour « *HyperText Transfer Protocol* », permet de transférer des médias, textes et graphiques sur le web.

I.6.3.B Protocoles de transport

- **UDP :** Pour « *User Datagram Protocol* », permet de transférer des paquets d'un hôte vers un autre sans accusé de réception.

Structure de l'entête UDP

Voici la structure de l'entête UDP basé sur 8 octets.

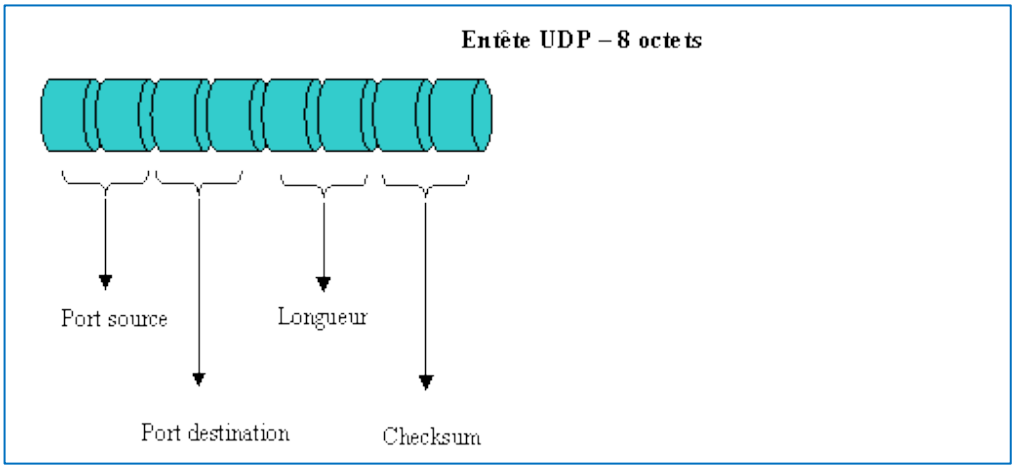

Figure I.5 : *Entête UDP [5].*

Définition des différents champs

- Port source UDP : Le champ Port source est codé sur 16 bits et correspond au port relatif à l'application en cours sur la machine source [5].
- Port destination UDP : Le champ Port destination est codé sur 16 bits et il correspond au port relatif à l'application en cours sur la machine de destination [5].
- Longueur : Le champ Longueur est codé sur 16 bits et il représente la taille de l'entête et les données. Son unité est l'octet et sa valeur maximale est 64 K octets (2^{16}) [5].
- Checksum : Le champ Checksum est codé sur 16 bits et représente la validité du paquet de la couche 4 [5].

❖ **TCP :** Pour « *Transmission Control Protocol* », permet une communication fiable entre les processus de deux hôtes distants avec accusé de réception.

Structure de l'entête TCP

Voici la structure de l'entête TCP basé sur 20 octets.

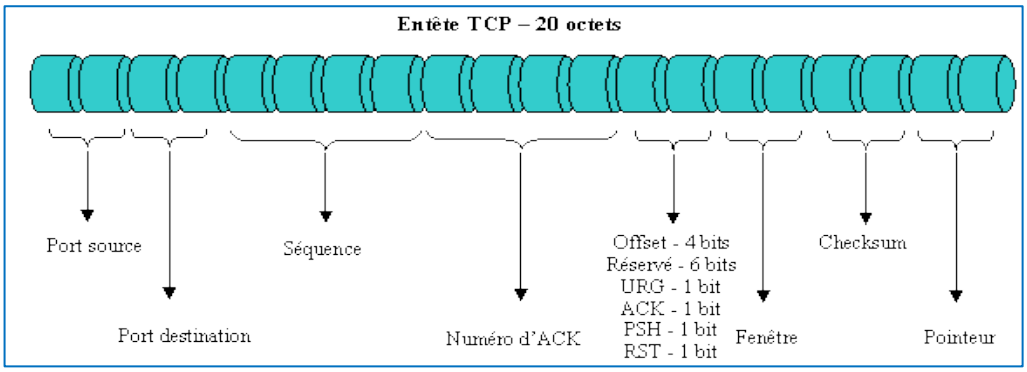

Figure I.6 : *Entête TCP [5].*

Définition des différents champs

- Port source TCP : Le champ Port source est codé sur 16 bits et correspond au port relatif à l'application en cours sur la machine source [5].
- Port destination TCP : Le champ Port destination est codé sur 16 bits et correspond au port relatif à l'application en cours sur la machine de destination [5].
- Numéro de séquence : Le champ Numéro de séquence est codé sur 32 bits et correspond au numéro du paquet. Cette valeur permet de situer à quel endroit du flux de données le paquet, qui est arrivé, doit se situer par rapport aux autres paquets [5].
- Numéro de l'accusé de réception : Le champ Numéro de séquence est codé sur 32 bits et définit un acquittement pour les paquets reçus. Cette valeur signale le prochain numéro de paquet attendu [5].
- Offset : Le champ Offset est codé sur 4 bits et définit le nombre de mots de 32 bits dans l'entête TCP. Ce champ indique donc où les données commencent [5].
- Réservé : Le champ Réservé est codé sur 6 bits et il servira pour des besoins futurs [5].
- Flags : Voici quelques flags [5] :
  - Le champ ACK est codé sur 1 bit et indique que le numéro de séquence pour les acquittements est valide.
  - Le champ PSH est codé sur 1 bit et indique au récepteur de délivrer les données à l'application et de ne pas attendre le remplissage des tampons.

- Le champ SYN est codé sur 1 bit et indique la synchronisation des numéros de séquence.
- Le champ FIN est codé sur 1 bit et indique fin de transmission.

- Fenêtre : Le champ Fenêtre « Windows » est codé sur 16 bits et correspond au nombre d'octets à partir de la position marquée dans l'accusé de réception que le récepteur est capable de recevoir [5].
- Checksum : Le champ Checksum est codé sur 16 bits et représente la validité du paquet de la couche 4 TCP [5].
- Pointeur de donnée urgente : Le champ Pointeur de donnée urgente est codé sur 16 bits et communique la position d'une donnée urgente en donnant son décalage par rapport au numéro de séquence [5].

I.6.3.C Protocoles Internet

- **ICMP :** Pour « *Internet Control Message Protocol* », permet de signaler à l'hôte distant les erreurs survenues lors de la transmission.
- **NAT :** Permet de convertir les adresses locales[9] en adresses globales[10] sur le réseau mondial
- **IP :** Pour « *Internet Protocol* », permet de regrouper les messages en paquets et indiquer l'adresse de destination.

Adressage IPv4 :

Une adresse IPv4 (IP version quatre) est un entier écrit sur quatre octets, elle peut donc prendre des valeurs entre 0 et 2^{32} - 1. Pour plus de commodité, on note les adresses en donnant les valeurs de chaque octet séparé par des points, par exemple :
11000000 10101000 00000001 00001101. S'écrit 192.168.1.13.

Une adresse IP est constituée de deux parties : l'adresse du réseau et l'adresse de la machine, elle permet donc de distinguer une machine sur un réseau. Deux machines se trouvant sur un même réseau possèdent la même adresse réseau mais pas la même adresse de machine.

[9] **Adresses locales** : appelées adresses privées sont utilisées par la plupart des entreprises dans leur réseau interne, ces adresses ne sont pas routables via internet et doit être traduites en adresses IPv4 publiques à l'aide de la traduction NAT.
[10]**Adresses globales** : appelées adresses publiques, sont routables via internet et sont acheminées de manière globale entre les routeurs des FAI (fournisseurs d'accès à Internet) et ils sont attribuées par ce dernier.

Entête de paquet IPv4

Un paquet IPv4 comporte deux parties :

- En-tête IP : indique les caractéristiques du paquet.
- Données utiles : contient les informations du segment de couche 4 et les données en elles-mêmes [5].

Comme le montre la figure, un en-tête de paquet IPv4 comporte des champs contenant des informations importantes sur le paquet. Ces champs contiennent des nombres binaires, examinés par le processus de couche 3. Les valeurs binaires de chaque champ indiquent divers paramètres du paquet IP [5].

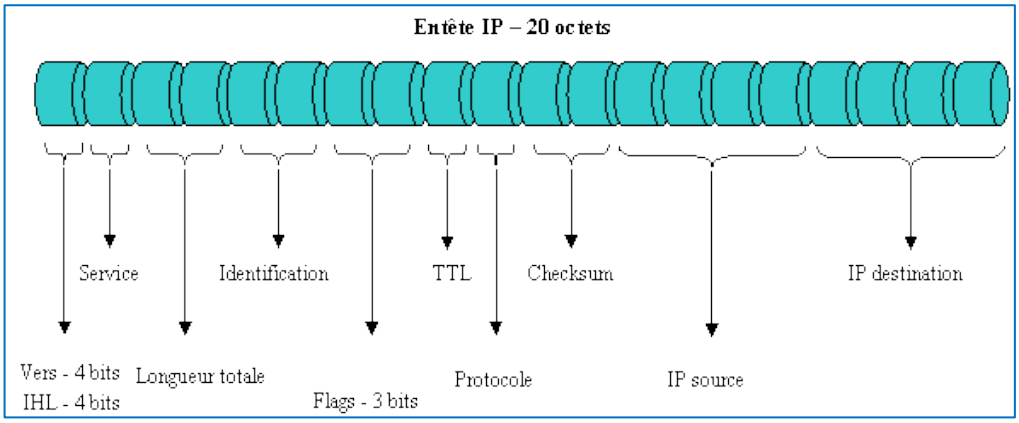

Figure I.7 : *Entête de paquet IPv4 [5].*

Les champs importants de l'en-tête IPv4 sont les suivants [5] :

- Version : contient une valeur binaire de 4 bits indiquant la version du paquet IP. Pour les paquets IPv4, ce champ est toujours 0100.
- Time-to-live (durée de vie, TTL) : contient une valeur binaire de 8 bits utilisée pour limiter la durée de vie d'un paquet. Cette durée est indiquée en secondes mais est généralement appelée « nombre de sauts ».
- Protocole : cette valeur binaire de 8 bits indique le type de données utiles transportées par le paquet, ce qui permet à la couche réseau de transmettre les données au protocole de couche supérieure approprié. Les valeurs habituelles sont notamment ICMP (1), TCP (6) et UDP (17).
- Adresse IP source : contient une valeur binaire de 32 bits qui représente l'adresse IP source du paquet.
- Adresse IP de destination : contient une valeur binaire de 32 bits qui représente l'adresse IP de destination du paquet.

Les champs restants sont utilisés pour identifier et valider le paquet, ou pour réassembler un paquet fragmenté.

I.6.3.D Protocoles d'accès réseau

- **ARP :** Pour (*Address Resolution Protocol*), Fournis un mappage dynamique entre une adresse IP et une adresse physique.
- **Ethernet :** Le Protocol le plus utilisé en local permettant de définir les règles de câblage et de signalisation.
- **Pilotes d'interface :** Donne les instructions à l'ordinateur pour communiquer avec ses interfaces réseau.

I.7 Principaux risques pour la sécurité du réseau

Les trois principaux risques pour la sécurité du réseau sont les failles, les menaces et les attaques [4] :

- Les failles correspondent au degré de vulnérabilité inhérent à tout réseau ou périphérique. Cela concerne les routeurs, les commutateurs, les ordinateurs de bureau, les serveurs et même les périphériques de sécurité.
- Les menaces viennent d'individus qui cherchent à exploiter les failles de sécurité et sont capables d'y parvenir. Il est prévisible que de tels individus continueront à rechercher de nouvelles faiblesses et de nouveaux exploits.
- Une attaque est l'exploitation d'une faille d'un système informatique (système d'exploitation, logiciel ou bien même de l'utilisateur) à des fins non connues par l'exploitant du système et généralement préjudiciables.

I.7.1 Types d'attaques

I.7.1.A Attaques de programmes malveillants

- **Virus :** Un virus est un logiciel malveillant intégré à un autre programme pour exécuter une fonction indésirable spécifique sur l'ordinateur de l'utilisateur [4].
- **Un cheval de Troie :** se distingue uniquement par le fait qu'il a été entièrement conçu pour ressembler à une application normale, alors qu'il s'agit d'un outil malveillant [4].
- **Les vers :** sont des programmes autonomes qui attaquent un système en tentant d'exploiter une faille spécifique. Lorsque l'exploitation de la vulnérabilité réussit, le

ver recopie son programme de l'hôte assaillant vers les systèmes nouvellement exploités et le cycle recommence [4].

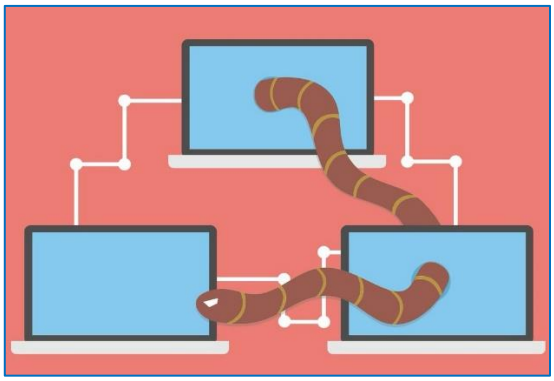

Figure I.8 : *Ver informatique.*

I.7.1.B Attaques de reconnaissance

Découverte et mappage non autorisés de systèmes, services ou vulnérabilités par exemple :

- **Analyseurs réseau (*sniffer)* :** c'est un dispositif permettant d'écouter le trafic sur un réseau c'est-à-dire d'y capturer les informations qui y circulent.

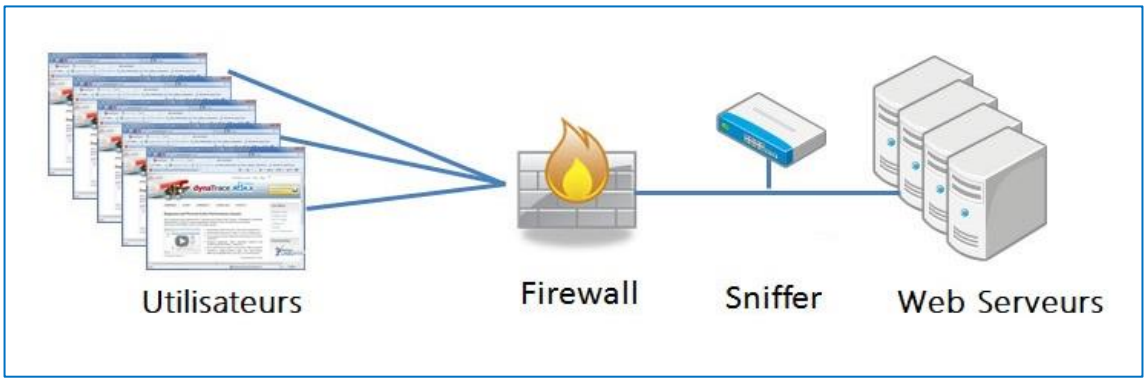

Figure I.9 : *Analyseur réseau (sniffer).*

- **Balayage de ports :** appelé aussi « scanner de vulnérabilité » est un utilitaire permettant de réaliser un audit de sécurité d'un réseau en effectuant un balayage des ports ouverts sur la machine ou le réseau tout entier.

I.7.1.C Attaques par accès

Ces attaques exploitent les vulnérabilités connues des services d'authentification, services FTP et services Web pour accéder à des comptes Web, des bases de données confidentielles et d'autres informations sensibles. Par exemple : attaques de mots de passe.

I.7.1.D Attaques DOS (déni de service)

Les attaques DOS peuvent prendre de nombreuses formes. Elles empêchent l'utilisation d'un service par les personnes autorisées en épuisant les ressources du système.

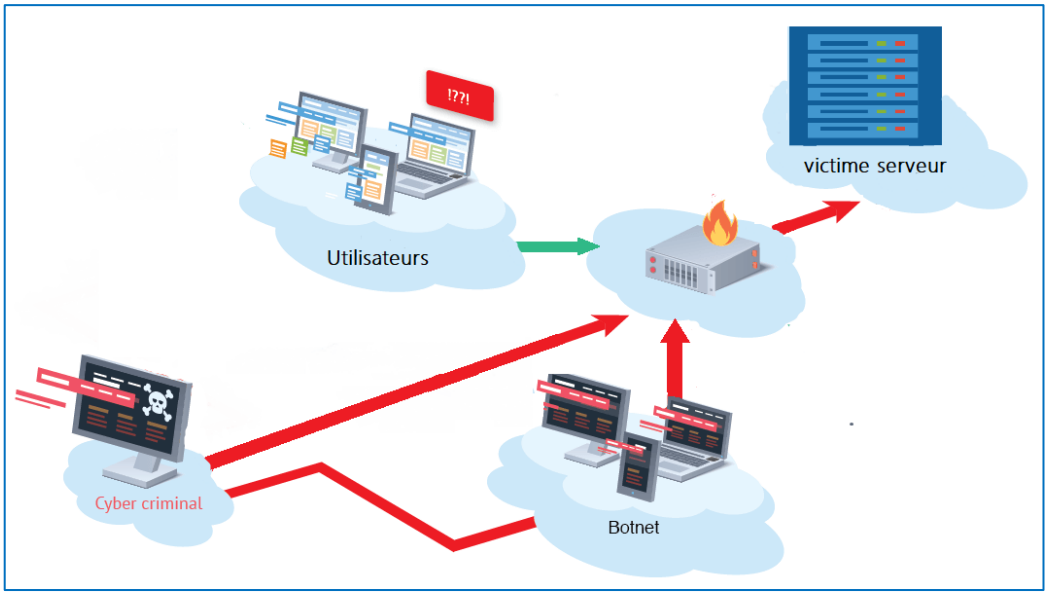

Figure I.10 : *Attaque Dos.*

- **Ping de la mort** : Le principe du Ping de la mort consiste tout simplement à créer un datagramme IP dont la taille totale excède la taille maximum autorisée (65536 octets). Un tel paquet envoyé à un système possédant une pile TCP/IP vulnérable, provoquera un plantage.
- **Attaque SYN** : L'attaque SYN (appelée également « *TCP/SYN Flooding* ») est une attaque réseau par saturation (déni de service) exploitant le mécanisme de poignée de main en trois temps (en anglais *Three-Way Handshake*) du protocole TCP.

I.8 Sécurité des réseaux informatique

Qu'ils soient filaires ou sans fil, les réseaux informatiques jouent un rôle essentiel dans la vie quotidienne. Les particuliers comme les entreprises sont dépendants de leurs ordinateurs et de leurs réseaux. Une intrusion par une personne non autorisée peut causer des pannes de réseau et des pertes de productivité coûteuses. Les attaques sur un réseau peuvent être dévastatrices et résulter en une perte de temps et d'argent, parce que des informations ou des ressources importantes sont endommagées ou volées.

I.8.1 Définition

La sécurité informatique est l'ensemble des moyens mis en œuvre pour réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles. Il convient

d'identifier les exigences fondamentales en sécurité informatique. Elles caractérisent ce à quoi s'attendent les utilisateurs de systèmes informatiques en regard de la sécurité [6].

I.8.2 Critères de la sécurité

Le système d'information est généralement défini par l'ensemble des données et des ressources matérielles et logicielles de l'entreprise permettant de les stocker ou de les faire Circuler, il représente un patrimoine essentiel de l'entreprise, qu'il convient de protéger [6].

La sécurité informatique vise généralement cinq principaux objectifs [6] :

- **L'intégrité** : c'est de garantir que les données sont bien celles que l'on croit être.
- **La confidentialité** : c'est d'assurer que seules les personnes autorisées aient accès aux ressources échangées
- **La disponibilité** : c'est de maintenir le bon fonctionnement du système d'information.
- **Le non répudiation** : c'est de garantir qu'une transaction ne peut être niée.
- **L'authentification** : c'est d'assurer que seules les personnes autorisées aient accès aux ressources.

I.8.3 Politique de sécurité

Une politique de sécurité ou stratégie de sécurité est une déclaration formelle des règles qui doivent être respectées par les personnes qui ont accès aux ressource et données de l'entreprise en vue de protéger son réseau contre les attaques menées soit de l'intérieur, soit de l'extérieur. Elle a pour objectif :

- Identifier les risques informatiques et leurs éventuelles conséquences.
- Elaborer des règles et des procédures à mettre en œuvre dans les services de l'organisation pour les risques identifiés.
- Surveiller et détecter les failles du système informatique.

I.8.4 Sécurisation de l'interconnexion des réseaux

Vu l'interconnexion des réseaux d'entreprise avec l'Internet, ou tout autre réseau, il est donc nécessaire de protéger les entrées et sorties sur le réseau. Différents équipements peuvent être mis en place pour protéger le réseau [7] :

I.8.4.A Programme antivirus :

Les logiciels antivirus sont des programmes informatiques qui détectent, empêchent et prennent des mesures pour désarmer ou supprimer des programmes informatiques malveillants, tels que des virus et des vers. Leur mode de fonctionnement est

basé sur une veille permanente. Pour empêcher les virus les plus courants, un logiciel antivirus doit être mis à jour régulièrement [8].

I.8.4.B Pare-feu :

Structure (logicielle ou matérielle) située entre l'utilisateur et le monde extérieur afin de protéger les données d'un réseau interne des intrus [9].

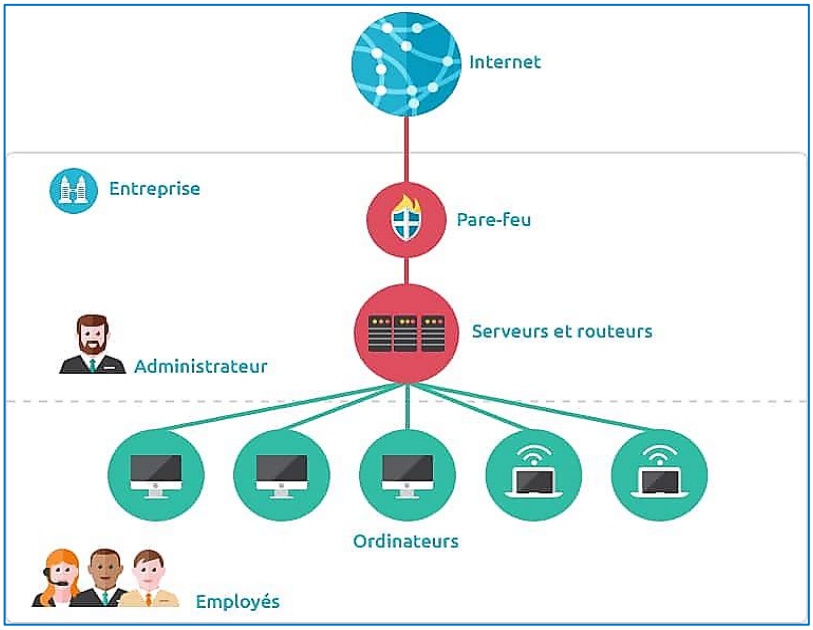

Figure I.11 : *Pare-feu.*

Rôles d'un pare-feu [10] :

- Déterminer le type de trafic qui sera acheminé ou bloqué.
- Limiter le trafic réseau et accroitre les performances.
- Contrôler le flux de trafic.
- Fournir un niveau de sécurité d'accès réseau de base.
- Autoriser un administrateur à contrôler les zones auxquelles un client peut accéder sur un réseau.

I.8.4.C Proxy :

a. Définition du serveur proxy

Le terme proxy se traduit littéralement par le mot procuration mais on lui préfère celui de mandat. Un serveur proxy se définit donc comme un serveur mandataire réalisant à votre place des requêtes réseaux protocolaires comme par exemple HTTP ou encore FTP. On dégage trois grandes fonctionnalités d'un serveur proxy [11] :

- Le partage de connexion Internet

- La mise en cache des éléments (images, pages HTML, sons…).
- Le filtrage des données.

Si la première fonctionnalité peut être réalisée autrement, la deuxième accélère les réponses pour les navigateurs clients et la dernière offre la possibilité de sélectionner les sites autorisés et ceux qui ne le sont pas [11].

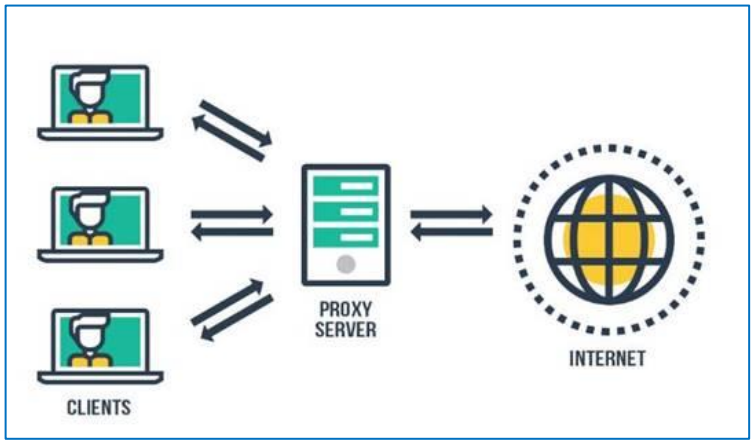

Figure I.12 : *Serveur proxy.*

Dernière chose, un proxy agit selon deux modes différents, serveur ou transparent [11] :

- En mode serveur, une modification se fera dans les paramètres de connexion du navigateur des postes clients afin d'indiquer l'adresse du serveur et le port sur lequel il doit s'y connecter.
- En mode transparent, aucune modification n'est nécessaire sur le poste client mais il ne peut plus y avoir alors d'authentification utilisateur.

b. Serveur de proxy dans une entreprise

En entreprise, un proxy peut servir de protection (vous pouvez vous connecter sur internet mais les ordinateurs d'internet ne peuvent pas accéder au votre). Un proxy peut aussi servir à masquer les informations contenues dans votre ordinateur (masquer adresse IP, masquer le système d'exploitation utilisé…) et aussi à mémoriser les pages les plus sollicitées par votre ordinateur. En entreprise, on peut dépeindre un proxy comme étant un « espion » puisqu'il enregistre toutes les requêtes demandées (vous savez qui a cherché quoi, quand et quel contenu a été visualisé) [12].

I.8.4.D Zone démilitarisée :

Une DMZ (*Demilitarized zone*) est une zone tampon d'un réseau d'entreprise, située entre le réseau local et Internet, généralement derrière le pare-feu. Il s'agit d'un réseau intermédiaire regroupant des serveurs publics (HTTP, DHCP, mails, DNS, etc.). Ces

serveurs devront être accessibles depuis le réseau interne de l'entreprise et, pour certains, depuis les réseaux externes. Le but est ainsi d'éviter toute connexion directe au réseau interne [12].

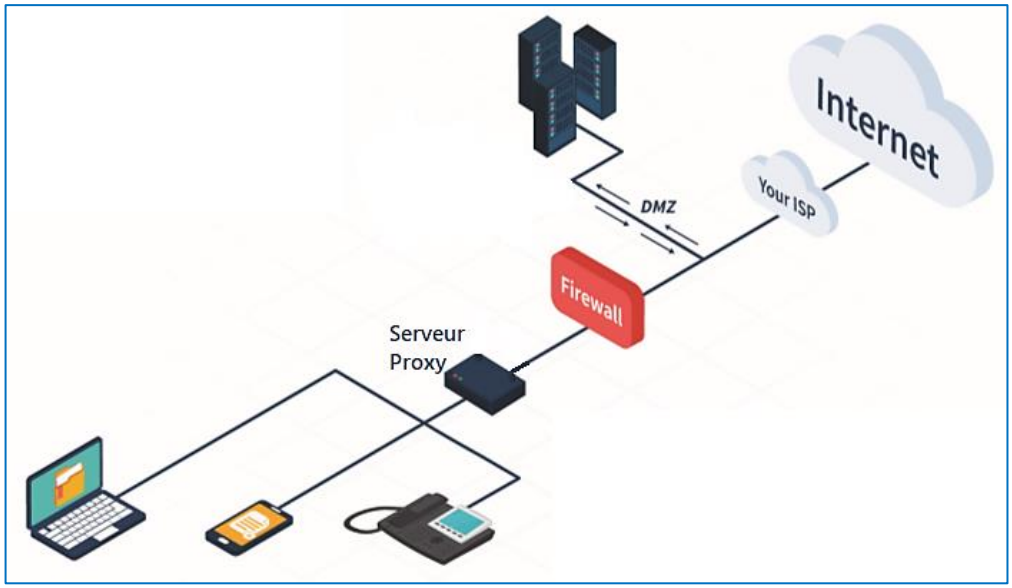

Figure I.13 : *Zone démilitarisée.*

I.8.4.E Système de détection d'intrusions (IDS) :

A l'origine, les premiers systèmes de détection d'intrusions ont été initiés par l'armée américaine, puis par des entreprises. C'est un ensemble de composants logiciels et matériels dont le rôle serait de surveiller les données qui transitent sur ce système, et qui serait capable de réagir si des données semblent suspectes [13].

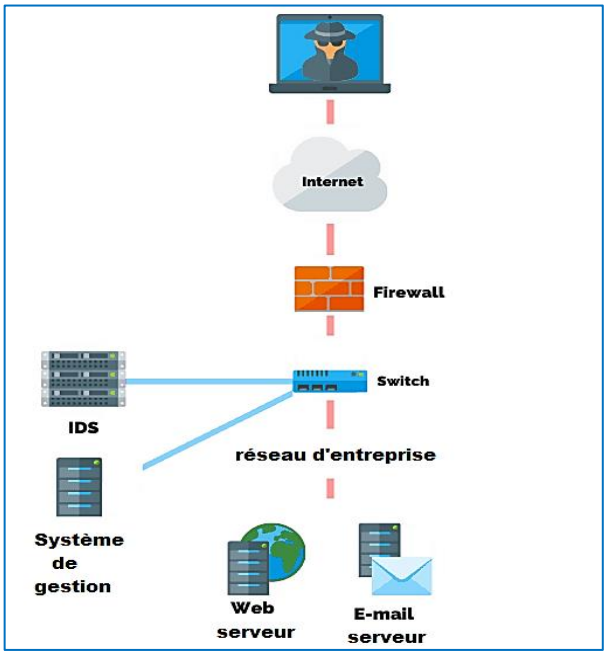

Figure I.14 : *Système de détection d'intrusions (IDS).*

Nous pouvons distinguer trois grandes familles d'IDS :

- **Les systèmes de détection d'intrusions réseaux (NIDS) :**

Il a pour objectif d'analyser de manière passive les flux en transit sur le réseau et détecter les intrusions en temps réel. Un NIDS[11] écoute donc tout le trafic réseau, puis l'analyse et génère des alertes si des paquets semblent dangereux.

- **Les systèmes de détection d'intrusions de type hôte (HIDS) :**

Un HIDS[12] se base sur une unique machine, il analyse l'activité qui se passe sur cette machine. Il analyse en temps réel les flux relatifs à une machine ainsi que les journaux.

- **Les systèmes de détection d'intrusions hybrides :**

Généralement utilisés dans un environnement décentralisé, ils permettent de réunir les informations de diverses sondes placées sur le réseau. Leur appellation " hybride " provient du fait qu'ils sont capables de réunir aussi bien des informations provenant d'un système HIDS qu'un NIDS.

I.9 Conclusion

Au cours de ce chapitre, nous avons parcouru des généralités sur les réseaux informatiques, ainsi que leurs différents équipements de base.

Ce chapitre nous a permis de comprendre le concept d'attaques informatiques puis la sécurité informatique et plus particulièrement la sécurité des réseaux, où nous avons présenté ses objectifs, et les différentes solutions qui permettent de garantir la sécurité,

Le prochain chapitre sera consacré à la présentation de la notion de la vie privée sur internet, ainsi que les outils qui permettent de garantir l'anonymat des échanges des données sur internet.

[11] **NIDS :** Network Intrusion Detection System.
[12] **HIDS :** Host Intrusion Detection System.

CHAPITRE II

Anonymat et vie privée

II.1 Introduction

Internet est le réseau informatique mondial qui rend accessible au public des services divers et variés comme le courrier électronique et le World Wide Web (plus couramment appelé Web).

Chaque fois qu'un internaute navigue sur les web il laisse derrière lui une multitude de trace permettant aux autres de recueillir des informations sur lui. Ce qu'on appelle les empreintes numériques, Ainsi, les risques associés aux traces numériques sont relatifs à l'utilisation des données personnelles des personnes à leur insu, et à l'atteinte à leur vie privée.

Les entités qui absorbent les données personnelles des internautes sont nombreuses et sont partout, parfois même cachées, c'est ici qu'entre le terme anonymat, ce droit à nécessairement besoin d'être appliqué sur internet, L'anonymat permet à des individus de s'exprimer sans crainte de représailles, et il est particulièrement important dans les pays où la liberté d'expression est lourdement censurée. L'anonymat permet de surfer sur le web sans conséquences et éviter toutes sortes de plans publicitaire sur le web.

Cependant, si les personnes restent anonymes, par définition, elles ne peuvent pas être identifiées, ce qui rend impossible leur responsabilisation. Les tenants des communications anonymes sur Internet ouvrent ainsi la porte à de nombreuses formes de comportements criminels et antisociaux, comme piratage informatique le harcèlement et le vol d'identité.

Il existe de nombreux outils qui garantissent la navigation anonyme sur le web qui préservent en même temps la sécurité et la fiabilité, qui sont : le réseau privé virtuel (VPN), Proxy, Projet Tor, Freenet et le réseau I2P. Nous avons opté d'utilisé le navigateur Tor et le réseau I2P pour notre projet.

II.2 Couches du Web

Les sites Web que nous consultons chaque jour ne représentent qu'un faible pourcentage d'Internet. Ces sites, sont appelés Web de surface, Au-delà du Web de surface, 96% du contenu en ligne se trouve dans le Web profond et le Darknet. On constate que le Web peut être divisé en trois couches [13] :

- **Le web surfacique** (ou indexable) : est la partie accessible en ligne, celle connue par le public. Elle contient toutes les ressources indexées par les moteurs de recherche classiques comme Google ou Bing.

- **Le web profond** (ou DeepWeb) : est la partie qui n'est pas indexée par les moteurs de recherches classiques. Elle représenterait plus de 90% du contenu d'internet. Elle est principalement constituée de données gouvernementales, médicales, financières, scientifiques, etc.
- **Le darknet** : est une partie du web profond dont les données sont volontairement cachées. Y accéder demande l'utilisation d'outils spéciaux assurant l'anonymat de leurs utilisateurs comme Projet Tor, I2P et Freenet.

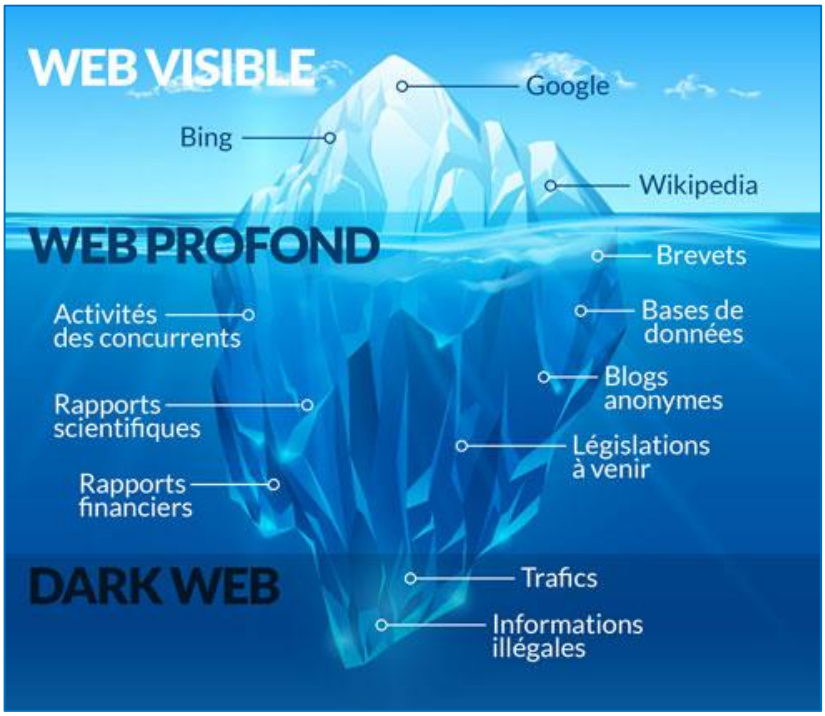

Figure II.1 : *Couches du Web [13].*

II.3 L'anonymat

II.3.1 L'anonymat sur internet

L'anonymat signifie globalement l'état de quelqu'un qui choisit de rester inconnu inaccessible, impossible à suivre. Être anonyme sur internet ne se résout pas à se cacher simplement derrière un pseudonyme, la tâche est bien plus complexe, l'anonymat réel requiert une non-traçabilité et le droit d'un individu de contrôler la collection des informations le concernant. Et évidemment la nature des réseaux informatiques rend cela difficile. C'est pourquoi de nombreux internautes ont choisis d'autres solutions techniques pour contourner cette problématique, ces derniers sont appelé « réseau anonyme » qui sont spécifiquement conçu pour assurer l'anonymat de leurs utilisateurs. Parmi ceux les plus

connus VPN (*Virtual Private Network*), TOR (*The Onion Router*), FreeNet et I2P (*Invisible Internet Project*).

Le côté sombre de l'anonymat et qu'elle pourrait être utilisée dans des activités criminelles ou d'autres types de téléchargement illégal, le harcèlement ou l'intimidation en ligne.

II.4 Systèmes de cryptage dans l'anonymat

II.4.1 Cryptage des données

Le cryptage des données consiste à convertir les données d'une information lisible (texte en clair ou texte ordinaire) en des chaînes inintelligibles (texte chiffré) au moyen d'une valeur dite clé de cryptage. Un second aspect important du cryptage : le protocole de sécurité réside dans la suite des étapes suivies pour échanger sans risque des clés et du texte crypté dans des messages.

II.4.2 Protocoles de sécurités

Un protocole est un ensemble de règles et de procédures à respecter pour émettre et recevoir des données sur un réseau. Tel que la plupart de ces protocoles ne sont pas sécurisés lors de la transmission des données sur le réseau. Les protocoles sécurisés ont été mis au point, afin d'encapsuler les messages dans des paquets de données chiffrées. On cite parmi ces protocoles les suivants [14] :

- **Protocole SSH (*Secure Shell*) :** c'est un protocole qui permet à des services TCP/IP d'accéder à une machine à travers une communication chiffrée appelée « tunnel ».
- **Protocole SSL (*Secure Socket Layer*) :** c'est un procédé de sécurisation des échanges, il a été conçu pour assurer la sécurité des transactions effectuées via Internet.
- **Protocole TLS (*Transport Layer Security*) :** c'est la version améliorée du protocole SSL.
- **IPSec (IP Security) :** IPSec (*Internet Protocol Security*) est conçu pour sécuriser le protocole IPv6. La lenteur de déploiement de ce dernier a imposé une adaptation d'IPSec à l'actuel protocole IPv4. On établit un tunnel entre deux sites et IPSec gère l'ensemble des paramètres de sécurité associés à la communication.

II.4.3 Chiffrement de données

Le chiffrement est un terme technique qui désigne la méthode par laquelle les communications (SMS, courriels, appels téléphoniques et vidéo) sont sécurisées afin

d'empêcher toute personne autre que le destinataire prévu d'y accéder. Le chiffrement est la manipulation mathématique des informations dans le but de les rendre lisibles uniquement par le ou les destinataires prévus [15].

Nous utilisons quotidiennement l'une des trois méthodes de chiffrement dès que nous utilisons des services connectés :

- **Le chiffrement complet des données d'un disque dur ou d'un appareil** est le procédé par lequel toutes les données stockées sur un ordinateur ou un smartphone sont chiffrées (PIN[1]) lorsqu'elles se trouvent sur l'appareil [15].

- **Le chiffrement de bout en bout** garantit que les communications transmises entre l'expéditeur et le destinataire ne peuvent pas être déchiffrées ou lues par une tierce personne ou par un fournisseur de services [15].

  - **Chiffrement symétrique (chiffrement à clé secrète) :** Le chiffrement symétrique est la première forme de chiffrement à être apparu. Elle consiste à chiffrer un message à l'aide d'une clé. Pour déchiffrer ce message, il faut utiliser la même clé. Les plus utilisés et les plus sûrs du monde est l'AES[2] [16].
  - **Chiffrement asymétrique (chiffrement à clé publique) :** Son principe repose sur l'utilisation de deux clés. L'une est publique et peut-être distribuée à tous et l'autre est privée et ne doit pas être divulguée. En chiffrant un message avec l'une de ces clés, on peut le déchiffrer avec l'autre et l'RSA[3] est l'un des algorithmes de chiffrement asymétrique [16].

  L'échange d'un message entre deux entités s'effectue en trois étapes :

  1. Le nœud A envoie sa clé publique au nœud B.
  2. B chiffre son message avec cette clé.
  3. B envoie ensuite le message chiffré à A qui peut le déchiffrer avec sa clé privée qu'il a conservée.

[1] **PIN :** Personal Identification Number
[2] **AES** : Advanced Encryption Standard.
[3] **RSA** : Au nom de Ronald Rivest, Adi Shamir et Leonard Adleman.

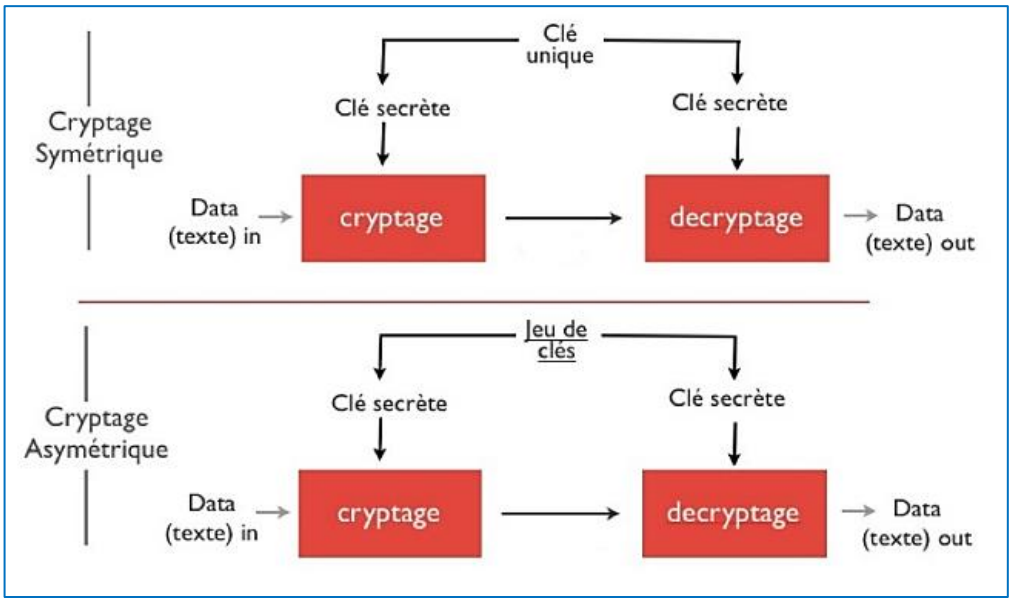

Figure II.2 : *Cryptage symétrique et asymétrique [17].*

- **Le chiffrement du transport ou chiffrement de la couche transport (dont l'implémentation la plus courante est le HTTPS avec le protocole TLS)** est le moyen par lequel les communications entre les sites Internet que vous consultez (par exemple le moteur de recherche Google, les boutiques en ligne) et votre navigateur sont chiffrées. Ainsi, lorsque vous saisissez un nom d'utilisateur et un mot de passe sur une page de connexion. Les sites Internet qui utilisent le HTTPS[4] garantissent une meilleure protection [15].

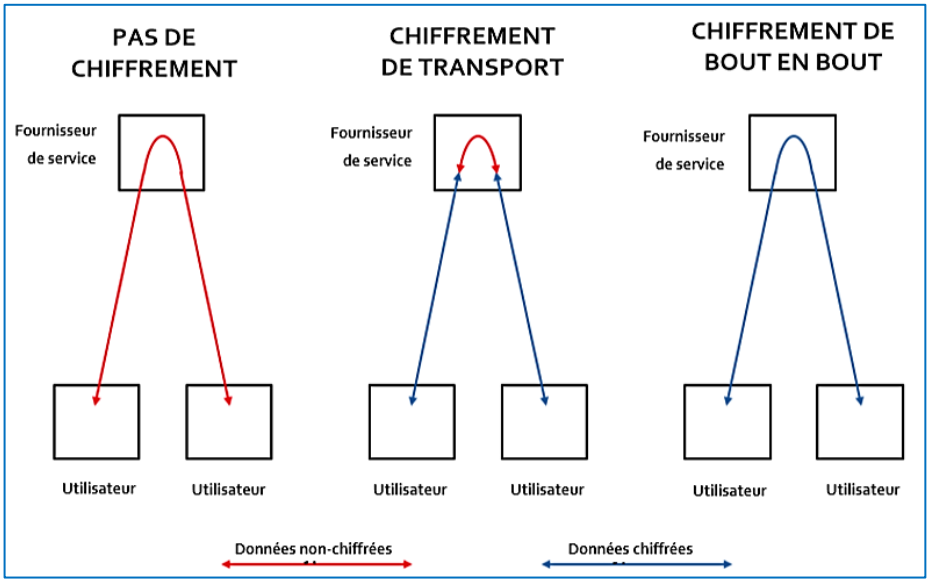

Figure II.3 : *Méthodes de chiffrement [15].*

[4] **HTTPS :** Hypertext Transfer Protocol Secure.

II.5 Outils d'anonymat

Il existe de multiples solutions pour une navigation sécurisée et anonyme. C'est à l'utilisateur individuel de déterminer la configuration là mieux adaptée à ses habitudes de navigation.

II.5.1 Navigation privée

La navigation privée empêche simplement que votre poste conserve les traces de vos activités en ligne. Elle supprime automatiquement les fichiers temporaires, les downloades, les mots de passe et l'historique des plateformes visitées. Tous les navigateurs proposent un mode de surf privé. Leur fonctionnement varie souvent d'une structure à une autre [18].

Figure II.4 : *Navigateur privé.*

II.5.2 VPN

Le VPN « *Virtual Private Network* » représente aujourd'hui le dispositif de référence pour contrer les principales menaces liées à la cybercriminalité. Sa mise en place vous permet de naviguer en toute confidentialité. Vous surfez via un serveur virtuel qui dissimule votre adresse IP [18].

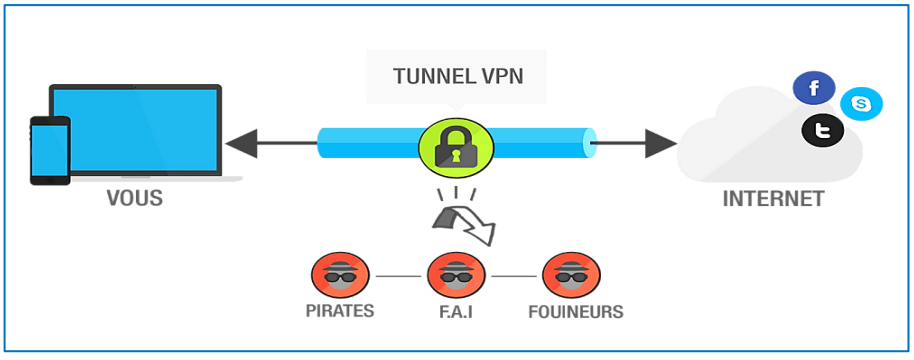

Figure II.5 : *Réseau privé virtuel.*

II.5.3 TOR

« *The Onion Router* », dont TOR est l'acronyme, est un peu comme une connexion proxy survitaminée. Il fonctionne sur la base d'une interconnexion entre votre TOR et celui d'autres navigateurs TOR à travers le monde entier. Concrètement, il transforme votre ordinateur en un serveur proxy pour des utilisateurs distants. Il opère donc de multiples liaisons et change votre identité toutes les fois que vous accédez à Internet via un nouvel onglet [18].

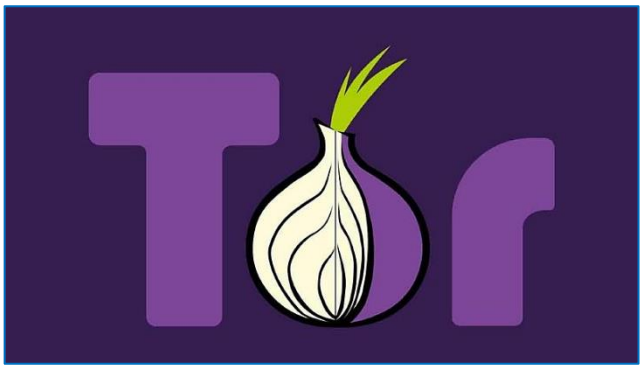

Figure II.6 : *Logo du réseau Tor.*

II.5.4 I2P

I2P signifie « *Le projet Internet invisible* » dont l'objectif principal est l'anonymat, c'est un réseau isolé des autres réseaux et agit en tant que réseau superposé aux infrastructures internet existantes. Ce réseau anonyme peut donc être utilisé pour créer des services web anonymes : blog, forum, courriel, SSH, proxys sortant…etc.

Dans I2P, les utilisateurs peuvent contrôler le niveau de sécurité, l'anonymat, la bande passante pour répondre à leurs besoins spécifiques. Ce qui garantit l'anonymat avec I2P est le fait que l'expéditeur et le destinataire ne communiquent jamais directement, mais via plusieurs routeurs nommé tunnels [19].

II.5.5 FreeNet

Freenet est un logiciel libre qui permet d'accéder au réseau du même nom qui est un réseau autonome anonyme distribué a pour objectif d'assurer la sécurité et l'anonymat à chacun, Freenet est décentralisé afin de la rendre moins vulnérable aux attaques et permet d'utiliser de façon anonyme les différents services proposés au sein de son propre réseau, il n'est donc pas possible de se connecter à des services comme Facebook ou Google avec Freenet [19].

II.6 Choix du logiciel

Les réseaux Tor, I2P et FreeNet sont conçus pour l'anonymat, les adresses IP des participants ne sont jamais révélées.

Pour notre projet nous avons choisi de faire la comparaison entre Tor et I2P, pour les raisons suivantes :

- TOR et I2P sont conçus pour permettre la mise en relation de deux ordinateurs. Ce sont des outils de mise en relation (connexion) anonymes. Freenet est conçu pour stocker et distribuer des documents. Il ne peut donc servir que du contenu statique, pas de pages dynamiques.
- TOR et I2P peuvent anonymiser des logiciels existants (email, chat, FTP[5], ssh...). Pour Freenet, les logiciels doivent être spécialement développés ou modifiés pour fonctionner avec Freenet.

En matière de navigation anonyme, Tor (Acronyme de « *The Onion Router* ») est devenu de par sa vocation grande publique un acteur dominant de ce domaine. Tor dispose également d'une interface facile à prendre en main qui permet à toute personne disposant d'une connexion Internet d'anonymiser son trafic tout en téléchargeant un navigateur Web. Cette combinaison entre convivialité et anonymisation a contribué à la popularité de Tor parmi les internautes, en faisant de celui-ci un outil utile pour passer sous les radars sur le Web, ou pour accéder à des contenus, des services ou des marchandises illégales.

Alors qu'I2P semble être un outsider dans cette « bataille pour la vie privée », Il ne vise pas à rendre anonymes les communications internet classiques comme le fait Tor, il se distingue par le fait est qu'il n'a pas été spécifiquement conçu pour exécuter des proxys sur Internet, mais qu'il a plutôt été développé comme un réseau interne. Cependant, des clients d'outproxies peuvent être utilisés pour permettre une forme de navigation incognito sur Internet. Il permet aux utilisateurs de créer ou d'accéder à du contenu, mais aussi de bâtir des communautés en ligne sur le réseau. I2P fonctionne avec des capacités équivalentes à Internet. Cependant, sa conception et sa décentralisation[6] créent un environnement qui résiste à la censure et favorise la libre circulation de l'information. Des sites miroir hébergés sur le réseau permettent d'accéder à des sources de nouvelles et à d'autres ressources dans les régions où l'information est filtrée ou bloquée.

[5] **FTP** : File Transfer Protocol.
[6] **Un réseau décentralisé :** ne dépend pas d'un pôle unique de décision, mais chaque membre du réseau n'est pas nécessairement autonome et peut dépendre de la disponibilité d'un serveur qui le relie au reste du réseau, tel le courrier électronique ou les réseaux de chat.

II.7 Explication du projet

Notre projet est intitulé « le réseau anonyme Tor VS I2P " comparaison et détection " ». Il a pour objectif de présenter le réseau Tor et I2P d'un point de vue technique et fonctionnel ainsi de comparer ces deux homologues ensuite détecter leurs utilisations. Pour cela nous allons suivre les étapes suivantes :

La première étape consiste à effectuer une présentation sur les deux réseaux d'anonymat, on se focalisant sur leurs fonctionnements et le chiffrement utilisé, ensuite on va aborder la comparaison entre TOR et I2P en termes de terminologie, d'usages et de techniques utilisés.

La deuxième étape se base sur l'architecture client-serveur où nous utilisons un PC serveur et un PC client. Nous installerons le logiciel CCProxy pour bloquer certains sites dont le but est d'imiter le réseau d'entreprise. Après on va installer le navigateur Tor pour contourner le proxy et naviguer librement sur internet, puis installer I2P et y'accéder à ses différents services.

La troisième étape se fonde sur l'utilisation de l'analyseur de paquets qui nous permettra d'extraire la différence entre Tor et I2P en termes de rapidité, latence, bande passante et signatures numériques.

Finalement ces signatures numériques nous amènent a créé des règles dans un système de détection d'intrusion « IDS » afin de pouvoir détecter l'utilisation du Tor et I2P.

II.8 Projet de l'Internet invisible (I2P)

II.8.1 Définition

I2P Invisible Internet Project (I2P) est un réseau anonyme accessible via les navigateurs Web habituels. Le réseau intègre sa propre reconfiguration dynamique en réponse aux diverses attaques, et a été conçu pour utiliser de nouvelles ressources au fur et à mesure de leur disponibilité [20].

Contrairement à de nombreux autres réseaux anonymes, I2P ne tente pas de procurer l'anonymat en masquant l'initiateur d'une communication et pas le destinataire, ou le contraire. I2P est conçu pour permettre aux pairs l'utilisant de communiquer anonymement (à la fois l'émetteur et le récepteur sont non-identifiables à l'un par l'autre comme par un tiers). La possibilité d'avoir des serveurs dans I2P est essentielle, car il est plus que certain que n'importe quel proxy sortant vers l'Internet sera surveillé, désactivé, ou même piraté pour tenter des attaques encore plus pernicieuses [20].

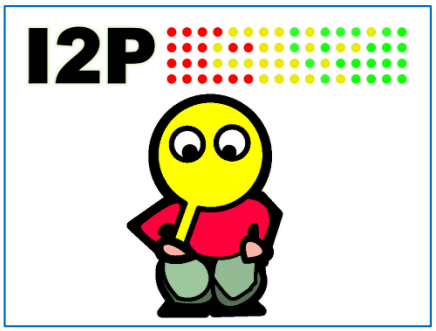

Figure II.7 : *Logo I2P.*

II.8.2 Architecture du réseau

Chaque nœud, ou utilisateur I2P, au sein du système déploie un Routeur I2P, formant ainsi la superposition du réseau I2P. Les routeurs I2P se connectent entre eux en formant des tunnels : un chemin multi-saut entre différents routeurs I2P. Le routeur I2P A envoie des messages via un tunnel à un saut utilisant Routeur I2P B (point final) et reçoit des messages utilisant également un tunnel à un saut utilisant le routeur I2P F (Passerelle). Le routeur I2P D utilise également des tunnels à un saut, où il envoie des données via le routeur I2P E (point final), et reçoit des données via le routeur I2P C (Passerelle). Le nombre de sauts dans un tunnel I2P varie entre 1 et 7, plus il y a de sauts dans un tunnel plus l'anonymat augmente, mais réduit les performances, étant donné que les données doivent traverser plus de nœuds intermédiaires [20].

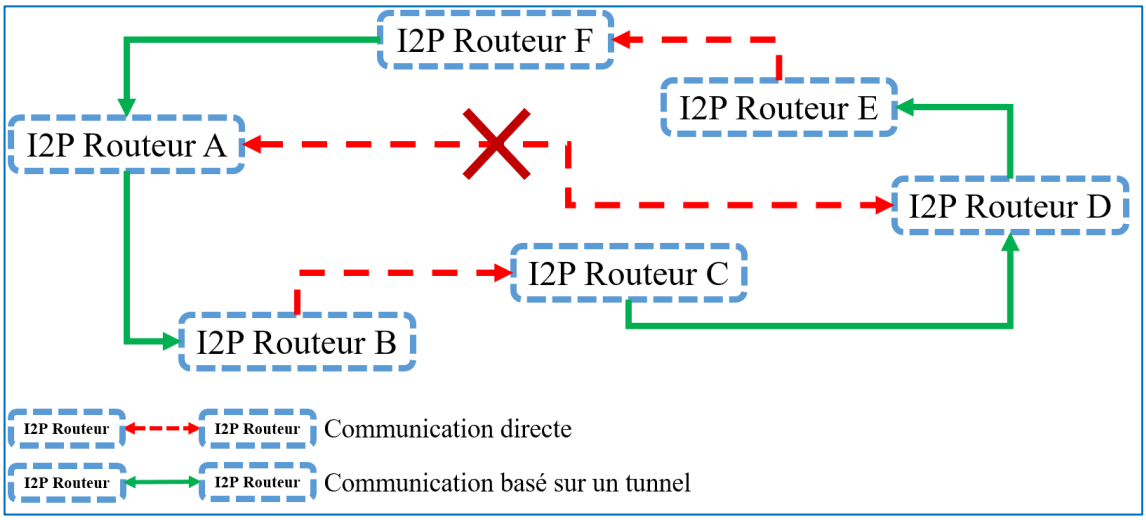

Figure II.8 : *Architecture du réseau I2P [20].*

II.8.3 Fonctionnement du réseau

Le réseau I2P repose sur trois composants clés : les routeurs (ou nœuds), les tunnels et la base de données réseau NetDB [21] :

- **Les routeurs** : sont les utilisateurs du logiciel I2P. Tous les utilisateurs font transiter des communications au travers de leur machine.
- **Les tunnels** : sont des chemins unidirectionnels constitués de plusieurs routeurs. Chaque routeur peut faire partie de plusieurs tunnels entrants et sortants.
- **La base de données réseau NetDB** : contient les informations sur les routeurs et les services disponibles sur le réseau. Des routeurs particuliers, nommés Floodfill, sont chargés de stocker et de maintenir cette base à jour.

Le réseau I2P est constitué d'un ensemble de routeurs virtuels. Chaque utilisateur qui rejoint le réseau fait office de routeur et communique avec les autres routeurs constituant le réseau. Néanmoins, afin de garantir l'anonymat des utilisateurs, l'expéditeur et le destinataire ne communiquent pas directement entre eux, mais passent par de multiples routeurs. Des données transitent donc en permanence dans tous les routeurs virtuels du réseau. Le système est conçu pour qu'aucun utilisateur n'ait le moyen de savoir si les données reçues proviennent du routeur précédent ou si celles-ci ont juste été relayées par ce dernier [21].

II.8.3.A Base de données NetDB

La base de données décentralisée NetDB contient toutes les informations permettant aux routeurs de communiquer. Ces métadonnées se résument à deux types d'enregistrement [21] :

- **RouterInfo** : données nécessaires pour communiquer avec un autre routeur (identité du routeur, passerelle, etc.).
- **LeaseSets** : données fournies aux routeurs pour communiquer avec un service, notamment les clés publiques, l'identité de passerelle d'entrée du tunnel du service et l'adresse de la destination finale.

La base NetDB est hébergée et alimentée par les routeurs Floodfill. C'est grâce à ces derniers que les autres routeurs sont en mesure de connaître les coordonnées des services exposés sur le réseau. Tous les utilisateurs du réseau peuvent configurer leur routeur en mode Floodfill, mais cela nécessite d'autoriser un trafic plus volumineux. Les LeaseSets ont une durée limitée à 10 minutes dans la NetDB. Le routeur de l'utilisateur aura besoin de récupérer les nouvelles informations liées au service pour pouvoir continuer à communiquer [21].

II.8.4 Tunnels

- **Les tunnels « exploratoires »** sont utilisés pour envoyer des requêtes aux bases de données (NetDB) afin de construire les tunnels clients, ainsi pour la maintenance de la NetDB du réseau et des tunnels [23].
- **Les tunnels « Clients »** sont utilisés pour échanger des données. Les tunnels sortants permettant d'envoyer des messages et les tunnels entrants permettant d'en recevoir. [23].

Le premier nœud d'un tunnel est nommé la passerelle, le dernier, le nœud de sortie et les autres, les nœuds intermédiaires [16].

Un routeur envoie son message dans un de ses tunnels sortants, son nœud de sortie transmet le message à la passerelle d'un des tunnels entrants du destinataire. Pour que le nœud de sortie puisse savoir à quel tunnel entrant envoyer le message, le routeur ajoute les instructions dans le message chiffré. Les nœuds intermédiaires ne peuvent pas savoir s'ils transmettent des données d'un tunnel entrant ou sortant. Chaque tunnel a un identifiant unique choisi aléatoirement par le routeur qui l'a créé. La durée de vie d'un tunnel est de dix minutes [16].

II.8.5 Construction des tunnels

La construction de ses tunnels s'effectue en collectant des données « RouterInfo » dans le NetDB, dont l'objectif est de créer une liste de nœuds qui pourrait être utilisée dans ses tunnels. Ensuite le nœud envoie un message de construction du tunnel au premier nœud choisi, ce denier retransmettra la demande de construction au nœud suivant qui fera de même pour le denier nœud [22].

Une fois ses tunnels construits, le nœud, pour contacter un autre nœud, va récupérer son " RouterInfo " dans NetDB et donc la liste des passerelles des tunnels entrants du nœud distant. Ensuite, le nœud envoie dans un de ses tunnels sortants un message contenant les informations d'un des tunnels entrants du nœud distant permettant au nœud de sortie du tunnel sortant de savoir à qui retransmettre le message [16].

II.8.6 Sécurité et chiffrement

Afin d'établir une communication anonyme et sécurisée, I2P utilise différentes couches de cryptage. La sécurité des communications repose donc sur la cryptographie symétrique et asymétrique. I2P utilise quatre chiffrements [23] :

- Pour les tunnels sortants, l'émetteur ajoute de multiples couches de chiffrement (une par nœud traversé). Chaque couche est ensuite supprimée (déchiffrée) par chaque nœud traversé. Pour les tunnels entrants, l'opération est inversée. Ce sont les nœuds traversés qui ajoutent la couche de chiffrement et le receveur déchiffre toutes les couches successives. En effet, celui-ci connaît les clés de chaque nœud au moment de la construction du tunnel.
- I2P utilise un chiffrement en « garlic » (tête d'ail) pour protéger le contenu des informations qui transitent entre les différents routeurs du tunnel. Le principe du routage en ail est de chiffrer plusieurs messages ensemble et de les faire circuler dans le même paquet, cela permet d'une part d'accélérer la vitesse de transfert mais également rendre plus difficile l'analyse du trafic. Le routage en ail consiste à regrouper plusieurs messages et leurs instructions en un seul block, ces messages ne seront exposés qu'une fois arrivée au bout du tunnel.
- L'algorithme de chiffrement symétrique AES256[7] est utilisé pour le chiffrement du message transmis de bout en bout (la clé est chiffrée à l'aide d'ElGamal).
- Le système ElGamal est utilisé pour le chiffrement asymétrique qui intervient de bout en bout entre l'émetteur et le destinataire. Celui-ci est également utilisé pour les enregistrements et les requêtes de la NetDB envoyées au routeur « *Floodfill* ».

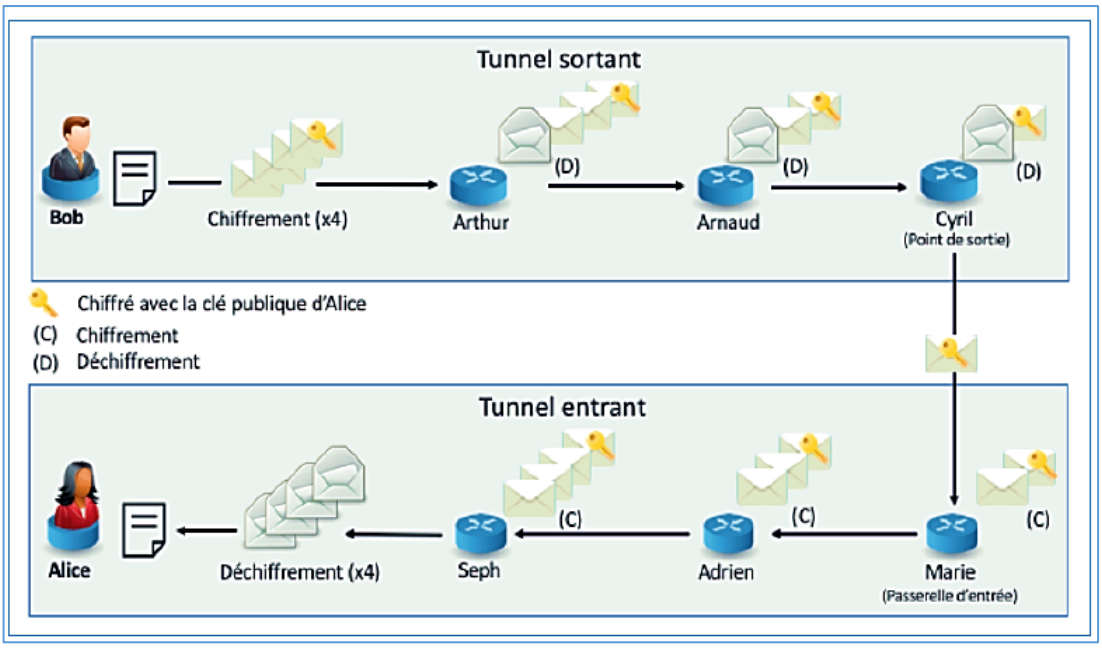

Figure II.9 : *Chiffrement des informations qui transitent dans les tunnels [21].*

[7] **AES :** Advanced Encryption Standard.

II.8.7 Services d'I2P

Les services disponibles via I2P sont similaires aux services Internet de base habituelles telles que la navigation Web, la messagerie électronique, les blogs et les forums, l'hébergement de sites Web, le chat en temps réel (IRC), le partage de fichiers Peer-to-Peer (*Torrents*). Ces services tendent à renforcer l'utilité du réseau dans son ensemble et à rendre le contenu du réseau plus susceptible d'être découvert. Parmi ces services [24] :

- **E-MAIL I2P** : Avec un service de messagerie sécurisé et décentralisé à l'intérieur d'I2P, il n'est pas possible d'envoyer des messages vers des comptes de messagerie en dehors d'I2P sur Internet normal. I2P utilise de solides techniques de cryptographie pour garantir la sécurité et l'anonymat.
- **I2PSnark** : anonymiser les téléchargements Bittorrent.
- **EEPSITE (SITES WEB)** : Les sites Web I2P sont l'équivalent du site Web sur Internet ordinaire. Ils peuvent contenir le même contenu que les sites Web habituels. De plus, en raison de la bande passante plus faible, la plupart des sites Web utilisent peu de supports gourmands en bande passante tels que des images ou des vidéos et un style très simple. Une différence importante avec l'Internet classique est l'absence d'un système de noms de domaine (DNS) central. Les domaines.i2p peuvent être enregistrés par n'importe qui et sont principalement communiqués aux autres utilisateurs via un service de carnet d'adresses.
- **Service Bittorrent** : pour télécharger des torrents (Le téléchargement d'un torrent[8] utilise le P2P).

II.9 Routage en oignon (Tor)

II.9.1 Définition

Tor est l'un des outils les plus connus et les plus célèbres parmi les solutions de confidentialité et d'anonymat sur Internet. Il est à la fois un logiciel libre et un réseau de surcouche distribué en nœuds sur la base du volontariat. Sous le terme décentralisé se cache des milliers de serveurs mis à disposition par des bénévoles que l'on appelle des nœuds et qui agissent comme des relais pour permettre l'anonymisation des connexions.

[8] **Torrent :** Un Torrent est un fichier « .torrent » contenant des informations pour savoir sur quels ordinateurs se connecter afin de télécharger un contenu donné sur Internet (vidéo, audio, image, fichier compressé, exécutable...) via le protocole Bittorrent.

Tor a été conçu à l'origine dans l'optique de respecter les considérations suivantes :

- **Déployabilité** : Tor doit être déployable et utilisé dans le monde réel. Ainsi, il ne doit pas être coûteux et ne doit pas être difficile dans l'implémentation.
- **Utilisabilité** : Un système difficile à utiliser a logiquement moins d'utilisateurs, et parce que les systèmes d'anonymisation cachent les utilisateurs parmi les autres utilisateurs, l'utilisabilité est une réelle nécessité de sécurité. A ce titre, Tor doit être le plus multiplateforme possible rendant son utilisation "confortable".
- **Flexibilité** : Le protocole de Tor doit être flexible et bien spécifié, de telle manière à ce que Tor puisse servir de base à de la recherche. Tor veut s'imposer comme un standard (et c'est tout de même assez le cas actuellement).
- **Conception simple** : Le protocole et les paramètres de sécurité doivent être bien compréhensibles. Des fonctionnalités supplémentaires imposent des coûts d'implémentation et de complexité. Tor a pour but de déployer un système simple et stable qui intègre les meilleures approches pour protéger l'anonymat.

II.9.2 Routage :

Le routage des oignons Tor est l'implémentation la plus réussie et la plus courante d'un type de mécanisme d'acheminement de l'oignon. On peut également se demander d'où vient l'analogie faite avec l'oignon. Celle-ci fait référence à la manière dont les données sont encapsulées puis "épluchées" au cours d'un trajet dans un circuit Tor [25].

Le réseau est constitué de nœuds nommés "Routeur Oignon". Un utilisateur établit un circuit de routeur oignon, la communication est cryptée en couche avec chaque clé publique des nœuds du circuit. Chaque nœud du circuit connaît uniquement son prédécesseur et son successeur. Le dernier nœud du circuit transmet le message au destinataire, il est le seul à le connaître, tout comme le premier nœud du circuit est le seul à connaître l'utilisateur.

Les nœuds au sein de Tor peuvent être de plusieurs types [25] :

- **Les Oignon Router** (OR, aussi appelés relais) : Ce sont les nœuds qui constituent les circuits utilisés au travers du réseau, ils sont le cœur fonctionnel de Tor, ce sont eux qui font transiter les paquets au travers du nuage Tor.
- **Les Nœuds clients** (aussi appelés oignons proxies, OP, par abus de langage) : Ce sont les nœuds qui se connectent au réseau, ou plus précisément les clients logiciels Tor.
- **Les Directory Servers** (aussi appelés authority Servers) : Il s'agit des serveurs qui référencent les OR connus, ils sont les annuaires du réseau.

La communication au travers du réseau Tor fonctionne en routant les paquets au travers d'un circuit de relais entre le client et le serveur, ce circuit a pour longueur trois relais, que l'on nomme Gardien - Nœud intermédiaire – Nœud de sortie. Chaque nœud partage avec le client une clef de chiffrement symétrique, et le message est donc enveloppé de trois couches de chiffrement, à la manière d'un oignon [25].

Ce circuit ainsi crée réduit drastiquement les risques d'analyses de trafic en disséminant les communications entre plusieurs endroits de l'Internet. On ne peut donc pas en observant un seul point, associer l'utilisateur à son destinataire. Au lieu d'emprunter un itinéraire direct entre la source et la destination, les paquets suivent une trajectoire qui semble aléatoire à travers plusieurs relais, de plus, les données étant chiffrées, personne ne peut savoir qui parle et ce qu'il dit dans le nuage.

Lorsqu'un client désire communiquer avec un serveur externe à travers le nuage Tor, le client va construire un circuit entre lui et sa destination. Ainsi l'OP du client va consulter les Directory Servers afin de connaître la liste des OR, avec diverses informations à leur sujet, notamment leur adresse IP, leur vitesse de connexion etc...

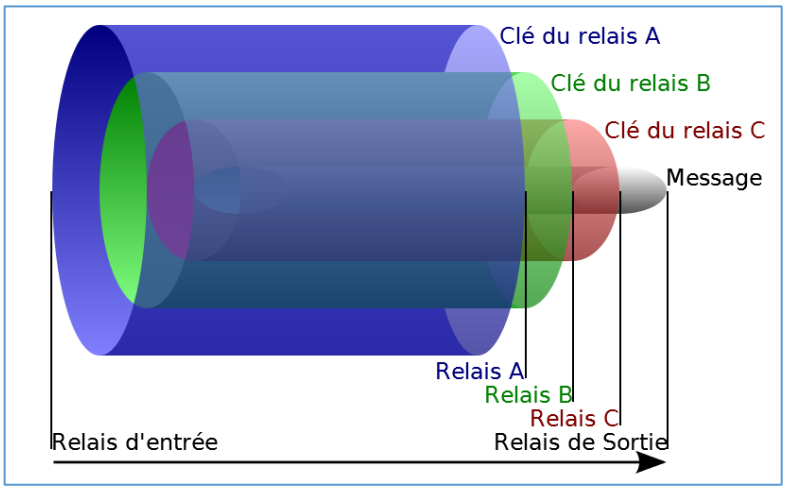

Figure II.10 : *Principe du routage en oignon.*

II.9.3 Serveurs annuaires

Les directory serveurs sont les nœuds spécifiques qui ont le rôle d'annuaire au sein du réseau. Ils sont redondants, et présentent les mêmes informations. Leur fonction est de référencer les OR connus, en présentant leur descripteur de routeur. Ces descripteurs regroupent des informations telles que ses clefs publiques, sa politique de sortie, son adresse IP, sa bande passante, la version de Tor utilisée, etc...

II.9.4 Circuit

Le client va choisir 3 relais qui constitueront le circuit. Il va inspecter la liste des nœuds pour choisir le nœud de sortie adéquat car ces derniers ont des politiques de sorties et n'acceptent que certains ports.

Ensuite, le client va construire un circuit entre lui et le serveur, chaque nœud apportant son chiffrement.

Une fois le circuit établit, il peut être utilisé pour anonymiser une application TCP, comme du trafic web, de la messagerie instantanée.

On peut noter que la liaison « nœud de sortie – Serveur » n'est pas chiffrée par Tor, ce qui implique qu'elle n'a que le chiffrement propre au protocole utilisé, plus simplement, utiliser du HTTP sans TLS ne procure aucune confidentialité, avec ou sans Tor.

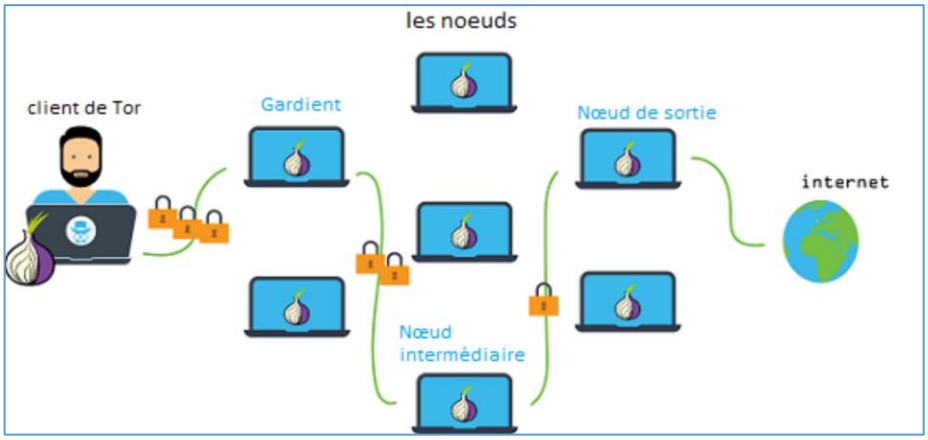

Figure II.11 : *Circuit Tor.*

II.9.5 Services cachés

Les services cachés permettent d'anonymiser le côté destinataire d'une connexion TCP. Dans ce que nous avons vu jusqu'à maintenant, seule la source était cachée du destinataire. Les services cachés permettent de cacher la destination. Cependant il faut noter que s'il n'est possible de cacher que la source d'une connexion TCP avec Tor, il n'est pas possible de ne cacher que la destination. En effet, l'utilisation des services cachés impose également la dissimulation de la source, ainsi, il est obligatoire d'utiliser Tor pour accéder aux services cachés [26].

Les services cachés fonctionnent sur l'utilisation de "point de rendez-vous" : le client choisit un point de rendez-vous qui sera communiqué de manière indirecte au service caché, et les deux communiqueront via ce point de rendez-vous. Il se pose le problème de savoir comment le client peut-il communiquer le point de rendez-vous au serveur, s'il ne

connaît pas le serveur justement. Un service caché doit donc afficher son existence dans le nuage Tor avant que des clients puissent le contacter. Pour ce faire, le service caché choisi des routeurs oignons, construit des circuits vers eux, et leur demande de se comporter comme étant des points d'introduction en leur fournissant sa clef publique.

L'utilisation d'un circuit Tor rend difficile d'associer un serveur à ses points d'introduction. Et bien que les points d'introduction disposent de la clef publique identifiant le service caché, ils n'ont aucune idée de l'IP de ce même service [26].

II.9.5.A Création du service caché

Publier un service caché sur le grand darknet permet dans l'exemple d'avoir un service en ligne plus discret que sur l'internet visible comme par exemple un FTP [27].

1. Le serveur choisit les points d'introductions en leur communiquant sa clef publique.
2. Le serveur fait un descripteur de service (Publique Key + IP des points d'introductions) et envoi le descripteur au serveur d'annuaire de TOR.
3. Lorsque le client renseigne le domaine oignon de la cible il télécharge le descriptif de service oignon, grâce à celui-ci il connaît les relais d'introductions et la clef publique. Il crée un circuit parmi les relais jusqu'à fixer un point de rdv sur un relai choisit.
4. Le client assemble un descriptif en mixant celui du service oignon avec le point de rdv puis envoi le descriptif aux points d'introductions du service oignon (le descriptif est chiffré avec la clef publique du service oignon).
5. Le point d'introduction communique le nouveau descriptif au service oignon, puis le service oignon crée un circuit entre les relais TOR jusqu'au point de rdv.
6. Le point de RDV interconnecte le client et le service.

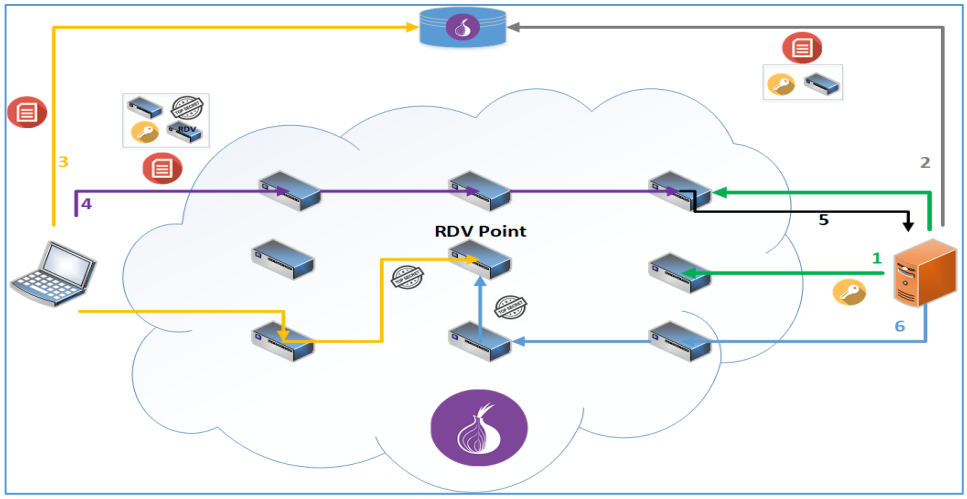

Figure II.12 : *Création du service caché [27].*

II.9.6 Contrôle du débit

Nous avons parlé de la présence d'une valeur de bande passante dans les descripteurs de routeurs associés aux nœuds. Il est en effet possible de spécifier une bande passante maximale à utiliser, pour minimiser la charge du serveur. Cela permet en partie d'assurer que le volontaire qui héberge le nœud ne donne pas plus que ce qu'il souhaite donner. Ce qui cultive la confiance envers Tor pour les hébergeurs [26].

Cette limitation se fait en utilisant un Token bucket. Cette technique consiste à modéliser la bande passante disponible sous forme de sceau, où régulièrement sont versés des jetons, qui peuvent s'accumuler jusqu'à une certaine limite, et d'où les jetons sont tirés à chaque donnée transitée [26].

Ce contrôle permet de fournir un service préférentiel aux services interactifs qui n'ont pas besoin de beaucoup de débit mais d'une latence faible.

II.10 TOR vs I2P

Tor et I2P permettent un accès anonyme au contenu en ligne, ils utilisent une structure de routage de type Peer-to-Peer et fonctionnent à l'aide d'un cryptage en couches. Mais ils diffèrent également à certains égards.

II.10.1 Terminologie :

Les éléments techniques des deux réseaux sont proches, mais diffèrent par leur appellation.

Réseau Tor	Réseau I2P
Client	Routeur ou client
Circuit	Tunnel
Liste des nœuds	NetDB
Nœud d'entrée	Mandataire entrant
Nœud de sortie	Mandataire sortant
Site web caché /service Onion	Eepsites ou Destination
Point d'introduction	Passerelle entrante
Nœud	Routeur
Relais	Routeur
Point Rendez-vous	un peu comme passerelle entrante + extrémité sortante (outbound endpoint)

Descripteur de routeur	RouterInfo
Serveur d'annuaire	Routeur Floodfill

Tableau II.1 : *Comparaison de terminologie entre Tor et I2P.*

II.10.2 Usage

I2P et Tor ont l'objectif commun d'anonymiser les connexions de leurs utilisateurs et d'héberger des services non accessibles depuis l'Internet standard. Néanmoins, leurs usages sont relativement différents.

- **Tor**

Tor a été conçu comme un service proxy pour accéder à Internet ordinaire de manière anonyme, maintenant Tor a implémenté des « services cachés » qui permettent l'utilisation d'un réseau au sein d'Internet normal. Dans le réseau Tor, les utilisateurs doivent délibérément choisir de devenir un nœud, pour transférer la communication des autres. Enfin, Tor est écrit en C.

- **I2P**

I2P n'a pas été conçu pour créer des mandataires vers l'Internet externe. Il est plutôt destiné à être utilisé comme réseau interne et toutes ses applications résident à l'intérieur de ses propres frontières. Cela dit, il existe des services fournis par des bénévoles qui agissent en tant que mandataires pour le contenu basé sur internet normal - ceux-ci sont appelés "outproxies" sur le réseau I2P. Il y a un outproxies configuré par défaut dans le tunnel client HTTP d'I2P « false.i2p ». Bien que ce service existe actuellement, rien ne garantit qu'il soit toujours là car il ne s'agit pas d'un service officiel fourni par le projet I2P. Toute personne utilisant le réseau I2P agit comme un « nœud » pour transférer la communication des autres. Enfin, I2P est écrit en Java.

Ainsi, alors qu'en théorie, les deux peuvent être utilisés pour les mêmes applications, la pratique apprend que Tor et I2P sont beaucoup plus efficaces lorsqu'ils sont utilisés de la manière dont ils étaient initialement destinés à être.

II.10.3 Fonctionnement

Tor et I2P semblent avoir de nombreux points techniques communs. Les deux sont des réseaux distribués et offrent des communications anonymes et privées grâce à des couches de chiffrement. Néanmoins, leur conception et leur fonctionnement sont pourtant différents.

❖ **Tor**

1. Tor dispose de trois différents types de nœuds :
   - Les serveurs d'annuaire qui constituent la base de données centrale du réseau.
   - Les relais internes par lesquels transite le trafic depuis Tor vers Tor.
   - Les nœuds de sortie vers Internet.
2. Tor protège chaque élément du message en trois couches de cryptage, une technique appelée « routage de l'oignon ». Le message quitte le réseau Tor dans son état d'origine (pré-cryptage).
3. Pour augmenter les performances, Tor configure plusieurs utilisateurs pour suivre le même chemin à travers le réseau.
4. Tor utilise des connexions cryptées bidirectionnelles pour relayer les données jusqu'aux nœuds de sortie ou aux services cachés (sites.onion), il utilise toujours trois relais pour traverser son réseau. Par ailleurs, Tor ne supporte pas le protocole UDP.
5. Étant donné que Tor utilise toujours des adresses IP, il utilise la résolution DNS pour naviguer sur internet normal.
6. Dans TOR, le nœud d'entrée est le seul à connaître l'identité de l'utilisateur et le nœud de sortie est le seul à connaître le destinataire.

❖ **I2P**

1. Chaque nœud est un routeur et il n'y a aucune distinction, contrairement à Tor. Chaque utilisateur I2P est un routeur d'un tunnel relayant du trafic.
2. I2P dispose d'une base de données réseau (NetDB), distribuée, maintenue par les routeurs Floodfill. Aucun serveur central n'existe.
3. I2P utilise une variante du routage des oignons nommée routage de l'ail pour créer des connexions anonymes.
4. I2P utilise des connexions unidirectionnelles entre chaque serveur de ses tunnels. Il supporte le protocole TCP (Transmission Control Protocol) et du protocole UDP (User Datagram Protocol) ce qui permet de fournir de meilleures performances pour certaines applications.
5. I2P n'utilise donc pas de correspondance adresse IP/nom de domaine pour accéder aux différents services. Ils sont principalement communiqués aux autres utilisateurs via un service de carnet d'adresses.
6. Dans I2P, un nœud ne peut pas savoir si les messages qu'il transmet, proviennent directement d'un utilisateur ou d'un autre nœud dans le tunnel.

II.10.4 Communauté

Tor est un réseau beaucoup plus grand et qui dispose d'un financement important et a déjà résolu certains des problèmes d'évolutivité qu'I2P n'a pas encore rencontrés. Tor bénéficie d'un nombre de développeurs bien plus important qu'I2P [21].

Le réseau Tor dispose d'une grande visibilité, très médiatisé depuis quelques années. I2P reste encore très anonyme malgré ses 13 ans d'existence et peu de personne, même dans une population du secteur informatique, ne le connait. Une petite équipe de développeurs répartie sur plusieurs continents gère l'avancement du projet. La documentation sur Tor est également mieux fournie et de nombreux investissements en recherche et développement y sont consacrés [21].

II.10.5 Avantages de Tor sur I2P

Parmi ces avantages [28] :

- Plus grand nombre d'utilisateurs ; beaucoup plus de visibilité dans les communautés universitaires et hackers.
- Conçu et optimisé pour le trafic sortant, avec un grand nombre de nœuds de sortie.
- Une meilleure documentation avec des articles et des spécifications formelles, un meilleur site Web, beaucoup plus de traductions.
- A un financement important.
- Les nœuds client de Tor ont une très faible surcharge de la bande passante.
- Assez grand pour avoir dû s'adapter aux tentatives de blocage et de DOS.

II.10.6 Avantages d'I2P sur Tor

Parmi ces avantages [28] :

- Conçu et optimisé pour les services cachés, qui sont beaucoup plus rapide que dans Tor.
- Entièrement distribué et auto-organisé.
- Tellement petit qu'il n'a pas été bloqué ni subi beaucoup de DOS, ou pas du tout.
- Tunnels unidirectionnels au lieu de circuits bidirectionnel, signifie qu'un attaquant doit compromettre deux fois plus de nœuds dans I2P que dans Tor pour obtenir la même quantité d'informations.
- Dans I2P, les tunnels ont une durée de vie courte, diminuant ainsi le nombre d'échantillons qu'un assaillant peut utiliser pour lancer une attaque active, contrairement aux circuits dans Tor, qui ont généralement une durée de vie longue.

- Par essence, tous les pairs participent à acheminer pour d'autres.
- Mécanisme intégré de mise à jour automatique.
- Utilise à la fois les transports TCP et UDP.

II.11 Entreprise et anonymisation

Bien que Tor et I2P était conçu pour répondre à des besoins liés à une utilisation bienveillante, les réseaux Tor/I2P présentent des limites dont il faut être conscient pour ne pas négliger certains risques de sécurité

Le réseau Tor est souvent utilisé pour la distribution de marchandises et contenus illégaux, allons jusqu'à la contrefaçon de cartes de crédit. Il a survécu principalement parce que ses mécanismes d'anonymisation et de chiffrement étaient suffisants pour contrecarrer toute tentative légale de faire cesser ses activités. Alors qu'I2P pourrait être aussi utilisé par des cybercriminels ou encore des terroristes pour mettre en place des réseaux IRC anonymes ou pour développer des applications de transfert de fichiers, tout en faisant en sorte que les autorités ne puissent pas savoir quels sont leurs sujets de discussion.

Dans une entreprise, la connexion internet d'un employé est censée être strictement professionnelle. L'employeur est en droit d'interdire l'usage privé de l'Internet au travail s'il estime que cet usage nuit au rendement du salarié et de l'entreprise. Cependant, cette règle n'est pas toujours respectée. Certains salariés se permettent de débloquer des sites dont l'accès est interdit par l'employeur. Plusieurs méthodes de déblocage sont disponibles, notamment celles utilisant des logiciels spécifiques comme TOR et I2P, ces logiciels dissimulent l'adresse IP permettent d'utiliser un autre chemin pour accéder à des sites bloqués. Cependant, utiliser Tor/I2P au sein d'entreprise peut engendrer des risques de sécurités de réseau. Parmi ces derniers :

- Les logiciels (Tor/I2P) peuvent être utilisés pour contourner des mesures de contrôles mises en place pour empêcher la divulgation de documents confidentiels.
- Tor/I2P peut chiffrer tout le trafic sur votre réseau et rendre la surveillance de vos activités très difficile.
- Tout utilisateur qui télécharge des contenus via Tor/I2P pourrait donc exposer le réseau d'entreprise à un risque d'infection par des malwares.
- Les nœuds de sortie de Tor peuvent surveiller le trafic qui transite par les appareils des employés et capturer toute information non chiffrée telle que le login ou mot de passe.

- Tor peut entraîner une forte utilisation de la bande passante de votre réseau d'entreprise. Ceci peut exposer en permanence votre organisation à une attaque DDoS (Distributed Denial of Service), laquelle peut rendre votre serveur, un service ou une infrastructure indisponible.

Vu les risques ci-dessus, Il est recommandé aux entreprises et aux administrations de détecter voire bloquer les communications qui pourraient être établies vers des nœuds Tor/I2P, même si son utilisation n'est pas explicitement proscrite par la PSSI (politique de sécurité du système d'information) de l'organisation.

II.12 Conclusion

Aujourd'hui, Internet met les gens au défi de préserver leur vie privée en ligne, l'anonymat permet à des individus de s'exprimer sans crainte de représailles. Et de surfer sur Internet librement.

Par conséquent, il existe plusieurs outils permettant de garder l'anonymat sur le Web, parmi ces derniers présentés précédemment navigateur privé, réseau Tor, VPN, et le réseau I2P. Les méthodes de chiffrement et de routage utilisés au sein des réseaux Tor et I2P les rendent les meilleurs réseaux anonymes. Comme susmentionné, Tor et I2P peuvent être le moyen idéal pour les utilisateurs qui veulent couvrir leurs traces, mais leurs utilisations dans un réseau d'entreprise peuvent exposer l'organisation à certains risques de sécurités.

Le premier pas vers la détection de ces réseaux anonyme consiste à extraire les signatures numériques de ces derniers, qui seront abordé dans le troisième chapitre.

CHAPITRE III

Extraction des signatures numérique du réseau

TOR & I2P

III.1 Introduction

Le réseau Tor et I2P peuvent être le moyen idéal pour les utilisateurs qui veulent couvrir leurs traces, mais leurs utilisations dans une entreprise peuvent l'exposer à divers risques de sécurités et de problèmes judiciaires. Pour cette raison l'administrateur réseau souhaite bloquer tout le trafic basé sur ces réseaux au sein d'une entreprise.

Il faut d'abord souligner qu'il est difficile de détecter et de bloquer Tor/I2P dans le réseau d'entreprise. Pour détecter le réseau Tor et I2P, l'administrateur doit effectuer une analyse du trafic web normal et du trafic provenant du réseau Tor/I2P, afin d'obtenir les différences entre ces derniers. Ce qui ramène à extraire les signatures numériques du réseau Tor/I2P qui permettent d'identifier ces derniers.

Au cours de ce chapitre nous allons réaliser en premier lieu une architecture client-serveur dont le but est de simuler le réseau d'une entreprise. Ensuite nous allons faire une comparaison entre le réseau Tor et I2P en terme de performance. Puis nous allons effectuer la détection du Tor/I2P dans notre réseau, en se basant sur l'analyse des différents trafics provenant des navigateurs normaux et du navigateur Tor/I2P qui va nous permettre de révéler les signatures de ces deux réseaux d'anonymat.

III.2 Environnement de travail

Un environnement de travail se réfère à une suite de matériels et de logiciels que nous avons utilisés pour atteindre le but de notre recherche.

Comme montré dans la figure III.1, nous avons utilisé deux ordinateurs avec les caractéristiques décrites dans le tableau III.1. Notre environnement est implémenté comme une architecture client-serveur en utilisant CCProxy[1] pour gérer les connexions dans notre réseau. Et les logiciels d'anonymat Tor[2] et I2P[3], dont I2P repose sur l'environnement JAVA[4]. Et Wireshark[5] pour l'analyse des paquets qui transitent dans le réseau ; ainsi que Snort[6] comme un système de détection d'intrusion « NIDS ».

[1] https://www.youngzsoft.net/
[2] https://www.torproject.org/
[3] https://geti2p.net/
[4] https://www.java.com/
[5] https://www.wireshark.org/
[6] https://www.snort.org/

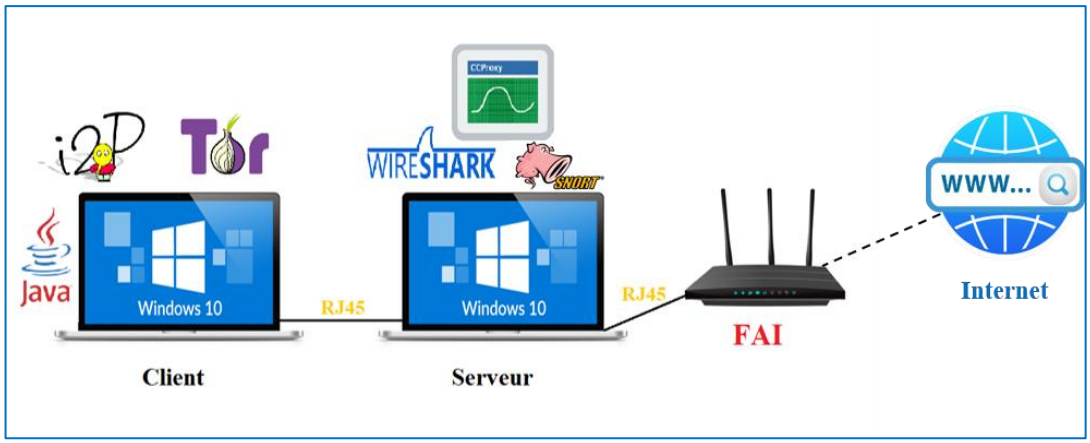

Figure III.1 : *Environnement de travail.*

Ordinateur	Serveur	Client
Système d'exploitation	Windows 10	Windows 10
RAM	4.00 Go	4.00 Go
Processeur	Intel® Core™ i5-4210U CPU @ 1.70 GHz 2.40 GHz	Intel® Celeron® CPU N2810 @ 2.00 GHz 2.00 GHz
Logiciels Installés	1. Wireshark 2. CCProxy 3. Snort	1. Navigateur Tor 2. I2P 3. Java
Fabriquant	Lenovo	TOSHIBA

Tableau III.1 : *Caractéristiques des machines.*

III.3 Architecture client-serveur

Nous avons réalisé une architecture client-serveur en utilisant deux ordinateurs dont l'adresse IP du client est « 192.168.137.210 » et ceux du serveur sont :

1. 192.168.1.16 : pour les communications entre le serveur et le fournisseur d'accès Internet « FAI ».
2. 192.168.137.1 : pour partager la connexion avec le client.

```
Carte Ethernet Ethernet :

   Suffixe DNS propre à la connexion. . . :
   Adresse IPv6 de liaison locale. . . . .: fe80::e142:8b02:65a7:700e%6
   Adresse IPv4. . . . . . . . . . . . . .: 192.168.137.210
   Masque de sous-réseau. . . . . . . . . : 255.255.255.0
   Passerelle par défaut. . . . . . . . . : 192.168.137.1
```

Figure III.2 : *Adresse IP du client.*

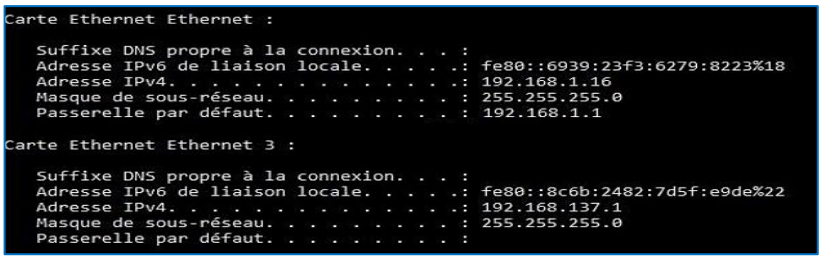

Figure III.3 : *Adresses IP du serveur.*

III.3.1 CCProxy

III.3.1.A Définition

CCProxy est un serveur proxy puissant et facile à utiliser. Il peut prendre en charge les connexions hautes débit, accès à distance, etc…. CCProxy comprend de nouvelles fonctions idéales cache Web, filtre Web, connexion à distance, etc…. Toutes ces fonctions vous permettent de partager la connexion Internet au sein du LAN de manière simple et efficace [29].

Après l'installation du CCProxy nous avons empêché le client d'accéder à certains sites Web (*https://www.facebook.com/* , *https://www.youtube.com/*) en bloquant ces derniers avec CCProxy.

III.3.1.B Configuration

La figure ci-dessous représente la configuration du CCProxy pour bloquer les sites mentionnés précédemment.

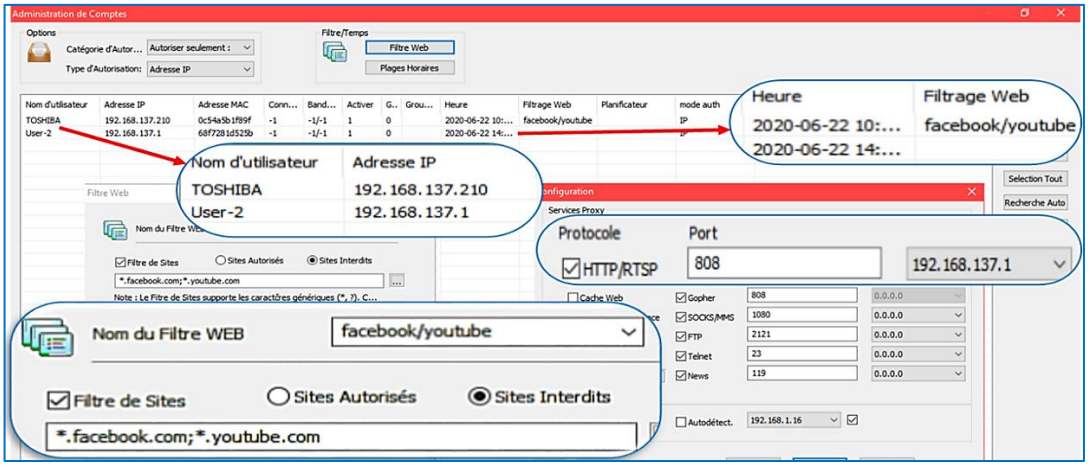

Figure III.4 : *Configuration du CCProxy.*

Au niveau du client nous avons testé d'y accéder aux sites interdis par le serveur en utilisant le navigateur Chrome, et nous pouvons voir dans la figure ci-dessous que les sites sont bloqués.

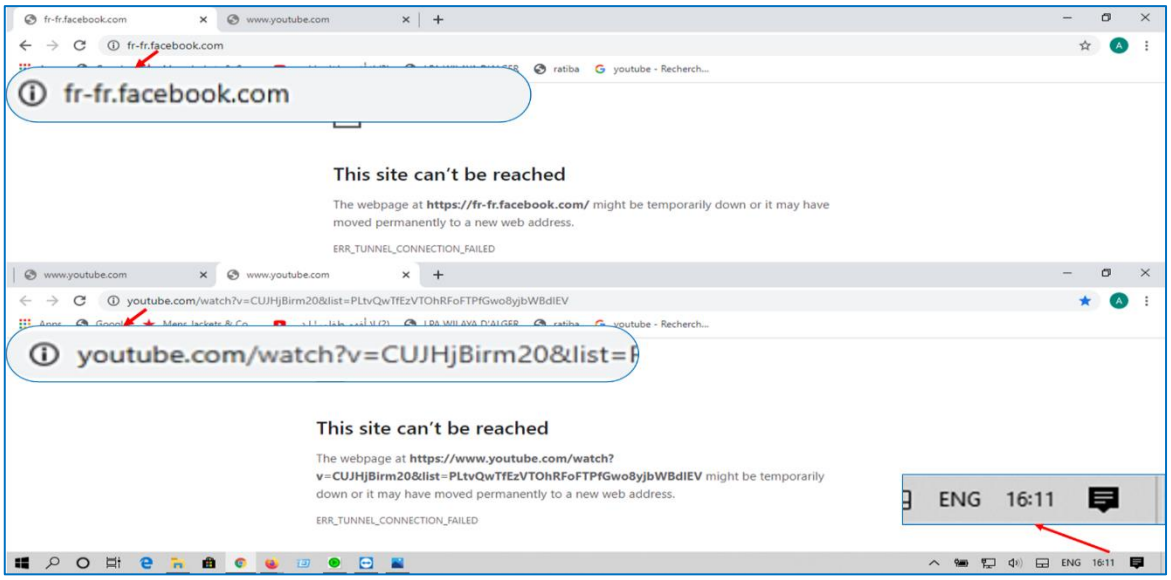

Figure III.5 : *Sites bloqués au niveau du client.*

Les tentatives pour y accéder aux sites « https://www.facebook.com/» «*https://www.youtube.com/*» ont été affichées dans le journal de CCProxy.

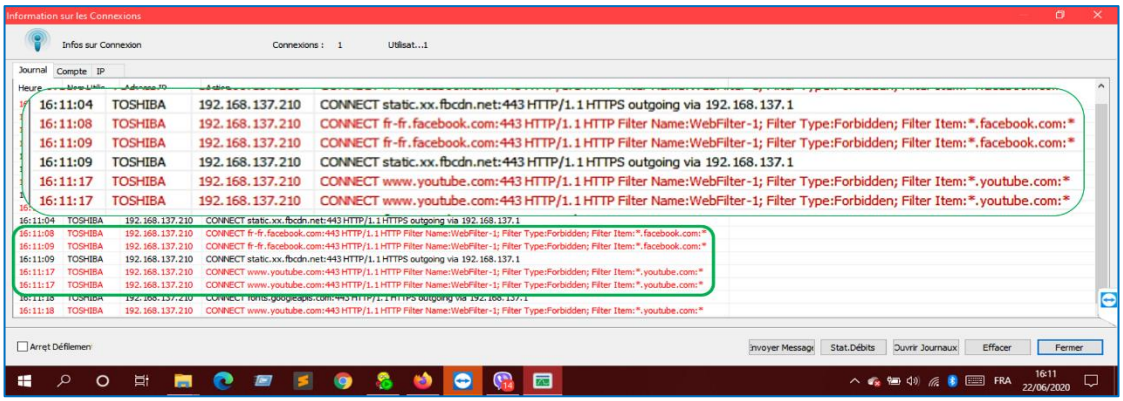

Figure III.6 : *Journal de CCProxy.*

Si nous souhaitons accéder aux sites interdits par le serveur nous devons utiliser les outils d'anonymats (Tor et I2P) pour contourner le proxy sans que l'administrateur se rend compte.

III.4 Réseau I2P

III.4.1 Lancement du réseau I2P

Pour lancer le réseau I2P nous cliquons sur l'icône « *Start I2P (Restartable*) ». Cela démarre le réseau I2P avec une fenêtre, le temps qu'I2P s'initialise, il ouvre ensuite automatiquement un navigateur avec la page d'accueil affichant des informations et des options pour la configuration.

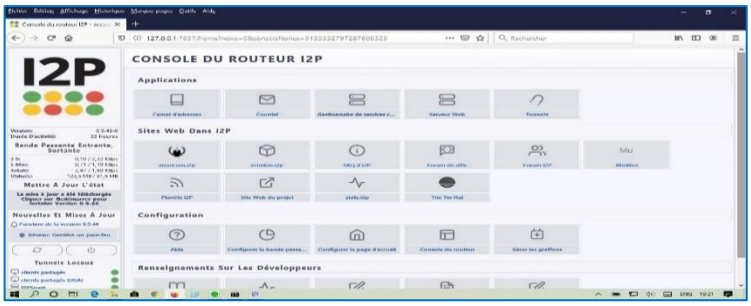

Figure III.7 *: Page d'accueil d'I2P.*

III.4.2 Configurer le navigateur pour pouvoir naviguer sur les sites I2P

La navigation sur les sites web I2P se fait par l'utilisation d'un proxy fourni dans l'installation d'origine. L'inconvénient d'une configuration manuelle du proxy dans le navigateur réside dans le fait qu'il faut le désactiver pour naviguer sur les sites web classiques et le réactiver pour naviguer sur les sites web I2P. Nous allons utiliser Firefox comme navigateur d'I2P.

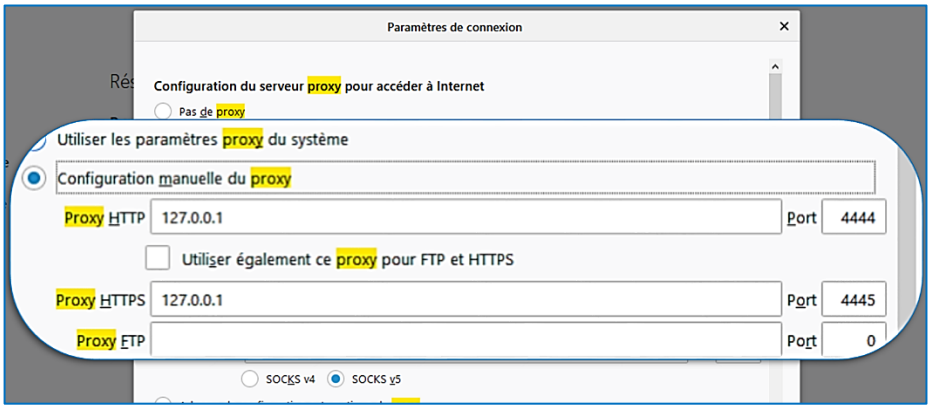

Figure III.8 : *Configuration du proxy du navigateur Firefox.*

Nous allons visiter quelques sites internes d'I2P :

1. forum.i2p : une connexion sécurisée et anonyme vers forum d'I2P « i2pforum.i2p ».
2. echelon.i2p : archive de software (I2P-Messenger, …etc.) et informations à propos d'I2P « echelon.i2p »

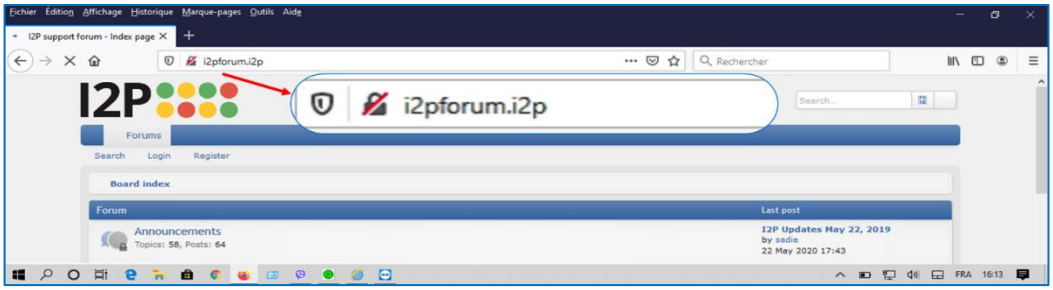

Figure III.9 *: Forum d'I2P.*

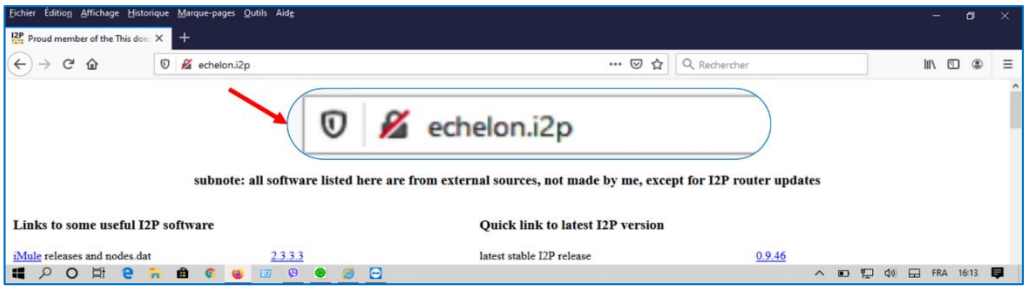

Figure III.10 : *Site Echelon d'I2P.*

III.4.3 I2PTunnel

I2PTunnel est un outil d'interfaçage et de fourniture de services sur I2P. Il est utilisé pour créer des tunnels vers des services et interagir avec eux. Afin de créer un service sur le réseau, il est nécessaire de fournir à I2PTunnel une adresse IP et un numéro de port pour que celui-ci génère une clé de destination et la publie sur le réseau. Une interface web pour la gestion I2PTunnel est disponible sur « 127.0.0.1:7657/i2ptunnel » [30].

Parmi les tunnels de client I2P :

1. **I2P HTTP Proxy** : 127.0.0.1: 4444 Un mandataire HTTP utilisé pour parcourir anonymement I2P et l'Internet ordinaire à travers I2P. La navigation sur Internet à travers I2P utilise un mandataire aléatoire précisé par l'option mandataires sortants comme l'outproxy « false.i2p ».
2. **Irc2P** : 127.0.0.1: 6668 Un tunnel IRC vers le réseau IRC anonyme par défaut, Irc2P.

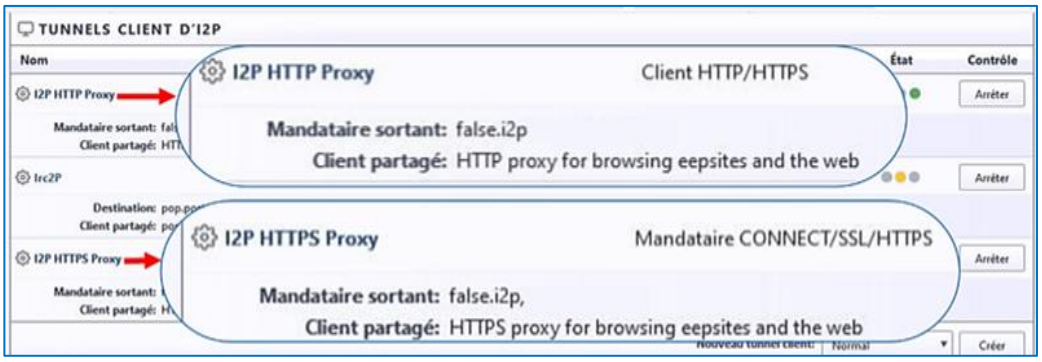

Figure III.11 *: I2PTunnel.*

III.4.3.A Mandataire sortant false.i2p

I2P n'est principalement pas destiné ni conçu pour être utilisé comme proxy pour l'Internet ordinaire. Cela dit, il existe des services qui sont fournis par des bénévoles qui agissent en tant que mandataires pour le contenu basé sur internet ceux-ci sont appelés "outproxies" sur le réseau I2P.

Il y a un outproxy configuré par défaut dans le tunnel client HTTP d'I2P « false.i2p ». Bien que ce service existe actuellement, rien ne garantit qu'il soit toujours là car il ne s'agit pas d'un service officiel fourni par le projet I2P [31].

Si la principale exigence à partir d'un réseau anonyme est la capacité d'accéder aux ressources internet, nous recommandons d'utiliser Tor [31].

Après avoir démarré les services I2PTunnel, Nous allons utiliser l'outproxy « false.i2p » pour contourner le proxy et y accéder aux sites interdis.

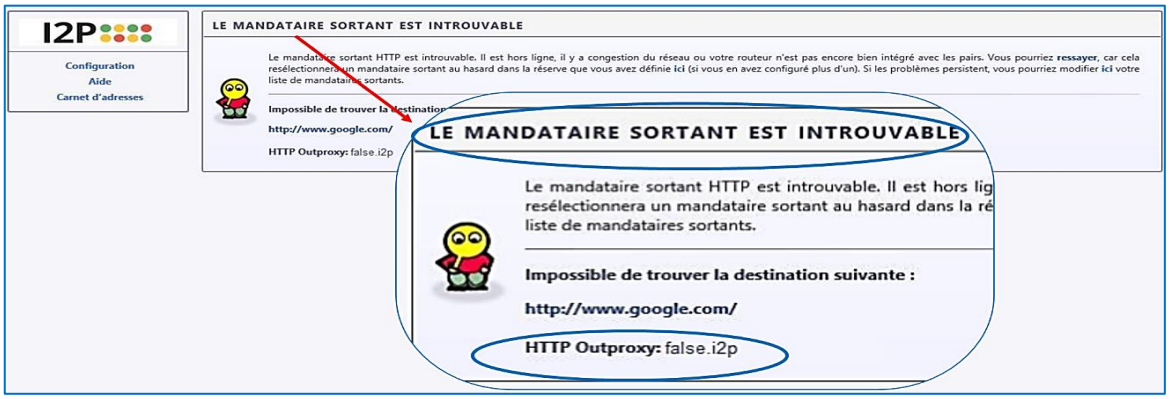

***Figure III.12** : Accès aux sites Web via l'outproxy false.i2p.*

Comme prévu l'outproxy false.i2p est hors ligne, cela signifie que nous ne pouvons pas y accéder au Web normal vie i2p, malgré plusieurs tentatives simultanées, l'outproxy est toujours introuvable. Nous allons tester un autre outproxy ci-dessous.

III.4.3.B Reddit

Reddit[7], est un site d'actualités sociales et de divertissement les utilisateurs peuvent publier et évaluer différents types de contenu. Les utilisateurs enregistrés soumettent du contenu sous la forme d'un lien ou d'un message texte. Les entrées de contenu sont organisées par domaines d'intérêt appelés « subreddits ». Il a été conçu au début du millénaire par deux étudiants de l'Université de Virginie, et maintenant il s'est développé en un site massif avec des centaines de milliers d'utilisateurs. Reddit est basé à San Francisco, en Californie [32].

Nous faisons déjà partie du Subreddits I2P qui est un subreddit pour les informations et les discussions liées au réseau anonyme I2P. Le « 20 juin 2020 » l'utilisateur « I2Pplus » qui est déjà un administrateur du subreddit I2P (développeur I2P) a publié un nouveau outproxy « purokishi.i2p » pour le réseau I2P, dont il explique ce mandataire sortant dans cette publication :

[7] https://www.reddit.com/

« Nouveau service outproxy I2P : purokishi.i2p. Il s'agit d'un nouveau service géré par des bénévoles pour fournir un outproxy fonctionnel et rapide aux utilisateurs I2P. Afin de fournir les meilleures performances possibles, le proxy externe est configuré pour s'exécuter à l'aide du nouveau cryptage ECIES-X25519 introduit avec I2P version 0.9.45, et va jusqu'à une certaine longueur pour bloquer les publicités et autres contenus indésirables. »

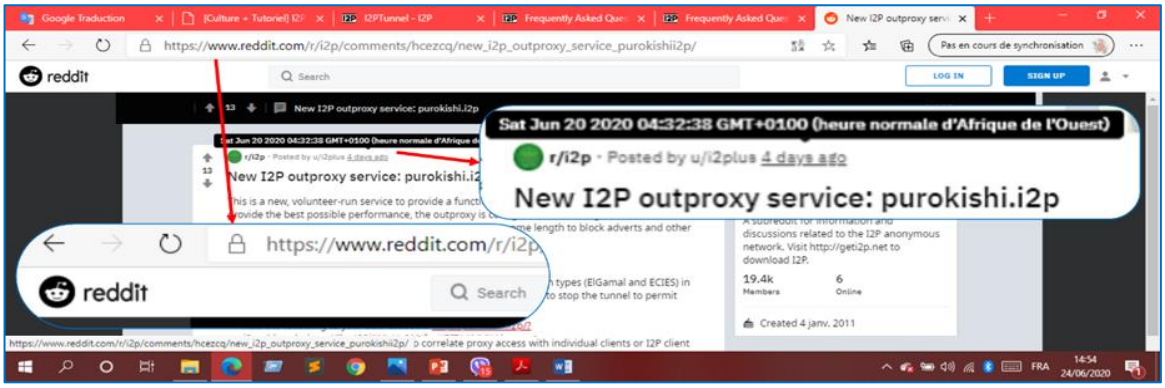

Figure III.13 : *L'outproxy Purokishi.i2p.*

III.4.3.C Mandataire sortant Purokishi.i2p

- **Fonctionnement**

Les mandataires sortants sont des services auxiliaires non officiels gérés par des bénévoles. Ce ne sont généralement que des proxys HTTP configurés pour écouter localement, puis transmis à un service I2P. Un serveur ou client est connu comme une « destination » et chaque destination a au moins un tunnel entrant et un tunnel sortant. Il y a par défaut 3 sauts par tunnel. Cela s'élève à 12 sauts (c.-à-d. 12 nœuds I2P différents) pour un aller-retour complet « client-serveur-client ». Cependant, si le client ne se soucie pas de l'anonymat, il peut configurer le site I2P pour utiliser uniquement 1 saut. Donc, si un utilisateur héberge un outproxy, le chemin de sortie (c'est-à-dire que l'outproxy est intrinsèquement visible). L'utilisateur a peu de raisons de protéger son anonymat dans ce cas, il choisit donc d'optimiser la vitesse et ne sélectionne qu'un seul saut pour ses entrées et sorties. Dans ce cas, la longueur du tunnel devient 4, 8 nœuds au total (4 entrants, 4 sortants). De plus, les mandataires externes ont toujours des adresses IP publiques. Dans le cas du proxy externe Purokishi.i2p, le trafic est acheminé vers Tor de manière sélectif, seul le trafic.onion est acheminé vers Tor.

Nous allons configurer ce nouveau outproxy à l'I2PTunnel spécifiquement au « I2P http proxy » et « I2P HTTPS proxy » en suivant les instructions de l'utilisateur « I2PPlus »

Configuration :

1. Nous nous assurons que notre client proxy http est bien configuré et démarré.
2. Dans la section "Outproxies" et "SSL Outproxies" du Gestionnaire de tunnel / Gestionnaire des services cachés, nous spécifions « false.i2p, purokishi.i2p ».

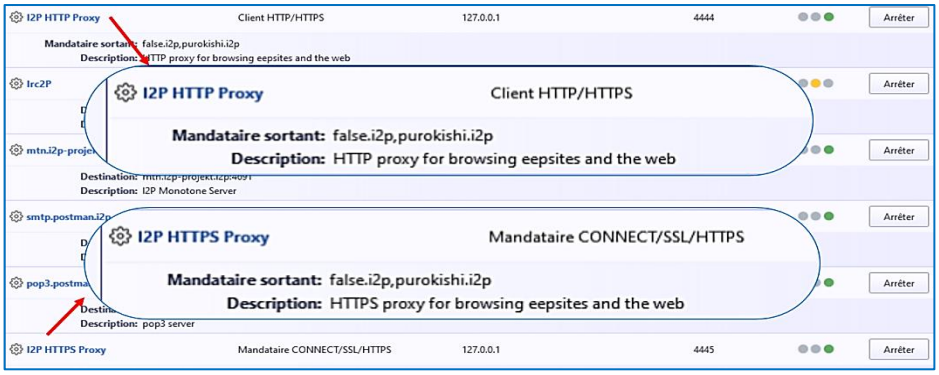

Figure III.14 : *Configuration de l'outproxy.*

Puis nous avons testé ce dernier pour y accéder aux sites web normal via I2P. Et nous avons pu y accéder aux sites Web comme : https://www.reddit.com/.

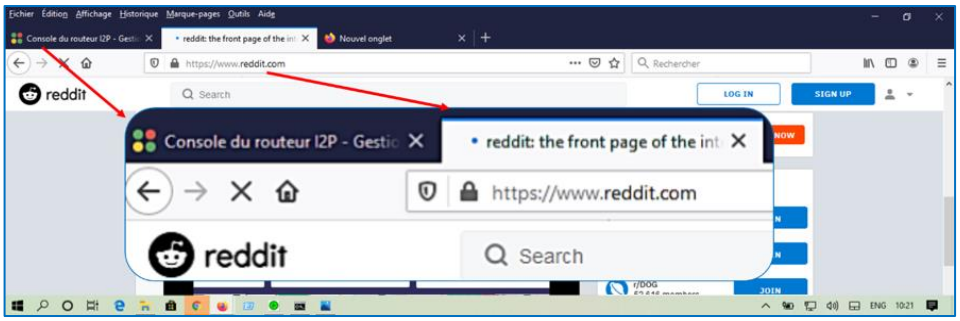

Figure III.15 : *Site Web Reddit via I2P.*

Puisque l'outproxy purokishi.i2p fonctionne bien nous allons l'utiliser pour contourner le proxy du serveur et visiter les sites Web interdis.

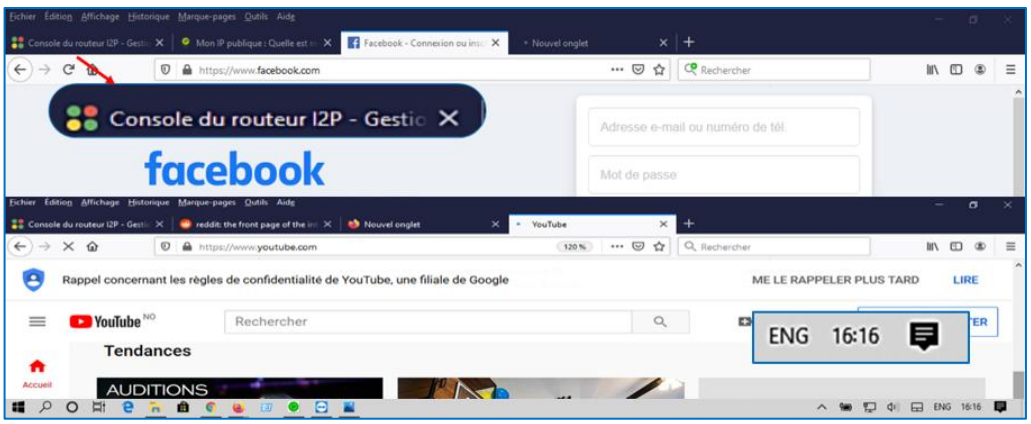

Figure III.16 : *Accès aux sites interdis via puroikishi.i2p.*

Alors comment I2P peut contourner le proxy bien que le proxy de l'ordinateur client est configuré avec l'adresse du serveur et le proxy du navigateur Firefox (navigateur I2P) est configuré avec l'adresse local « 127.0.0.1 et le port 4444 » ?

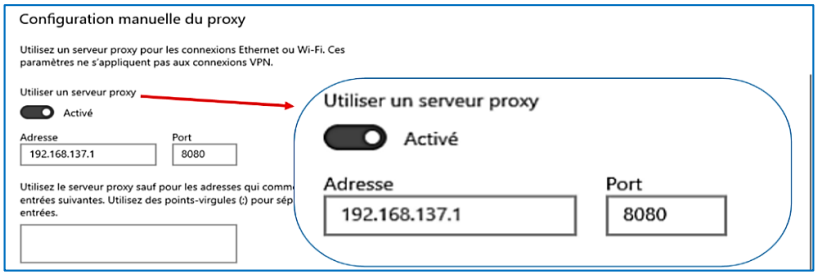

Figure III.17 : *Configuration du proxy de l'ordinateur client.*

Comme déjà mentionné précédemment le Firefox est configuré en adresse local et malgré ça il peut accéder à internet, il faut trouver l'adresse IP publique utilisé par le réseau I2P et la comparer avec l'adresse publique utilisé par l'ordinateur client (navigateur Chrome). Pour connaitre ces adresses publiques nous consultons le site d'identification d'adresse : https://www.monippublique.com/.

- Navigateur Chrome (celui qui utilise le proxy du serveur « 192.168.137.1 »)

Enabled WAN Connections:

Interface	Description	Connection Status	IPv4 Address	IPv6 Address
ADSL	br_0_0_35	Connected	0.0.0.0	
ADSL	pppoe_0_0_38	Connected	41.109.182.35	

Figure III.18 : *ADSL Router.*

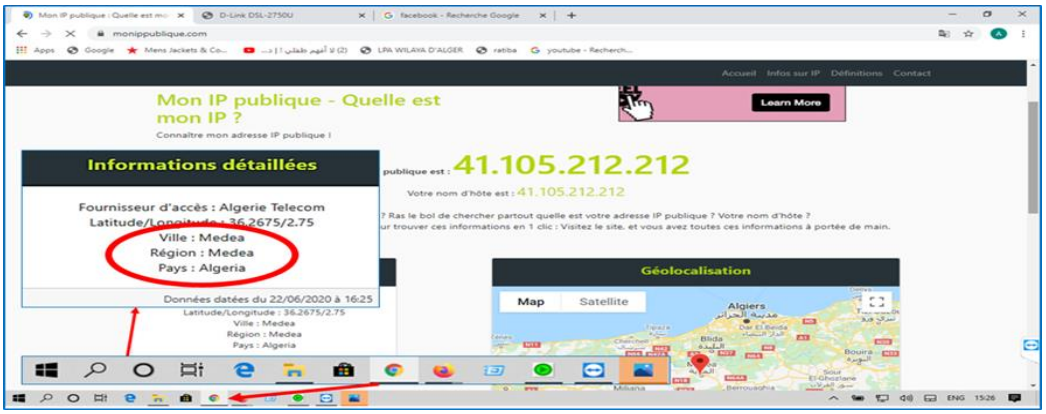

Figure III.19 : *L'adresse IP publique de l'ordinateur client.*

Nous remarquons que l'adresse publique utilisé par le navigateur Chrome est la même que l'adresse publique de l'ADSL[8] routeur, ce qui est logique puisque le chrome utilise le proxy du serveur, et que ce dernier utilise l'adresse publique du routeur pour sortir.

[8] **ADSL :** Asymmetric Digital Subscriber Line.

- Navigateur Firefox (celui qui utilise le proxy d'I2P « 127.0.0.1»)

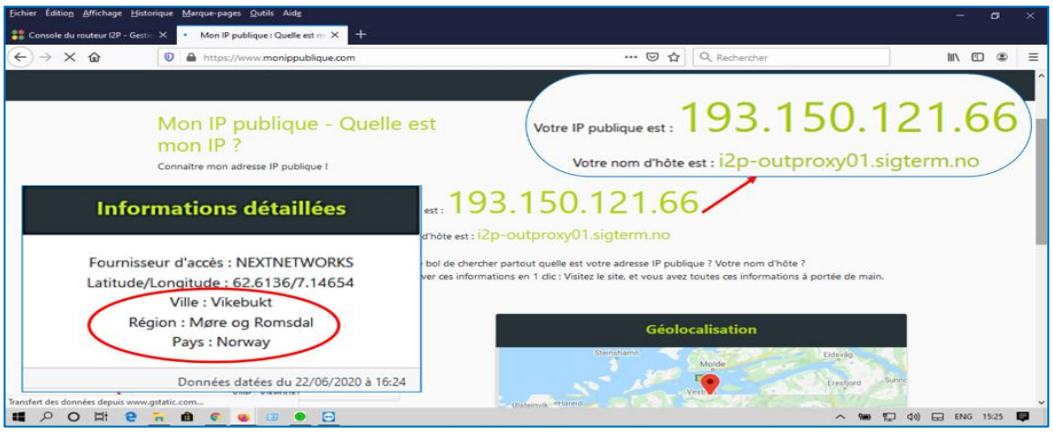

Figure III.20 : *L'adresse IP publique du purokishi.i2p.*

Nous remarquons que l'adresse IP publique détecter dans le navigateur Firefox « 193.150.121.66 » est différente de celle détecté dans navigateur Chrome « 41.105.212.212 », cela signifie que le réseau I2P n'utilise pas l'adresse publique fournis par FAI « Algérie-télécom » mais il utilise l'adresse publique du « purokishi.i2p » fournis par le FAI « *NEXTNETWORKS* » pays « *Norway* », ce qui a permis à I2P de contourner le filtrage des sites Web effectué par le serveur et d'accéder à ces derniers. Et I2P utilise l'adresse locale « 127.0.0.1 » configuré dans le proxy du navigateur Firefox pour naviguer sur les sites interne d'I2P « sites.i2p ».

III.5 Réseau Tor

III.5.1 Lancement du navigateur Tor

Pour lancer le navigateur Tor, nous cliquons sur l'icône « Start Tor Browser », une fenêtre s'ouvre affichant une barre verte qui illustre la connexion du Navigateur Tor au réseau Tor.

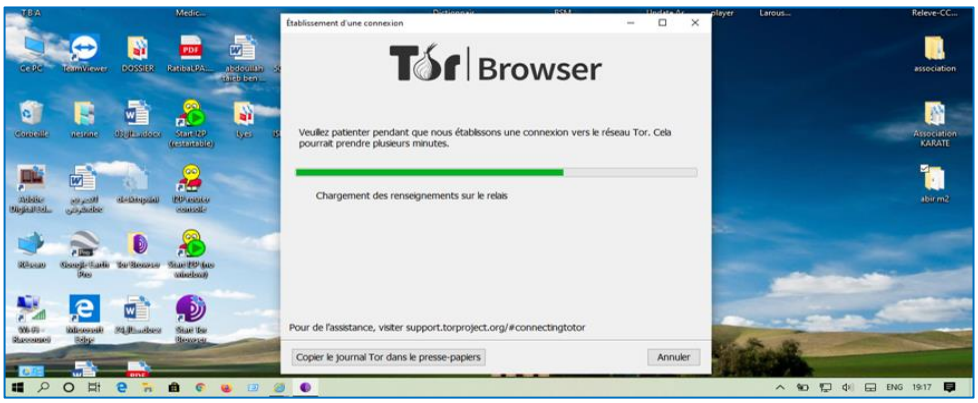

Figure III.21 : *Lancement du navigateur Tor.*

Après le premier démarrage du navigateur Tor qui pourrait être long, la page d'accueil du navigateur Tor s'ouvrira.

III.5.2 Circuit Tor

Lorsque nous cliquons sur l'oignon en haut à gauche pour consulter un tutoriel sur le fonctionnement de Tor. Ce dernier explique le fonctionnement des circuits de Tor et comment ils nous permettent de naviguer en ligne tout en protégeant les informations.

Les circuits sont composés de relais attribués au hasard. Ce sont des ordinateurs disséminés dans le monde entier, configurés pour acheminer le trafic de Tor.

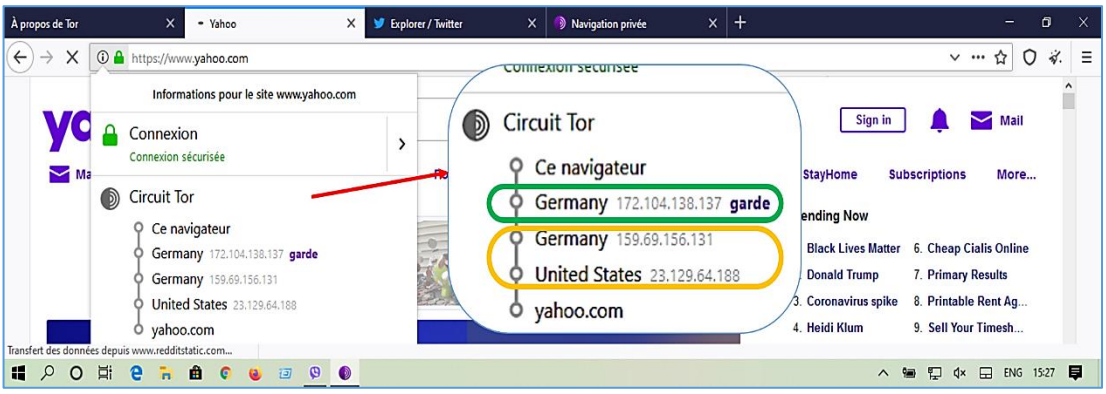

Figure III.22 : *Circuit Tor pour le site Web Yahoo.*

Les adresses mentionnées dans la figure ci-dessus sont les adresses des relais qui composent le circuit du site Web « https://www.yahoo.com/».

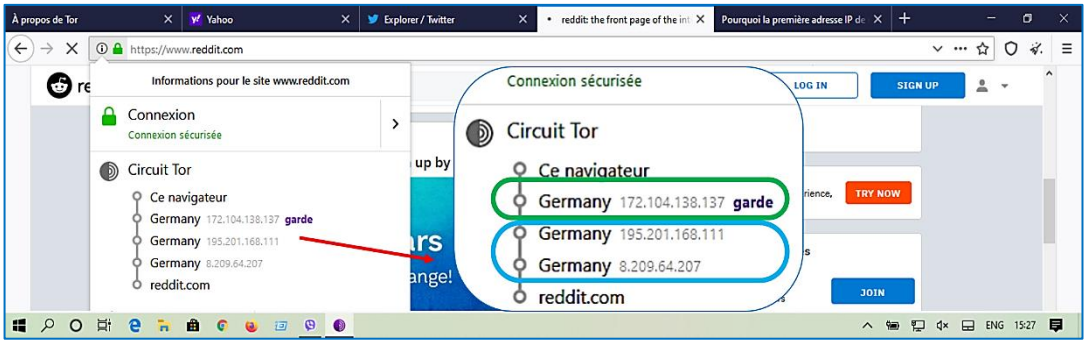

Figure III.23 : *Circuit Tor pour le site Web Reddit.*

Les adresses mentionnées dans la figure ci-dessus sont les adresses des relais qui composent le circuit du site Web « https://www.reddit.com/».

Nous remarquons que l'adresse du relais **garde** « 172.104.138.137 » du circuit Tor attribué pour le site Web Yahoo est la même que celle attribué pour le site Web Reddit, par contre le reste des relais qui constituent les deux circuits sont différents pour les deux sites Web. C'est un comportement normal de Tor. Le premier relais du circuit est appelé « garde

d'entrée ». Il s'agit d'un relais rapide et stable qui reste le premier relais du circuit pendant 2 à 3 mois. Le reste du circuit change pour chaque nouveau site Web que nous visitons. Ensemble, ces relais fournissent la protection complète de la vie privée et des données personnelles offerte par Tor.

III.5.3 Accès aux sites interdis via Tor

Comme nous avons déjà mentionné le navigateur Tor est conçu pour la navigation anonyme au Web normal, nous essayerons d'accéder aux sites interdis par le serveur.

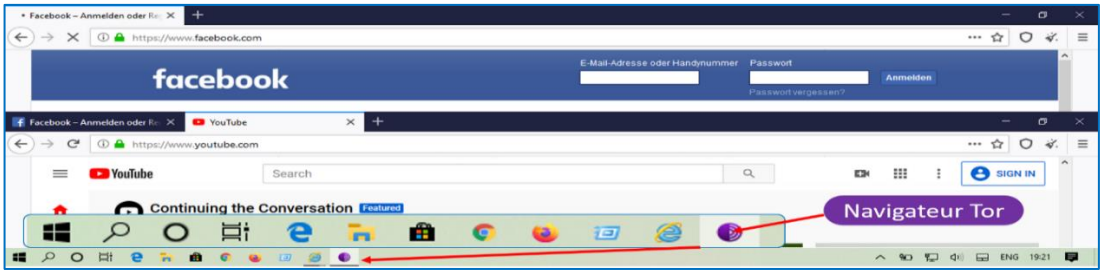

Figure III.24 : *Accès aux sites interdis via Tor.*

Comme nous montre la figure ci-dessus, nous avons pu y accéder aux sites Web interdis par le serveur via le navigateur Tor grâce aux circuits différents utilisé pour chaque sites web consulté.

Nous allons visiter le site d'identification d'adresse https://www.monippublique.com/. Pour savoir l'adresse publique utilisé pour consulté ce dernier.

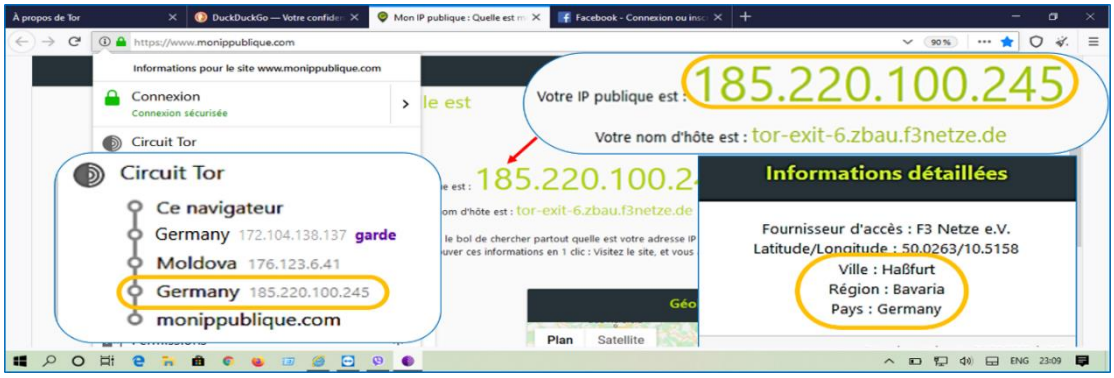

Figure III.25 : *Informations détaillées du site Monippublique.*

Nous remarquons que l'adresse IP publique « 185.220.100.245 » détecté dans le site Monippublique est la même adresse du dernier relais du circuit Tor qui l'a utilisé pour atteindre ce site, ce qui est logique puisque nous avons déjà mentionné que Tor construit un circuit diffèrent pour chaque nouveau site Web consulté.

Nous remarquons aussi que le géolocalisation du site Monippublique affiche « *Germany* » comme pays et c'est le même pays affiché avec le dernier relais du circuit Tor. Avec un FAI « F3 Netze e.V ».

III.6 Tor vs I2P

III.6.1 Comparaison basée sur les tests effectués

Nous allons comparer les deux réseaux Tor et I2P en se basant sur les constatations obtenues durant la navigation vers les sites Web normaux.

- Le démarrage du réseau I2P est plus lent par rapport au démarrage du navigateur Tor cela est dû au temps nécessaire pour que les routeurs du réseau I2P commence à fonctionner.

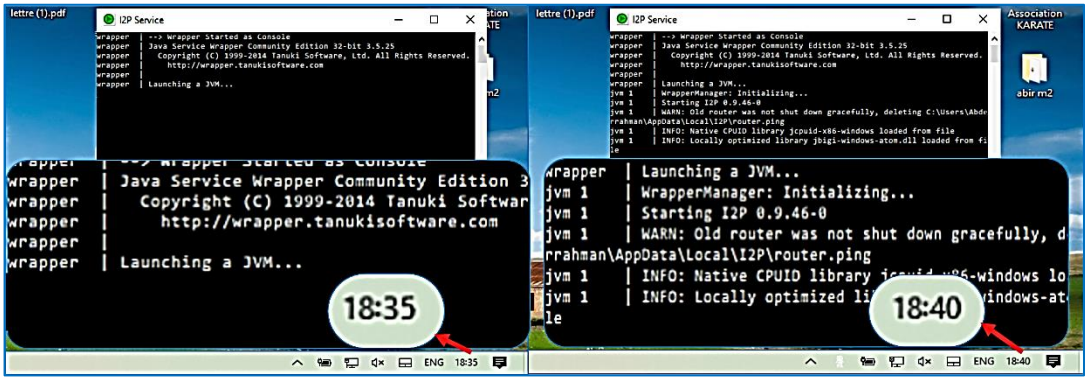

Figure III.26 : *Démarrage des routeurs I2P.*

- Une fois que la page d'accueil du navigateur Tor s'affiche nous pouvons facilement surfer sur les sites Web sans aucune configuration, par contre au réseau I2P il faut attendre le rechargement des I2PTunnel préconfigurés (au minimum deux minutes) pour pouvoir surfer sur les sites.i2p. Pour la navigation anonyme au Web normal nous aurons besoin de configuré le Outproxy d'I2P.

Figure III.27 : *Temps de rechargement des Tunnels I2P.*

- Dans le réseau I2P nous trouvons un seul mandataire sortant « outproxy (récemment ajouté) » qu'il faut configurer pour l'utiliser. En revanche le navigateur Tor emploie un circuit de trois relais pour chaque site consulté.
- Les relais de Tor sont des ordinateurs disséminés dans le monde entier, configurés et toujours prêts pour acheminer le trafic de Tor. Ces derniers sont toujours accessibles, à l'opposé d'I2P qui a un seul mandataire sortant qui n'est pas toujours accessible car il est utilisé par tous les utilisateurs du réseau I2P. ce qui rend l'accès aux sites Web via Tor plus rapide par rapport à I2P.
- Nous pouvons accéder à des sites.onion via I2P grâce à l'outproxy sélectif Purokishi déjà mentionné en haut, par rapport au Tor nous ne pouvons pas y accéder au sites.i2p via Tor.

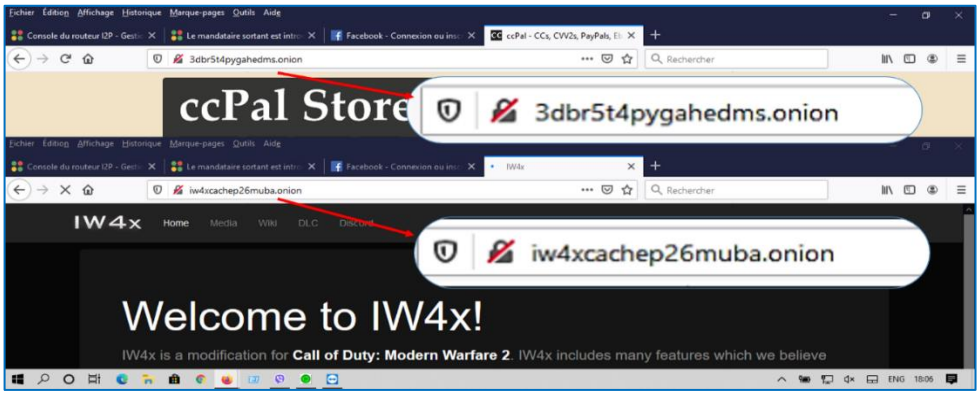

Figure III.28 : *Sites.onion.*

III.6.2 Bande passante

❖ Définition

La bande passante est la capacité de transmission d'une liaison de transmission de données, la bande passante représente la quantité d'informations (en bits/s) qui peut être transmise sur une voie de transmission [33].

❖ Navigateur Tor

Nous allons générer le graphique de bande passante en utilisant Wireshark (un logiciel d'analyse réseau, qui sera décrit au cours de ce chapitre), nous pouvons également déduire à partir du graphique un potentiel encombrement de la bande passante,

Après que nous ayons lancé le navigateur Tor et lancer le téléchargement de l'application Viber (pour Windows de taille 97 Mo). Par défaut Wireshark affiche la bande passante de tous les paquets, nous allons ensuite appliquer un filtre qui correspond au flux capturé lors de l'échange entre le nœud de garde du Tor « 172.104.138.137 » et notre machine client

« 192.168.137.210 » qui a effectué le téléchargement, pour avoir une courbe du téléchargement via Tor durant une heure.

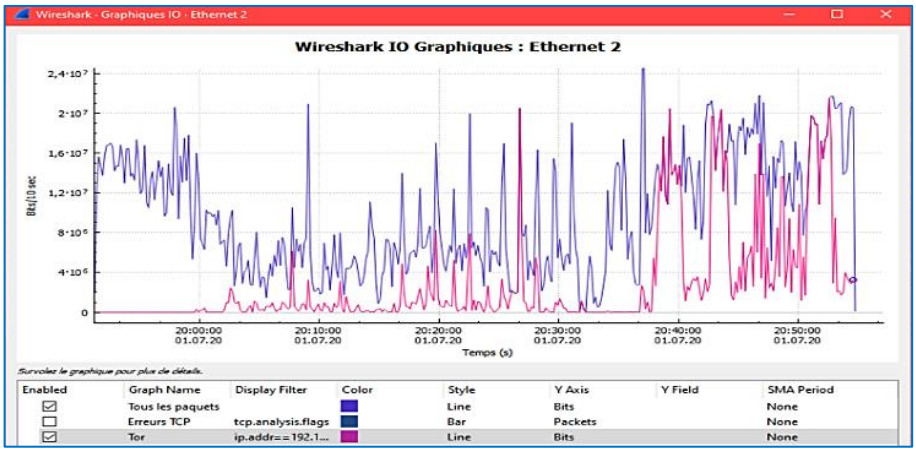

Figure III.29 : *Graphique de la bande passante du Tor avec Wireshark.*

Nous remarquons que dans ce graphique qui représente les statistiques de bande passante en Mbit/s en fonction du temps. Les deux courbes (paquets Tor et tous les paquets) présentent des successions de piques et de descentes qui est typiquement le symptôme d'un engorgement de la bande passante due au téléchargement via Tor en parallèle avec l'utilisation des autres applications ce qui provoque une décrémentation de la courbe de téléchargement Tor.

- **Réseau I2P**

Avant tous nous définissons le pourcentage de bande passante (par défaut : 80%). Ce réglage correspond à la bande passante partagée que nous avons attribuée à I2P. C'est à dire au trafic qui transite par notre machine mais dont nous ne sommes ni la destination, ni l'émetteur. Il faut mentionner que plus nous partageons de bande passante, plus le réseau sera rapide et efficace. Il n'y a aucun serveur de routage à haut débit fourni par l'équipe I2P. Les utilisateurs créent et fournissent eux-mêmes la bande passante du réseau.

Nous allons générer le graphique de bande passante en utilisant Wireshark, après que nous avons lancé les tunnels I2P et démarrer le téléchargement de l'application Viber (pour Windows de taille 97 Mo) en utilisant le outproxy pour y accéder au site du téléchargent. Par défaut Wireshark affiche la bande passante de tous les paquets, nous allons ensuite appliquer un filtre qui correspond au flux de téléchargement capturé lors de l'échange entre « 173.230.128.232 » et notre machine client « 192.168.137.210 » qui a effectué le téléchargement, pour avoir une courbe du téléchargement via I2P.

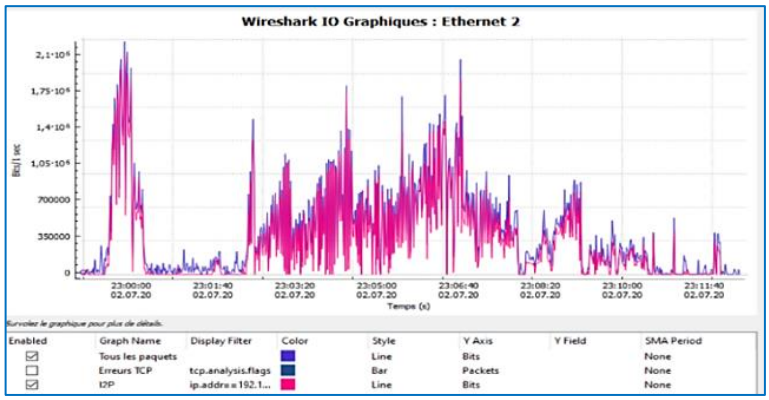

Figure III.30 : *Graphique de la bande passante d'I2P avec Wireshark.*

L'outil Graphiques d'I2P nous a permis de voir les statistiques l'utilisation de la bande passante d'I2P en Ko/s en fonction du temps.

Figure III.31 : *Graphique de la bande passante d'I2P.*

❖ **Comparaison**

Nous remarquons que la consommation de la bande passante lors du téléchargement via i2p est plus grande par rapport à celle du téléchargement via Tor cela revient au nombre de sauts (nœud) utilisé par I2P qui sont 8 nœuds au total (4 entrants, 4 sortants), comparer à ceux du Tor qui sont 3 nœuds au total.

III.6.3 Latence

❖ **Définition**

C'est le temps nécessaire pour véhiculer un paquet au travers d'un réseau. La latence peut être mesurée de plusieurs façons : temps d'aller-retour RTT « *Round-Trip Time* », le temps de retard aller-retour RTD « *Round-Trip Delay* » [33].

❖ **Navigateur Tor**

Nous allons nous intéresser au temps de repense plus particulièrement au RTT nous allons travailler sur la même capture du flux qui concerne le téléchargement d'application Viber via Tor.

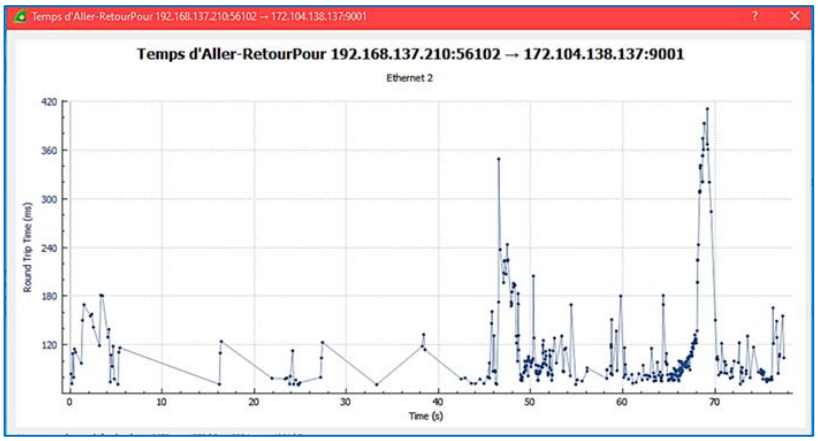

Figure III.32 : *Temps d'aller-retour du flux Tor.*

Cette courbe représente les statistiques de la variation du RTT en fonction d'un intervalle du temps de 80s. Le RTT atteint sa valeur maximale qui est « 380ms » et il a « 20ms » comme valeur minimal, le graphique représente seulement 2 grandes piques du RTT, et le reste est entre « 20ms-180ms », pour une communication donnée plus le RTT est grand plus les requêtes sont lentes.

- **Réseau I2P**

Nous allons nous intéresser au temps de réponse plus particulièrement au RTT nous allons travailler sur la même capture du flux qui concerne le téléchargement d'application Viber via I2P.

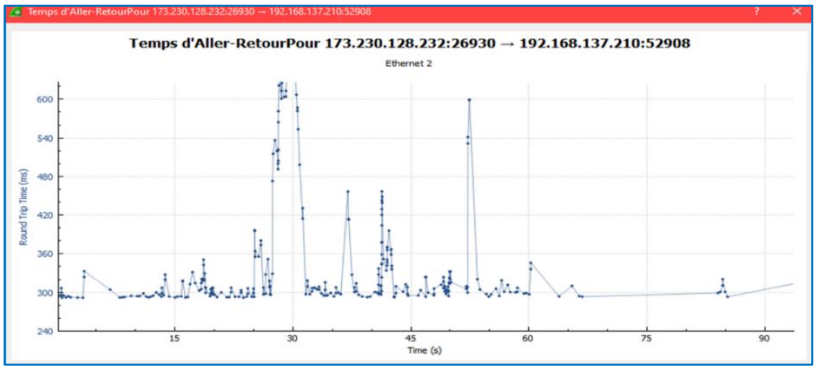

Figure III.33 : *Temps d'aller-retour du flux I2P.*

Cette courbe représente les statistiques de la variation du RTT en fonction d'un intervalle du temps de 90s. Le RTT atteint sa valeur maximale qui est « 600ms » et il a « 290ms » comme valeur minimal, le graphique représente plusieurs piques du RTT entre « 380ms-600ms », et le reste est entre « 290ms-340ms ».

❖ **Comparaison**

Comme nous pouvons le voir, le RTT dans Tor est environ la moitié du RTT dans I2P, ce qui signifie que Tor fonctionne mieux qu'I2P en terme de téléchargement effectué et que les performances du Tor sont meilleures que ceux d'I2P lors de la navigation anonyme au Web normal.

III.7 Extraction des signatures du réseau Tor

III.7.1 Wireshark

Wireshark est un outil open source pour profiler le trafic réseau et analyser les paquets. Un tel outil est souvent appelé analyseur de réseau, analyseur de protocole réseau ou renifleur. Wireshark, anciennement connu sous le nom d'Ethereal, peut être utilisé pour examiner les détails du trafic à différents niveaux, allant des informations au niveau de la connexion aux bits qui composent un seul paquet. La capture de paquets peut fournir à un administrateur réseau des informations sur des paquets individuels tels que l'heure de transmission, la source, la destination, le type de protocole et les données d'en-tête. Ces informations peuvent être utiles pour évaluer les événements de sécurité et résoudre les problèmes de périphérique de sécurité réseau [34].

La fenêtre principale de l'interface de Wireshark est divisée en trois sections :

- La première, en haut, affiche la liste des paquets capturés.
- La deuxième, au milieu, donne des détails sur le paquet sélectionné de la liste du haut.
- La troisième reproduit le contenu en hexadécimal, du même paquet.

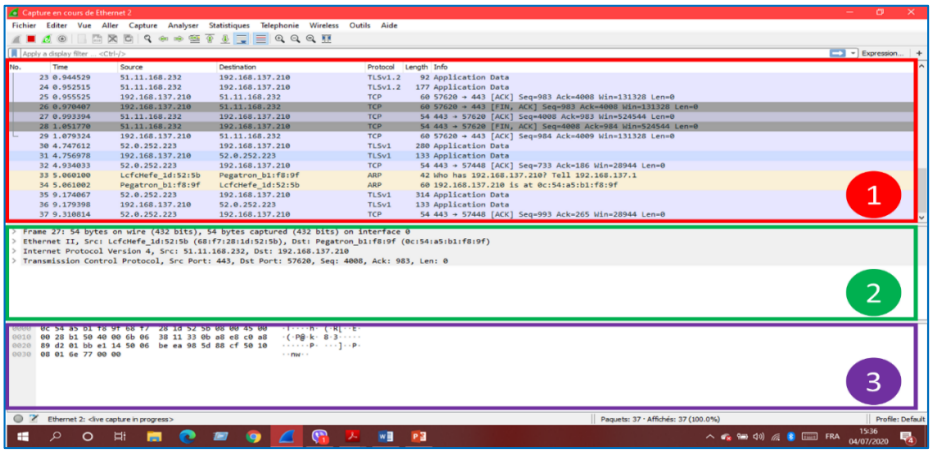

Figure III.34 : *Interface de Wireshark.*

III.7.2 Démarche suivi pour notre analyse

Comme mentionné précédemment, tout le trafic réseau TOR est chiffré à l'aide de TLS, ce dernier nécessite un transport fiable donc il est basé sur le protocole TCP. D'abord le client Tor établit une connexion TCP avec le nœud d'entrée, ensuite, il établit une session TLS. Pour cette raison, nous ne pouvons pas voir à l'intérieur du paquet lorsque le cryptage TLS s'établit. Pour voir les anomalies et les caractéristiques du trafic TOR, nous ne pouvons analyser que son processus d'établissement de connexion TLS.

Nous allons utiliser Wireshark pour analyser les deux processus, le premier est le processus d'établissement de connexion ou généralement appelé « TCP *Three-Way Handshake* » entre les utilisateurs et le réseau TOR, et le second est le processus de création d'une session sécurisée du protocole TLS ou généralement appelé « *TLS Handshake* ».

Nous allons analyser et comparer les différences et les anomalies des paquets TOR avec d'autres paquets ordinaires. Lors de l'établissement d'une connexion « TCP *Three-Way Handshake* » et « *TLS Handshake* ».

III.7.3 Analyse du trafic Web des différents navigateurs

III.7.3.A Etablissement d'une connexion « TCP *Three-Way Handshake* »

Comme son nom l'indique, le Three-Way Handshake se déroule en trois étapes :

1. SYN : Le client va envoyer un premier paquet SYN (*synchronized*) au serveur.
2. SYN-ACK : Le serveur va répondre au client à l'aide d'un paquet SYN-ACK (*synchronize*, *acknowledge*).
3. ACK : Pour terminer, le client va envoyer un paquet ACK au serveur qui va servir d'accusé de réception,

Maintenant, nous allons analyser et comparer les paquets TCP (SYN, SYN-ACK, ACK) résultant de l'établissement d'une connexion en trois étapes entre :

- Navigateur Tor et le nœud d'entrée, son adresse IP est « 172.104.138.137 ».
- Navigateur Firefox et le site « https://openclassrooms.com/ », son adresse IP est « 104.22.65.200 ».
- Navigateur Chrome et le site « https://openclassrooms.com/ », son adresse IP est « 104.22.65.200 ».

Nous allons analyser le premier paquet d'une connexion TCP entre le navigateur Tor et le nœud d'entrée, nous sélectionnons le paquet 94 qui correspond au SYN.

Cela permet de visualiser les différentes couches d'encapsulation de ce dernier dans la deuxième section du Wireshark.

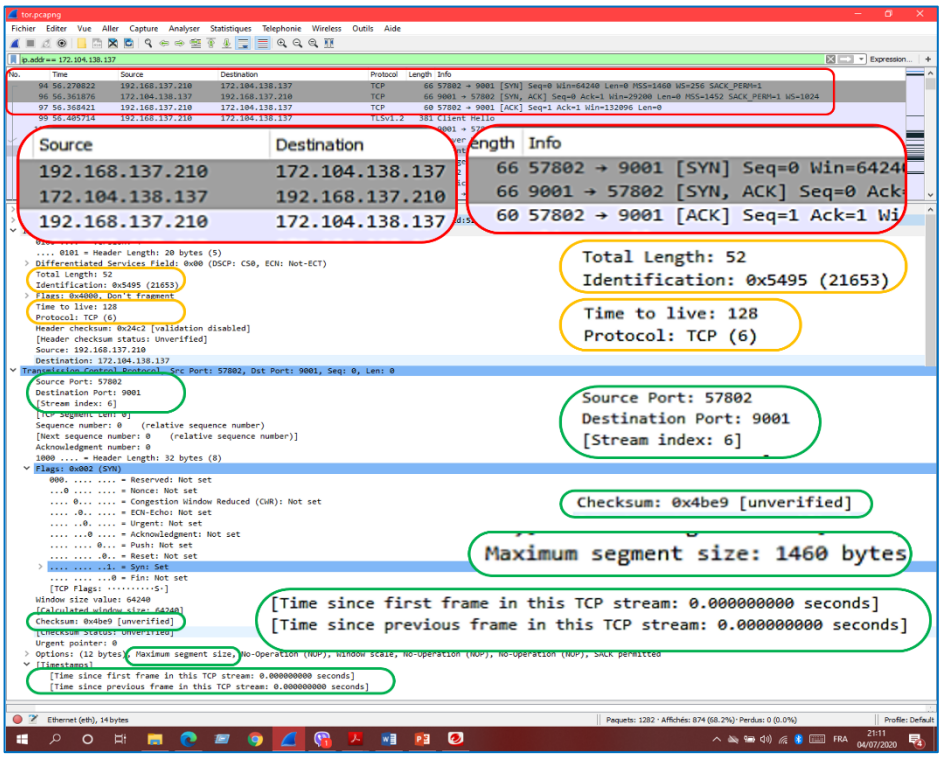

Figure III.35 : *Paquet SYN d'une connexion TCP avec le nœud Tor.*

A partir de ces différentes couches d'encapsulation nous pouvons extraire les informations suivantes [35] :

- **Les adresses IP source et destination, numéro de port source et destination, le type d'adressage et le protocole utilisé.**
- **Identification** : constitue l'identification utilisée pour reconstituer les différents fragments. Chaque fragment possède le même numéro d'identification, les entêtes IP des fragments sont identiques à l'exception des champs longueur totale, Checksum et Position fragment.
- **TTL** (*Time To Live*) : indique la durée de vie maximale du paquet. Il représente la durée de vie en seconde du paquet. Si le TTL arrive à 0, alors l'équipement qui possède le paquet, le détruira.
- **Protocole** : Le champ Protocole est codé sur 8 bits et représente le type de Data qui se trouve derrière l'entête IP « 06 – 00110 – TCP ».
- **Stream Index** : L'index de flux est un mappage Wireshark interne qui identifie un flux TCP unique. Tous les paquets pour le même flux TCP doivent avoir les mêmes valeurs dans ce champ.

- **Checksum** : Le champ Checksum représente la validité du paquet de la couche 4 TCP.
- **Sequence Number** : Numéro de séquence cette valeur fournit le numéro de séquence du paquet.
- **MSS (***Maximum Segment Size***)** : la taille maximale du segment des données applicatives que l'émetteur accepte de recevoir. Au moment de l'établissement d'une connexion (paquet comportant le flag SYN), chaque partie annonce sa taille de MSS. Ce n'est pas une négociation. Pour l'Ethernet la valeur est 1460.
- **Windows size value** : Valeur de la taille de la fenêtre indique la taille de la fenêtre (taille du tampon) sur l'ordinateur source afin que le destinataire puisse déterminer le nombre de paquets pouvant être envoyés à un moment donné.
- **Analyse SEQ/ACK** : RTT to ACK (*Round Time Trip to Acknowledgement*) cette valeur indique que le paquet est une réponse ACK à une autre trame. La valeur est fournie par Wireshark et ne fait pas partie de l'en-tête réel. Si le paquet est le premier d'une session TCP, ce champ n'apparaît pas.

Les informations du paquet SYN mentionnées ci-dessus seront classés dans un tableau avec les autres paquets SYN des deux autres navigateurs, puis nous allons aborder la même procédure avec les paquets SYN-ACK et ACK.

- **Paquet TCP (SYN) :**

	Tor avec « 172.104.38.137 »	Chrome avec « 104.22.65.200 ».	Firefox avec « 104.22.65.200 ».
Type	IPv4 (0x0800)	IPv4 (0x0800)	IPv4 (0x0800)
Protocole	TCP (6)	TCP (6)	TCP (6)
Port	57802	58573	58737
Port	9001	443	443
MSS (octets)	1460	1460	1460
La langueur totale (IP)	52	52	52
Identification	0x5495 (21653)	0xff3e (65342)	0xff59 (65369)
TTL	128	128	128
Numéro de séquence	0	0	0
Stream index	6	28	16
Flags	0x002 (SYN)	0x002 (SYN)	0x002 (SYN)
Windows size value	64240	64240	64240

Tableau III.2 : *Paquet SYN de l'établissement d'une connexion « TCP Three-Way Handshake ».*

❖ **Paquet TCP (SYN-ACK) :**

	Tor avec « 172.104.38.137 »	Chrome avec « 104.22.65.200 ».	Firefox avec « 104.22.65.200 ».
Type	IPv4 (0x0800)	IPv4 (0x0800)	IPv4 (0x0800)
Protocole	TCP (6)	TCP (6)	TCP (6)
Port	9001	443	443
Port	57802	58573	58737
MSS	1452	1400	1400
La langueur totale (IP)	52	52	52
Identification	0x0000 (0)	0x0000 (0)	0x0000 (0)
TTL	49	51	51
Numéro de séquence	0	0	0
Stream index	6	28	16
Flags	0x012 (SYN, ACK)	0x012 (SYN, ACK)	0x012 (SYN, ACK)
Windows size value	29200	65535	65535
Analyse SEQ/ACK			
RTT to ACK	0.097599000	0.052884000	0.052235000

Tableau III.3 : *Paquet SYN-ACK de l'établissement d'une connexion « TCP Three-Way Handshake ».*

❖ **Paquet TCP (ACK) :**

	Tor avec « 172.104.38.137 »	Chrome avec « 104.22.65.200 ».	Firefox avec « 104.22.65.200 ».
Type	IPv4 (0x0800)	IPv4 (0x0800)	IPv4 (0x0800)
Protocole	TCP (6)	TCP (6)	TCP (6)
Port	57802	58573	58737
Port	9001	443	443
MSS			
La langueur totale (IP)	40	40	40
Identification	0x5496 (21654)	0xff3f (65343)	0xff5a (65370)
TTL	128	128	128
Numéro de séquence	1	1	1
Stream index	6	28	16
Flags	0x010 (ACK)	0x010 (ACK)	0x010 (ACK)
Windows size value	516	514	514
Analyse SEQ/ACK			
RTT to ACK (s)	0.006545000	0.001366000	0.000707000

Tableau III.4 : *Paquet ACK de l'établissement d'une connexion « TCP Three-Way Handshake ».*

❖ **Constatation**

D'après l'analyse effectuée nous avons pu trouver les différences suivantes :

1. Les ports sources : ces ports sont >1024, donc ce sont des ports dynamiques attribués d'une manière différente et aléatoire pour chaque application.
2. Les ports destinations : le navigateur Chrome et Firefox utilise le port 443 qui est un port dédié au protocole HTTPS (http+SSL), par contre le nœud d'entrée Tor utilise le port 9001.
3. Identification : puisque chaque connexion est identifiée d'une manière unique, ce champ change pour chaque paquet.
4. RTT to ACK : ce champ varie d'une connexion à l'autre.

Lorsque l'en-tête des paquets TCP dans l'établissement de la connexion TOR et l'établissement de la connexion normale a été comparé, aucune anomalie n'a été observée. Tor effectue une connexion TCP « *Three-Way Handshake* » avec les indicateurs « SYN, SYN - ACK et ACK » comme les autres connexions normales.

Maintenant nous allons procéder à l'analyse de la connexion TLS Handshake en se basant sur les messages « Client Hello, Server Hello, Certificate, Server Hello Done, Client Key Exchange, Change Cipher Spec ».

III.7.3.B Etablissement d'une connexion « TLS *Handshake* »

TLS est un protocole de cryptage conçu pour sécuriser les communications internet. TLS Handshake est le processus qui démarre une session de communication qui utilise le cryptage TLS. Au cours du TLS Handshake, les deux parties communicantes échangent des messages pour se reconnaître, se vérifier, établir les algorithmes de chiffrement qu'elles utiliseront et se mettre d'accord sur les clés de session. Le TLS Handshake est une partie fondamentale du fonctionnement de HTTPS [36].

TLS Handshake [37] **:**

1. Le premier message est appelé « ClientHello ». Ce message répertorie les capacités du client afin que le serveur puisse choisir la suite de chiffrement (*the cipher suite*) que les deux utiliseront pour communiquer. Il comprend également un grand nombre premier choisi au hasard appelé « client aléatoire (*client random*) ».
2. Le serveur répond avec un message "SeverHello". Dans ce message, il indique au client les paramètres de connexion qu'il a sélectionnés dans la liste fournie et renvoie

son propre nombre premier sélectionné au hasard appelé « serveur aléatoire (*server random*) ».

3. Dans le message « Certificate », le serveur envoie sa chaîne de certificats au client. Pour fournir une authentification à la connexion, un certificat SSL est signé par une autorité de certification, ce qui permet au client de vérifier que le certificat est légitime. A la réception, le client effectue plusieurs vérifications pour authentifier le certificat. Cela inclut la vérification de la signature numérique du certificat, le client s'assurera également que le serveur est en possession de la clé privée du certificat. Cela se fait pendant le processus d'échange / génération de clés (*the key exchange/generation process*)
4. Il s'agit d'un message facultatif, uniquement nécessaire pour certaines méthodes d'échange de clés qui nécessitent que le serveur fournisse des données supplémentaires.
5. Le message « Server Hello Done » indique au client qu'il a envoyé tous ses messages.
6. Le client fournit ensuite sa contribution à la clé de session. Les spécificités de cette étape dépendent de la méthode d'échange de clés qui a été décidée dans les premiers messages « Hello ». Dans cet exemple, nous examinons RSA[9]. Le client va donc générer une chaîne d'octets aléatoire appelée secret pré-maître (*pre-master secret*), puis la crypter avec la clé publique du serveur et la transmettre.
7. Le message « Change Cipher Spec » indique à l'autre partie qu'il a générée la clé de session et va passer à une communication cryptée.
8. Le message « Finished » est ensuite envoyé pour indiquer que le Handshake est terminée côté client. Le message « Finished » est crypté et constitue les premières données protégées par la clé de session.
9. C’est maintenant au tour du serveur de faire de même. Il déchiffre le secret pré-maître (*pre-master secret*), et calcule la clé de session. Il envoie ensuite son message « Change Cipher Spec » pour indiquer qu'il passe à une communication cryptée.
10. Le serveur envoie son message « Finished » en utilisant la clé de session symétrique qu'il vient de générer.

[9] **RSA :** Rivest, Shamir, et Adelman.

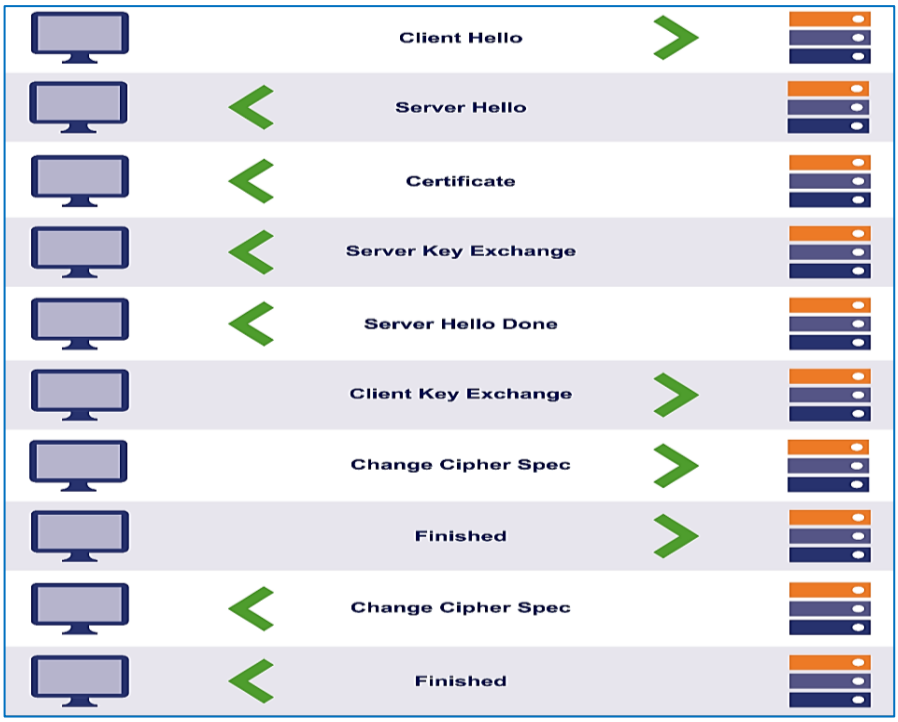

Figure III.36 : *TLS Handshake [37].*

Nous allons filtrer les paquets d'une connexion TLS Handshake entre le navigateur Tor et le nœud de garde Tor pour les analyser et les comparer avec les paquets d'une connexion TLS Handshake établie lors de l'accès aux sites Web mentionné ci-dessous via le navigateur Chrome et Firefox.

- Navigateur Tor et le nœud d'entrée « garde », son adresse IP « 172.104.138.137 ».
- Navigateur Chrome et les deux sites Web « https://reddit.com/» son adresse IP est « 151.101.1.140 », et « https://amazon.com/ » son adresse IP est « 151.101.1.16 ».
- Navigateur Firefox et les deux sites Web « https://reddit.com/» son adresse IP est « 151.101.1.140 », et « https://amazon.com/ » son adresse IP est « 151.101.1.16 ».

Dans Wireshark nous avons utilisé le filtre « **ip.addr == <adresse de serveur destination> && tls.handshake** » pour isoler les paquets de la connexion TLS Handshake entre le navigateur Tor et le nœud d'entrée, et entre les navigateurs Chrome et Firefox et les sites Web déjà mentionnés :

Nous nous intéressons aux paquets TLS Handshake présentés dans la figure ci-dessous.

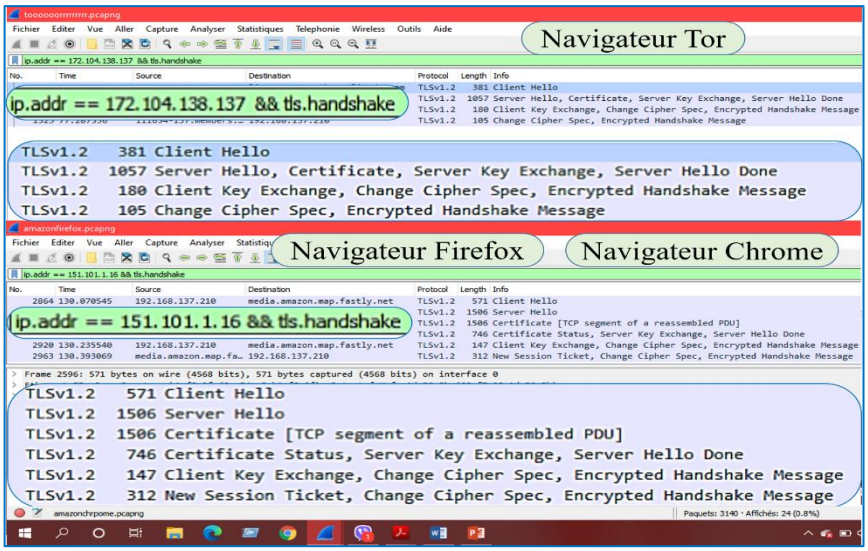

Figure III.37 : *TLS Handshake pour les trois navigateurs.*

a. Message « Client Hello »

Client Hello est le premier message envoyé par le client pour démarrer ou initier une connexion TLS Handshake. La structure des messages Client Hello est décrite ci-dessous [38] :

1. Handshake Type : type du message Handshake expédié.
2. Length : longueur du message Handshake.
3. Version : version TLS que le client souhaite / propose utiliser.
4. Random : une information se compose de temps et de 28 octets aléatoires.
5. Session ID : ID de la dernière session que le client peut et veut utiliser, de sorte qu'il n'a pas à créer une nouvelle session.
6. Cipher Suites : Liste des algorithmes de chiffrement proposés par le client.
7. Compression Method : méthode de compression qui peut être utilisée par le client pour réduire l'utilisation de la bande passante.
8. Extension : se compose d'une extension ou d'informations supplémentaires envoyées pour le serveur. Le protocole TLS a tellement d'extensions qui peuvent être utilisées, l'une des extensions importantes est "nom_serveur" qui permet de donner des informations sur le nom du serveur, nom d'hôte visité par le client.

Nous allons comparer les champs des trois messages « Client hello » des navigateurs sauf les champs « suites de chiffrements, Méthode de compression et Extension » ces derniers seront comparés par la suite.

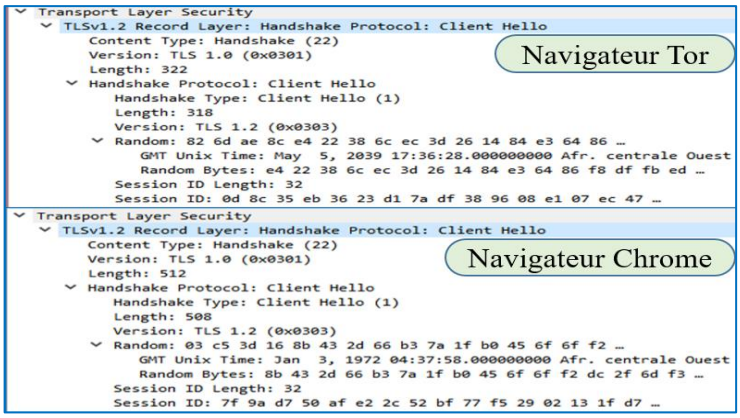

Figure III.38 : *Première partie du message « client hello » pour les deux navigateurs.*

Nous remarquons qu'il n y'a aucune différence entre les trois navigateurs dans les champs présentés dans la figure ci-dessus sauf au niveau du champ de la langueur totale (Length) qui est fixe pour le navigateur Chrome et Firefox (Length =512 octets), et que cette dernière est petite (Length = 322 octets) pour le navigateur Tor, ainsi le temps mentionnés dans le champ Random est totalement déréglé pour les trois navigateurs.

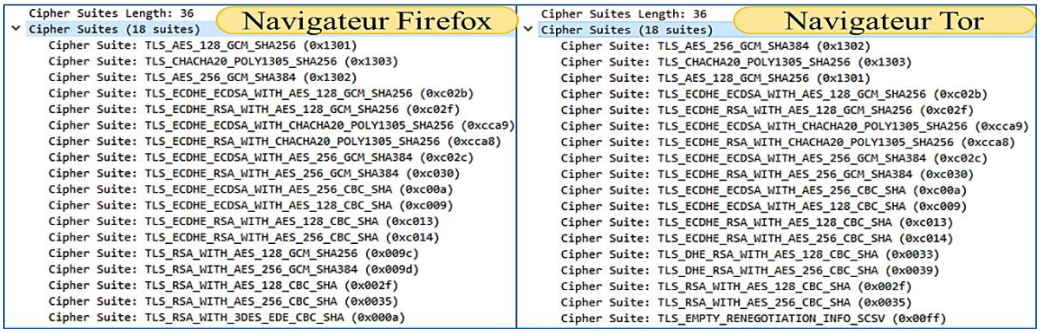

Figure III.39 : *Suites de chiffrement proposées par le navigateur Tor et Firefox.*

Le navigateur Tor et Firefox proposent 18 suites de chiffrement, par contre le navigateur Chrome propose 16 suites de chiffrement ; les suites de chiffrement proposées par les trois navigateurs sont fixes et dans un ordre unique pour chaque navigateur.

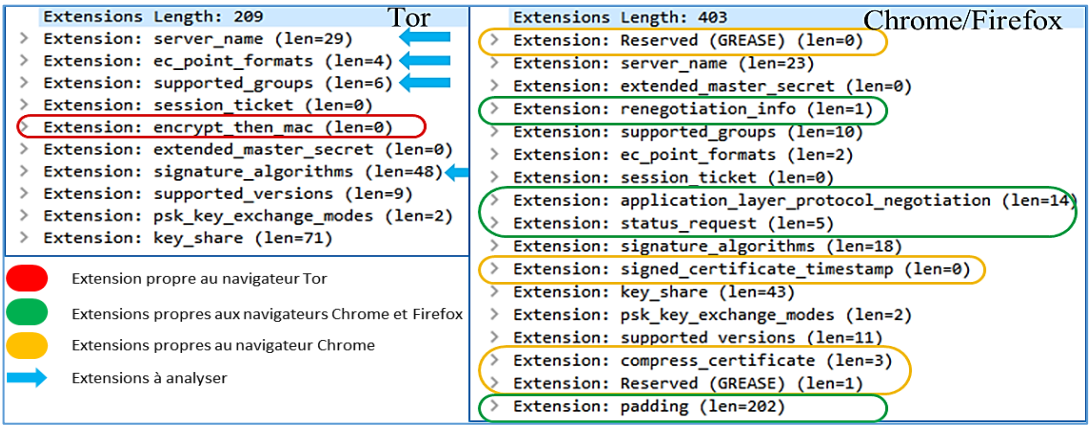

Figure III.40 : *Extensions des trois navigateurs.*

Le navigateur Tor utilise 10 extensions, par contre le navigateur Chrome utilise 17 extensions, et le navigateur Firefox utilise 14 extensions. On trouve des extensions communes entres Chrome, Firefox et Tor, et d'autres extensions différentes entre ces derniers, et une extension unique utilisé par le navigateur Tor. Les extensions qui nous intéressent sont indiquées par une flèche dans la figure III.40 et l'extension utilisée seulement par navigateur Tor est encadrée en rouge.

1. **Extension « server_name »** : Elle permet au client de préciser le nom de domaine du serveur qu'il souhaite joindre. Dans cette extension on trouve le nom de domaine de serveur web de destination [39].

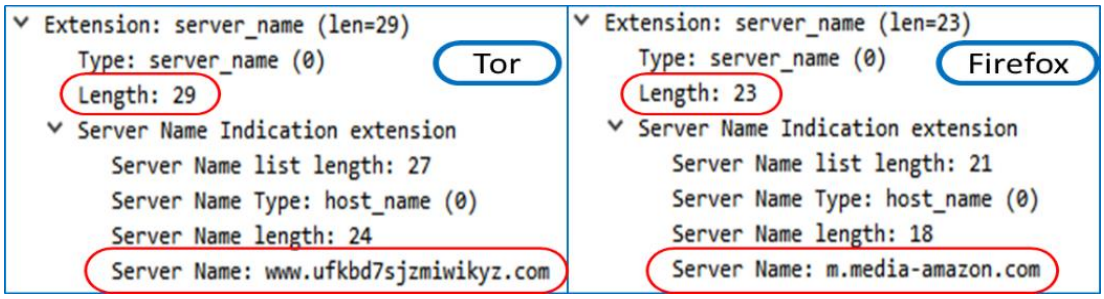

Figure III.41 : *Extension « server_name ».*

Le nom du domaine désigné par Tor a un format toujours le même : « www.Chaîne-aléatoire.com », qui est générer de façon aléatoire. Ce nom de domaine n'existe pas.

```
C:\Users\Abir>nslookup www.ufkbd7sjzmiwikyz.com
Serveur :   DSL.2750U
Address:  192.168.1.1

*** DSL.2750U ne parvient pas à trouver www.ufkbd7sjzmiwikyz.com : Non-existent
domain
```

Figure III.42 : *Nom de domaine généré par navigateur Tor.*

2. **Extension « ec_point_formats »** : cette extension signale les formats de point de courbe elliptique[10] pris en charge par le client ou le serveur (s'il en existe). Il est en effet possible de représenter les points de courbe elliptique sous une forme compressée. En l'absence de cette extension, il est attendu que les coordonnées de points soient transmises dans leur totalité [39].

[10] La cryptographie aux courbes elliptiques regroupe un ensemble de techniques cryptographiques qui utilisent une ou plusieurs propriétés des courbes elliptiques L'usage des courbes elliptiques permet d'améliorer les primitives cryptographiques existantes, par exemple en réduisant la taille des clés cryptographiques,

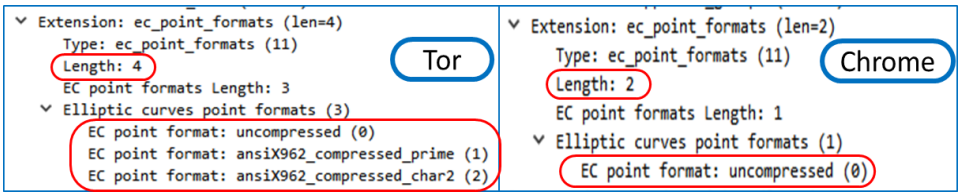

Figure III.43 : *Extensions « ec_point_formats »*

Le navigateur Tor propose trois formats fixes de courbes elliptiques, par contre le navigateur Chrome et Firefox proposent un seul format de courbes elliptiques.

3. **Extension « Supported_groups »** : cette extension informe le serveur des courbes elliptiques prises en charge par le client (s'il en existe), sa prise en charge et son usage sont obligatoires dès lors que le client ou le serveur souhaitent exploiter des fonctions ECC[11] [39].

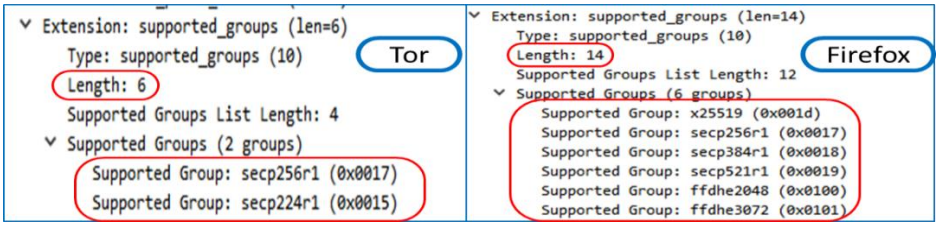

Figure III.44 : *Extensions « Supported_groups ».*

Le navigateur Tor supporte deux groupes qui restent fixes, par contre le navigateur Chrome supporte 4 groupes et le navigateur Firefox supporte 6 groupes.

4. **Extension « signature_algorithms »** : cette extension signale les algorithmes de signature pris en charge pour vérifier l'authenticité de futurs messages du Handshake, notamment le « Server Key Exchange ». Dans le cadre recommandé d'une session TLSv1.2 avec échange de clé authentifié, sa prise en charge et son usage sont obligatoires par le client et le serveur [39].

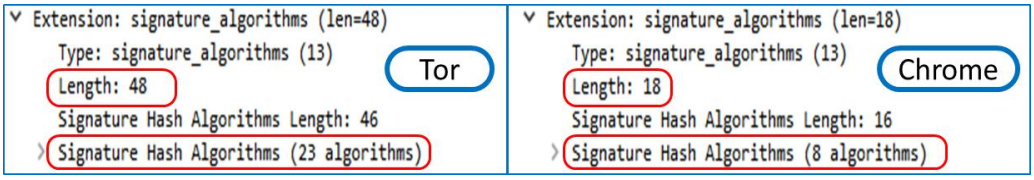

Figure III.45 : *Extensions « signature_algorithms».*

Le navigateur Tor propose 23 algorithmes de signatures qui sont fixes, par contre le navigateur Chrome propose 8 algorithmes et le navigateur Firefox propose 11 algorithmes de signatures.

[11] **ECC :** Elliptic Curve Cryptography.

5. **Extension « encrypt_then_mac »** : lors de la connexion, le client inclut l'extension encrypt_then_mac[12] dans son ClientHello s'il souhaite utiliser encrypt_then_mac. Si le serveur est capable de répondre à cette exigence, il répond par un encrypt_then_mac dans son SeverHello. La valeur "extension type" pour cette extension devra être 22 (0x16) [40].

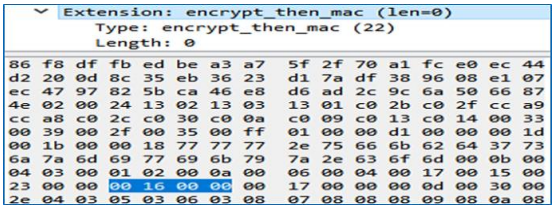

Figure III.46 : *Extension « encrypt_then_mac ».*

Cette extension est utilisée seulement par le navigateur Tor.

b. Message « Server Hello »

Après avoir obtenu ClientHello, le serveur fournira des réponses en envoyant un message SeverHello. Comme celle de l'illustration de la négociation TLS. Le cas du navigateur Tor la remise du message SeverHello est également accompagnée d'un certificat et d'une remise ServerHelloDone dans, la structure du message SeverHello est représenté dans la figure ci-dessous : (Handshake Type, Length, Random Session ID, Cipher Suites, Compression, Extension) [38].

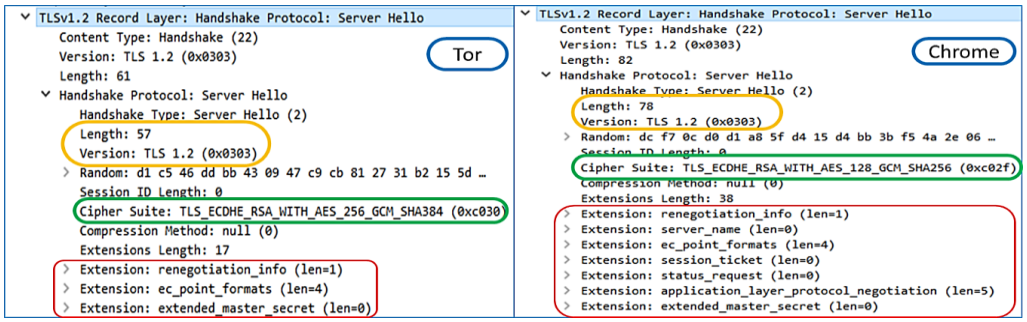

Figure III.47 : *Message « SeverHello ».*

Les suites de chiffrement choisi par les serveurs sont mentionnées dans le tableau suivant :

Serveur destination	Suite de chiffrement choisi
Nœud d'entrée Tor	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
www.amazon.com	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
www.reddit.com	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

Tableau III.5 : *Suites de chiffrements des serveurs destination.*

[12] **MAC :** Message Authentication Code.

Le nœud d'entrée Tor a choisi une suite parmi les suites proposé dans le message ClientHello, « TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 » sous le code 0xC030, qui représente un mécanisme d'échange de clé, qui précise un algorithme d'échange « ECDHE[13] » et éventuellement l'algorithme de signature « RSA » utilisé pour authentifier les échanges. Et un mécanisme assurant la confidentialité et l'intégrité des données échangées après le Handshake : « AES_256_GCM[14] » (avec une taille de clé de 256 bits). Et le « [15]SHA-384 » utilisée pour la dérivation des secrets à partir du pre master secret [39].

- L'ECDHE est l'algorithme utilisé pour l'échange de clés. Il permet de générer des paires de clés asymétriques temporaires pour l'échange de clé de session.
- L'algorithme à clé symétrique AES en mode GCM (*Galois Counter Mode*) est utilisé pour générer des clés de session de 256 bits.

Nœud d'entrée Tor utilise trois extensions dans le message SeverHello, par contre les autres serveurs web utilisent 7 extensions ou plus. On trouve que les trois extensions utilisées par le nœud de garde Tor sont les mêmes utilisées par les autres serveurs sans aucune différence.

c. Message « Certificate »

Dans le message « Certificate », le serveur envoie sa chaîne de certificats au client. Pour fournir une authentification à la connexion, La structure du message Certificate est la suivante (Version, Serial Number, Signature Algorithm, Extension) [38] :

- Issuer : nom / informations sur qui a émis et vérifié les informations dans le certificat.
- Validity : plage de dates de validité du certificat.
- Subject : Objet du serveur de destination (le propriétaire du certificat).
- Subject Public Key Info : se composent de la clé publique et de l'algorithme utilisé dans le chiffrement de cette clé publique.

[13] **ECDHE :** Elliptic Curve Diffie–Hellman Ephemeral.
[14] **GCM :** Galois Counter Mode.
[15] **SHA :** Secure Hash Algorithm.

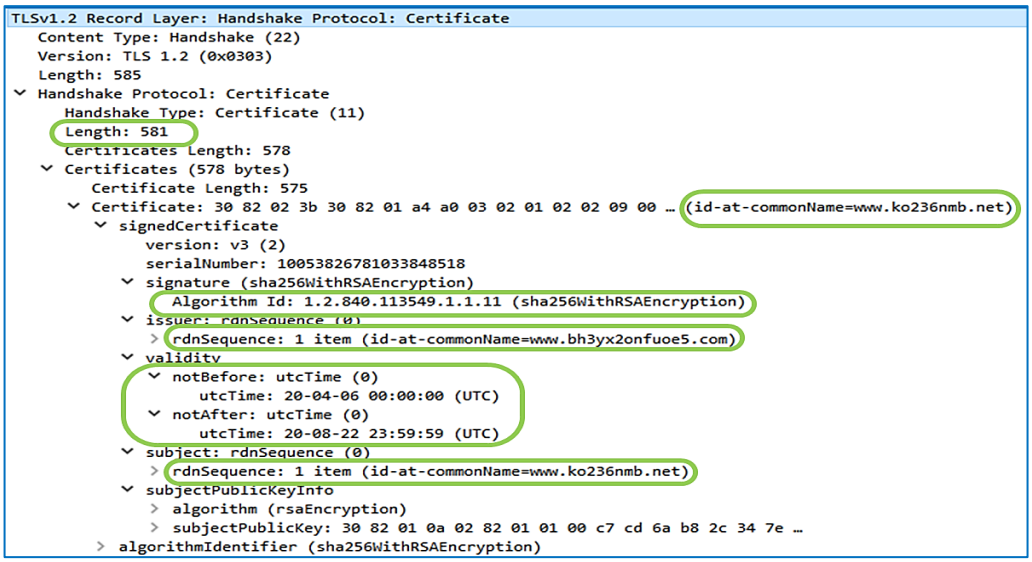

Figure III.48 : *Message « Certificate » envoyé par le nœud d'entrée Tor.*

Figure III.49 : *Message « Certificate » envoyé par le serveur Reddit.*

Tor utilise des certificats un peu différemment des services normaux, ce qui pourrait révéler que l'utilisateur exécute Tor :

- La langueur totale du message « Certificate » envoyé par le nœud d'entrée Tor est plus petite « 587 octets » par rapport à la langueur de message « Certificate » envoyé par les autres serveurs de destinations « généralement entre 2000 et 4000 octets ».
- Tout comme les messages « ClientHello », un message « Certificate » peut avoir également des extensions. Le nombre d'extensions dépend de l'utilisation du certificat. Le certificat que le nœud d'entrée Tor donne à l'utilisateur n'a aucune extension tandis que tous les sites Web que nous avons observés ont trois extensions ou plus.

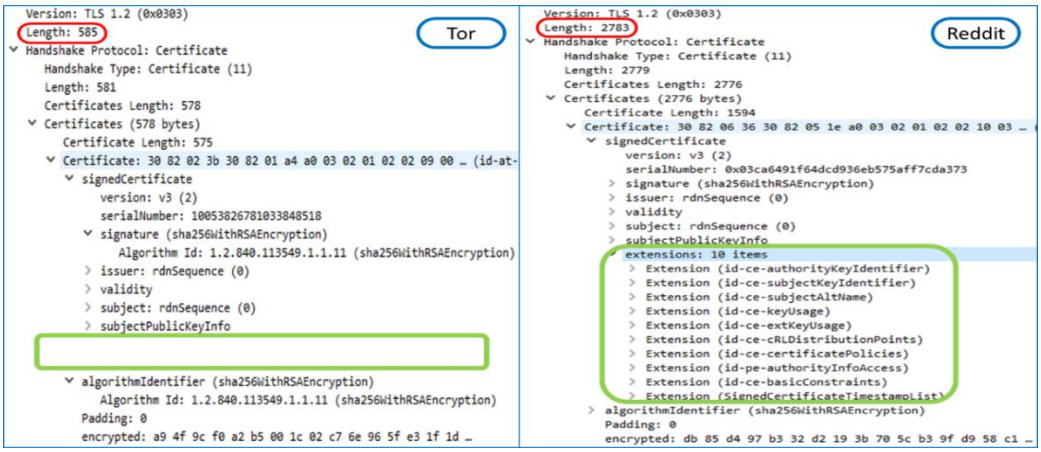

Figure III.50 : *Extensions du message « Certificate ».*

- Le nom commun est une partie importante du certificat car il décrit l'URI que le certificat authentifie. Il s'agit généralement de l'URI sur lequel réside le service. Un nom lisible par l'homme est presque toujours utilisé comme URI et aussi comme nom commun [41]. Tor utilise un certificat auto-signé et un nom commun qui ne résout aucun site. Le nom commun est une chaîne de lettres aléatoire qui commence par "www" et se termine par ".net" dans le certificat de Tor « www.ko236nmb.net ». Lorsque Tor utilise une chaîne aléatoire, cette dernière n'est pas conforme aux autres services que nous avons testés « *.redditmedia.com ».

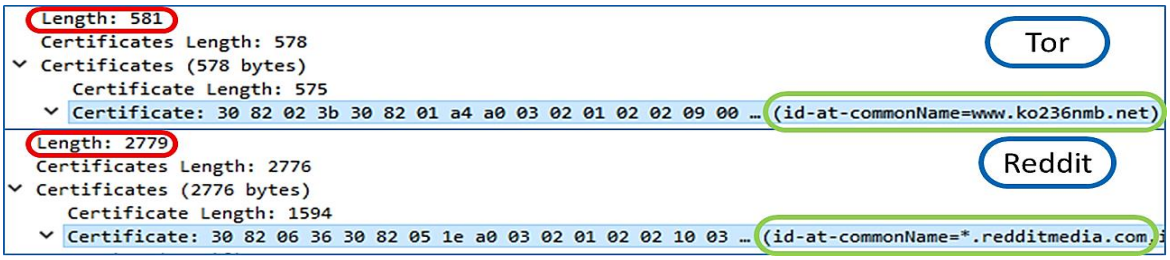

Figure III.51 : *Champ « Common Name ».*

- Sur Internet, un certificat est utilisé pour authentifier qu'une clé publique appartient à une identité. Ceci est obtenu en utilisant une signature numérique pour signer le certificat qui contient la clé publique et l'identité. Un certificat est généralement lié à une identité avec le nom commun, tel que "www.google.com". Le champ nom commun (*Common Name*) (CN) du certificat doit correspondre à l'entité qui est authentifiée, généralement un URI[16]. L'autorité de certification racine (CA) est le niveau le plus élevé dans un schéma de chaîne de confiance et utilise un certificat

[16] **URI :** Uniform Resource Identifier, est une courte chaîne de caractères identifiant une ressource sur un réseau physique ou abstraite, et dont la syntaxe respecte une norme d'Internet mise en place pour le World Wide Web.

auto-signé [41]. Le certificat que Tor utilise est auto-signé et le créateur du certificat a également signé sa légitimité. Cela signifie qu'aucune autorité de certification ne signe le certificat et que nous devons s'assurer qu'il s'agit du nœud correct sans chaîne de confiance. Par contre les serveurs web normaux utilisent deux certificats, le premier est celui de serveur Web qui est auto-signé, le second est celui de l'autorité de certification qui a signé le premier certificat

- Un certificat a un champ de validité qui indique le délai de validité de ce certificat. Ce champ est utilisé pour décrire la durée pendant laquelle l'autorité de certification conservera les informations sur la validité du certificat. Une fois le délai expiré, le CA ne suivra plus si un certificat est révoqué ou valide et si la communication est considérée comme non sécurisée [41]. La durée de validité de Tor est similaire à la durée de validité des serveurs Web que nous avons testé, elle est entre « 130 à 365 jours »
- Le numéro de port que Tor utilise par défaut est le port TCP 9001 pour la communication. Le trafic TLS est souvent vu sur le port TCP 443 qui est utilisé pour le trafic HTTPS. Le fait que Tor utilise le port 9001 est révélateur car aucun autre service n'utilise ce port pour le trafic TLS.

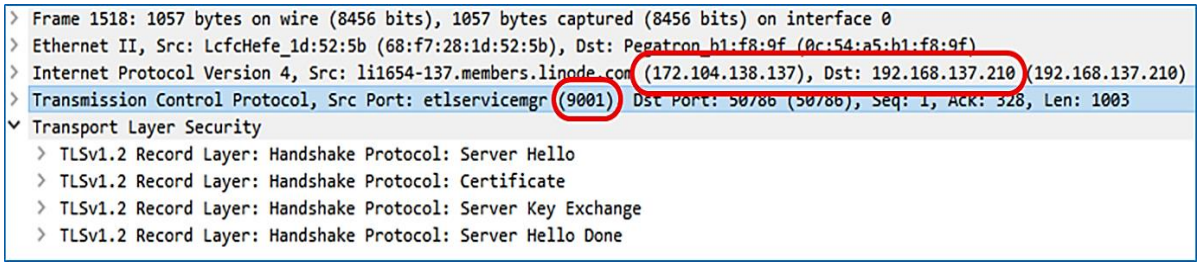

Figure III.52 : *Port TCP utilisé par le nœud d'entrée Tor pour le trafic TLS.*

- Le champ Sujet (*Subject*) se compose de plusieurs sous-champs qui contiennent des informations sur l'organisation et l'emplacement. Le seul sous-champ contenant des informations utiles pour le processus d'authentification est le champ Nom commun. Ce champ lie le nom d'hôte à la clé publique du certificat. Le champ Common Name correspond au nom d'hôte qu'il authentifie [41]. Sur les certificats Tor, le champ suit le même modèle de chaîne aléatoire que le champ émetteur. Le certificat Tor communique qu'il partage très peu d'informations car il a omis tous les sous-champs à l'exception du nom commun.

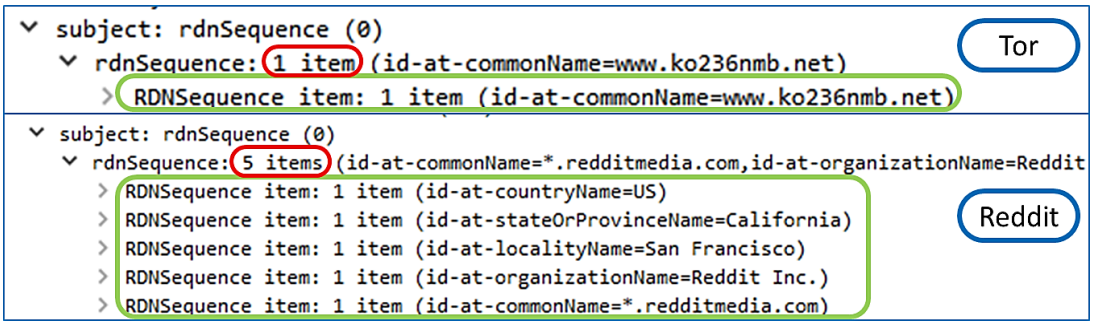

Figure III.53 : *Champ « Subject ».*

Les rdnsequences utilisées pour les champs de titres et d'émetteurs, ce module implémente un sous-ensemble couramment utilise des valeurs RDNSequence. Il ne prend en charge qu'une seule paire de type/valeur d'attribut pour chaque élément de la séquence et implémente les types d'attributs (*CountryName*, *OrganizationName* etc…) [42].

d. Message « Server Key Exchange »

Le message « Server Key Exchange » est un message complémentaire pour l'échange des clés. Ce message contient la clé publique du serveur utilisée par le client pour chiffrer les informations de clé de session

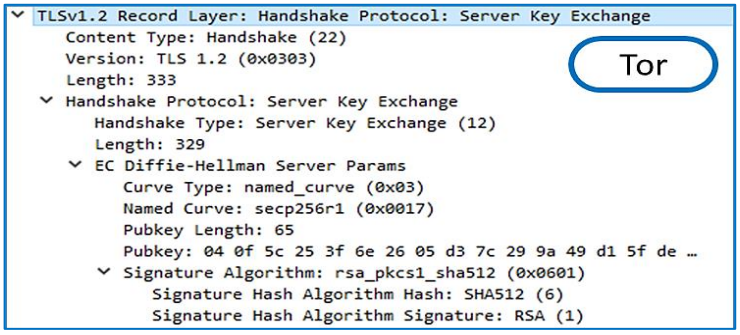

Figure III.54 : *Message « Server Key Exchange ».*

Dans ce processus, aucune anomalie n'a été observée par rapport à une connexion régulière.

e. Message « Server Hello Done »

Ce message indique que le message « Server Hello » est terminé, donc le processus peut poursuivre jusqu'à la phase suivante.

f. Message « Client Key Exchange, Change Cipher Spec (Client) »

Le client crée la clé principale « *pre-master*» à l'aide de son « client random » et de celui du serveur « server random » puis chiffrée avec la clé publique du serveur avec un algorithme qui a été précédemment approuvé dans la négociation.

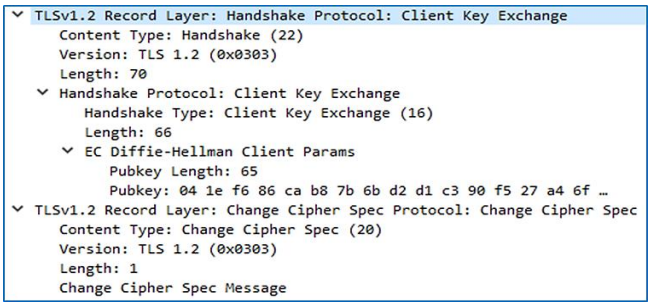

Figure III.55 : *Message « Client Key Exchange, Change Cipher Spec (Client) ».*

Dans ce processus, aucune anomalie n'a été observée par rapport à une connexion régulière.

III.7.4 Constatation

Après avoir analysé les paquets des connexions « TCP Handshake » et « TLS Handshake », nous avons pu trouver plusieurs identifiants Tor qui peuvent caractériser le trafic Tor par rapport aux trafics ordinaires, nous fournissons la liste des identifiants ci-dessous :

1. Le port TCP.
2. Les suites de chiffrements « Cipher Suites » dans le message « ClientHello ».
3. Les groupes supportés « Supported groups » dans le message « ClientHello »
4. Les algorithmes de signatures « Signature Algorithm » utilisées dans « ClientHello ».
5. Type d'extension dans le message « ClientHello ».
6. La langueur du message « Certificate ».
7. Le nom commun « Common Name » dans le message « Certificate ».
8. Le nombre d'extension du message « Certificate ».

Ces identifiants seront utilisés comme signatures numériques pour détecter le trafic Tor, la détection est effectuée à l'aide d'un système de détection d'intrusion réseau « Snort ».

III.8 Extraction des signatures du réseau I2P

III.8.1 Démarche suivi pour notre analyse

Contrairement à Tor lors du démarrage le routeur I2P n'effectue pas une connexion TCP/TLS Handshake.

La planification du concept pour identifier le trafic I2P a abouti à analyser le trafic du réseau pendant trois états différents : lors du démarrage du routeur I2P, après l'avoir exécuté pendant un certain temps et pendant le processus d'arrêt.

Les trois états du réseau ont ensuite été analysés avec Wireshark pour trouver les anomalies et les caractéristiques du trafic I2P qui permettent de l'identifier.

III.8.2 Analyse du trafic du réseau I2P

III.8.2.A 1er état « démarrage du routeur I2P »

a. Pool.ntp.org

Les réseaux isolés peuvent avoir une heure décalée sans problème, mais dès que nous nous connectons à Internet, les effets deviennent visibles. (Exemple d'un courriel arrive cinq minutes avant d'avoir été envoyé, où recevoir une réponse deux minutes avant qu'il a été envoyé).

NTP (*Network Time Protocol*) est un protocole qui permet de synchroniser, via un réseau informatique, l'horloge locale d'ordinateurs sur une référence d'heure [43].

Le projet pool.ntp.org a été démarré en 2003 en réponse à la rapide croissance de la consommation des ressources de serveurs NTP populaires. Le projet a grandi jusqu'à être aujourd'hui central dans le fonctionnement de millions de systèmes dans le monde. Au lieu de créer et de maintenir votre propre ensemble de serveurs NTP ou de négocier de manière individuelle avec des administrateurs de serveur, nous pouvons utiliser pool.ntp.org [43].

- L'utilisation de pool.ntp.org est généralement gratuite. Si nous avons un très grand nombre de clients, il est possible que nous demandions une contribution pour aider au passage à l'échelle de pool.ntp.org afin de répondre à la demande.
- Les projets Open Source sont bien entendu les bienvenus pour l'utilisation de pool.ntp.org dans leur configuration par défaut. Il suffit cependant de demander et d'utiliser une zone fournisseur lorsque nous utilisons pool.ntp.org dans la configuration par défaut (exemple : dz.pool.ntp.org pour l'Algérie).

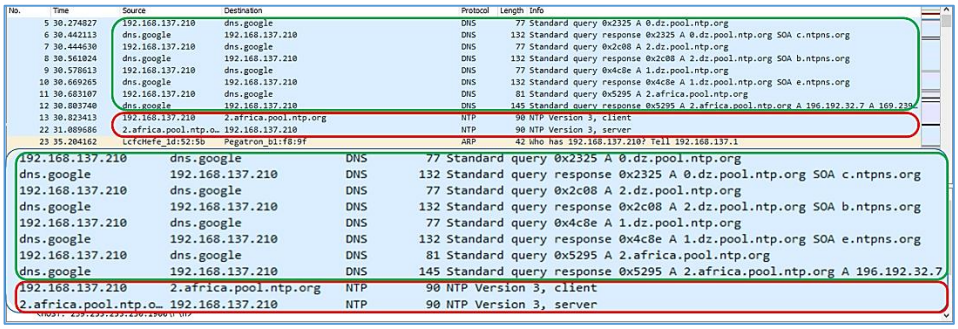

Figure III.56 : *Démarrage du routeur I2P.*

Dès que le routeur I2P démarre, l'analyseur Wireshark affiche les paquets encadrés dans la figure ci-dessus. Et comme nous pouvons voir, le routeur I2P envoie des

requêtes au serveur DNS pour l'interroger sur les adresses IP des pool.ntp.org. Car le routeur I2P a besoin du protocole NTP (Network *Time Protocol*) pour fonctionner correctement.

Au début le routeur I2P contacte le serveur DNS pour avoir l'adresse IP du serveur NTP de la « dz.pool.ntp.org » (de l'Algérie). Le système essaiera de trouver les serveurs disponibles les plus proches pour le routeur I2P.

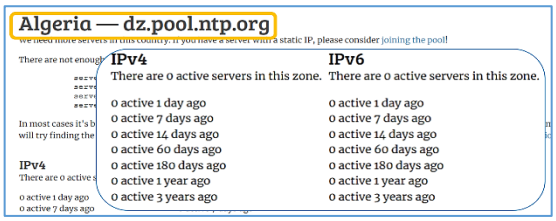

Figure III.57 : *dz.pool.ntp.org.*

Comme nous montre la figure ci-dessus, il n'y a aucun serveur NTP en Algérie. Alors le routeur I2P demande l'adresse IP du serveur NTP de la zone Afrique « africa.pool.ntp.org », ensuite le serveur DNS lui répond avec 4 adresses IP de la « africa.pool.ntp.org ».

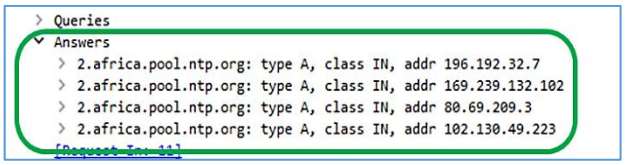

Figure III.58 : *Adresses IP « africa.pool.ntp.org ».*

Après avoir reçu la réponse du serveur DNS le routeur I2P choisi l'une des adresses et contacte le serveur NTP correspondant à cette dernière, le serveur NTP lui répond, ceci est nécessaire pour la synchronisation de l'heure interne d'I2P.

Nous remarquons que dans la figure ci-dessous que le routeur I2P envoie une requête de type client au serveur NTP du « africa.pool.ntp.org », ce dernier lui répond avec une requête de type serveur, celui-ci lui retourne l'heure courante. On peut voir aussi que le protocole NTP utilise le port 123.

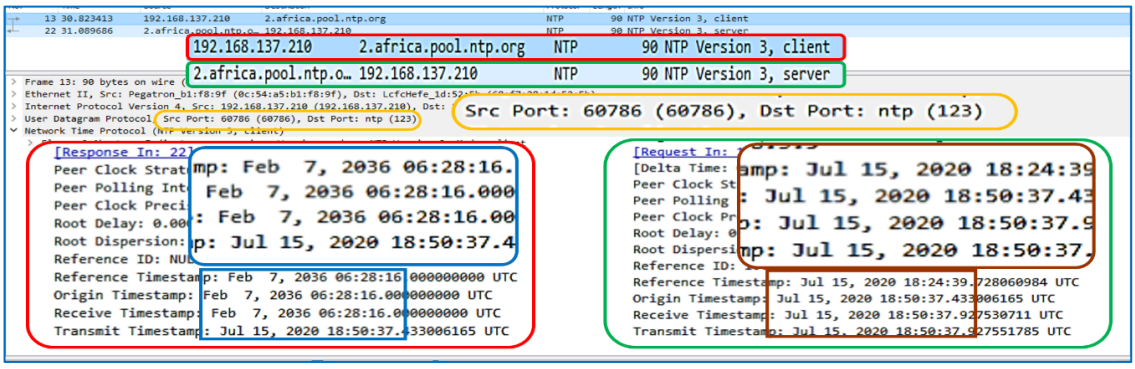

Figure III.59 : *Synchronisation via le protocole NTP.*

b. SSDP

SSDP (*Simple Service Discovery Protocol*) est un protocole de communication informatique en réseau qui fournit un mécanisme par lequel les clients peuvent découvrir des services disponibles sur le réseau [44].

Nous avons remarqué l'apparition des paquets SSDP dans l'analyseur Wireshark, ce protocole est utilisé dans les réseaux informatiques, pour voir les anomalies des paquets SSDP envoyé par le routeur I2P, nous allons effectuer une comparaison entre ces derniers et les paquets SSDP ordinaire (paquets SSDP captés lors de la navigation via le navigateur Chrome et Firefox).

SSDP effectue des annonces périodiques NOTIFY ou des requêtes de découvertes M-SEARCH. Ces annonces et requêtes sont limitées au LAN et sont basées sur des échanges HTTPMU (HTTP sur UDP en multicast). Le groupe « multicast » utilisé est « 239.255.255.250 » et le port UDP est « 1900 ».

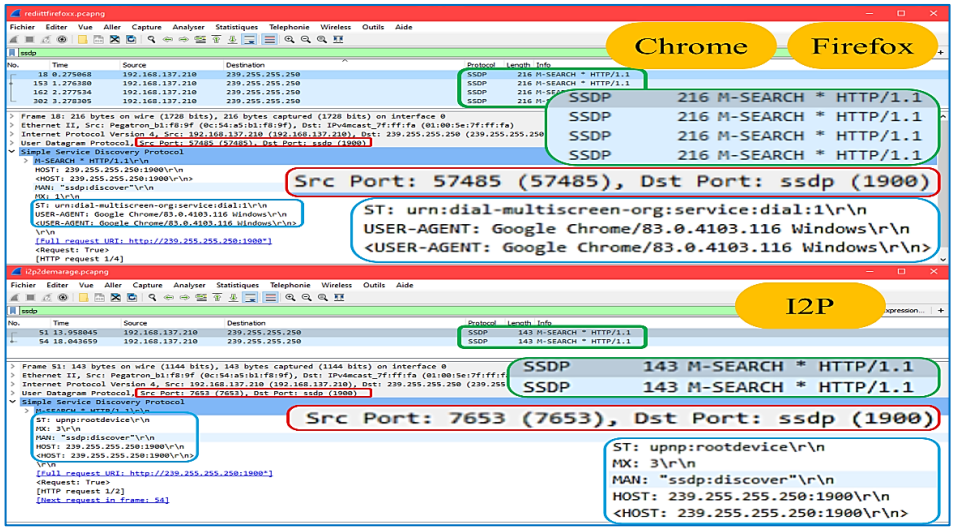

Figure III.60 : *Paquets SSDP via le réseau I2P et le navigateur Chrome/Firefox.*

Le nombre de paquets SSDP captés lors de la navigation via le navigateur Chrome et Firefox est égale à 4 paquets par contre le nombre de paquets SSDP captés via le réseau I2P est égale à 2 paquets.

La longueur des paquets SSDP est égale à 216 octets pour les navigateurs Chrome et Firefox, contrairement au réseau I2P la longueur des paquets SSDP est 143 octets.

Le port utilisé par notre machine pour envoyer la requête M-SEARCH est « 57485 et 58365 » pour les deux navigateurs, ces derniers sont des ports aléatoires par contre le port utilisé par le routeur I2P pour envoyer la même requête est «7653 » qui est un port prédéfini par I2P pour envoyer les requêtes SSDP M-SEARCH.

Le champ « USER-AGENT » qui existe dans les paquets SSDP envoyé via le navigateur Chrome et Firefox n'existe pas dans le paquet SSDP envoyé via le routeur I2P. Et le champ ST (ST : type d'élément recherché) est le même pour les deux navigateurs mais ce dernier est diffèrent pour le paquet SSDP envoyé par le routeur I2P, qui contient le protocole UPnP[17]. Donc l'utilisation du protocole UPnP SSDP à travers le port « 7653 » confirme que le paquet SSDP est envoyé par le routeur I2P.

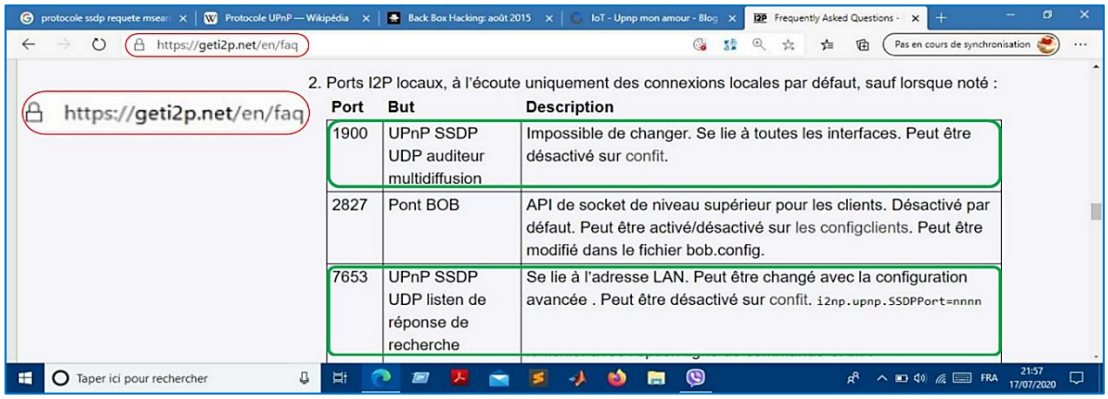

Figure III.61 : *Ports SSDP utilisé par le réseau I2P.*

III.8.2.B 2ème état « après un certain temps d'exécution »

Lorsque le routeur I2P est amorcé (Un routeur I2P nouvellement installé n'a pas de NetDB et nécessite une pré-configuration. Il doit rechercher les RouterInfo de ses pairs dans la NetDB. Cette phase est également appelée phase d'amorçage ou d'initialisation), il peut communiquer avec un autre routeur I2P participant et propage le reste de la NetDB, que nous appelons la phase opérationnelle.

Dans cette phase nous avons remarqué que le routeur I2P transmet de nombreux paquets TCP d'I2P utilisent l'indicateur PUSH pour indiquer au destinataire de transmettre le paquet dès qu'il le reçoit, et de nombreux paquets UDP. I2P utilise des numéros de port aléatoires supérieurs à 9000. Si un hôte communique avec de nombreux pairs différents sur des ports élevés différents, cela peut être considéré comme inhabituel.

Bien que ce ne soit pas des caractéristiques uniques d'I2P, ils peuvent être utilisés pour durcir davantage un soupçon.

Nous allons voir les adresses IP des contacts auxquelles notre routeur I2P peut atteindre dans RouterInfo. Ces derniers changent régulièrement.

[17] **UPnP :** *(Universal Plug and Play*) est composé d'une suite de protocoles destinée à simplifier la configuration et l'interopérabilité des équipements pour l'utilisateur. UPnP est très présent dans les équipements grand public comme les Box Internet, Smart TV, etc….

Figure III.62 : *RouterInfo.*

Nous allons filtrer les paquets dans Wireshark pour voir s'il y'a une communication entre les adresses mentionné dans la figure ci-dessus et notre routeur I2P.

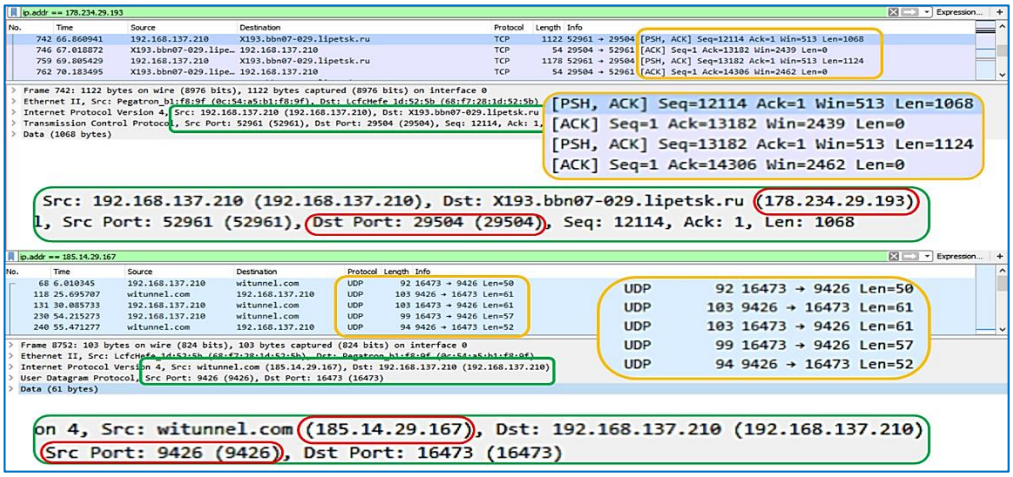

Figure III.63 : *Filtrage des adresses IP trouvées dans RouterInfo.*

Nous avons remarqué qu'il y a une communication avec les adresses « 185.14.29.167 – 178.234.29.193 » et notre routeur I2P, les ports utilisés par ces dernières sont le même port mentionné dans RouterInfo et ils sont supérieurs à 9000.

Pour confirmer cette constatation nous avons répété l'opération à chaque changement d'adresses remarqué dans RouterInfo. Et on a obtenu les mêmes résultats qui est le transfert de nombreux paquets TCP utilisent l'indicateur PUSH et le transfert de nombreux paquets UDP.

III.8.2.C 3ème état « pendant le processus d'arrêt »

Pour arrêter le router I2P il suffit de cliquer sur l'icône d'I2P puis sur arrêté immédiatement et pendant ce processus aucune anomalie n'a été détecter dans Wireshark, juste quelques paquets TCP avec des indicateurs FIN/ACK pour indiquer la fin des sessions entre le routeur I2P et quelques adresses IP qui étaient en cours de communication.

III.8.3 Constatation

Après avoir analysé les paquets du réseau I2P pendant ses trois états différents nous avons pu trouver des identifiants I2P qui peuvent caractériser le trafic I2P par rapport aux trafics ordinaires, nous fournissons la liste des identifiants ci-dessous :

1. L'envoie des requêtes « dz.pool.ntp.org » au serveur DNS pour trouver les serveurs NTP.
2. L'envoie des requêtes « africa.pool.ntp.org » au serveur DNS pour trouver les serveurs NTP.
3. Demande de synchronisation au serveur NTP « africa.pool.ntp.org ».
4. Longueur totale du paquet SSDP et le port source SSDP/UPnP utilisé.
5. Le nombre de paquets UDP avec un port supérieur à 9000.
6. Le nombre de paquets TCP avec l'indicateur PSH/ACK avec un port supérieur à 9000.

Ces identifiants seront utilisés comme signatures numériques pour détecter le trafic I2P, la détection est effectuée à l'aide d'un système de détection d'intrusion réseau « Snort ».

III.9 Signatures du réseau Tor et I2P

Après l'analyse expérimentale effectuée précédemment nous avons pu obtenir les signatures du réseau Tor et I2P présenté dans le tableau ci-dessous.

Signature	Tor
1	Le port TCP.
2	Les suites de chiffrements « Cipher Suites » dans le message « ClientHello ».
3	Les groupes supportés « Supported groups » dans le message « ClientHello ».
4	Les algorithmes de signatures « Signature Algorithm » utilisées dans « ClientHello ».
5	Type d'extension dans le message « ClientHello ».
6	La langueur du message « Certificate ».
7	Le nombre d'extension du message « Certificate ».

8	Le nom commun « Common Name » dans le message « Certificate ».
9	Le nombre de certificat dans le message « Certificate ».
Signature	**I2P**
1	L'envoie des requêtes « dz.pool.ntp.org » au serveur DNS pour trouver les serveurs NTP.
2	L'envoie des requêtes « africa.pool.ntp.org » au serveur DNS pour trouver les serveurs NTP
3	Demande de synchronisation au serveur NTP « africa.pool.ntp.org ».
4	Longueur totale du paquet SSDP et le port source SSDP/UPnP utilisé.
5	Le nombre de paquets UDP avec un port supérieur à 9000.
6	Le nombre de paquets TCP avec l'indicateur PSH/ACK avec un port supérieur à 9000.

Tableau III.6 : *Signatures du réseau Tor & I2P.*

III.10 Conclusion

Dans ce chapitre nous avons effectué une comparaison entre le réseau I2P et le navigateur Tor.

Puis nous avons identifié le trafic Tor en se basant sur les résultats obtenus de la comparaison des paquets « ClientHello » et « Certificate » du navigateur Tor avec les paquets ordinaires.

Ensuite nous avons identifié le trafic I2P en se basant sur l'analyse du trafic réseau pendant les trois états différents « lors du démarrage du routeur I2P », « après l'avoir exécuté pendant un certain temps » et « pendant le processus d'arrêt ».

Les caractéristiques du trafic TOR et I2P obtenues à partir de l'analyse décrite au cours de ce chapitre seront utilisées comme signature pour détecter le trafic de ces derniers.

Certains identifiants peuvent être réalisé au niveau de l'IDS Snort, tandis que d'autre sont simplement irréalisables.

Dans le chapitre suivant, nous allons créer des règles en se basant sur les signatures du Tor et I2P pour détecter l'utilisation de ces derniers à l'aide de l'IDS Snort, et tester la fiabilité des règles déjà créé.

CHAPITRE IV

Détection du réseau Tor et I2P

IV.1 Introduction

Aucun système informatique n'est totalement protégé, pour une entreprise connectée à internet, le problème aujourd'hui, n'est plus de savoir si elle va se faire attaquer mais quand cela va arriver, une solution possible est la mise en œuvre de divers moyens destinés à augmenter le niveau de sécurité. Pour contrer les menaces d'intrusion, les entreprises se tournent de plus en plus vers les solutions de détection d'intrusion IDS (*Intrusion Detection Systems*).

On appelle l'IDS un mécanisme écoutant le trafic réseau de manière furtive afin de repérer des activités suspectes et permettant ainsi d'avoir des alertes d'intrusion.

Au cours de ce chapitre nous allons créer en premier lieu des règles basées sur les signatures obtenues précédemment dans un système de détection d'intrusion « Snort », puis tester la fiabilité de nos règles afin de détecter l'utilisation du réseau Tor et I2P.

IV.2 Système de détection d'intrusion

Un système de détection d'intrusion ou IDS est un équipement matériel ou bien logiciel permettant de surveiller l'activité d'un réseau ou d'un hôte donné, afin de détecter toute tentative d'intrusion (volontaire ou non) et éventuellement de réagir à cette tentative [45].

Certains termes sont souvent employés quand on parle d'IDS :

- Faux positif : une alerte provenant d'un IDS, mais qui ne correspond pas à une attaque réelle.
- Faux négatif : une intrusion réelle qui n'a pas été détectée par l'IDS.

IV.2.1 Fonctions d'un IDS

Un IDS possède quatre fonctions principales : l'analyse, la journalisation, la gestion et l'action [45].

1. **Analyse :** Analyse des journaux du système pour identifier des intentions dans la masse de données recueillie par l'IDS. Il y a deux méthodes d'analyse : une basée sur les signatures d'attaque, et l'autre sur la détection d'anomalies.
2. **Journalisation :** Enregistrement des événements dans un fichier de log.
3. **Gestion :** Les IDS doivent être administrés de manière permanente. On peut assimiler un IDS à une caméra de sécurité.
4. **Action :** Alerter l'administrateur quand une attaque dangereuse est détectée.

Un NIDS est un type d'IDS qui surveille l'ensemble du réseau, et se découpe en trois grandes parties : La capture, les signatures et les alertes.

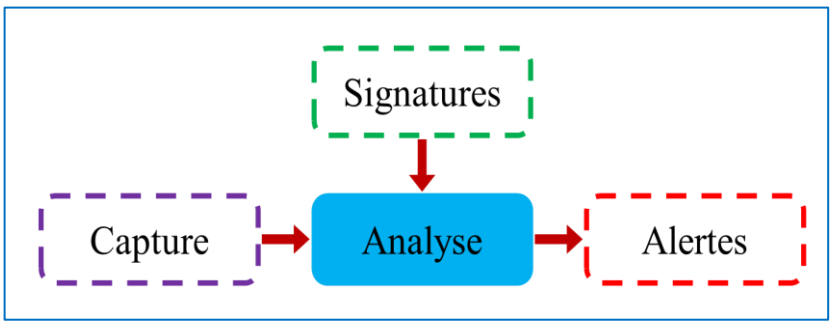

Figure IV.1 : *Fonctionnement du NIDS.*

1. La capture : La capture sert à la récupération de trafic réseau. En général cela se fait en temps réel, bien que certains NIDS permettent l'analyse de trafic capturé précédemment, la plupart des NIDS utilisent l'interface de programmation PCap « *packet capture* » pour capturer un trafic réseau [45].

2. Les Signatures : Les bibliothèques de signatures rendent la démarche d'analyse similaire à celle de l'antivirus quand ceux-ci s'appuient sur des signatures d'attaques. Ainsi, le NIDS est efficace s'il connaît l'attaque, mais inefficace dans le cas contraire. Les outils à base de signatures requièrent des mises à jour très régulières [45].

3. Les Alertes : Lorsqu'un IDS détecte une intrusion, il doit signaler à l'administrateur, et ce à travers les alertes. Ces alertes peuvent être enregistrées dans des fichiers logs ou dans une base de données où il est possible de les consulter plus tard par un expert de sécurité [45].

IV.2.2 Modes de détection

Nous notons deux modes de détection qui sont [46] :

1. La détection d'anomalies : Elle consiste à détecter des anomalies par rapport à un profil "de trafic habituel". La mise en œuvre comprend toujours une phase d'apprentissage au cours de laquelle les IDS vont découvrir le fonctionnement normal des éléments surveillés. Ils sont ainsi en mesure de signaler les divergences par rapport au fonctionnement de référence.
2. La reconnaissance de signature : Cette approche consiste à rechercher dans l'activité de l'élément surveillé les empreintes ou signatures d'attaques connues (ensemble de caractéristiques permettant d'identifier une activité intrusive). Ce type d'IDS est purement

réactif ; il ne peut détecter que les attaques dont il possède la signature. De ce fait, il nécessite des mises à jour fréquentes.

Il faut noter que la reconnaissance de signature est le mode de fonctionnement le plus implémenté par les IDS du marché. Cependant, les nouveaux produits tendent à combiner les deux méthodes pour affiner la détection d'intrusion.

IV.3 Snort

Snort est un des plus actifs NIDS open source et possède une communauté importante contribuant à son succès. Il est capable d'effectuer une analyse en temps réel du trafic entrant et sortant. Snort est disponible pour la plupart des systèmes d'exploitation (Windows et linux comme Ubuntu, Debian, CentOS), et les mises à jour des règles sont gratuites.

Ces derniers, sont les raisons pour lesquelles nous avons choisi le NIDS Snort pour la détection de l'utilisation du réseau Tor et I2P.

IV.3.1 Présentation générale

Snort est un système de détection d'intrusion libre (ou NIDS) publié sous licence GNU GPL. À l'origine écrit par Martin Roesch, Snort est capable d'effectuer en temps réel des analyses de trafic et de logger les paquets sur un réseau IP. Il peut effectuer des analyses de protocole, recherche et correspondance de contenu et peut être utilisé pour détecter une grande variété d'attaques et des tentatives. Snort est basé sur libpcap (pour la capture de paquets de bibliothèque), un outil largement utilisé dans les détecteurs de trafic TCP / IP et les analyseurs grâce à l'analyse du protocole et à la recherche de contenu [48].

SNORT permet d'analyser le trafic réseau de type IP, il peut être configuré pour fonctionner en trois modes [46] :

- **Le mode sniffer :** dans ce mode, SNORT lit les paquets circulant sur le réseau et les affiche d'une façon continue sur l'écran.
- **Le mode « packet logger » :** dans ce mode SNORT journalise le trafic réseau dans des répertoires sur le disque.
- **Le mode détecteur d'intrusion réseau (NIDS) :** dans ce mode, SNORT analyse le trafic du réseau, compare ce trafic à des règles déjà définies par l'utilisateur et établi des actions à exécuter.
- **Le mode Prévention des intrusions réseau (IPS) :** c'est SNORT-inline.

IV.3.2 Architecture de Snort

L'architecture de Snort est organisée en modules qui sont [47] :

- **Décodeur de paquet** « *Packet Decoder* » : il capture les paquets de données des interfaces réseaux, les prépare afin d'être prétraitées ou envoyées au moteur de détection.
- **Pré processeur** « *Pre processor* » : ce sont des composants utilisés avec Snort afin d'améliorer les possibilités d'analyse. Ils reçoivent les paquets, les retraitent et les envoient au moteur de détection.
- **Moteur de détection** « *Detection Engine* » : c'est le composant le plus important de Snort. Son rôle consiste à détecter les éventuelles intrusions qui existent dans un paquet.
- **Système d'alerte et d'enregistrement des logs** « *Logging and Alerting System* » : il permet de générer les alertes et les messages log suivant ce que le moteur de détection a trouvé dans le paquet analysé.
- **Modules de sortie** « *Output ou plugins* » : permet de traiter l'intrusion générée par le système d'alertes et de notation de plusieurs manières : envoie vers un fichier log, génère un message d'alerte vers un serveur syslog, ou stocke cette intrusion dans une base de données.

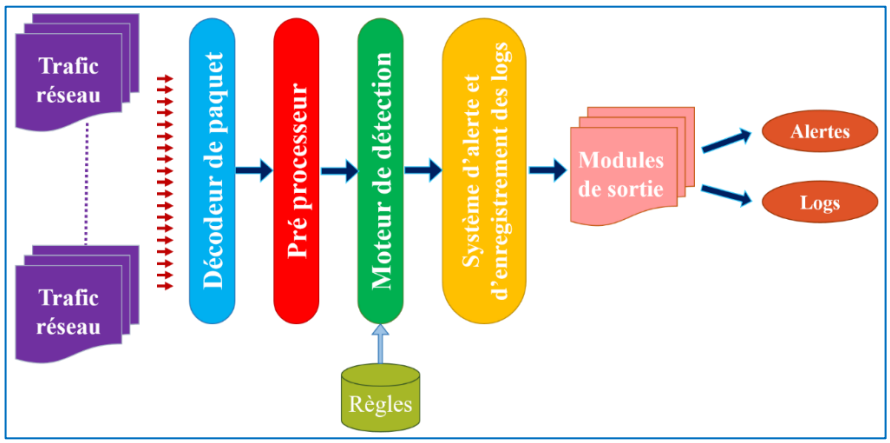

Figure IV.2 : *Architecture d'un IDS [47].*

IV.3.3 Format des règles Snort

Les signatures jouent un rôle très important dans Snort. Dans la plupart des cas, les gens utilisent des jeux de règles existants. Une règle/signature se compose des éléments suivants [48] :

- **L'action :** qui détermine ce qui se passe lorsque la signature correspond
- **En-tête :** définition du protocole, des adresses IP, des ports et de la direction de la règle.
- **Options de règle :** définissant les spécificités de la règle.

Un exemple de règle est le suivant :

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Likely Bot Nick in IRC (USA +..)"; flow:established,to_server; flowbits:isset,is_proto_irc; content:"NICK "; pcre:"/NICK .*USA.*[0-9]{3,}/i"; reference:url,doc.emergingthreats.net/2008124; classtype:trojan-activity; sid:2008124; rev:2;)
```

Dans cet exemple, le rouge est l'action, le vert est l'en-tête et le bleu sont les options.

IV.3.3.A Action

Toutes les signatures ont des propriétés différentes. L'un d'eux est la propriété Action. Celui-ci détermine ce qui se passera quand une signature correspond. Il existe quatre types d'action [48] :

1. **Pass :** Si une signature correspond et contient un laissez-passer, Snort cesse de scanner le paquet et passe à la fin de toutes les règles (uniquement pour le paquet en cours).
2. **Drop :** Cela ne concerne que le mode IPS « *Intrusion Prevention System* », si le programme trouve une signature qui correspond, contenant le « *drop* », il s'arrête immédiatement.
3. **Reject :** Il s'agit d'un rejet actif du paquet. Le récepteur et l'expéditeur reçoivent un paquet de rejet.
4. **Alert :** Si une signature correspond et contient une alerte, le paquet sera traité comme n'importe quel autre paquet non menaçant, à l'exception de celui-ci une alerte sera générée par Snort. Seul l'administrateur système peut remarquer cette alerte.

IV.3.3.B En-tête

Elle est composée de 4 champs [48] :

1. **Protocole :** Ce mot clé dans une signature indique à Snort quel protocole il s'agit.
2. **IP Source et IP destination**
3. **Ports (source et destination)**
4. **Direction :** La direction indique de quelle façon la signature doit correspondre. Presque chaque signature a une flèche vers la droite. Cela signifie que seuls les paquets

ayant la même direction peuvent correspondre. Cependant, il est également possible d'avoir une correspondance de règle dans les deux sens : « -> < > ».

- source -> destination.
- source < > destination (les deux directions).

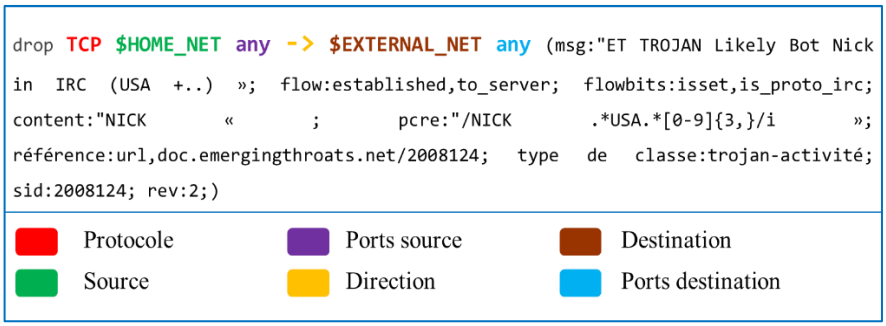

Figure IV.3 : *Les composent de l'en-tête des règles Snort.*

IV.3.3.C Options de règle

Le reste de la règle se compose d'options. Celles-ci sont placées entre parenthèses et séparées par des points-virgules. Certaines options ont des paramètres qui sont spécifiés par le mot-clé de l'option, suivi de deux points, suivi des paramètres. Voici quelques exemples [48] :

1. **msg** (*message*) : Le mot clé msg donne des informations textuelles sur la signature et l'alerte possible.
2. **Sid :** le mot Sid mot clé donne à chaque signature son propre id.
3. **Id :** Avec le mot clé d'identification, vous pouvez correspondre sur une valeur d'ID IP spécifique. L'ID identifie chaque paquet envoyé par un hôte.
4. **Seq :** le mot clé Seq peut être utilisé dans une signature pour rechercher un numéro de séquence TCP spécifique.
5. **Dsize :** Avec le mot clé Dsize, vous pouvez correspondre à la taille de la charge utile du paquet.
6. **Content :** Ce mot clé est très important dans les signatures. Entre deux « | | » vous pouvez écrire sur ce que vous souhaitez que la signature correspondre.
7. **Rev (***revision***) :** Rev représente la version de la signature. Si une signature est modifiée, le nombre de Rev sera incrémenté par les auteurs de signature.
8. **Offset :** modifie l'option « content », fixe le décalage du début de la tentative de correspondance de motif.

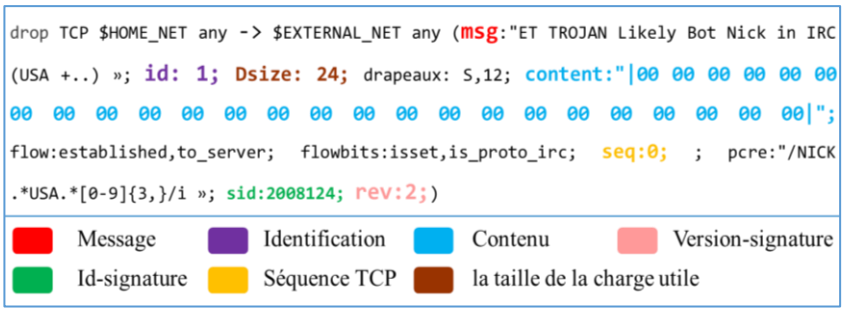

Figure IV.4 : *Options de règle.*

IV.4 Implémentations des signatures numérique dans Snort

IV.4.1 Signatures numérique du navigateur Tor

Nous avons dévoilé plusieurs identifiants révélateurs lors de l'expérimentation. Certains sont faciles à mettre en œuvre et à tester, tandis que d'autres peuvent être difficiles et peut-être irréalisables à mettre en œuvre.

Nous avons essayé d'implémenter certains des identifiants qui pourraient impliquer l'utilisation de Tor. Pour ce faire, nous avons utilisé le système de détection d'intrusion open source appelé Snort. Snort utilise des signatures pour analyser le trafic réseau, nous devons d'abord créer des signatures qui déclenchent l'utilisation de Tor. Pour ajouter de nouvelles signatures, un langage simple est utilisé où l'on spécifie le sens du trafic et les octets qui identifient le trafic. Nous avons écrit des signatures basées sur certains des identifiants que nous avons trouvés.

Les règles se trouvent dans trois dossiers. Il y a le dossier « rules » qui est composé des règles Snort. Ce dossier contient les règles que Snort va utiliser pour surveiller le réseau ce dossier « rules » contient les règles basées sur un texte standard et qui vont être exécutées lorsqu'on lance Snort.

Il est important de noter que le fichier « local.rules » est le fichier qui va être spécifique à chaque hôte du système. Donc il sera mis-à-jour par le fournisseur de sécurité et donc par un expert qui pourra spécifier la règle à implémenter qui sera spécifique à l'entreprise. Dans notre cas nous allons écrire les règles pour le réseau Tor et le réseau I2P dans le fichier « local.rules ».

- La première signature détecte les suites de chiffrement exactes utilisées dans le message « ClientHello » par le navigateur Tor. Il vérifie que le « ClientHello » a les 18 suites de chiffrement que Tor utilise et qu'il a le bon ordre.

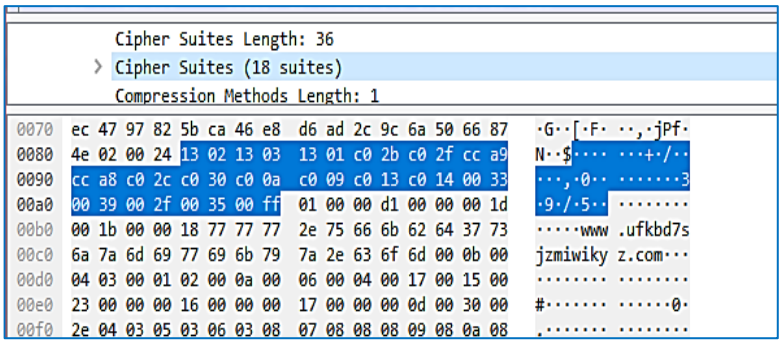

Figure IV.5 : *Première signature du navigateur Tor «suites de chiffrement ».*

Pour écrire cette alerte nous avons utilisé les octets (en hexadécimal) qui représentent les 18 suites de chiffrements du message « ClientHello », qui se trouvent à partir du 30éme octet (offset) de la charge utile du message « ClientHello », avec le port destination 9001.

```
alert tcp any any -> any 9001 (msg: "Possibilité de création 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;)
```

- La deuxième signature détecte les extensions « Supported_groups » exactes utilisées par le navigateur Tor. Il vérifie que le « ClientHello » a les deux groupes que Tor utilise.

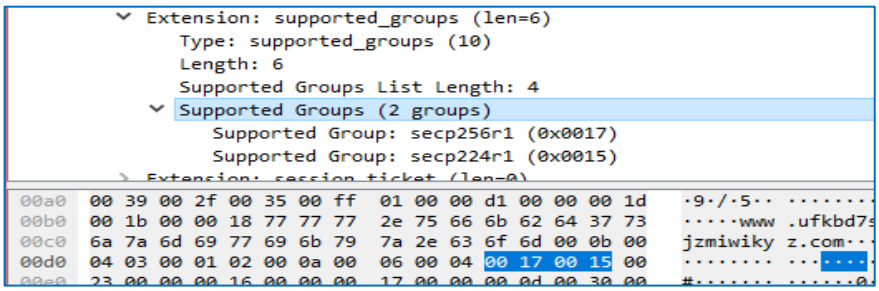

Figure IV.6 : *Deuxième signature du navigateur Tor «Supported_groups».*

Pour écrire cette alerte nous avons utilisé les octets (en hexadécimal) qui représentent les deux groupes du message « ClientHello ».

```
alert tcp any any -> any 9001 (msg: "Possibilité de création du circuit Tor : CleintHello Extension supported_groups"; content:"|00 17 00 15|"; offset:30; sid:10000005 ;)
```

- La troisième signature détecte les extensions « signature_algorithms » exactes utilisées par le navigateur Tor. Il vérifie que le « ClientHello » a les 23 algorithmes de signature que Tor utilise.

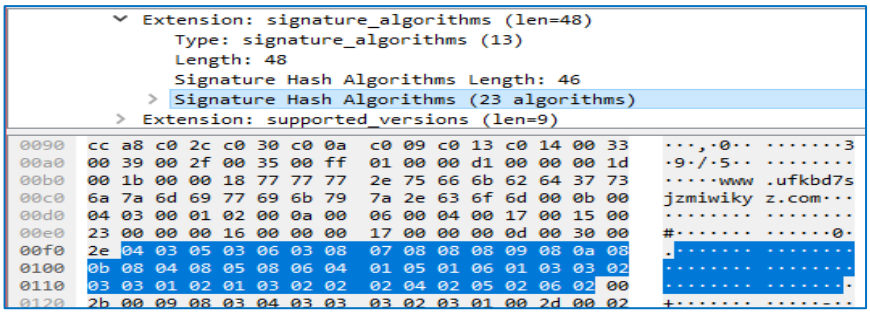

Figure IV.7 : *Troisième signature du navigateur Tor « signature_algorithms».*

Pour écrire cette alerte nous avons utilisé les octets (en hexadécimal) qui représentent les 23 algorithmes de signature du message « ClientHello ».

```
alert tcp any any -> any 9001 (msg: "Possibilité de création circuit Tor : CleintHello Extension signature_algorithms"; 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 ;)
```

- La quatrième signature détecte le nombre de certificat dans le message « Certificate » envoyé par le nœud d'entrée Tor, il vérifie que le nombre de certificat est égal à 1. Ceci est assez rare pour le trafic TLS normal.

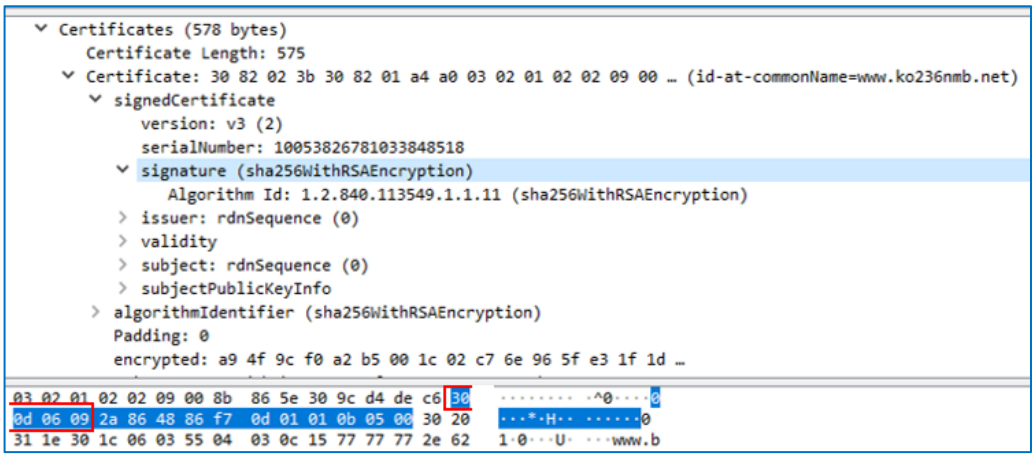

Figure IV.8 : *Quatrième signature du navigateur Tor « Nombre de certificat».*

Le nœud d'entrée Tor envoie le message « Certificate » avec un seul certificat avec une signature (*sha256WithRSAEncryption*) dont la représentation hexadécimale est « 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0b 05 00 », nous avons remarqué que d'après l'analyse effectuée que tous les messages certificats envoyés lors de la navigation avec le navigateur Chrome et Firefox (sites : Reddit, Amazon) ont la même signature mentionnée.

Les 4 premiers octets sont fixes pour « sha224, sha256, sha384 et sha512 » :

sha224WithRSAEncryption: 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0d 05 00

sha256WithRSAEncryption: 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0b 05 00

sha384WithRSAEncryption: 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0c 05 00

sha512WithRSAEncryption: 30 0d 06 09 2a 86 48 86 f7 0d 01 01 0d 05 00

Pour écrire cette alerte nous avons utilisé les 4 premiers octets (en hexadécimal) de la signature (*sha256WithRSAEncryption*) utilisé par le nœud d'entrée Tor, pour identifier les paquets Tor qui possède un seul certificat. Car donc le cas normal on trouve ces 4 premiers octets 2 fois dans le message « Certificate ».

Puisque dans le cas du Tor on trouve ces 4 octets une seule fois dans le message « Certificate » nous allons écrire la règle en spécifiant le début « 30 0d 06 09 » pour trouver le 1er certificat, puis on ignore un certain nombre d'octets avec l'option « distance » (ces octets appartient au 1er certificat), une fois ignorée les 500 octets. S'il ne trouve pas une deuxième fois les 4 premiers octets « 30 0d 06 09 », l'alerte sera déclenchée, (cela signifie que le message certificat ne contient qu'un seul certificat).

```
alert tcp any 9001 -> $HOME_NET any (msg: "Possibilité de création du circuit Tor : Nombre de certificat = 1"; content:"|30 0d 06 09|"; offset:80; content:!"|30 0d 06 09|"; distance: 500 ; sid: 1000007;)
```

- La cinquième signature vérifie les extensions de certificat. S'il n'y a aucune extension, l'alerte se déclenche.

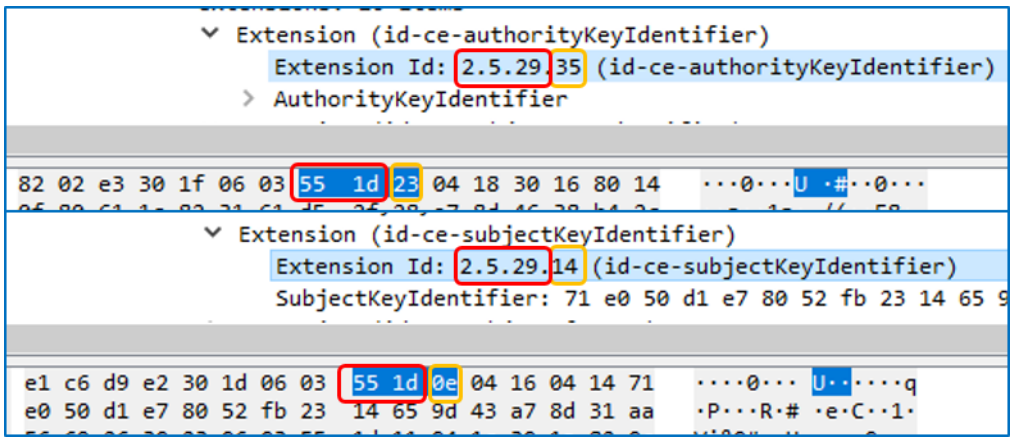

Figure IV.9 : *Extension du message « Certificate » du site Web Reddit.*

Pour écrire cette alerte nous avons utilisé les 2 premiers octets (en hexadécimal) fixes pour tous les extensions du message « Certificate », ces derniers représentent le début

d'ID de chaque extensions. En allons au message « Certificate » (par les 4 premiers octets « 30 0d 06 09 ») et en cherchant à partir de l'octet 250 les 2 premiers octets fixe du champ extension, S'il ne trouve pas ces derniers, l'alerte sera déclenchée, (cela signifie que le message certificat ne contient pas des extensions).

```
alert tcp any 9001 -> $HOME_NET any (msg: "Possibilité de création du circuit Tor : Nombre d'extensions certificat = 0"; content:"|30 0d 06 09|"; offset:80; content: !"|55 1d|"; offset:250; sid: 1000008;)
```

- La sixième signature vérifie la taille du message « Certificate ». Si elle est inférieure à 1200 l'alerte se déclenche.

Puisque la langueur du message « Certificate » du Tor est inferieur par rapport à la langueur de message « Certificate » envoyé par les autres serveurs destinations qui est généralement entre « 2000 et 4000 octets ». Nous avons choisi 1200 comme référence car le 2ème paquet du TLS Handshake établie par Tor contient les 4 messages « SeverHello, Certificate, serveur key exchange, ServerHelloDone », donc la taille totale du paquet et inférieur à 1200 (langueur du message « Certificate » est de 578).

```
alert tcp any 9001 -> $HOME_NET any (msg: "Possibilité de création du circuit Tor : Taille de certificat"; content:"|30 0d 06 09|"; offset:20; dsize: <1200; sid: 1000009;)
```

IV.4.2 Signatures numérique du réseau I2P

- La première signature détecte si le routeur I2P contacte le serveur DNS pour avoir l'adresse IP du serveur NTP de la « dz.pool.ntp.org » (de l'Algérie). S'il y'aura une requête DNS demandant les adresses IP des « 0.dz.pool.ntp.org, 1.dz.pool.ntp.org et 2.dz.pool.ntp.org » l'alerte se déclenche.

```
alert udp any any -> any 53 (msg: "Possibilité de démarrage 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;)

alert udp any any -> any 53 (msg: "Possibilité de démarrage 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;)

alert udp any any -> any 53 (msg: "Possibilité de démarrage 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;)
```

- La deuxième signature détecte si le routeur I2P contacte le serveur DNS pour avoir l'adresse IP du serveur NTP de la « africa.pool.ntp.org » (de l'Afrique). S'il y'aura une requête DNS demandant les adresses IP de l'une des Pool d'Afrique soit « 0.africa.pool.ntp.org» ou « 1.africa.pool.ntp.org» ou bien « 2.africa.pool.ntp.org », l'alerte se déclenche.

```
alert udp any any -> any 53 (msg: "Possibilité de démarrage 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;)

alert udp any any -> any 53 (msg: "Possibilité de démarrage 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;)

alert udp any any -> any 53 (msg: "Possibilité de démarrage 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;)
```

- La troisième signature détecte si le routeur I2P contacte le serveur NTP par l'une des adresses IP proposées par le serveur DNS.

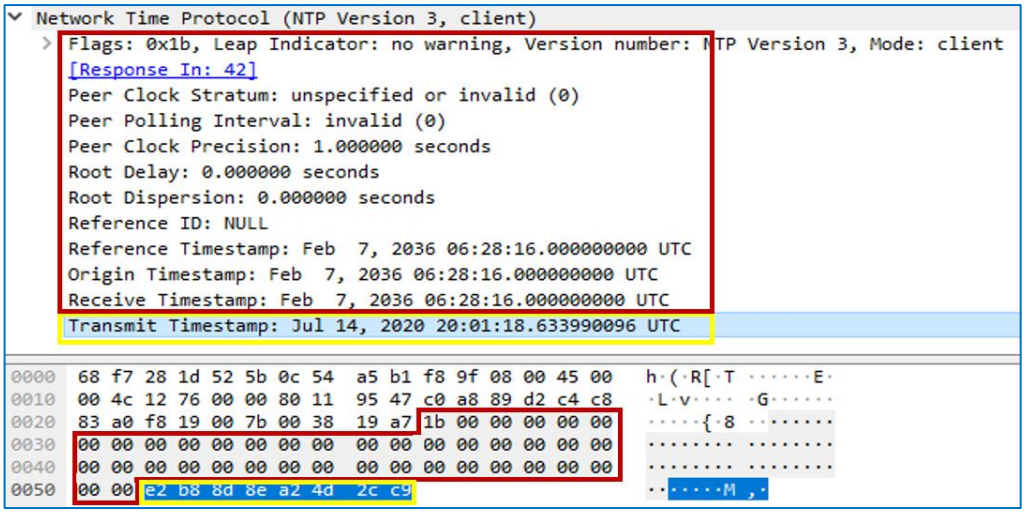

Figure IV.10 : *Paquet NTP du routeur I2P*

Pour écrire cette alerte nous avons utilisé les 40 premiers octets (en hexadécimal) fixes représentant la requête client NTP. Ces 40 octets regroupent tous les champs sauf le champ « Transmit Timetamp » qui change, il représente le temps de l'envoie de la requête NTP puis nous avons spécifié le port associé au NTP « 123 ».

```
alert udp any any -> any 123 (msg: " Possibilité de démarrage 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 00 00 00 00 00 00 |";  sid: 10000016;)
```

- La quatrième signature détecte si le routeur I2P contacte le groupe multicast « 239.255.255.250 » en effectuant une requête SSDP M-SEARCH.

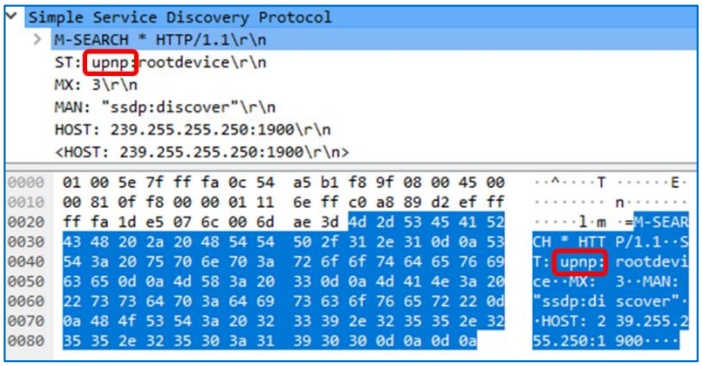

Figure IV.11 : *Paquet SSDP envoyé par le routeur I2P.*

Pour écrire cette alerte nous avons utilisé le port UDP destination « 1900 » du SSDP et le port source « 7653 » déjà mentionné dans le chapitre 3 comme étant le port source spécifique utilisé par le routeur I2P pour lancé les requête SSDP M-SEARCH en multicast. Puis nous avons le contenu « UPnP » trouvé seulement dans le paquet SSDP envoyé par le routeur I2P ainsi que la langueur de ce dernier qui est « 143 » inferieur a la langueur du paquet SSDP ordinaire « 216 ».

```
alert udp any 7653 -> 239.255.255.250 1900 (msg: " Possibilité de démarrage du routeur I2P: SSDP "; content:"upnp"; dsize: <150;  sid: 10000017;)
```

IV.5 Environnement de test

IV.5.1 Présentation de l'architecture de test

Comme mentionné dans le chapitre 3, nous allons utiliser deux ordinateurs. Notre environnement de test est implémenté comme une architecture client-serveur en utilisant CCProxy dans le PC-serveur pour gérer les connexions. Et les logiciels d'anonymat Tor et I2P sont installés au niveau du PC-client, ainsi que Snort comme un système de détection d'intrusion « NIDS » sera installé au niveau du PC-serveur pour détecter l'utilisation du Tor et I2P par le PC-client.

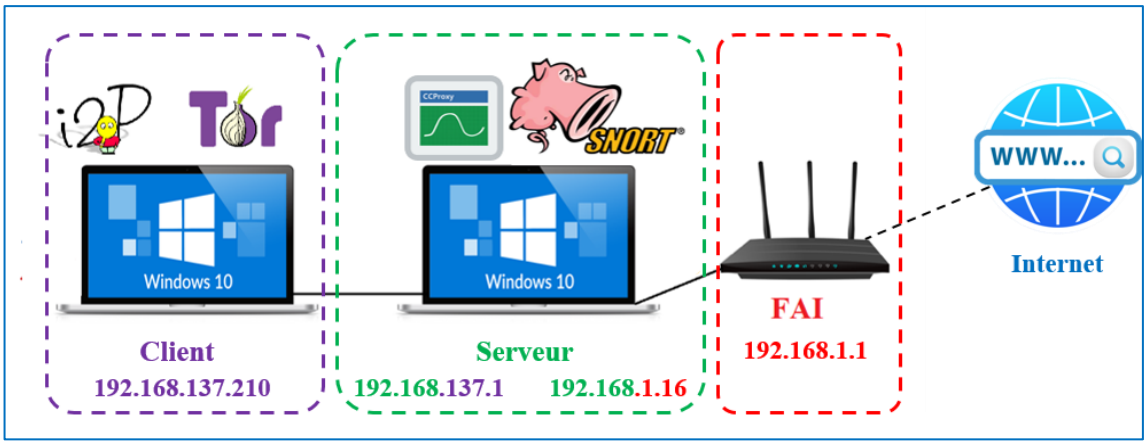

Figure IV.12 : *Architecture de test.*

IV.5.2 Lancement de l'IDS Snort

L'installation de Snort, bien que relativement simple, ne permet pas d'utiliser directement l'IDS sans modification préalable de la configuration par défaut. Après avoir modifié quelques commandes dans le fichier « Snort.conf » pour que Snort fonctionne sous Windows. Nous avons démarré Snort en suivant les étapes ci-dessous :

De nombreux systèmes ont plusieurs interfaces réseau, c'est donc important de déterminer laquelle nous voulons que Snort surveille en exécutant la commande « **Snort – W »** pour voir les interfaces disponibles. Dans notre cas nous utilisons l'interface N° 4.

```
:\Snort\bin>snort -W
  ,,_     -*> Snort! <*-
 o"  )~   Version 2.9.16-WIN64 GRE (Build 118)
  ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
          Copyright (C) 2014-2020 Cisco and/or its affiliates. All rights reserved.
          Copyright (C) 1998-2013 Sourcefire, Inc., et al.
          Using PCRE version: 8.10 2010-06-25
          Using ZLIB version: 1.2.11

Index   Physical Address       IP Address      Device Name     Description
-----   ----------------       ----------      -----------     -----------
    1   00:00:00:00:00:00       disabled        \Device\NPF_{5E7003F2-950C-474B-940A-0680560924DE}      NdisWan Adapter
    2   00:00:00:00:00:00       0000:0000:fe80:0000:0000:0000:6958:a0fc \Device\NPF_{C0E24FC2-A745-4C1B-ABAA-C3DE3155360F} Microsoft
    3   00:00:00:00:00:00       0000:0000:fe80:0000:0000:0000:b52a:405c \Device\NPF_{4422SAC4-49C4-4AED-8429-292A4993D2EE} Microsoft
    4   68:F7:28:1D:52:5B       0000:0000:fe80:0000:0000:0000:b15a:795f \Device\NPF_{453EB42A-2D39-472A-BD32-458A1B9C8D1A} Realtek PCIe GBE Family Controller
    5   00:00:00:00:00:00       disabled        \Device\NPF_{340000F8-FFC5-4D3D-8BCB-8DA83DAF2B08}      NdisWan Adapter
    6   00:00:00:00:00:00       0000:0000:fe80:0000:0000:0000:35db:02e8 \Device\NPF_{AE432656-9761-409F-B99D-56C8C30B1B25} Microsoft
    7   00:00:00:00:00:00       0000:0000:fe80:0000:0000:0000:c06b:1c3e \Device\NPF_{E9EC242D-30B0-4E90-96A3-FA3DF3E88535} Microsoft
    8   00:00:00:00:00:00       disabled        \Device\NPF_{665C03CF-5F6C-4165-B9D8-47883F262425}      NdisWan Adapter
    9   00:00:00:00:00:00       0000:0000:fe80:0000:0000:0000:545e:0511 \Device\NPF_{1E100D05-D871-4862-9835-96F70E5AF884} MS NDIS 6.0 LoopBack Driver
```

Figure IV.13 : *Interfaces réseau disponible.*

1. L'exécution de l'invite de commande en tant qu'administrateur.
2. L'accès au répertoire dans lequel Snort est installé :

   c: \ Windows \ system32> cd \ Snort \ bin.
3. Test de la configuration actuelle de Snort :

   c: \ Snort \ bin> snort -i 4 -c c: \ Snort \ etc \ snort.conf –T.

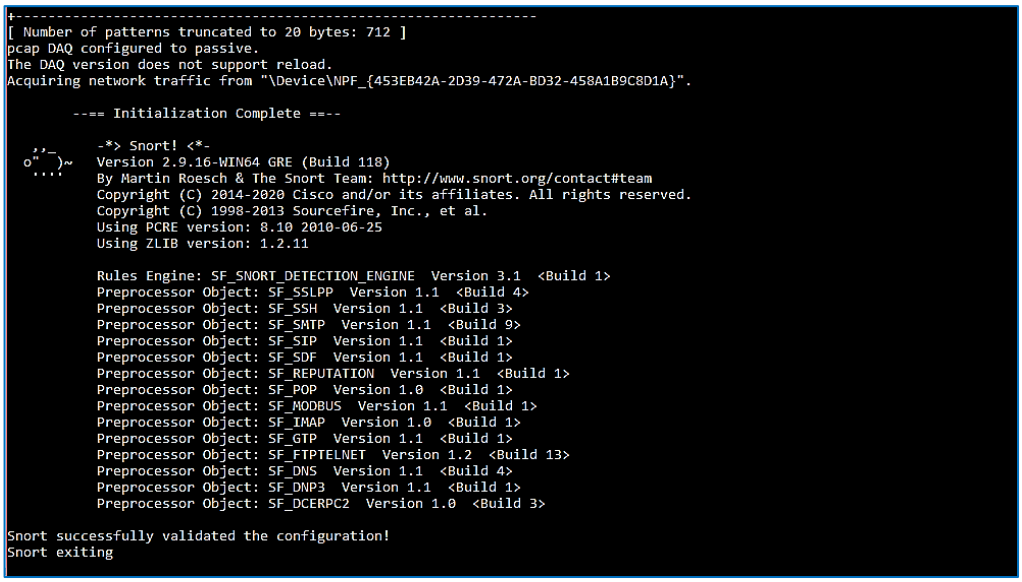

Figure IV.14 *: Test de configuration actuelle.*

4. Démarrage Snort :

c: \ Snort \ bin> snort -i 4 -c c: \ Snort \ etc \ snort.conf -A console.

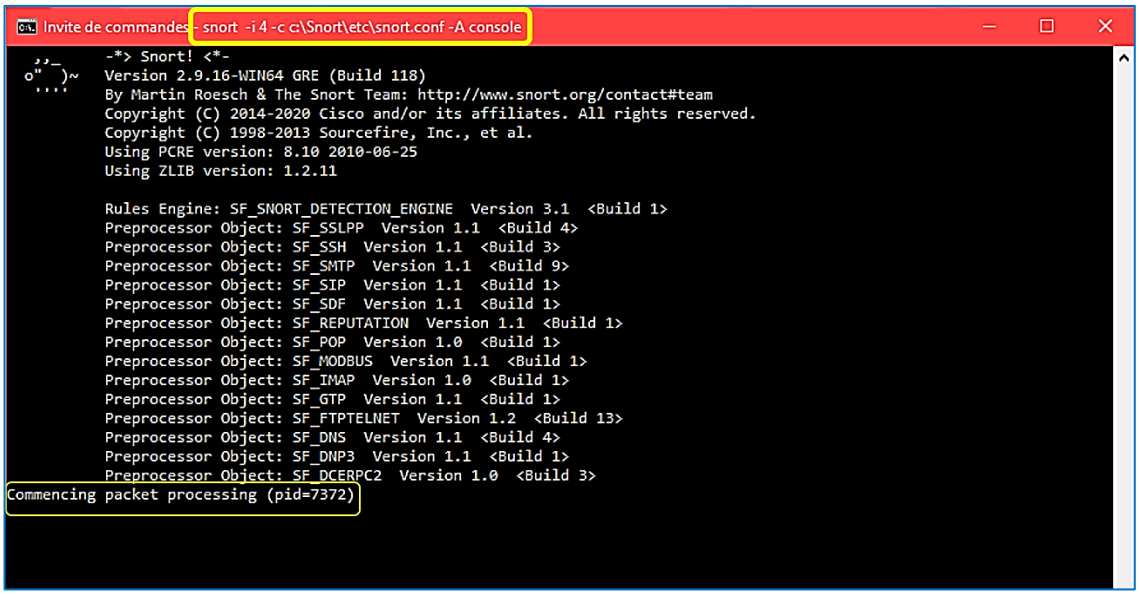

Figure IV.15 : *Démarrage du Snort.*

IV.5.3 Scenario de test

Le tableau ci-dessous représente les règles crées précédemment :

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 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;)

Tableau IV.1 : *Régles Tor et I2P.*

Pour tester la fiabilité de nos règles, Nous allons effectuer deux testes :

1. Naviguer avec le navigateur Firefox et le navigateur Chrome en utilisant le PC-client pour voir si Snort génère des fausses alertes (faux positifs).
2. Naviguer avec le navigateur Tor et démarrer le routeur I2P alternativement en utilisant le PC-client pour voir si Snort détecte l'utilisation de ces derniers.

IV.5.4 Résultat

IV.5.4.A Résultat du 1er test

Après le démarrage de Snort et la navigation vers différents sites Web à travers le navigateur Firefox et le navigateur Chrome, nous avons remarqué que Snort n'a pas généré des fausses alertes.

IV.5.4.B Résultat du 2ème test

Dès que nous lançons le navigateur Tor nous avons remarqué que Snort génère les six alertes au même temps, ces dernières correspondent aux règles du navigateur Tor présentées dans le tableau ci-dessus. Ce qui signifie la détection de l'utilisation du navigateur Tor.

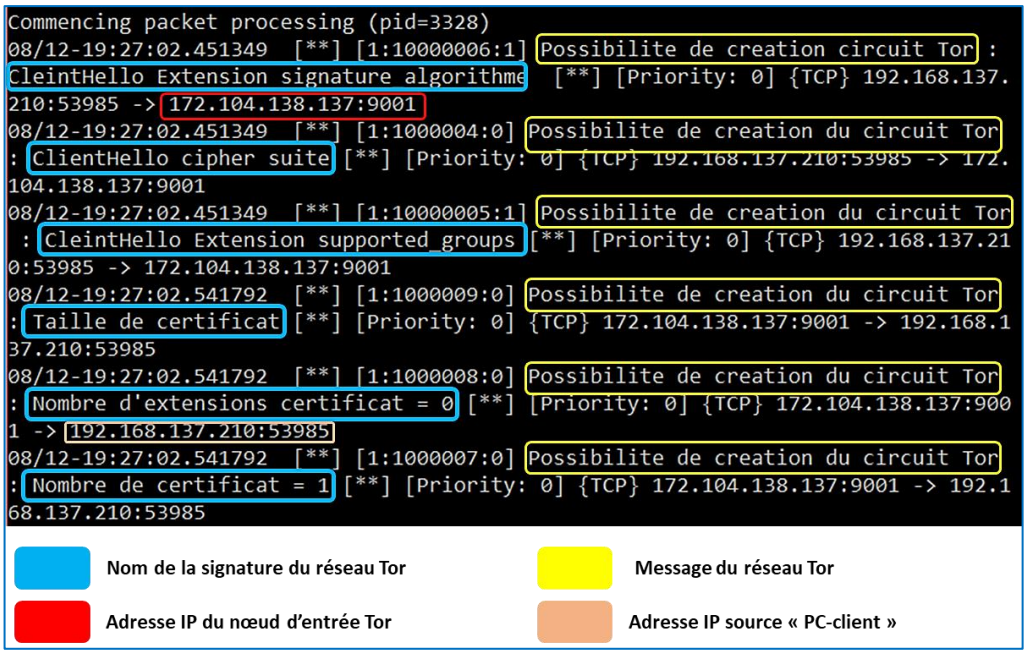

Figure IV.16 : *Détection de l'utilisation du navigateur Tor.*

Dès que nous démarrons le routeur I2P nous avons remarqué que Snort génère les six alertes au même temps, ces dernières correspondent aux règles du réseau I2P présentées dans le tableau ci-dessus. Ce qui signifie la détection de l'utilisation du réseau I2P.

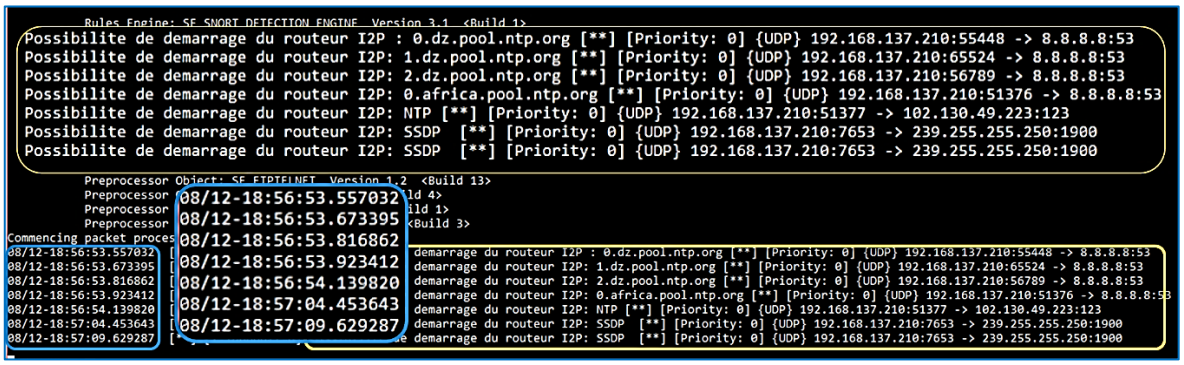

Figure IV.17 : *Détection de l'utilisation du réseau I2P.*

IV.5.4.C Syslog server

Pour pouvoir surveiller et analyser les alertes et autres sorties produites par Snort nécessite généralement un traitement et une présentation supplémentaires. Une façon de faciliter la surveillance de la sortie Snort est de la diriger vers un serveur de journal système « Syslog » afin obtenir un affichage conviviale et qu'un administrateur puisse surveiller l'activité de Snort à l'aide d'un serveur Syslog [49].

Syslog est un protocole définissant un service de journaux d'événements d'un système informatique. Il est un composant courant dans de nombreux environnements Unix

et Linux, mais ne se trouve généralement pas sous Windows, Plusieurs serveurs Syslog sont disponibles pour fonctionner sous Windows, y compris plusieurs produits gratuits [49].

1. Pour que Snort dirige la sortie vers le serveur Syslog, nous ouvrons le fichier Snort.conf et nous modifions la configuration de sortie pour Syslog :
   output alert_syslog: host = 127.0.0.1: 514, LOG_AUTH LOG_ALERT
2. Démarrage Snort :
   c: \ Snort \ bin> snort -i 4 -c c: \ Snort \ etc \ snort.conf -s

L'option **« -s »** de la commande de démarrage Snort dirige la sortie vers Syslog, en utilisant les paramètres du fichier Snort.conf. Sans sortie à l'écran, aucune activité n'apparaîtra dans la fenêtre de l'invite de commandes où Snort est en cours d'exécution, mais au fur et à mesure que les alertes se produisent, elles apparaîtront dans le serveur Syslog, comme illustré dans les figures ci-dessous [49].

Nous pouvons décomposer la fenêtre du serveur Syslog en 3 parties :

1. 1er partie représente les hôtes disponibles dans le réseau.
2. 2eme partie représente tous les évènements captés et envoyés par Snort au serveur Syslog par ordre chronologique.
3. 3eme partie représente les détails de l'évènement sélectionné.

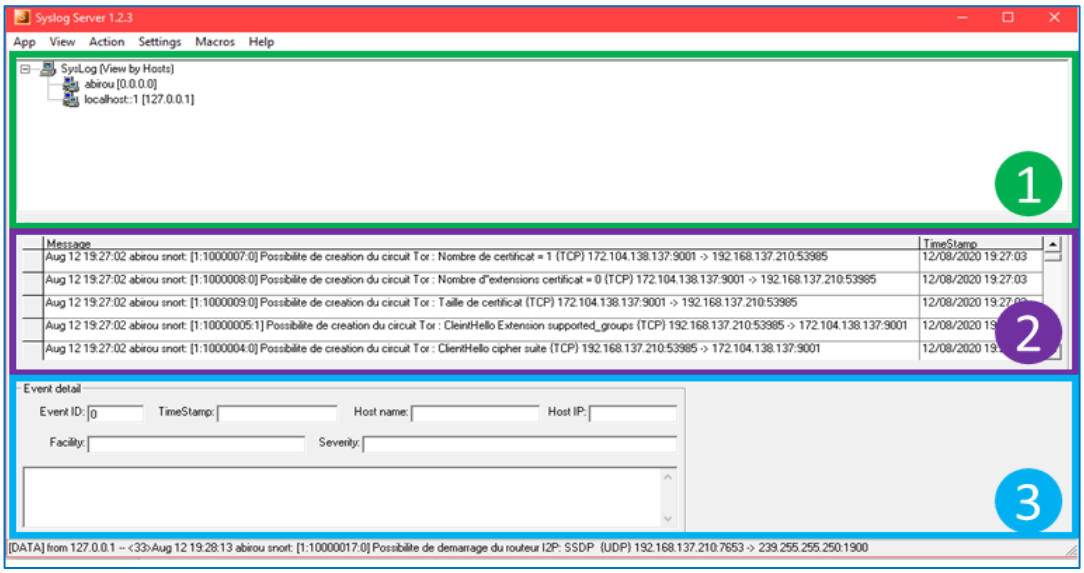

Figure IV.18 : *Fenêtre du serveur Syslog.*

La figure ci-dessous représente les évènements captés lors du lancement du navigateur Tor.

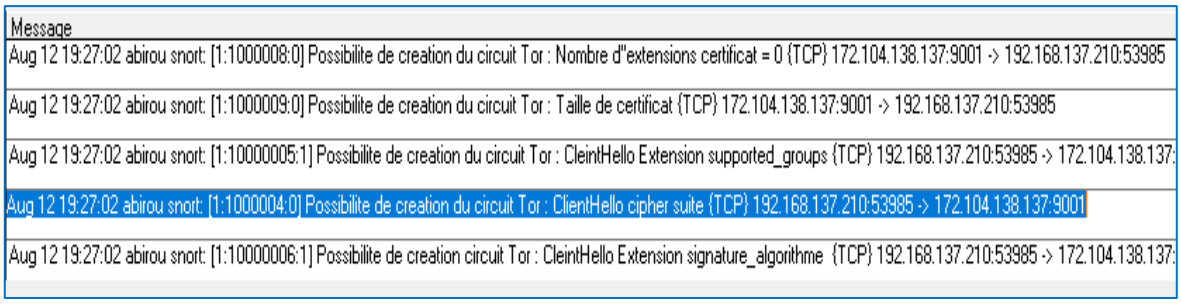

Figure IV.19 : *Les alertes Tor affichées dans serveur Syslog.*

La figure ci-dessous représente les détails de l'évènement sélectionné dans la figure ci-dessus.

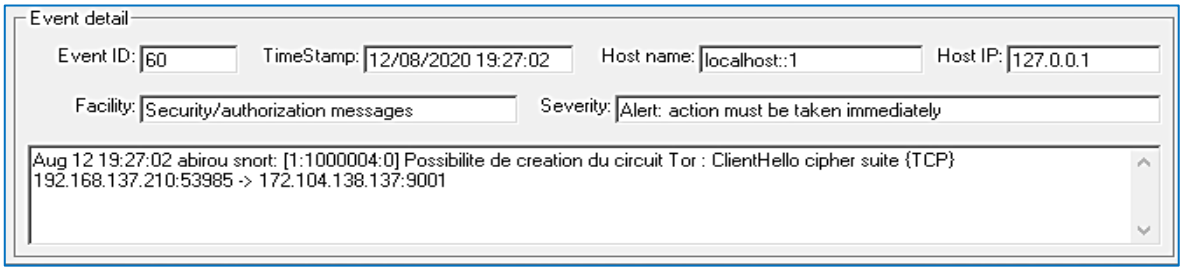

Figure IV.20 : *Détails de l'évènement Tor sélectionné.*

La figure ci-dessous représente les évènements captés lors du lancement du routeur I2P.

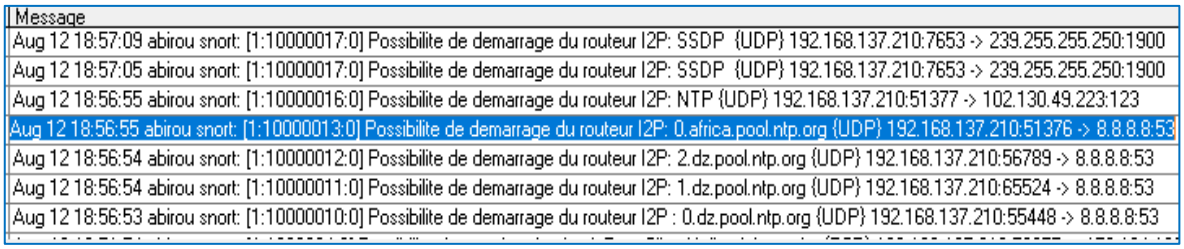

Figure IV.21 : *Les alertes I2P affichées dans serveur Syslog.*

La figure ci-dessous représente les détails de l'évènement sélectionné dans la figure ci-dessus.

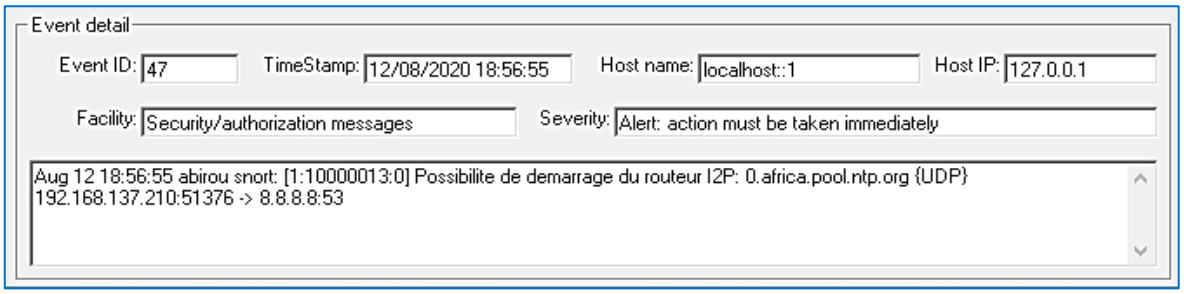

Figure IV.22 : *Détails de l'évènement sélectionné.*

IV.5.5 Constatation

D'après plusieurs tests effectués avec le navigateur Tor, nous avons obtenu la même adresse IP du nœud d'entrée, cela signifie que cette derniere est fixe pendant un certain temps, dans notre cas depuis que nous avons installé le navigateur Tor, notre adresse IP du nœud d'entrée n'a pas changé (deux à trois mois) cela confirme la théorie du chapitre précédant.

La combinaison des deux conditions dans nos règles Tor (numéro de port et signature) nous a permis d'eviter les fausses alertes, car ces deux conditions dans les six alertes déclenchées à la fois ne peuvent pas etre généré que par le navigateur Tor.

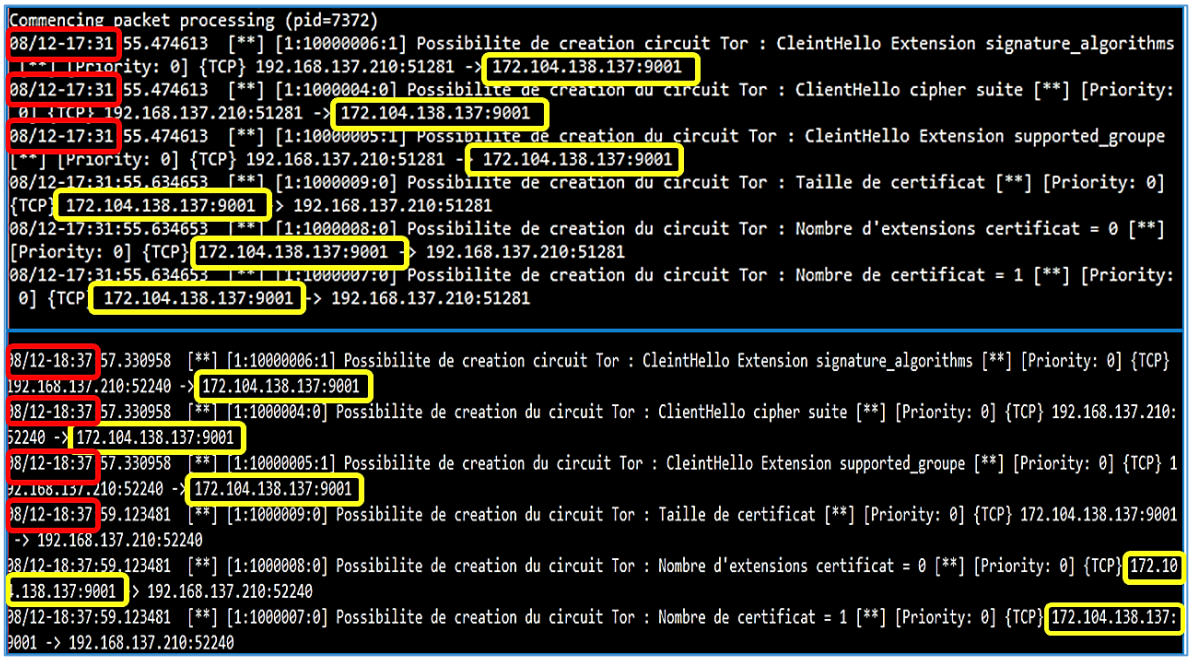

Figure IV.23 : *Alertes Tor declanchées dans des horaires differents.*

Nous pouvons constater des expériences qu'il est possible d'identifier le trafic I2P pendant la phase d'initialisation / d'amorçage car nous avons pu détecter rapidement les requêtes DNS vers internet. Le démarrage du routeur I2P commence toujours par demander les adresses IP des serveurs NTP de dz.pool.ntp.org, ce comportement sera distingué rapidement car généralement les PC dans une entreprise ne demande pas une requête NTP pour synchroniser l'heure.

IV.6 Discussion

Dans la gestion du réseau, les règles de pare-feu sont souvent utilisées pour autoriser ou filtrer le trafic. Les techniques de blocage populaires sont souvent basées sur le numéro de port, la signature du protocole et l'adresse IP. Cependant, les réseaux d'anonymat,

y compris Tor et I2P, sont conçus pour résister à la censure. En conséquence, toute tentative de bloquer définitivement ces réseaux pourrait causer des dommages collatéraux.

Pour la censure basée sur les ports, le blocage des ports de relais oignon (orports) ou des ports d'échange d'informations d'annuaire (dirports) est efficace pour bloquer les relais Tor momentanément, et le blocage du port UDP 123 empêcherait I2P de fonctionner correctement car le logiciel du routeur I2P a besoin du protocole NTP pour fonctionner correctement.

Une approche plus efficace est le filtrage des destinations. Pour implémenter cette approche, un censeur doit compiler une liste d'adresses homologues I2P actives et bloquer l'accès à toutes. Cette approche de blocage basée sur l'adresse aura un impact sévère sur le processus de formation de nouveaux tunnels I2P, empêchant ainsi les utilisateurs d'accéder au réseau I2P momentanément. En outre, un moyen plus simple mais toujours efficace d'empêcher les nouveaux utilisateurs d'accéder à I2P consiste à bloquer l'accès aux serveurs de réamorçage I2P, qui sont requis pour le processus d'amorçage. Par conséquent, les nouveaux utilisateurs ne pourront pas accéder au réseau I2P s'ils ne peuvent pas récupérer RouterInfo d'autres pairs.

IDS sauvegarde également les paquets TOR détectés (alertes Tor) dans un fichier log qui sera utilisé pour crée une liste des adresses IP destinations des paquets Tor, cette derniere sera implémenté dans un serveur proxy pour bloquer la connexion.

IV.7 Conclusion

Au cours de ce chapitre, nous avons pu détecter l’utilisation du réseau Tor et I2P en temps réel à l’aide des règles présentées précédemment, ce qui confirme la fiabilité de ces dernières. Cette détection a été effectuée en utilisant Snort en tant qu’un système de détection d'intrusion réseau (NIDS) sous Windows. Ensuite nous avons surveillé l'activité de Snort à l'aide d'un Syslog server pour une présentation supplémentaire de la sortie du Snort.

Conclusion générale

Tout au long de la préparation de notre projet de fin d'études, nous avons essayé de mettre en pratique les connaissances acquises durant nos études universitaires et cela dans le but de la comparaison et la détection de l'utilisation du réseau Tor et du réseau I2P dans une entreprise, pour protéger cette dernière des risques de sécurités liés à l'utilisation de ces deux réseaux.

Nous avons pu résoudre la problématique en répondant aux questions secondaire :

- Qu'est-ce que l'anonymat et la vie privée sur internet ?
- Quels sont les outils permettant l'anonymat ?
- Qu'est-ce qu'un réseau Tor et un réseau I2P et quel est leurs fonctionnement ?
- Y-a-t-il une difference entre le trafic Tor/I2P et le trafic du web ordinaire ?
- L'implémentation des signatures Tor et I2P dans un système de détection d'intrusion est-elle possible ?

Nous avons présenté l'anonymat et la vie privée sur internet, les différents outils qui peuvent la garantir et les deux réseaux anonymes Tor et I2P ainsi une comparaison théorique entres ces derniers par une étude bibliographique.

Nous avons pu comprendre que malgré Tor et I2P fournissent des fonctionnalités similaires, il existe des différences majeures entre eux. Tor fonctionne au niveau du flux TCP, tandis que le trafic I2P peut utiliser à la fois TCP et UDP. Tor a une architecture centralisée dans laquelle un ensemble d'autorités de répertoire assurent le suivi du réseau, tandis qu'aucune entité n'a une vue complète du réseau I2P en raison de sa nature décentralisée. En conséquence, nous avons effectué une détection du réseau Tor basé sur l'analyse et la comparaison entre le trafic Web des navigateurs ordinaires et le trafic du navigateur Tor. Tor déguise sa connexion pour qu'elle ressemble à une connexion HTTPS ordinaire. Nous avons analysé son processus de prise de contact TLS « TLS Handshake » afin d'obtenir les caractéristiques que nous avons utilisé pour identifier le trafic Tor.

Ensuite, nous avons effectué une détection du réseau I2P basé sur l'analyse du trafic lors du démarrage du routeur I2P. Au cours de la phase d'initialisation le routeur I2P a besoin du protocole NTP pour fonctionner correctement en envoyant des requêtes DNS

pour obtenir les adresses IP des serveurs NTP les plus proches. Nous avons utilisé ces caractéristiques pour identifier le trafic I2P.

Enfin, nous avons utilisé les caractéristiques des deux réseaux comme signatures numériques en les implémentant comme des règles dans un système de détection d'intrusion Snort, ce dernier a généré des alertes qui sont déclenchées dès le démarrage des deux réseaux. Cela nous a permis de confirmer que la détection de l'utilisation du réseau Tor et du réseau I2P dans une entreprise est possible.

Par ailleurs, notre méthode de détection soulève un certain nombre de questions ouvertes intéressantes telles que :

- Les mélanges de relais qui sont fournis au Tor client pour établir les circuits et les connexions ont besoin de plus de recherche, une évaluation approfondie de la politique d'allocation et de son potentiel de fuite de localisation ou d'autres informations d'identité pourrait être effectuée.
- Etudier les flux du réseau I2P en détail afin de fournir une compréhension plus complète du réseau dans son ensemble. Cela contribuera à une enquête efficace sur les activités d'I2P.

Enfin, les problèmes de sécurité dans une entreprise demeurent toujours des problèmes ouverts en conséquence beaucoup de pistes restent à explorer.

Bibliographie

[1] Récupéré sur : https://www.edx.org/

[2] J-L.Archimbaud : *Cours Interconnexion et conception de réseaux,* cel.archives ouvertes, CNRS (France), 11 janvier 2018.
Disponible sur : https://cel.archives-ouvertes.fr/cel-00561873

[3] G. Pujolle : *Cours réseaux et télécoms,* Edition Eyrolles, 2004.

[4] Cisco : *Cisco Certified Network Associate 1-chapitre 3,* Communications et protocoles réseau. Disponible sur : http://cisco.ofppt.info/ccna1/

[5] Entête TCP/UDP : https://www.frameip.com/entete-tcp/

[6] ACISSI : *sécurité informatique-Ethical Hacking-Apprendre l'attaque pour mieux se défendre,* 3ème édition Broché, 12 septembre 2012.

[7] J.Dordiogne : *Réseaux informatiques - Notions fondamentales*, 8e édition, ENI, Novembre 2019.

[8] P.Atelin, J.Dordiogne : *TCP/IP et les protocoles Internet*, 2ème édition, ENI. septembre 2008.

[9] S.Lohier, A.Quidelleur : *Le réseau Internet-Des services aux infrastructures*, DUNOD, 08 septembre 2010.

[10] R.Raveaux : *Pare-feux ou firewalls*, support de cours, 18 juillet 2007.
Disponible sur : http://romain.raveaux.free.fr/teaching/coursR4RT1parefeuRR.pdf

[11] G.Chamillard, S.Bobillier : *Ubuntu Linux - Création, configuration et gestion d'un réseau local d'entreprise (BTS, DUT Informatique)*, 3ème édition, Broché, 13 février 2013.

[12] J.Krier : *Les systèmes de détection d'intrusions*, support de cours, 21 juillet 2006.

[13] R. Stamboliyska : *La face cachée d'internet -hackers, dark net...*, Larousse, 2017.

[14] L.Poinsot : *Introduction à la sécurité informatique*, support de cours, Université Paris. Disponible sur :
https://lipn.univ-paris13.fr/~poinsot/save/INFO%203/Cours/Cours%201.pdf

[15] Amnesty International : *CHIFFREMENT UNE QUESTION DE DROITS HUMAINS*, mars 2016. Disponible sur : https://www.amnesty.org/download/Documents/POL4036822016FRENCH.PDF

[16] G.Pillot : *Anonymat et vie privée sur internet*, Mémoire, Université LAVAL, Québec, Canada, 2018. Disponible sur : https://corpus.ulaval.ca/jspui/bitstream/20.500.11794/32469/1/34754.pdf

[17] Récupéré sur : https://sites.google.com/site/enjeuxdunumerique/cours-2011

[18] Samuel : *Quelles sont les meilleures solutions pour surfer dans l'anonymat ?,* 26 aout 2019. Disponible sur: http://www.calvados-strategie.com/quelles-sont-les-meilleures-solutions-pour-surfer-dans-lanonymat%E2%80%89/

[19] A.AJDINI : *Étude et conception d'un service assurant l'anonymat*, mémoire, Haute école de gestion de Genève, 30 septembre 2019. Disponible sur : https://core.ac.uk/download/pdf/286405998.pdf

[20] J-P.Timpanaro, T.Cholez, I.Chrisment, O. Festor : *Evaluation of the Anonymous I2P Network's Design Choices Against Performance and Security,* Proceedings of the 1st International Conference on Information Systems Security and Privacy, SciTePress, Angers France. pp.46-55, février 2015. Disponible sur : https://hal.inria.fr/hal-01238453/file/I2P-design-vs-performance-security.pdf

[21] Les consultants du cabinet de conseil XMCO : *XMCO-ActuSecu-45-ShadowBrokers_I2P*, magazine numérique N° 45, 69 rue de Richelieu 75002 Paris – France, janvier 2017. Disponible sur : https://www.xmco.fr/actu-secu/XMCO-ActuSecu-45-ShadowBrokers_I2P.pdf

[22] P.Liu, L.Wang, Q.Tan, Q.Li, X.Wang, J.Shi : *Empirical Measurement and Analysis of I2P Routers*, Institute of Computing Technology, China, 9 septembre 2014.

[23] I2P : Cryptography : https://geti2p.net/en/docs/how/cryptography

[24] I2P : Applications Supported : https://geti2p.net/fr/docs/applications/supported

[25] A.Duquenoy, A.Durot : *Exploration du réseau Tor*, porjet IMA 4, Polytech Lille, 15 mai 2018.

[26] R.Dingledine, N.Mathewson, P-S.Naval : *Tor : The Second-Generation Onion Router*, article, The Free Haven Project, Massachusetts Institute of Technology. Juin 2004. Disponible sur : https://www.researchgate.net/publication/2910678

[27] Tor : hidden service : https://k-lfa.info/hidden-service-tor/.

[28] I2P : Comparaison Tor : https://geti2p.net/fr/comparison/tor.

[29] CCProxy : https://www.youngzsoft.net/ccproxy/

[30] I2P : I2PTunnel : https://geti2p.net/fr/docs/api/i2ptunnel

[31] I2P : Outproxy : https://geti2p.net/en/faq#outproxy

[32] Reddit : https://www.reddit.com/

[33] Performance-réseau : https://accedian.com/fr/blog-fr/

[34] Wireshark : https://www.wireshark.org/

[35] R.Weaver, D.Weaver, D.Farwood : *Guide to Network Defense and Countermeasures*, Cengage Learning,1 janvier. 2013.

[36] TLS Handshake : https://www.cloudflare.com/fr-fr/learning/ssl

[37] TLS Handshake : https://www.thesslstore.com/blog/explaining-ssl-handshake/

[38] F-A.Saputra, I-U.Nadhori, B-F.Barry : *Detecting and blocking onion router traffic using deep packet inspection*, International Electronics Symposium, Denpasar Indonesia, 29-30 Septembre 2016.

[39] ANSSI : *Recommandations de sécurité relatives à TLS*, LATEX, 19 août 2016.

[40] [RFC 7366] Encrypt-then-MAC for Transport Layer Security (TLS): https://tools.ietf.org/html/rfc7366.

[41] A-O.Granerud : *Identifying TLS abnormalities in Tor*, mémoire, Université Gjøvik, 2010.

[42] Rdnsequences : https://hexdocs.pm/x509/X509.RDNSequence.html

[43] NTP Pool Project : https://www.ntppool.org/fr/

[44] Protocole SSDP : https://www.lemondeinformatique.fr/actualites/lire-le-protocole-ssdp-nouveau-levier-pour-lancer-de-puissantes-attaques-ddos-58906.html

[45] S.Northcutt, J.Novak : *Détection d'intrusion de réseau*, 3ème Edition, Broché, 29 juillet 2017.

[46] K.Cox, C.Greg : *Sécurité réseau avec Snort et les IDS*, Broché, 15 décembre 2004.

[47] A.TOUATI : *Détection d'intrusions dans les réseaux LAN : installation et configuration de l'IDS-SNORT*, Mémoire, Université Bejaïa, 2016.

[48] Snort : rules : https://www.snort.org/resources#documents

[49] Server Syslog : https://www.securityarchitecture.com/learning/intrusion-detection-systems-learning-with-snort/installing-syslog/