

Rapport Projet Deuxième Année

1) Le réseau TOR

a) Qu'est ce que c'est ?

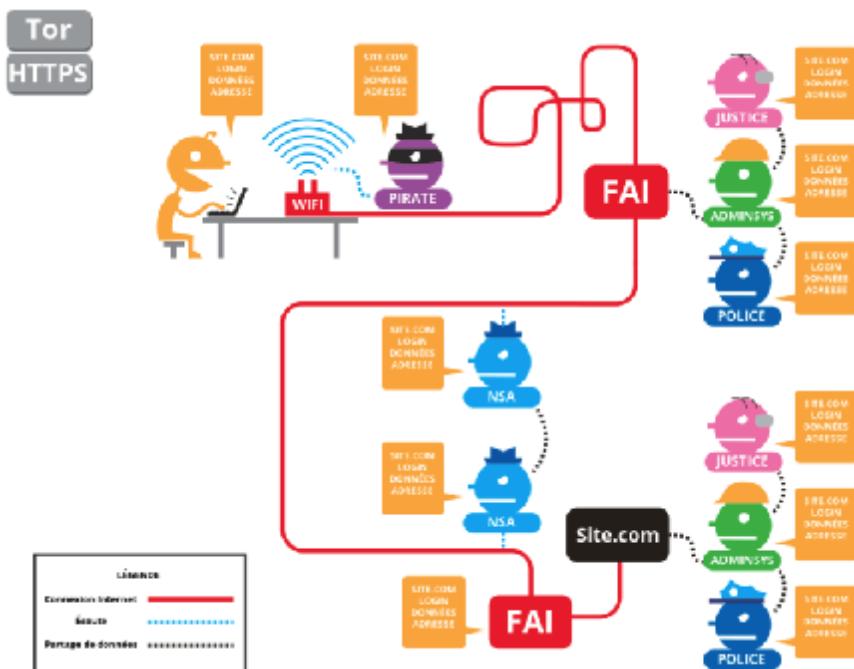
TOR est l'accronyme de The Onion Router. Le réseau a été conçu par la marine américaine dans le but de protéger les communications gouvernementales pendant les opérations secrètes. Le département de la défense a financé le projet à plus de 60% tandis que les autres financements proviennent des organisations qui protègent la vie privée tel que des associations de journalistes, des groupes d'activistes ou l'EFF. TOR est désormais une organisation à but non lucratif qui se concentre sur la protection des données personnelles et l'anonymat des internautes sur le web. C'est un projet open source, c'est à dire que tout le monde peut voir le code et le corriger si nécessaire.



Le réseau TOR permet d'accéder au “*DEEP WEB*” ou l'internet profond. Comme cela est illustré dans la figure précédente, une partie du réseau internet est enfouie, inaccessible depuis les moteurs de recherche communs tels que GOOGLE, SAFARI... Le deep web correspond à tout ce qui n'est pas indexé par ces moteurs de recherche. C'est bien sûr le repère de tout commerce illégal. Par exemple, en octobre 2013, Silk Road, site de traffic de drogue trouvable uniquement sur le deep web a été “trouvé” et fermé. C'est aussi le repère des organisations gouvernementales ou des groupes d'activistes.

b) Comment TOR garde l'anonymat?

L'objectif de TOR est que l'utilisateur puisse utiliser internet tout en restant totalement anonyme. Pour cela, il faut que personne ne soit capable de retrouver la source de la demande ou la destination de l'information. Lorsque nous nous connectons à un site web, notre ordinateur essaie de se connecter directement à son serveur par la route la plus courte. Ce qui prime est la rapidité et l'efficacité. Notre adresse IP est donc recensée comme point de départ de la communication et il n'y a aucun anonymat.

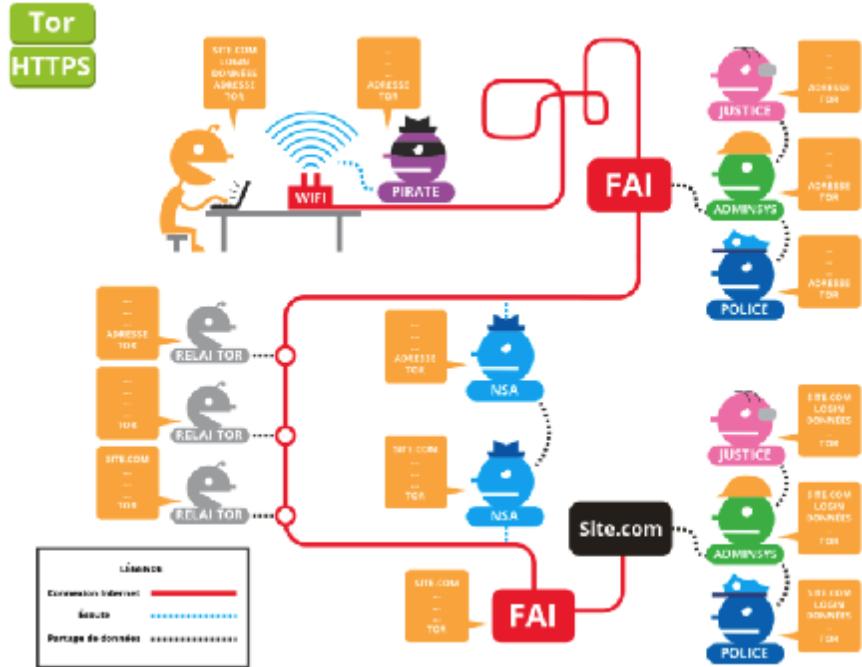


Toute personne ayant accès à la requête connaît l'émetteur, le destinataire et les données.

Pour éviter cela, TOR rompt celle ligne entre notre ordinateur et ce serveur distant. TOR utilise l'Onion Routing qui est une technique de communication anonyme sur un réseau. Les messages sont chiffrés en continu en passant de noeuds en noeuds. Ces noeuds sont aussi appelés routeurs Onions. Le terme onion se réfère aux différentes couches de chiffrement effectuées par des relais anonymes qui protègent les messages. Le routage devient alors totalement invisible.

Si une personne A envoie une requête au serveur B, la requête va passer par plusieurs noeuds. Cette requête va d'abord être chiffrer avec le clé publique du noeuds de sortie puis re-chiffrer par la clé publique de l'avant dernier noeud et ainsi de suite jusqu'au premier noeud auquel elle va être envoyé. Ce principe per-

met que le premier noeud connaisse seulement l'expéditeur mais pas la destination, les noeuds intermédiaires ne connaissent que le noeuds précédent et le suivant et le noeud de sortie ne connaît que le destinataire. Tor utilise ce même chemin pendant plusieurs minutes puis le change pour qu'aucun lien ne puisse être établi.



2) Familiarisation avec les technologies

a) Markdown

un langage simple et léger basé sur un balisage moins lourd que celui utilisé pour html. Utilisé à la base pour écrire un document html il a été adapté et permet aujourd'hui via l'intermédiaire de Pandoc (décrit ci-dessous) de créer à partir d'un document .md une multitude de document tel que .doc , .odt... Ce langage permet une grande facilité de lecture et d'écriture.

b) Pandoc

Un outil de conversion supportant un très grand nombre de formats.Dnas notre cas très utile car il supporte le format Markdown et contient de plus des éléments

c) Vagrant

Permet de créer une machine virtuelle avec tous les paramètres souhaités de manière très rapide. Ainsi notre environnement de travail est très rapidement disponible. Possibilité de récupérer les boxs sur internet facilement en choisissant sa distribution (nous avons choisi debian 7.2). On tape la commande : vagrant init NomMachine LienBox Cela entraîne la création d'un fichier vagrantfile

3) Administration web

1) Heberger un hidden service

- Télécharger Tor sudo apt install tor
- Configurer TOR Dans le fichier etc/tor/torrc : Décommenter ces lignes : DataDirectory /var/lib/tor HiddenServiceDir /var/lib/tor/hidden_service/ HiddenServicePort 80 127.0.0.1:80
Recharger TOR : sudo service tor reload
- Changer les clefs données par defaut Dans /var/lib/tor/hidden_service, modifier les fichiers hostname et private_key. Obtention de ces clefs avec scallions : <https://github.com/lachesis/scallion/raw/binaries/scallion-v2.0.zip> Commande set list .
- Telecharger apache sudo apt install apache2
- Configurer apache Dans le fichier ports.conf (ect/apache2): Remplacer la ligne Listen 80 par Listen 127.0.0.1:80 Dans sites-enabled, Remplacer par ainsi que Servername.

2) Devenir un relais tor de sortie

- Modifier le fichier etc/tor/torrc # Paramètres du relais Tor RunAsDaemon 1 # Démarrer Tor en tant que tâche de fond DirPort 9030 # Port pour le référencement du relais ORPort 9001 # Port du relais SocksPort 0 # Ajoutez cette ligne si vous n'utiliserez pas Tor sur votre réseau local Nickname RelayName # Nom du relais ContactInfo contact@domain.tld # Une adresse e-mail de contact Exitpolicy reject : # Rejette le trafic sortant afin de devenir un nœud intermédiaire RelayBandwidthRate 1250 KBytes # Limite de la bande passante pour le relais RelayBandwidthBurst 1450 KBytes # Burst de bande passante pour le relais (au cas où la bande passante maximum est atteinte)
- Relancer Tor : service tor restart

- Vérifier que les ports 9001 et 9030 sont ouverts

<https://themimitoof.fr/mettre-en-place-un-relais-tor/> <https://blog.torproject.org/lifecycle-new-relay>

2) VirtualHost avec Apache2

Il s'agit simplement d'associer plusieurs noms DNS à une seule adresse IP.

- Dans un premier temps nous allons définir des noms DNS pour nos sites : monsite1.fr et monsite2.fr Il faut ajouter ces informations au fichier /etc/hosts pour que la résolution DNS pointe sur la machine locale (127.0.0.1) : 127.0.0.1 monsite1.fr 127.0.0.1 monsite2.fr
- Puis nous allons créer deux dossiers ,dans le dossier /var/www qui est par défaut la racine d'apache, via ces deux commandes : sudo mkdir /var/www/msite1 sudo mkdir /var/www/msite2
- Nous allons créer les fichiers de configurations pour apache. Dans /etc/apache2/sites-available se trouve un fichier nommé default. Modifiez les lignes suivantes :

```
ServerName monsite1.fr ServerAlias www.monsite1.fr DocumentRoot
/var/www/msite1
```

- enregistrez le fichier sous le nom monsite1.conf puis modifiez-le en mettant cette fois monsite2 à la place de monsite1, puis ré-enregistrez sous le nom de monsite2.conf.
 - Pour terminer, il vous suffit de créer des liens des deux fichiers nouvellement créés dans le dossier /etc/apache2/sites-enabled. Pour ce faire, une commande a été faite spécialement :
- ```
sudo a2ensite monsite1.conf sudo a2ensite monsite2.conf
```
- Puis, afin de recharger la configuration d'Apache :
- ```
sudo /etc/init.d/apache2 reload
```

3) Tor : The onion rooter

1) Pourquoi ?

Le projet Tor est une organisation à but non lucratif dédiée à la recherche, le développement et l'éducation sur l'anonymat et la vie privée en ligne. Il répond aux problèmes de confidentialités des données personnelles ainsi que de contourner la censure. Le TOR permet de naviguer sur le web de façon anonyme et sécurisé. En effet, quand nous utilisons le web les sites visités peuvent enregistrer notre adresse IP et donc facilement remonter à nous par

ce biais. Le réseau TOR permet de contourner ce problème en cachant notre adresse IP. En Juin 2017, on estime le nombre d'utilisateurs de Tor à 2 375 000 par jour. 21,52% des utilisateurs se trouvent aux Emirats Arabes Unis ou Tor est illégal.

2) Le routage

Tor fonctionne grâce à la participation de ses utilisateurs. Certains acceptent d'être des relais TOR (ou des noeuds). Quand un utilisateur souhaite accéder à internet via le TOR, sa requête doit passer par trois noeuds TOR choisis au hasard. Avant d'atteindre le premier relais, la requête sera chiffrée trois fois et chaque noeud va enlever une couche de chiffrement (peler une couche). Il existe trois types de relais. Chacun a des missions distinctes et des modes de fonctionnement différents : * Le noeud d'entrée * Noeud Gardien * Bridge * Le noeud intermédiaire * Le noeud de sortie

Le circuit TOR désigne les trois relais par lesquels va transiter le flux de données.

Le premier noeud reçoit la requête chiffré 3 fois et il est chargé de faire transiter les données jusqu'au destinataire. Il est le seul à connaître l'émetteur. Ce noeud peut-être un Noeud Gardien ou un Bridge. La liste des Noeuds Gardiens est publique donc si l'on utilise le TOR avec un noeud gardien comme noeud d'entrée, le FAI ou toute personne se trouvant entre nous et le premier relais sait que l'on utilise le réseau TOR mais si elle ne connaît ni le contenu des échanges ni le destinataire. TOR garde secret la liste des bridges. S'il l'on se trouve dans un pays où TOR est bloqué ou illégal, il peut être utile d'utiliser comme premier relais un bridge pour ne pas être bloqué par le FAI. Pour cela il faut le demander au TOR project.

Cette liste de noeuds TOR est disponible sur le site Tor Metrics. Pour être à jour, le TOR a dû développer des serveurs particuliers appelés autorités d'annuaire. Ces neuf autorités sont chargées de mettre à jour un annuaire de tous les relais Tor disponibles. En effet, les noeuds peuvent rentrer et sortir à leur gré du réseau ou changer d'état. Toutes les heures, 8 autorités mettent à jour l'annuaire des relais publics et 1 autorité gère les bridges. Cet annuaire est appelé le consensus. Pour rendre possible ce répertoriation, les relais envoient périodiquement leurs données aux autorités. Ces dernières analysent les informations relatives à ces relais et décident s'ils peuvent devenir noeud gardien ou alors s'ils sont compromis et doivent alors recevoir le statut Bad Exit.

3) Etablissement du circuit

Supposons que Alice veuille se connecter anonymement à monsite.fr. Elle utilise pour cela le TOR. Son ordinateur commence par télécharger le consensus pour récupérer la liste des noeuds TOR. Ensuite sa machine va créer un circuit TOR en sélectionnant aléatoirement 3 noeuds. Elle va alors négocier une connexion

chiffrée avec chaque relai du circuit utilisé. Pour cela elle va récupérer les 3 clés publique des trois relais depuis un serveur de clés. Par sécurité, Tor renouvelle le circuit toutes les 10 minutes pour brouiller les pistes et limiter les informations qu'un attaquant contrôlant un noeud pourrait récupérer. Cependant, Le noeud d'entrée est fixe et ne change que tous les 2 à 3 mois pour un noeud gardien. Ce temps est appelé la "période de rotation".

Tout d'abord la machine d'Alice doit communiquer avec le noeud gardien et donc entamer une poignée de main. Le noeud signe cette poignée de main avec sa clé privée et le client vérifie la signature à l'aide de la clé publique récupérée précédemment. Cela permet au noeud gardien de s'authentifier. La seconde étape de communication avec ce noeud est le partage d'une clé de chiffrement symétrique (appelée clé de session 1) suivant le protocole d'échange Diffie-Hellman. Le noeud gardien connaît donc l'adresse IP du client.

Il faut maintenant négocier une clé symétrique avec le noeud intermédiaire. Le client envoie au travers du noeud gardien, une demande de connexion au noeud intermédiaire. Cette demande est chiffrée avec la Clé 1. Le noeud gardien déchiffre la demande, la transmet au noeud intermédiaire. Ce dernier lui répond en signant avec sa clé privée. Le noeud gardien chiffre cette réponse avec la clé 1 et la transmet au client. Le client déchiffre et vérifie la signature pour pouvoir ensuite négocier une deuxième clé de session symétrique avec le noeud intermédiaire. Cette négociation se fait de la même façon que l'authentification c'est à dire en passant par le noeud gardien.

Pour finir, il faut négocier la clé symétrique 3 avec le noeud de sortie. Cela est le même procédé qu'avec le noeud intermédiaire mais les données sont chiffrées deux fois. Une fois avec la clé 1 et une fois avec la clé 2. Le noeud Gardien ne voit uniquement passé que des données chiffrées avec la clé 1 et 2. Il ne sait donc pas qui est le destinataire des échanges (le noeud de sortie).

4) Connexion anonyme à un service publique

Le client est donc maintenant en possession de trois clés de chiffrement symétrique correspondant à chacun des noeuds. L'échange des données peut donc commencer. Le client envoie sa requête pour se connecter à monsite.fr mais il n'effectue pas lui-même la requête DNS pour éviter de compromettre son anonymat. C'est donc le noeud de sortie qui va se charger d'effectuer la requête DNS. La requête qui quitte le client est chiffrée avec la clé 3 puis la clé 2 et pour finir la clé 1. Chaque noeud va enlever une couche de chiffrement pour que l'information puisse parvenir au destinataire.

5) Connexion anonyme à un service caché

Un service caché n'est visitable qu'en utilisant le TOR. Il n'est pas enregistré dans le DNS et n'est pas localisable. Un service caché est un service en .onion.

C'est ce qui est communément appelé le deep web.

Maintenant Alice veut se connecter à mon service caché : wslgdkhq.Onion

a) HSDIR

Le service caché nécessite d'établir des circuits vers des points d'introduction. Ces points d'introduction vont permettre au client de pouvoir se connecter au service. Ce dernier établit donc plusieurs circuits de trois noeuds et demande aux relais de sortie de servir de points d'introduction. Il récupère ainsi leur IP. [Voir annexe 1]

Ensuite, le service ne peut pas être répertorié dans le DNS s'il veut garder son anonymat. Cependant il doit signaler sa présence pour permettre au client de se connecter. Pour cela il demande aux points d'introduction de maintenir la connexion et pendant ce temps il établit un autre circuit vers un Hidden Service Directory(HSDir). Il fournit alors au HSDir son descripteur composé des IP des points d'introduction, de la clé publique du service caché ainsi que de la signature des deux éléments précédant faite avec la clé privée correspondante. (6 HSDir seront en possession de ce descripteur). [Voir annexe 2]

Le descripteur est calculé comme ceci : descriptor-id = H(permanent-id | H(time-period | descriptor-cookie | replica)) Fonction de hashage sha1 en 2013. Est-ce que c'est passé à sha256? replica : 0 ou 1 time-period : (current-time + permanent-id-byte * 86400 / 256) / 86400 permanent-id : c'est un dérivé de la clé publique du service caché.

| 1 | Onion Address | Descriptor ID | Requests |
|----|-------------------|----------------------------------|----------|
| 2 | ----- | | |
| 3 | silkroadvb5piz3r | cjzls3i2mbj4hjnquamuvznihues4xh4 | 16387 |
| 4 | silkroadvb5piz3r | m6yz6gqrmu35twduumixzr2mqtxdo3er | 10891 |
| 5 | Sonwnspjvuk7cwvk | 6t44eim223ypmb2ueokcsfco5vzvryfm | 1413 |
| 6 | silkroadvb5piz3r | hadco5o7rmh2vcamg7mdzqk1prqffyyh | 558 |
| 7 | silkroadxmx45vk4 | 6tyqo2bf7xclfbmrtrxwm7mgb3z4s5ui | 197 |
| 8 | atlantisrky4es5q | hdj7wkuaigt7iicqf77gyzbo7zyvq7wf | 165 |
| 9 | atlantisrky4es5q | m6y4s2utv4kxgdczv7t3gbmoloezbzf | 161 |
| 10 | atlantisrky4es5q | 6r3z4tlr2vv15z34v5lcuaqckgjvtr7s | 129 |
| 11 | silkroadxmx45vk4 | m6eczdmpjse3jdfw54cv4nxcc6s34eku | 107 |
| 12 | sheep5u64fi457aw | c17dpz6emlh2pxpshovnxjbyjqnp3luo | 59 |
| 13 | silkrovafuce2ur2 | 6r6w7ncln5mo4hdwq6yh7f2hf3hlag2u | 14 |
| 14 | sheep5u64fi457aw | rzq5pd4ehayz4yker7dhmvpubm4we7om | 9 |
| 15 | silkroadopn752d1 | cja5ppzzmkrzvr2pgp2c2z6mtuc7m7yk | 4 |
| 16 | silkroaddr5cd6wbz | m4n2afulpiln4n42l7wlg36wy6okkqrh | 3 |
| 17 | silkroadfqmteec4 | cjcyh7mm6lzzuqka3vlq67upyt5zwd7b | 2 |

[8]

Un HSDir est un noeud Tor comme les autres mais il remplit une fonctionnalité

supplémentaire : il reçoit les informations sur les services cachés pour signaler leur existence et permettre aux clients de les contacter. Le consensus permet à un noeud de devenir HSDir.

b) Etablissement de la connexion

La machine d'Alice a téléchargé les consensus et elle possède donc les IP des HSDir. Elle peut donc, toujours en utilisant le TOR, télécharger le descripteur du service caché et vérifier la signature. La machine d'Alice connaît donc maintenant les IP d'introduction de mon service caché. [Voir annexe 3]

Comme pour se connecter à un service public, la machine d'Alice va créer aléatoirement un circuit avec trois noeuds TOR. Le noeud de sortie sera le point de rendez-vous. La machine fournira sous forme de cookie un secret à ce point de rendez-vous. Ce dernier permettra d'authentifier le service caché. [Voir annexe 4]

Le machine d'Alice garde en attente ce cette connexion. Par ailleurs, elle crée un nouveau circuit de façon à ce que le noeud de sortie communique avec un point d'introduction du service caché. Elle peut ensuite communiquer à ce service : l'IP du point de rendez-vous, le secret qui a été dit à ce point de rendez-vous et la première partie de l'échange de Diffie-Hellman pour la création d'une clé symétrique entre la machine d'Alice et le service caché. Toutes ces informations sont chiffrées avec la clé publique du service caché. [Voir annexe 5]

Le service caché contacte ensuite le point de rendez-vous pour s'authentifier puis en passant par ce noeud communique avec le client pour terminer l'échange de la clé symétrique de Diffie-Hellman. Maintenant, Alice et le service caché peuvent communiquer de façon sécurisé. Il n'y a pas trois noeuds Tor sur leur circuit mais 6. Les trois premiers enlèvent chacun une couche de chiffrement et les trois derniers en remettent chacun une. La connexion est chiffrée du client au service caché. [Voir annexe 6]

6) Les Vulnérabilités de Tor

a) Vulnérabilité exploitant le JavaScript

Le JavaScript est un langage de programmation utilisé par la majorité des sites web. Ce langage est exécuté côté client. Il est dangereux car il existe des exploits javascript qui peuvent être envoyés par le serveur pour faire exécuter du code malicieux par votre ordinateur. Par exemple, il est possible d'injecter du code javascript exploitant la vulnérabilité par l'hébergeur de service cachés. Ensuite, le code exécuté par la machine cible de l'utilisateur (le payload) récupère le nom de la machine et l'adresse mac, et l'envoie sur un serveur via une connexion non torrifiée, ce qui permet également de récupérer l'IP réelle. Il est donc vivement conseiller de désactiver le JavaScript si l'on veut utiliser le TOR en toute sécurité. Cela peut être un gros inconvénient pour consulter des sites publics car la plupart

ne fonctionne pas sans JavaScript, cependant les services cachés, pour la plupart, n'utilise pas ce langage. réf : [1]

b) Ecoute du noeud de sortie.

Si l'on contrôle un noeud de sortie ou que l'on est l'homme du milieu entre un site web public et un noeud de sortie TOR, nous avons accès au trafic en clair. Dans les requêtes, il peut se trouver des informations sensibles tel que des identifiants, mots de passe ou informations personnelles qui permettraient de vous lier à cette requête. Cela ne fonctionne pas pour les services cachés car les données sont chiffrées bout en bout. Exitmap et HoneyConnector sont deux procédés de scan des noeuds de sorties, ils ont pour but de distinguer les noeuds qui seraient corrompus. Exitmap permet de détecter les manipulations du trafic en établissant une connexion Tor vers un leurre contrôlé par le Torproject. On sait ce qu'on envoie dans le réseau, et on regarde ce qui arrive sur le leurre. Si le trafic a été modifié, alors ça veut dire que le noeud de sortie modifie les trames réseau. Honeyconnector permet de détecter le sniffage passif du trafic (C'est à dire la récupération des informations sans les modifier). Concrètement, on envoie via Tor un couple unique "identifiant/mot de passe" sur un leurre contrôlé par le Torproject. Ensuite, si une tentative de connexion a lieu ultérieurement sur ce leurre, alors on sait que le noeud de sortie par lequel ce couple d'identifiant est passé l'a intercepté. réf : [1]

c) Analyse de trafic

Grâce aux multiples couches de chiffrement utilisé par TOR le message même s'il est intercepté n'est normalement pas lisible. Cependant la taille ou la fréquence d'envoi peuvent donner des informations sur le type de communication ou le destinataire. Pour réaliser une analyse du trafic il faut dans la plupart des cas avoir accès à 2/3 du trafic soit le noeud d'entrée et le noeud de sortie. Ainsi on observe un certain motif en entrée et si l'on retrouve ce motif en sortie alors on sait d'où vient le message. Une attaque un peu plus complexe consiste à faire transiter un trafic important sur un relais Tor spécifique à destination d'un serveur détenus par l'attaquant. Quand l'attaquant reçoit le trafic il peut déduire la latence induite par son trafic sur le relais et ainsi chercher d'autres trafics sur le réseau qui seraient passer par ce noeud donc affecter par la même latence. Il existe des moyens pour empêcher ce genre d'attaque de fournir des résultats cohérents et utilisables tels que rajouter du trafic parasite ou introduire des perturbations aléatoires de débit. Le problème de ces solutions est leur impact sur la rapidité des connexions Tor qui sont déjà reconnues comme relativement lentes par les utilisateurs. Tor a pris le parti de considérer que la complexité de l'analyse et de l'exploitation des résultats garantissait un niveau de sécurité assez important. réf : [1]

c) Fingerprint de la souris

Lorsque l'on navigue sur le web cela peut paraître banal mais la façon de déplacer sa souris ou la manière de cliquer est différentes selon les équipements mais aussi selon les personnes. L'analyse de ces différentes caractéristiques pourrait permettre d'identifier de manière unique un individu. Pour cela il faudrait disposer d'une base de données, regroupant les caractéristiques collectées sur le web classique (de manière non anonyme), et l'utiliser sur Tor pour désanonymiser l'utilisateur. Ce type de vulnérabilité n'est que théorique mais il met en évidence une éventuelle faille. De plus les processus utilisés pour tracker ces informations nécessitent Javascript, raison de plus pour désactiver JavaScript lorsque l'on se trouve sur Tor. réf : [1]

c) Faiblesse des clés d'authentification

De très nombreux relais Tor utilisent des versions de Tor anciennes qui ne sont pas mise à jour et qui utilisent donc des clés de 1024 bits qui sont actuellement obsolètes. Si un attaquant arrive à casser la clé du relais Tor alors il pourrait usurper l'identité de ce dernier et forcer les connexions à passer par lui. Certaines clés publiques permettraient même de remonter par des procédés mathématiques à la clé privée ce qui ne devrait jamais être possible. Les recherches menées sur les clés permettant l'authentification montre qu'un nombre non négligeable de relais ont vu leurs clés volontairement modifiées afin de nuire à l'anonymat de certains services cachés. réf : [1]

Comment bloquer Tor

Liste des IP TOR avec Proxy Squid

Au démarrage Tor va vous demander si votre réseau est équipé d'un proxy. Si vous cochez non il va essayer de contacter des « IP Tor » à travers la passerelle configurée sur votre ordinateur et va scanner tous les ports qui peuvent potentiellement sortir sur internet.

La première sécurité est donc d'empêcher les clients de sortir directement sur internet et de mettre en place un proxy. Les clients ne pourront donc pas sortir sur internet sans passer par lui. Seul le proxy sera autorisé à sortir sur internet. Tor va donc scanner et ne jamais trouver de « porte de sortie ».

Mais Tor (s'il est bien configuré) peut aussi passer par un proxy pour contacter directement des « IP Tor ». Son principe de fonctionnement est le suivant : Tor contacte une adresse IP par le proxy et ne ferme pas la connexion, il la laisse ouverte et fait transiter la navigation Tor sur ce « tunnel » ouvert.

La solution est donc de bloquer les connexions vers les IP de Tor. Ce site <https://www.dan.me.uk/torlist/> liste la plupart des adresses IP Tor disponibles sur le net si l'on veut établir une blacklist des IP. Le meilleur moyen resterait

de constituer une whitelist pour n'autoriser qu'un lot fini d'adresses mais cela peut-être compliqué dans certains contextes.

Exemple avec Squid : Pour contrôler tout ce qui passe par votre serveur proxy, vous pouvez utiliser ce que l'on appelle les ACL (Access Control List). Les ACL sont des règles que le serveur applique. Cela permet par exemple d'autoriser ou d'interdire certaines transactions. On peut autoriser ou interdire en fonction du domaine, du protocole, de l'adresse IP, du numéro de port, d'un mot, on peut aussi limiter sur des plages horaires.

On récupère la liste des ip tor via ce lien : <https://www.dan.me.uk/torlist/>

On les stock dans un fichier :

/etc/squid3/iptor

On créer notre acl dans :

/etc/squid3/squid.conf:

et on ajoute l'acl avec le http_acces qui va avec :

acl tor dst "/etc/squid3/torlist" http_access deny tor

Apache2 et la commande deny

Dans un premier temps, nous allons télécharger sur notre serveur la liste des adresses ip des noeuds. A partir de cette liste nous allons générer un fichier avec des deny sur chaque adresses IP. Cette liste est générée par le site www.dan.me.uk et peut être téléchargé toutes les 30 minutes.

```
wget -q https://www.dan.me.uk/torlist/ -O - | sed 's/^/deny from /g' > /etc/apache/tor-ip.conf;
```

Une fois la liste téléchargé, il faudra rajouter ces quelques lignes dans votre fichier de configuration du virtualhost :

```
Order allow,deny Include /etc/apache2/tor-ip.conf allow from all
```

Il ne reste plus qu'à effectuer un reload du serveur apache :

```
/etc/init.d/apache2 reload
```

Pour améliorer encore il faudrait faire un cron pour mettre à jour régulièrement la liste des IPs TOR.

Définition cron : C'est un programme qui permet aux utilisateurs des systèmes Unix d'exécuter automatiquement des scripts, des commandes ou des logiciels à une date et une heure spécifiées à l'avance, ou selon un cycle défini à l'avance.

Détection Tor

Utilisation d'un IDS

L'IDS est un mécanisme destiné à repérer des activités anormales ou suspectes sur la cible analysée (un réseau ou un hôte). Snort est un IDS open source avec une communauté importante qui garantit son bon fonctionnement et sa fiabilité. Il permet ,sur la base d'un ensemble de règles, de créer des alertes qui seront répertoriées dans un fichier de Logs. Il suffit de créer une règle spécifiques pour TOR et l'ajouter au fichier de conf de L'IDS. Snort possède une version améliorée : Snort Inline qui est un IPS. Ce dernier permet, en plus de détecter, d'effectuer des actions pour stopper des éventuelles actions malveillantes. Pour mettre en place un IDS Snort couplé à un dispositif d'alertes par mails le tutoriel suivant est très détaillé et facile de mise en place. <http://jacquesgoueth.blogspot.fr/2017/07/comment-mettre-en-place-un-systeme-de.html>

Les logs Windows

Deuxième approche, si vous avez un moyen de collecte et de traitement des logs des vos machines, et que vous avez activé les logs qui trace les démarrage de processus : cherchez simplement toutes les lignes de journaux contenant « tor.exe ». Ca paraît simple, mais vous allez attraper tous les utilisateurs qui ont installé tor sans se poser plus de question. Autre avantage, vous détecterez aussi les exécutables tor « portables » lancé depuis une clé USB. Cette technique est très efficace en contrôle de vos utilisateurs. Cependant, les malwares utilisant TOR en exfiltration de données ne laisserons pas trainer un fichier tor.exe bien en évidence.

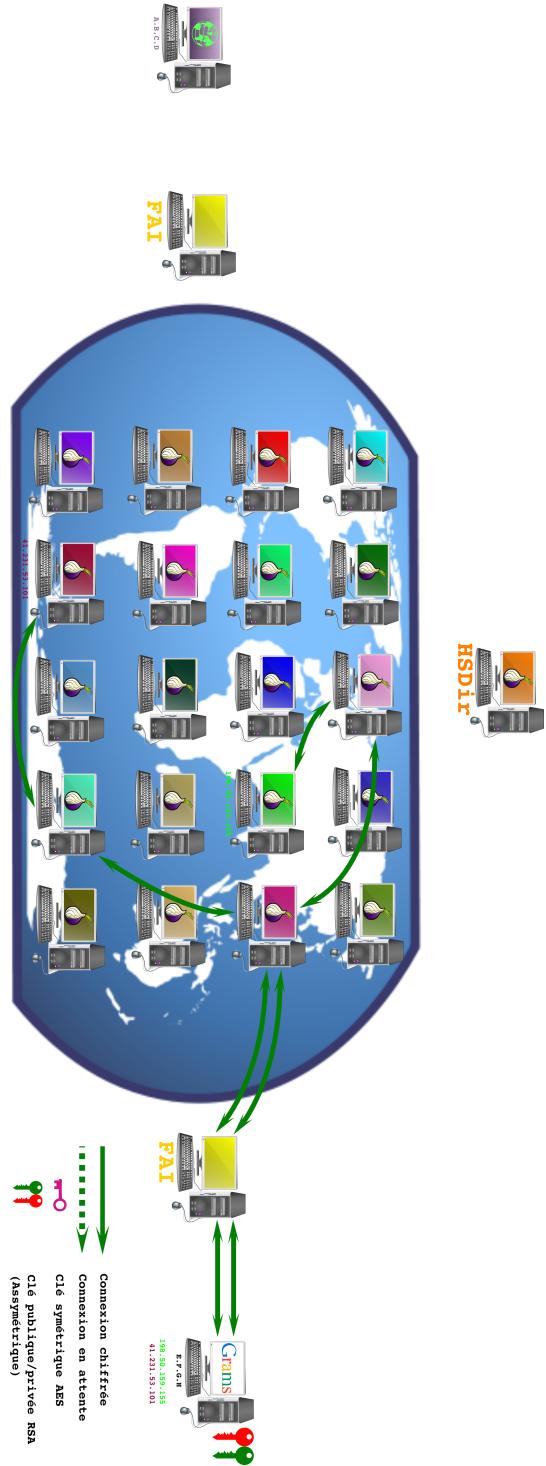
Le Réseau

TOR fait bien attention à utiliser un protocole HTTPS relativement standard pour ne pas attirer l'attention de tous les firewalls et IDS sur son chemin. On a quelques options tout de même : - La liste des relais TOR permet de savoir si un couple IP/Port correspond a un routeur TOR. Et de plus, si les logs sont anciens le Tor garde une liste des relais dans le temps à cette adresse : <https://exonerator.torproject.org/> Pour récupérer cette liste : - sudo pip install OnionPy - script python : from onion_py.manager import Manager from onion_py.caching import OnionSimpleCache manager = Manager(OnionSimpleCache()) sd = manager.query('details') len(sd.relays) for relay in sd.relays: for addr in relay.or_addresses: print(addr) il ne vous reste plus qu'à croiser cette liste IP:Port avec vos journaux de sondes, routeurs ou firewalls pour avoir des bonnes pistes des clients TOR chez vous. - Analyser le trafic

: notamment via les certificats SSL généré par TOR qui présente un pattern particulier.

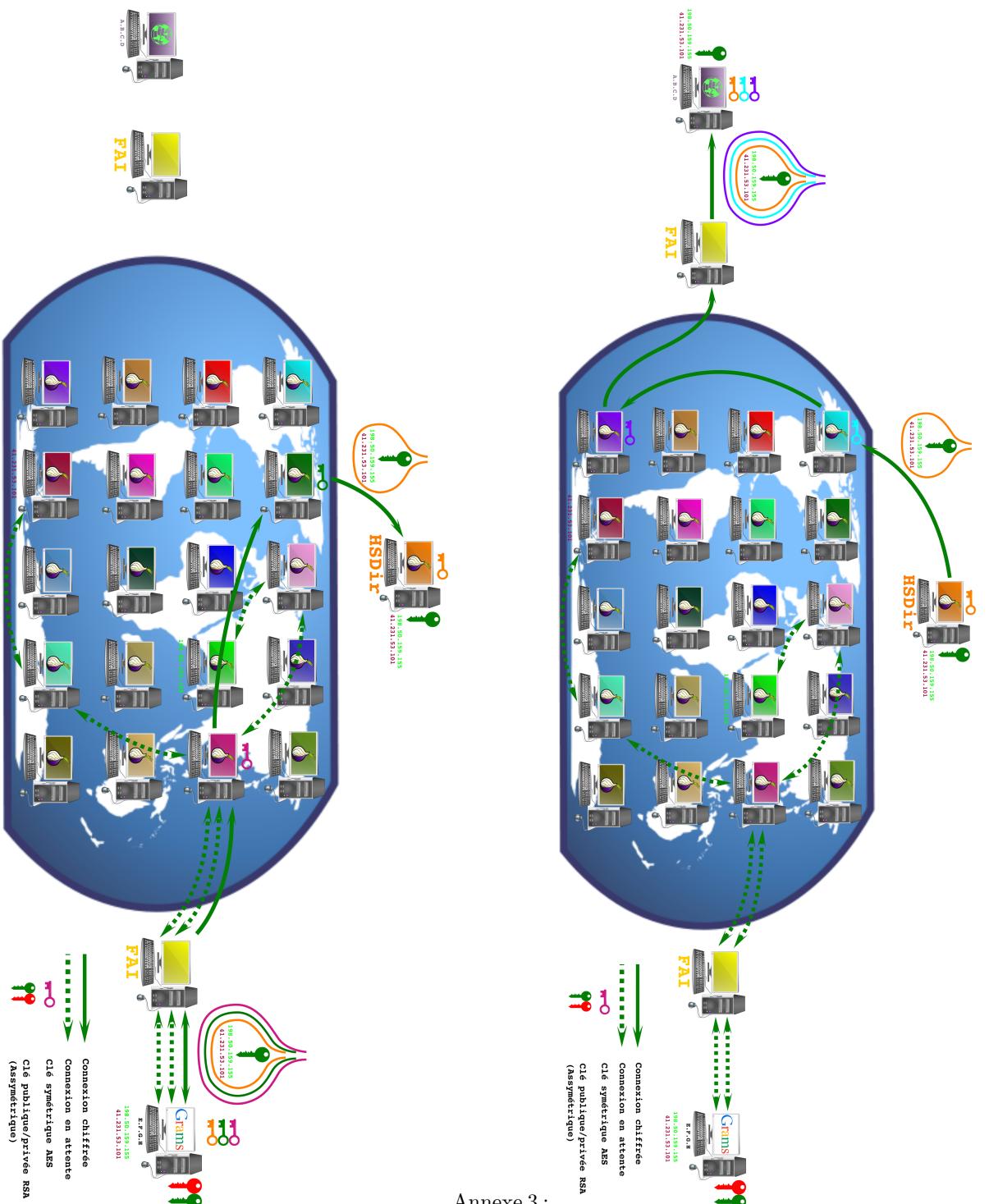
Conclusion

Le meilleur moyen d'avoir des résultats est de croiser ces différentes techniques pour obtenir un maximum d'information et ainsi prendre les bonnes décision.

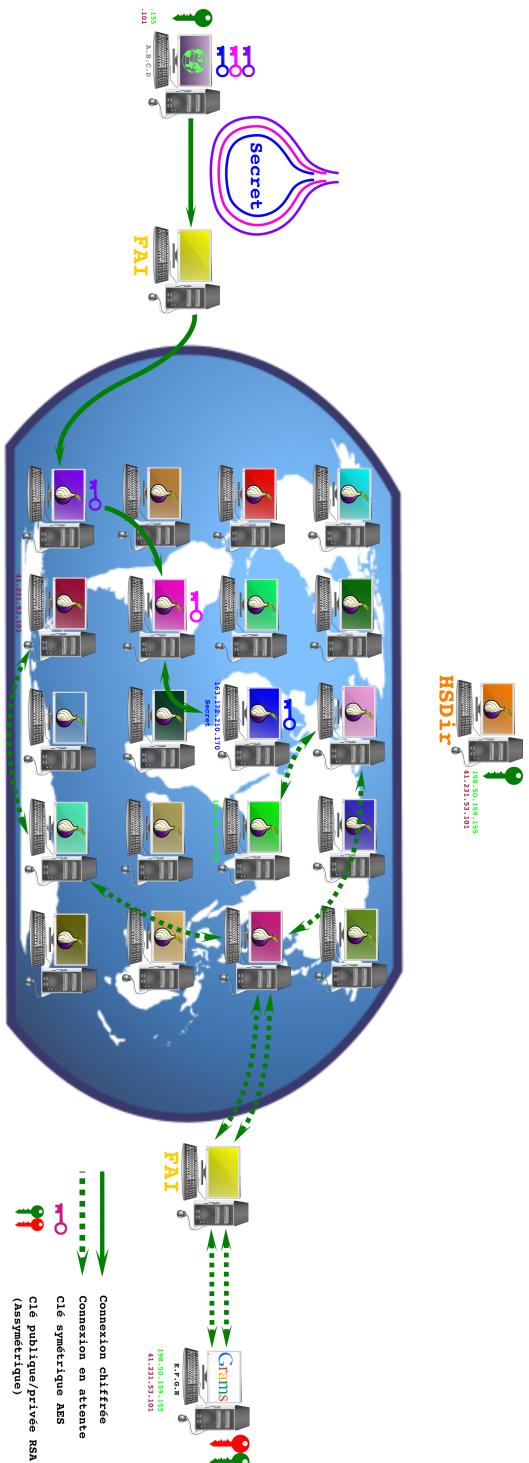


Annexe 1 :

Annexe 2 :

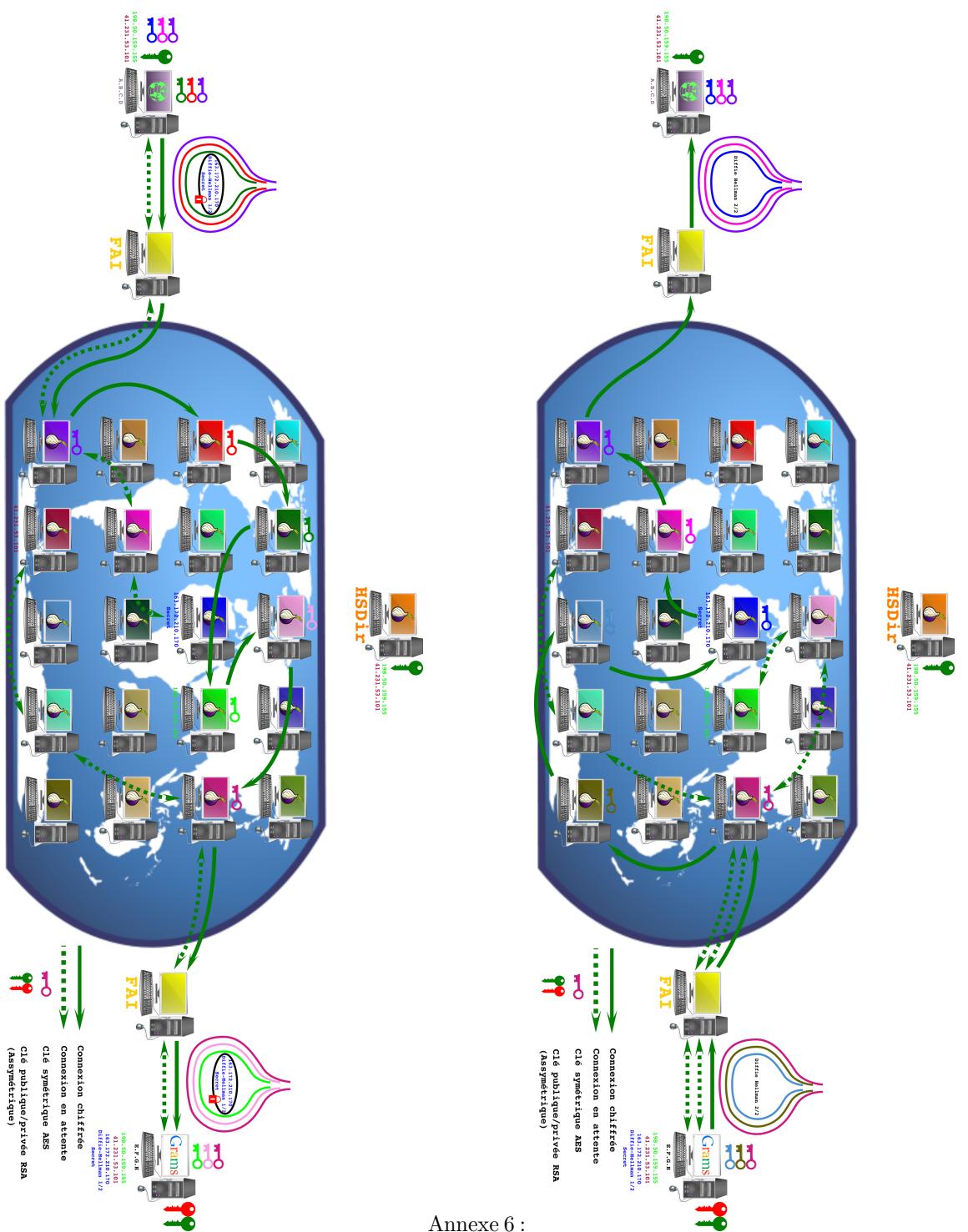


Annexe 3 :



Annexe 4 :

Annexe 5 :



Annexe 6 :

reference : [1] : https://www.psychoactif.org/psychowiki/index.php?title=Tor,_conception,_fonctionnement_et
[2] : <https://fr.softonic.com/articles/tor-outil-navigation-anonyme> [3] :
<https://www.torproject.org/docs/onion-services> [4] : <https://www.torproject.org/docs/tor-doc-relay.html.en#setup> [5] : <https://themimitoof.fr/mettre-en-place-un-relais-tor/> [6] : <http://www.supinfo.com/articles/single/277-creer-hidden-service-reseau-tor> [7] :<http://www.ieee-security.org/TC/SP2013/papers/4977a080.pdf> [8] :<https://donncha.is/2013/05/trawling-tor-hidden-services/> [9] :<https://framablog.org/2016/05/06/anonymat-en-ligne-nos-oignons/> [10] : <https://blog.torproject.org/lifecycle-new-relay> [11] : <https://blog.lesfourmisduweb.org/bloquer-le-reseau-tor/> [12] :<https://foxinou.fr/empecher-les-noeuds-du-reseau-tor-daccéder-a-votre-serveur-apache-2/> [12] :<https://geekeries.org/2016/07/detecter-du-trafic-tor-sur-son-reseau/>