Le petit livre du hacker

Par: Simon Lévesque

Version

Alpha (2016-08-31)

Site web

http://lpldh.foilen.com

Textes et illustrations
© Simon Lévesque, 2014-2015
Tous droits réservés
http://simonlevesque.com
https://foilen.com

Illustration de couverture
© Caroline Bissonnette, 2014-2015
Tous droits réservés
http://www.caro-art.ca

Contenu

Illustration Index

1 Préface

Dans la francophonie, nous avons la chance d'avoir deux mots distinctifs pour deux concepts qui ont souvent été associés: "Piratage" et "Hacking". Ces deux notions sont ambigües étant donné qu'en anglais, le mot "Piratage" n'est pas employé alors que le mot "Hacking" est utilisé à tord et à travers indépendamment pour parler des criminels ou des explorateurs de technologies.

Lorsque les médias parlent en mal de personnes qui utilisent la technologie, c'est habituellement du piratage. Que ce soit parce qu'une entreprise s'est faite attaquée, pillée, vandalisée ou parce que des gens téléchargent des biens virtuels tels de la musique, des films ou des jeux vidéos sans payer, tout cela est illégal et c'est du piratage. Ce l'est au même titre qu'un loup des mers qui attaquent et pillent d'autres navires.

Le hacking, c'est avoir le désir d'acquérir le plus de connaissances possible et de les appliquer de façon inventive. C'est de ne pas se limiter à utiliser un grille-pain pour griller du pain, mais au contraire, de vouloir savoir de quoi il est fait et comment il fonctionne pour pouvoir ensuite le réparer, l'améliorer ou prendre des pièces pour le transformer en quelque chose d'autre d'inusité. C'est donc s'intéresser à la science qui compose les biens qui nous entourent et ce peu importe le champ d'expertise. Le hacking est souvent associé à l'électronique, à l'informatique et aux logiciels, mais comme je viens de le montrer en exemple, même le fonctionnement d'un grille-pain peut être intéressant et accaparer notre curiosité. Cette dernière est la seule limite à ce que nous désirons hacker puisqu'elle peut s'étendre aux sciences sociales avec la psychologie des humaines, à l'économie pour trouver les failles de notre système financier et même à la botanique pour croiser des espèces et obtenir de nouvelles variétés.

Inévitablement, à force de fouiller partout et de tenter de percer les principes de bases derrière tout ce qui existe, des failles seront trouvées. Cette cette zone grise qui fait peur, car entre les mauvaises mains, ces failles permettent des actes illégaux. Tout dépend alors de la moralité et de l'éthique de ceux qui découvrent les problèmes. Ils peuvent se taire, avertir les personnes concernées ou en abuser.

C'est avec cet état d'esprit de curiosité et de désir de découvrir que j'aimerais vous amener à vous familiariser avec la machine merveilleuse qu'est l'ordinateur. Cela sera fait en expliquant les parties physiques et logicielles qui composent l'ordinateur, les parties du réseau mondial qu'est l'Internet et les mathématiques qui font fonctionner le tout. Ce sera entrecoupé d'exemples de limitations, d'exploitation possible et de manières de s'en protéger. Le but est de vous faire découvrir toutes les possibilités de la machine sans nécessairement aller profondément dans les détails puisque ce document se veut une manière facile et agréable de faire un tour d'horizon complet pour bien s'outiller en connaissances informatiques. Par la suite, vous avez bien entendu la possibilité d'approfondir les facettes qui vous intéressent le plus avec d'autres livres plus spécialisés ou en visitant les nombreux sites mis à votre disposition en bas de pages.

Commençons la découverte...

2 Nouveautés

Depuis la version 2013, voici ce qui a été ajouté :

- Chapitre 23:Cryptographie avancée
  - La théorie, les techniques, les attaques, les algorithmes, etc.
- Chapitre 26 :Les licences d'utilisation
  - Source libre versus propriétaire et les sections importantes.

Ce qui a été modifié :

- Chapitre 19:Les différents préfixes de grandeurs
  - Clarification des préfixes en parlant de la puissance 2.
- Les liens
  - Tous les liens en bas de pages ont été modifiés en allant vers un site de redirection. Les buts principaux sont d'avoir des liens courts à écrire et de pouvoir les mettre à jour sans avoir à réimprimer ce livre.
- Plusieurs précisions ont été ajoutées dans les sections déjà présentes suite aux commentaires reçus.
  - D'ailleurs merci à tous ceux qui offert des commentaires constructifs pour améliorer ce livre.

3 L'état d'esprit d'un hacker

Un hacker, c'est tout simplement une personne qui aime apprendre comment fonctionne tout ce qui l'entoure que ce soit les humains, les lois, les finances, l'électronique, l'informatique et autres technologies. C'est une personne curieuse de nature et qui aime trouver les limites pour mieux les dépasser.

Pour ceux qui ont vu le film *Hackers*[1] sorti en 1995, les effets visuels ne vous ont sûrement pas impressionnés, mais la culture qui émane de cette histoire est celle qui nous intéresse. Bien entendu, dans le film, les personnages dépassent parfois la simple curiosité et s'en vont dans le côté plus sombre en téléphonant gratuitement, en piratant l'ordinateur d'une entreprise ou en manipulant les informations pour déranger la vie de l'agent du FBI, mais à la base, le désir de tout connaitre et de partager avec ses pairs est l'essence même du hacking. D'ailleurs dans ce film, il y a un moment où un agent du FBI lit un extrait du manifeste d'un hacker. Voici le texte écrit par *The Mentor* en 1986 dans la revue *Phrack* traduit par *NeurAlien* pour le magazine *No Way* Volume I, numéro 3. Je l'ai un peu modifié, car il manquait quelques bouts de la version originale et il y avait plusieurs fautes.

Un autre s'est fait prendre aujourd'hui, c'est partout dans les journaux. "Un adolescent arrêté pour un scandale de crime informatique", "Arrestation d'un hacker après le piratage d'une banque"...

Satanés gosses. Tous les mêmes.

Mais avez-vous, dans votre psychologie à trois pièces et votre profil technocratique de 1950, un jour pensé à regarder le monde derrière les yeux d'un hacker? Ne vous êtes-vous jamais demandé ce qui l'avait fait agir, quelles forces l'avaient modelé?

Je suis un hacker, entrez dans mon monde...

Le mien est un monde qui commence avec l'école... Je suis plus astucieux que la plupart des autres enfants, les conneries qu'ils m'apprennent me lassent...

Satanés sous performants. Tous les mêmes.

Je suis au collège ou au lycée. J'ai écouté les professeurs expliquer pour la quinzième fois comment réduire une fraction. Je l'ai compris. "Non, Mme Dubois, je ne peux pas montrer mon travail. Je l'ai fait dans ma tête..."

Satanés gosses. Il l'a certainement copié. Tous les mêmes.

J'ai fait une découverte aujourd'hui. J'ai trouvé un ordinateur. Attends une minute, c'est cool. Ça fait ce que je veux. Si ça fait une erreur, c'est parce que je me suis planté. Pas parce qu'il ne m'aime pas...

Ou parce qu'il se sent menacé par moi...

Ou parce qu'il pense que je suis petit filou...

Ou parce qu'il n'aime pas enseigner et qu'il ne devrait pas être là...

1 https://r.foilen.com/film-hacker : Film Hacker

Satanés gosses. Tout ce qu'il fait c'est jouer. Tous les mêmes.

Et alors c'est arrivé... une porte s'est ouverte sur le monde... Se précipitant à travers la ligne téléphonique comme de l'héroïne dans les veines d'un accro, une impulsion électronique est envoyée, un refuge de l'incompétence quotidienne est demandé... un forum est trouvé.

"Voilà... c'est ici que j'appartiens..."

Je connais tout le monde ici... même si je ne les ai jamais rencontrés, ne leur ai jamais parlé, n'entendrai peut-être jamais parler encore d'eux... Je vous connais tous...

Vous pouvez gager que nous sommes tous les mêmes... On a été nourri à la petite cuillère de bouffe pour bébé à l'école quand on avait faim d'un steak... Les fragments de viande que l'on nous a laissée étaient prémâchés et sans goût. On a été dominé par des sadiques ou ignoré par des apathiques. Les seuls qui avaient des choses à nous apprendre trouvèrent des élèves volontaires, mais ceux-ci sont comme des gouttes dans le désert.

C'est notre monde maintenant... Le monde de l'électron et de l'interrupteur, la beauté du baud. Nous utilisons un service déjà existant, sans payer ce qui pourrait être bon marché si ce n'était pas la propriété de gloutons profiteurs, et vous nous appelez criminels. Nous explorons... et vous nous appelez criminels. Nous recherchons la connaissance... et vous nous appelez criminels. Nous existons sans couleurs de peau, sans nationalités, sans stéréotypes religieux... et vous nous appelez criminels. Vous construisez des bombes atomiques, vous financez les guerres, vous assassinez et trichez, vous manipulez et nous mentez en essayant de nous faire croire que c'est pour notre propre bien-être, et nous sommes encore des criminels.

Oui, je suis un criminel. Mon crime est celui de la curiosité. Mon crime est celui de juger les gens par ce qu'ils pensent et disent, pas selon leur apparence. Mon crime est de vous surpasser, quelque chose que vous ne me pardonnerez jamais.

Je suis un hacker, et ceci est mon manifeste. Vous pouvez arrêter cet individu, mais vous ne pouvez pas tous nous arrêter... après tout, nous sommes tous les mêmes.

+++The Mentor+++

C'est avec cet esprit d'ouverture que je vous invite à découvrir l'ordinateur...

L'ordinateur

4 Les composantes d'un ordinateur

4.1 Introduction

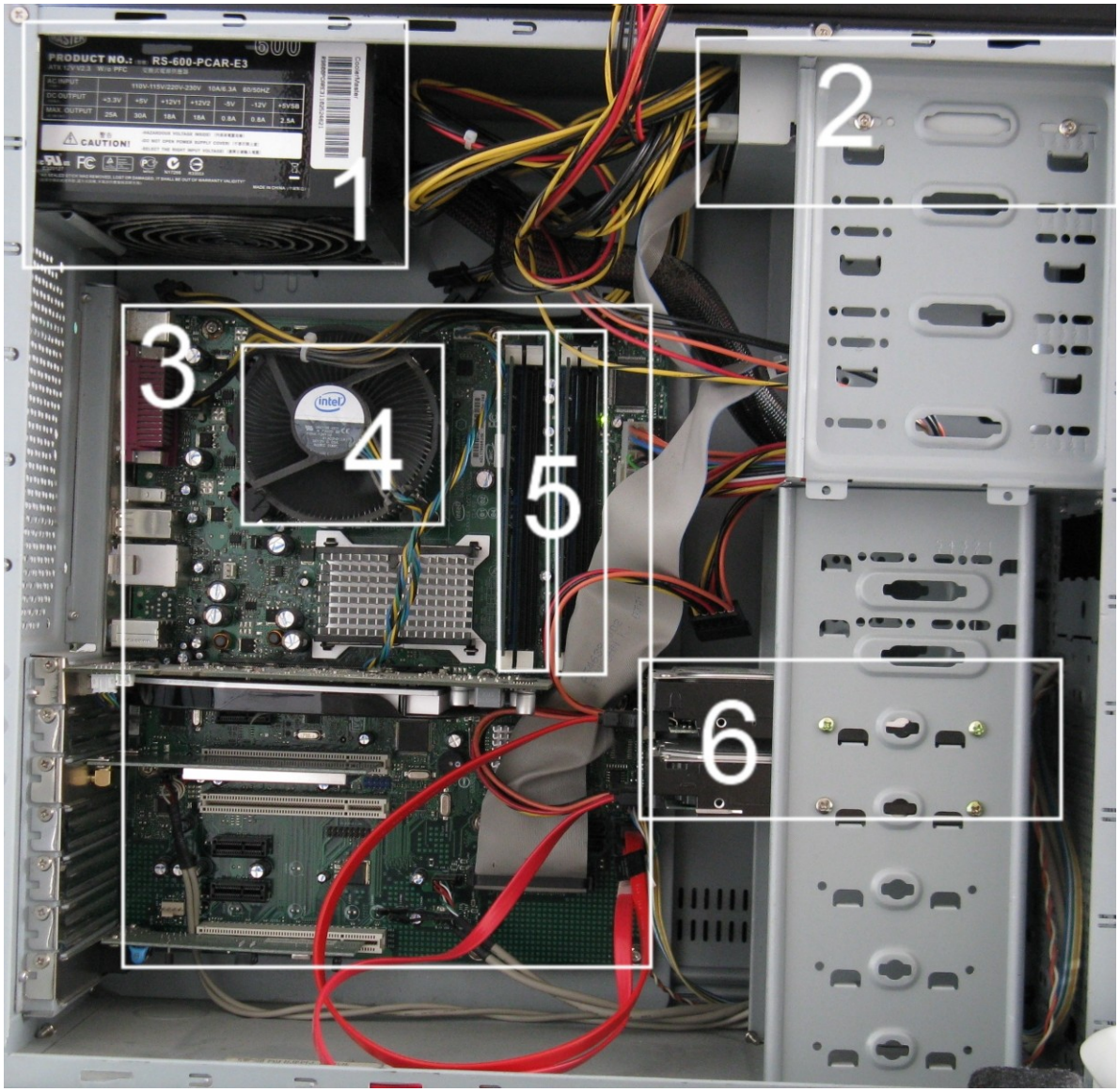

Illustration 1: Les composantes d'un ordinateur : Photos de l'intérieur d'un ordinateur

Lorsque nous achetons un ordinateur, les pièces que nous voyons sont l'écran, la tour, les haut-parleurs, le clavier et la souris. Tout est relié à la tour et c'est à l'intérieur d'elle que toute la magie s'opère. Pour voir comment brancher les différents périphériques à la tour, vous pouvez aller voir une vidéo que j'ai tournée[2]. C'est la tour qui coordonne toutes les données transmises aux périphériques pour former une expérience cohérente. Allons voir de plus près ce qu'il y a à l'intérieur.

1. Boite d'alimentation

2 https://r.foilen.com/f-brancher-ordi : Brancher un ordinateur

2. Lecteur optique CD/DVD
3. Carte mère
4. Processeur caché par son ventilateur. Le processeur est en dessous du ventilateur
5. Barrettes de mémoire vive
6. Disques durs

Étant donné que les pièces s'améliorent assez rapidement, je tiens à noter que ce chapitre est à jour en janvier 2013 pour ce qui est des modèles. De plus, les périphériques se dépréciant trop rapidement, j'ai préféré faire une comparaison par gamme plutôt que par prix.

4.2 Processeur (CPU)

Le processeur est le cerveau de l'ordinateur. C'est lui qui va exécuter toutes les instructions des programmes et qui va coordonner la transmission des données entre les pièces. C'est donc important d'en avoir un qui offre de bonnes performances pour l'utilisation désirée. Par exemple, il n'est pas nécessaire d'avoir un ordinateur ultra-puissant pour lire des courriels, mais pour jouer aux derniers jeux vidéos avec des graphiques réalistes ou pour faire de l'édition vidéo, il faut être prêt à sortir un peu plus d'argent.

L'unité de mesure d'un processeur est le Hertz et représente le nombre de cycles d'instructions par secondes qui peuvent être exécutés. Par exemple, 2 GHz veut dire qu'il peut rouler 2 000 000 000 cycles d'instructions par seconde (vous pouvez aller voir l'annexe des grandeurs pour connaitre les différentes grandeurs[3]). Selon le type de processeur, un cycle peut permettre d'exécuter plus d'instructions qu'un cycle d'un autre processeur, alors lorsque le même fabricant compare ses différents modèles, il est possible de comparer avec cette unité puisque cette quantité est semblable, mais ce n'est pas si simple pour comparer entre deux vendeurs. En d'autres termes, si pour effectuer une action spécifique, un fabricant a besoin d'exécuter deux fois plus d'instructions-machine pour obtenir le même résultat qu'un autre vendeur, alors il faut comparer en divisant sa vitesse par deux. De plus, chaque fabricant a des spécialités dans les actions alors un processeur peut être meilleur chez un vendeur pour encoder des vidéos, mais ce même processeur pourrait être plus lent pour jouer à des jeux vidéos. Normalement, ce n'est pas un facteur important pour les particuliers, mais pour les gens qui veulent faire une tâche spécialisée, il peut être intéressant de regarder des graphiques de comparaisons entre les processeurs d'une même marque et ses concurrents.

Pour les ordinateurs de bureau, il y a deux grandes entreprises qui se disputent le marché: Intel et AMD. Ce sont des noms de marques et chacun a des processeurs de plusieurs gammes. Le choix est donc purement personnel à moins d'avoir un logiciel puissant en particulier à utiliser. Dans ce cas, mieux vaut regarder les comparaisons puisque selon la tâche à effectuer, une entreprise peut être meilleure qu'une autre. Si c'est pour une utilisation domestique, à gammes égales, AMD est souvent le choix le moins dispendieux.

En plus de simplement exécuter des instructions, certains processeurs ont des fonctionnalités propres qui permettent d'améliorer les performances de certaines opérations. Les technologies importantes à connaitre sont les multicoeurs et le nombre de bits. Pour la première, il faut savoir que toutes les applications qui roulent en même temps sur l'ordinateur demandent un peu du temps de processeur. Techniquement parlant, il n'est pas faux de dire que deux logiciels pourraient fonctionner en même

3 Voir chapitre 19 à la page 93

temps puisqu’ils n’ont rien de partagé entre eux, mis à part le temps du processeur. Alors le multicoeur permet de partager du temps simultanément entre deux programmes. Par exemple, avec un double coeur, il est possible de rouler deux programmes exactement en même temps comme s’ils étaient sur deux ordinateurs différents plutôt que d’avoir chacun une seconde une après l’autre. C’est donc pour avoir des exécutions parallèles plutôt que concurrentes.

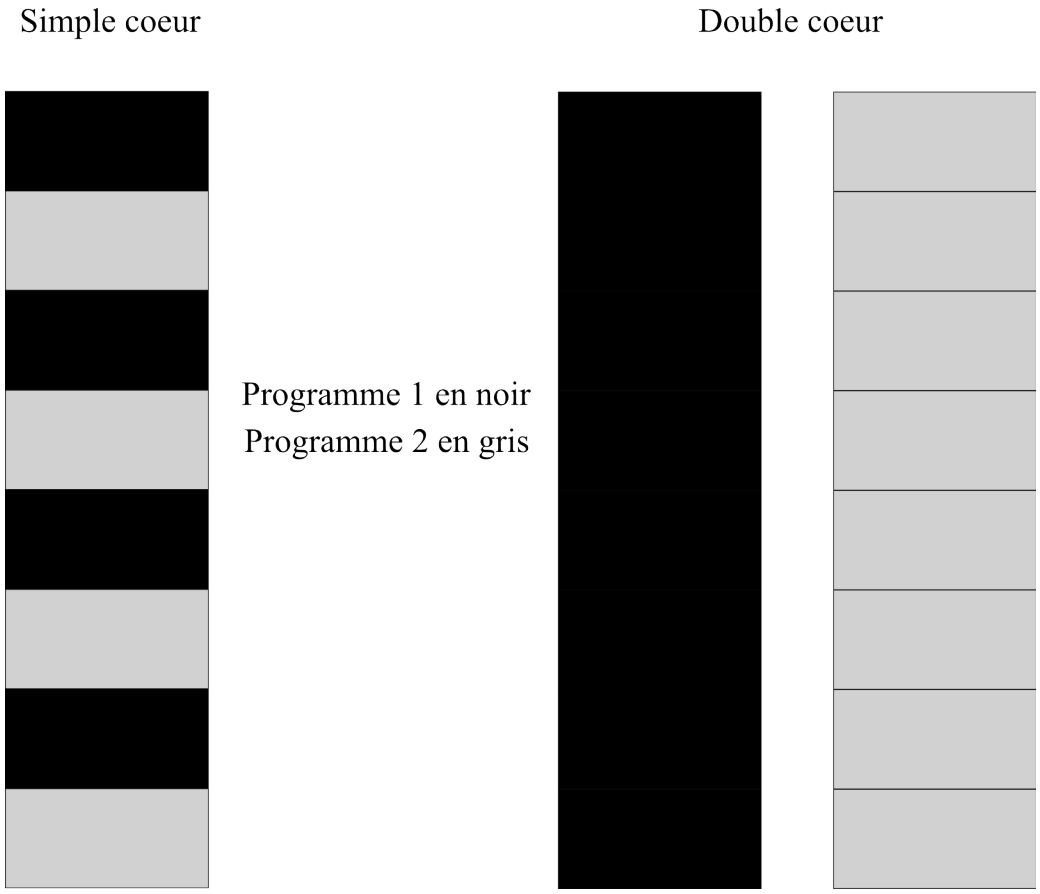

Illustration 2: Processeurs : Différence d'exécution entre simple et double coeur

Le graphique ci-dessus montre la différence entre un simple coeur et un double coeur. Supposons que nous avons un processeur à 8 Hz (8 instructions par secondes, d'où les 8 cases verticales), dans le premier cas, les programmes 1 et 2 se partagent la moitié du temps. Ils vont donc rouler 4 instructions par secondes chacun. Par contre, avec un double coeur, nous avons deux fois 8 Hz et chaque programme peut utiliser les 8 cases par secondes. La vitesse théorique est alors deux fois plus rapide. Si je précise "théorique", c'est parce qu'en pratique, le processeur partage d'autres composantes, alors pour avoir véritablement deux exécutions indépendantes, il faudrait deux ordinateurs. Chose certaine, il y a quand même des gains importants avec les doubles coeurs et bien entendu les quads coeurs (quatre coeurs).

La seconde technologie concerne le choix entre 32 bits ou 64 bits. Nous sommes actuellement en transition pour aller vers le 64 bits, mais pour en tirer pleinement avantage, il faut un système d’exploitation qui est précisé comme tel comme Windows XP 64, Windows 7 64 ou Windows 8 64. En plus du système d’exploitation, il faut aussi que les logiciels installés soient 64 bits pour vraiment tirer profit de ce gain de vitesse. Il est possible de rouler des applications 32 bits sur un système d’exploitation 64 bits, mais les performances sont réduites étant donné que le système d'exploitation

doit traduire du 32 aux 64 bits. Si vous achetez un ordinateur neuf, il va supporter le 64 bits (et le 32 bits). Dans ce cas, vous n'avez pas besoin de vous en faire. Par contre, si vous l'achetez usagé, mieux vaut vérifier à moins que vous vous contentiez d'utiliser des logiciels 32 bits.

Étant donné qu'il y a beaucoup de modèles de processeurs et qu'ils ont des spécialités, voici trois graphiques qui montrent les prix, la puissance pour un jeu 3D et la puissance pour encoder une archive. Ensuite, nous allons décortiquer le tout.

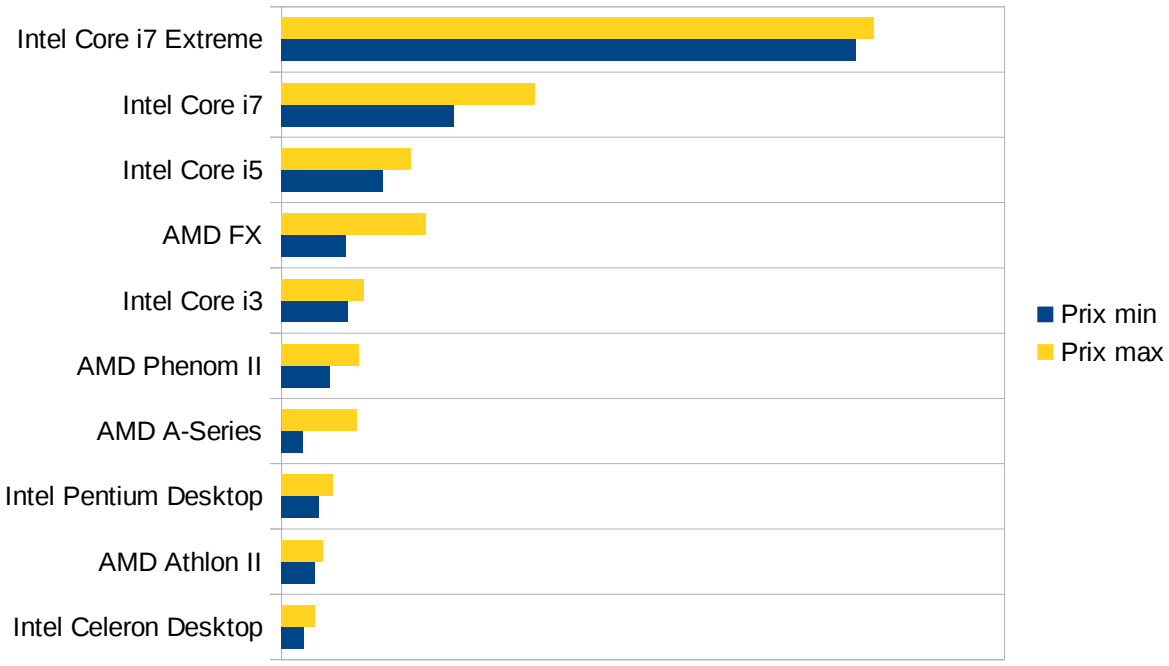

Illustration 3: Les composantes d'un ordinateur : Prix des processeurs

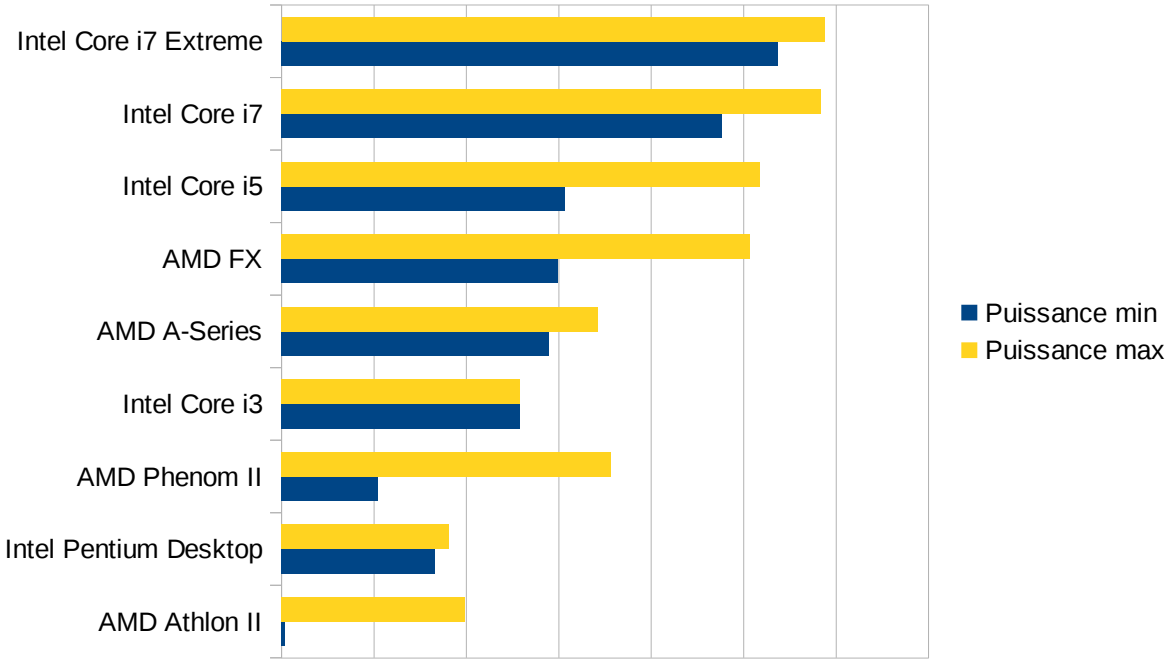

Illustration 4: Les composantes d'un ordinateur : Puissance de jeux

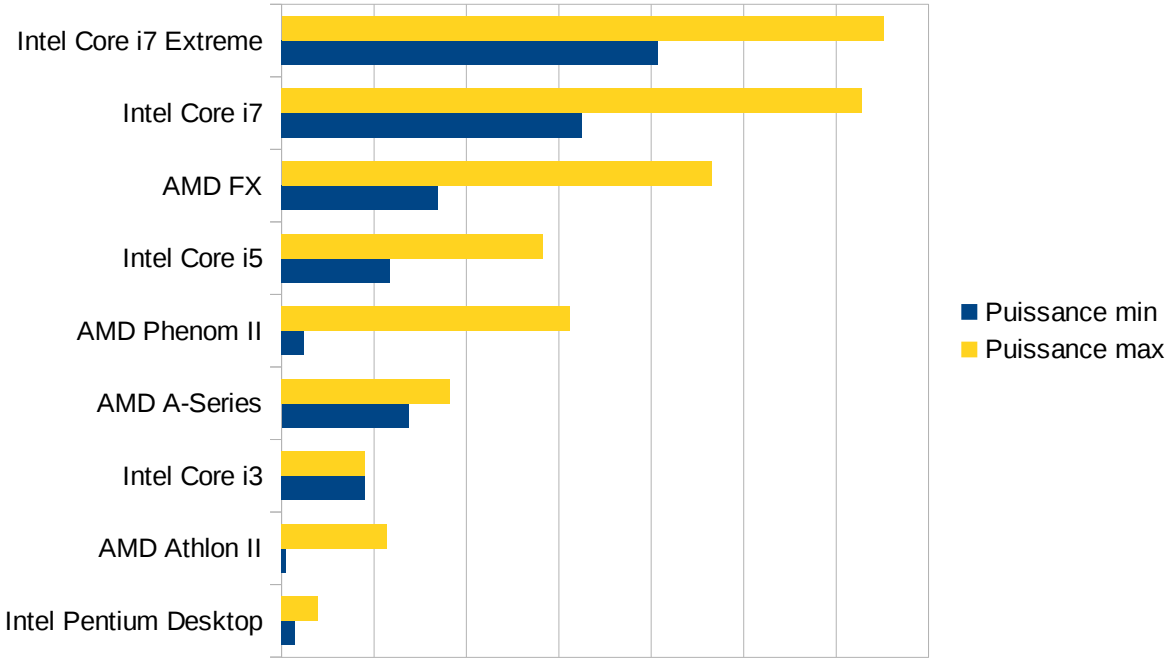

Illustration 5: Les composantes d'un ordinateur : Puissance d'encodage

Au début, je voulais faire l'analyse en regardant les prix et en donnant des catégories comme haut, milieux et bas de gamme, mais je me suis rendu compte que les AMD n'étaient pas comparables aux Intel, car ces processeurs coûtent très peu cher et seraient au mieux milieu de gamme. Par contre, en regardant les deux graphiques de performances, il est évident qu'AMD accote celles d'Intel et par

conséquent, AMD est beaucoup moins chère qu'Intel pour les mêmes performances. Ensuite, pourquoi ne pas faire les hauts, milieux et bas de gamme en se basant sur les résultats des deux dernières illustrations? Parce que si nous prenons le AMD Phenom II, le moins puissant serait dans le bas de gamme tandis que le plus puissant serait dans le haut de gamme.

Si vous avez besoin d'un ordinateur qui durera longtemps et qui sera **facile de mettre à jour le processeur**, le mieux est d'aller dans les modèles les plus récents comme les Intel Core i5 et i7 et les AMD FX. Par contre, si vous désirez ne pas payer trop cher tout en conservant la possibilité de mettre à jour dans les années à venir, vous devriez préférer un AMD FX puisque son prix le plus bas est comparable à celui d'un i3, ce qui donne une grande marge.

Pour une simple **utilisation de tous les jours**, un Intel Core i3, un AMD A-Series ou un AMD Phenom II feront très bien l'affaire. Si vous optez pour un Phenom II, les prix des différents modèles sont assez proches alors que la marge de performances est très large. Mieux vaudrait alors payer un léger supplément pour avoir le top qui dépasse les A-Series en performances, mais pas en prix.

Si vous désirez une **mini machine** qui pourrait par exemple servir de serveur de fichiers et qui ne nécessite pas beaucoup de puissance de calculs, mieux vaut payer le moins cher avec de l'AMD A-Series, Intel Celeron ou AMD Athlon II.

Pourquoi tant de différences entre les performances et les prix? Simplement parce qu'il y a aussi d'autres facteurs qui entrent en compte comme la consommation d'énergie et le choix du type de carte mère qui pourrait par ricochet demander d'autres périphériques plus chers.

4.3 Mémoire vive (RAM)

Lorsqu'un programme est ouvert, il est chargé dans la mémoire pour que ses données soient plus rapides à utiliser que si elles étaient sur le disque dur. Il est important d'en avoir en quantité suffisante pour que le système d'exploitation et les programmes que vous utilisez quotidiennement soient fonctionnels.

Il est possible d'acheter de la mémoire comme une seule barrette ou avec un ensemble de deux barrettes. La meilleure est la seconde option puisqu'elle permet de faire fonctionner l'ordinateur en doubles canaux plutôt qu'en simple. Cela veut dire que les deux barrettes sont utilisées en même temps de manière alternée, ce qui donne un rendement deux fois plus rapide. Le système n'attend donc pas d'être rendu à la fin d'une barrette avant d'aller sur la seconde; il partage les blocs de manières continues entre les barrettes. Ainsi, il est possible d'écrire simultanément deux blocs d'informations sur deux modules différents au lieu de les écrire l'un après l'autre sur le même.

Pour les types existants, il y a la DDR2 et DDR3 qui se vendent. Si vous avez le choix, préférez la seconde puisqu'à même grosseur, celle-ci se vend au même prix ou moindre et elle est bien entendu beaucoup plus rapide.

Pour ce qui est de la grosseur à acheter, Windows 7 et 8 recommande au minimum 1 Go pour la version 32 bits et 2 Go pour la 64 bits, mais vous devriez vérifier quels programmes vous désirez utiliser en même temps. Plus il y a de programmes qui tournent en parallèle, plus il faut avoir de la mémoire vive. Un point important à considérer, c'est qu'il est inutile d'avoir plus de 4 Go de mémoire vive si le système d'exploitation est en 32 bits, car c'est le maximum qu'il pourra utiliser. Il faut être en 64 bits pour profiter pleinement de toute la mémoire. Il est important de noter que Windows XP 32 bits n'ira pas au-dessus de 3 Go, malgré qu'il pourrait théoriquement aller à 4 Go. C'est Microsoft qui l'a choisi ainsi puisqu'il n'était pas commun au début des années 2000 d'avoir autant de mémoire vive.

4.4 Carte vidéo

Pour tous les adeptes des jeux vidéos, cette composante est la plus importante. Si ce n'est que pour travailler, vous pouvez vous contenter de celle incorporée à la carte mère. Pour cette dernière option, il est important de vérifier si elle a la mention de « mémoire partagée » puisque cela signifie que la mémoire pour la carte vidéo est prise directement de la mémoire vive. Ce genre de mention n'est que pour les cartes intégrées; celles achetées à part viennent avec leur propre mémoire.

Le choix d'une carte pour les jeux vidéos en 3D peut sembler très complexe dû à la quantité de choix disponible, mais il suffit de regarder les spécifications recommandées pour les jeux vidéos qui vous intéressent. Ensuite, vous pouvez décider de payer un peu plus maintenant pour ne pas avoir à changer de carte dans les mois ou années à venir.

Voici quelques points importants à vérifier pour bien comparer les cartes:

L'interface, c'est la sorte de connecteur dans lequel la carte se branche sur la carte mère. La plus commune est la PCI qui est blanche, mais c'est la moins intéressante pour les cartes vidéos à cause de sa lenteur. Il y a eu par la suite un port AGP qui était spécifiquement utilisé pour les cartes vidéos, mais maintenant, le summum est au PCI Express. La version 1 est sortie en 2003, la version 2 en 2007 et la version 3 existe depuis 2010. C'est donc cette dernière qui est la plus rapide.

La mémoire, c'est dans la mémoire que toutes les textures et les images sont emmagasinées pour un affichage rapide. Plus elle est grande, plus les textures peuvent être détaillées sans ralentir l'exécution par un incessant transfert d'images entre le disque dur et la mémoire de la carte vidéo.

Vitesse du GPU, un GPU (Graphical Processor Unit) est semblable à un CPU (Central Processor Unit). La différence est que le GPU n'est utilisé que pour calculer l'affichage 3D. Se faisant, le CPU n'a plus à utiliser son temps pour gérer le 3D et peut passer plus de temps pour le son, l'intelligence artificielle et les prises de données de la manette. Sur certaines cartes vidéos, il est aussi possible aux développeurs d'exécuter du code arbitraire et ainsi faire tous les calculs pour la physique directement sur la carte vidéo pour libérer du temps de processeur.

DirectX est la technologie de rendu vidéo de Microsoft pour son système d'exploitation. Windows XP supporte jusqu'à la version 9; Vista, Windows 7 et 8 la version 11.

Shader Model est une technologie de Microsoft qui est incluse dans DirectX. Sa version la plus haute est la 4.

OpenGL est la technologie de rendu vidéo qui est standardisée. Elle fonctionne sous Windows, Linux, Mac, quelques téléphones cellulaires intelligents et certainement d'autres appareils. La version la plus récente est la 4.3 qui est sortie le 2012-08-06.

Pour ce qui est des marques et modèles, il y avait la compagnie ATI qui a été achetée par AMD (la même entreprise qui fait les processeurs). Vous retrouverez donc des ATI Radeon et AMD Radeon qui sont en fait de la même entreprise, mais le nom ATI commence simplement à s'effacer pour être remplacé par AMD. L'autre grande marque est Nvidia GeForce. Pour choisir entre AMD et Nvidia, si vous utilisez un processeur AMD, cette entreprise va plus optimiser ses Radeon pour qu'elles fonctionnent mieux avec son matériel, alors cela peut être un choix logique. Sinon, choisissez selon les spécifications de vos jeux et votre budget.

4.5 Carte de son

À moins d’être un créateur de musique, de devoir l'enregistrer avec le moins de bruits possible et d’avoir un temps de latence extrêmement petit, la carte incluse avec la carte mère est amplement suffisante. Il existe aussi des hauts parleurs et des casques d'écoute qui ont leur propre contrôleur et utilisent simplement un port USB. Les bienfaits de cette dernière technique est que la musique est transférée de manière numérique plutôt qu'analogique, ce qui fait un son avec moins de grichements.

Pour les consommateurs de base, ce qui est important est le nombre d'hauts parleurs désirés, car il est possible de créer un cinéma maison avec un ordinateur. Un système 2.1 va avoir deux hauts parleurs satellites ainsi une caisse de bases. Un système 5.1 va créer une ambiance qui entoure l'utilisateur avec un centre-avant, deux satellites avant, deux satellites arrière et une caisse de bases. Ce dernier système est quand même un peu trop pour un ordinateur de bureau avec une seule personne assise devant. Autant prendre d'excellents écouteurs à la place.

Comme dernier élément, il y a des sièges qui incluent des hauts parleurs derrière la tête et autour de l'utilisateur. Ce genre d'équipement est plus pour les joueurs qui désirent ressentir l'action puisque les bases font vibrer le siège et cela entraîne une plus grande immersion.

4.6 Disque dur

Une des pièces centrales d’un système informatique est le disque dur. Il faut en prendre soin et ne pas oublier de faire des sauvegardes des données inscrites dessus, car une fois perdues, c’est pour de bon. Le disque dur est l’endroit où le système d’exploitation, les applications, les fichiers textes, les musiques, etc. sont emmagasinés.

Pour les disques durs internes, vous devez choisir le type de connecteur. Il y a IDE, SATA, SATA2 et SATA3. Les nouvelles cartes mères permettent le SATA3 et il y a moins de connecteurs IDE. Ces derniers sont surtout utilisés pour les lecteurs optiques qui ne sont pas tous passés au SATA. Les prix, tout comme les grosseurs disponibles, sont très variés.

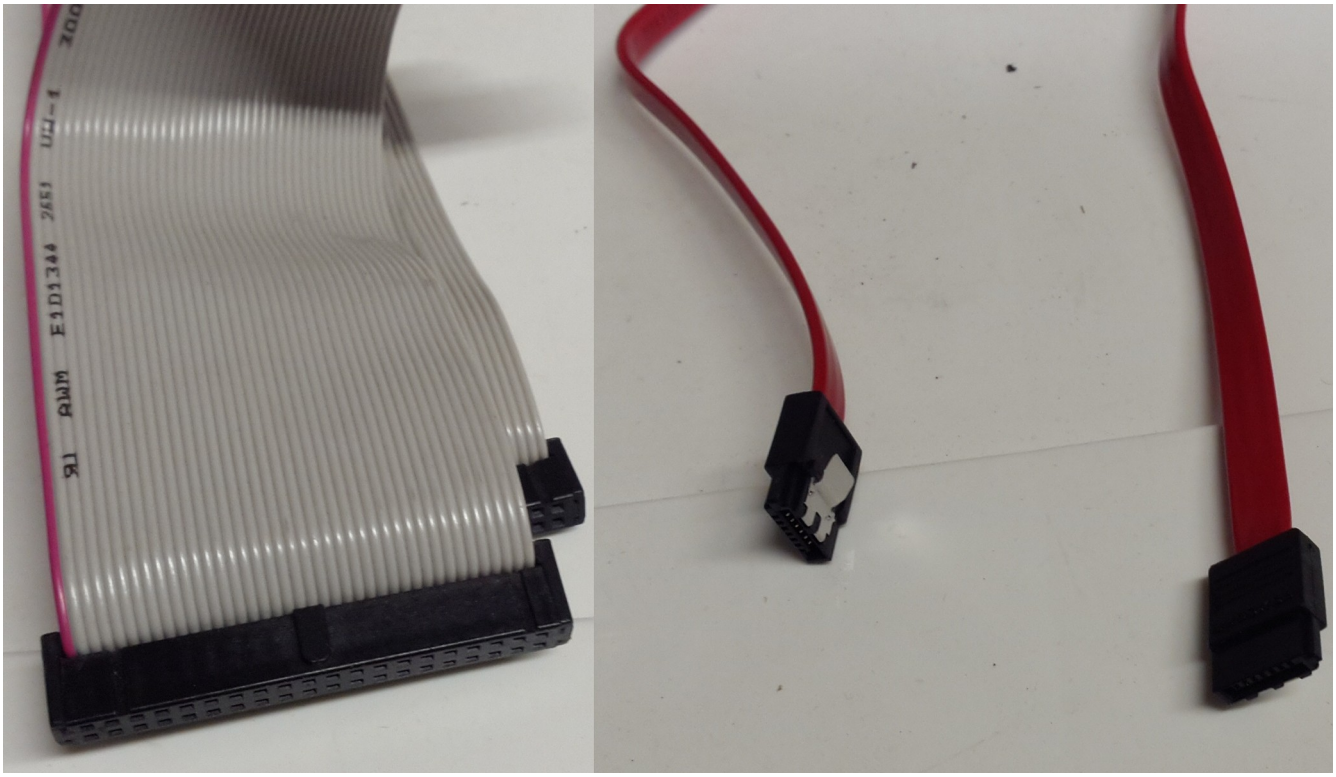

Illustration 6: Les composantes d'un ordinateur : Câble IDE à gauche et cable SATA à droite

Pour les disques standards (un disque rotatif), vous devez privilégier une vitesse de rotation de 7200 RPM (rotations par minute) pour avoir la vitesse maximale. Ensuite, les grandeurs vont d'une centaine de Go à 4 To (4000 Go). Pour avoir une idée de quoi prendre, Windows 7 utilise autour de 20 Go et les jeux 3D populaires vont de 8 à 12 Go chacun. C'est sans compter votre collection de musique. Pour celle-ci, vous pouvez entrer environ 285 chansons par tranche de Go d'espace. N'oubliez pas non plus de compter vos logiciels préférés comme votre suite de bureau et vos éditeurs d'images.

Si vous désirez un disque dur plus rapide, il y a une nouvelle technologie assez chère qui existe et qui s'appelle le "Solid State Drive". C'est comme une clé USB, donc de la mémoire flash, mais qui a beaucoup plus d'espace que ces dernières. Les grosseurs vont de 30 Go à 512 Go et vous risquez de payer très cher. Pour le moment, c'est surtout les portables de luxe qui utilisent cette technologie puisqu'elle supporte mieux les transports, qu'elle prend moins de place et qu'elle consomme moins d'énergie.

4.7 Lecteurs optiques

Pratiquement tous les lecteurs optiques peuvent lire et graver des CD, DVD et Blu-Ray et les nouveaux modèles utilisent l'interface SATA. Le prix est environ la moitié de celui d'un disque dur normal pour les deux premiers.

Si vous êtes prêt à débourser le prix d'un disque dur normal, vous pouvez regarder du côté des graveurs Blu-ray qui permettent de graver 25 ou 50 Go de données sur un seul disque. Ces modèles lisent et gravent aussi les CD et DVD, alors c'est un bon tout-en-un pour un prix abordable.

4.8 Lecteurs de disquette

Vous pouvez avoir ce genre de périphérique pour très peu cher, mais à moins d'avoir déjà des données sur disquettes à conserver, c'est inutile de gaspiller son argent pour cela. À 1.44 Mo la disquette, vous ne pouvez même pas mettre une chanson de trois minutes dessus. Mieux vaut acheter une clé USB qui offre des Go de données et risque de durer plus longtemps sans perdre les données.

4.9 Boitier et alimentation

J'ai choisi de mettre ces deux morceaux ensemble, car il n'est pas rare que les deux viennent dans le même paquet. C'est donc à vérifier lors de l'achat d'un boitier pour ne pas avoir de surprise.

Pour ce qui est du boitier, certes il y en a qui sont tape-à-l'oeil avec de la lumière, mais pour un côté purement fonctionnel, il faut simplement que la carte mère, les disques durs et les lecteurs optiques entrent dans le boitier. Libre à vous de choisir si vous voulez un design sobre ou si vous êtes prêt à dépenser un peu plus pour le côté esthétique.

Pour l'alimentation, il faut savoir qu'elle varie normalement avec le nombre de périphériques inclus dans l'ordinateur. Pour une configuration de bureau normal, un 450 Watts est suffisant. La meilleure façon de savoir s'il est correct pour l'ordinateur, c'est de regarder si les câbles qui sortent de l'alimentation sont en nombre suffisant pour être branchés sur toutes les pièces. Vérifiez d'ailleurs qu'il y a assez de prises pour le type de périphériques que vous avez (prises pour IDE ou SATA). Le second facteur qui fait varier le prix est le bruit qu'il émet. Payer un petit extra pour en avoir un plus silencieux est somme tout intéressant. Petite note, les alimentations qui viennent avec un boitier ne sont pas silencieuses, mais pas énormément bruyantes non plus.

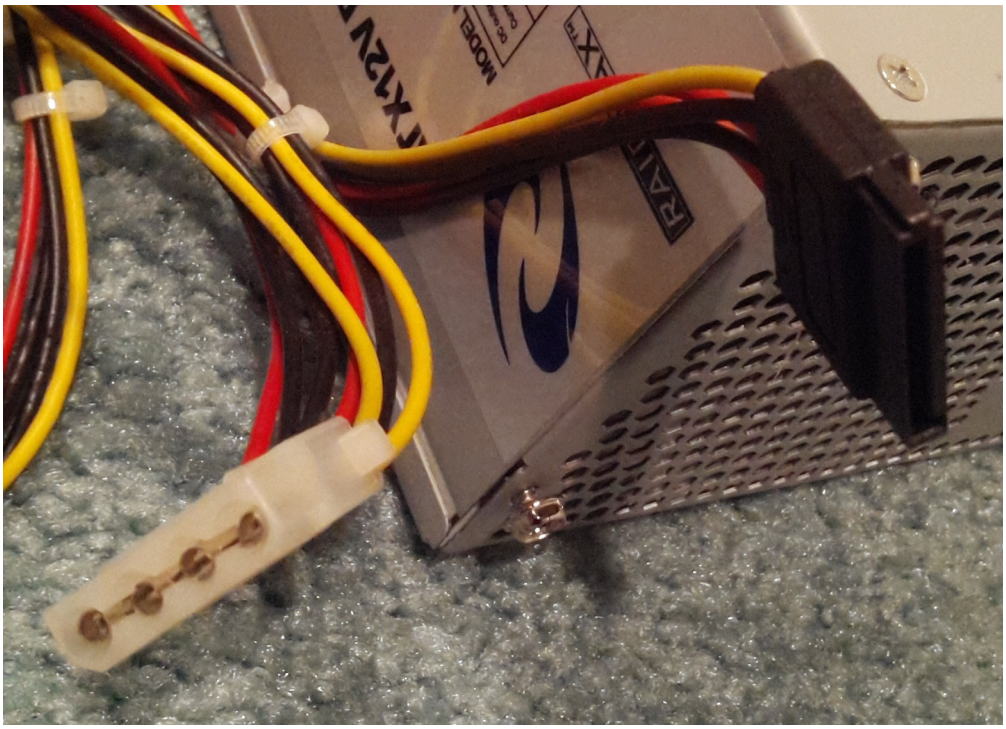

Illustration 7: Les composantes d'un ordinateur : Câble alimentation IDE (blanc) et SATA (noir)

4.10 Carte mère

La carte mère est le centre nerveux d'un ordinateur. Tous les périphériques viennent s'installer ou se brancher dessus alors il faut la choisir en fonction de ce que nous voulons brancher dans l'ordinateur. Il faut aussi garder en tête que beaucoup de pièces de bases peuvent être incluses.

Il n'est pas rare de voir des cartes mères avec un port réseau, un port vidéo et un port audio. Ce sont donc des composantes qui n'ont pas à être achetées sauf si nous désirons de meilleures performances. Par contre, il est inutile de payer pour ce que nous n'utilisons pas, alors si nous choisissons de prendre une meilleure carte graphique, autant ne pas acheter une carte mère l'incluant et de plutôt se tourner vers une qui est de meilleure qualité pour le même prix.

J'ai gardé cette pièce pour la fin puisqu'elle dépend de tout le reste. Quand vous avez choisi le type de disque dur, le processeur, la mémoire vive, si vous voulez une carte vidéo ou non, etc., vous devez regarder tous les ports et interfaces dont vous avez besoin. N'oubliez pas de penser pour le futur en prenant quelques ports libres de plus en réserve pour vos ajouts futurs.

5 Les systèmes d'exploitation

5.1 Introduction

Le système d'exploitation est le premier logiciel qui est démarré lorsque vous ouvrez votre ordinateur. C'est lui qui permet de lancer d'autres applications, de se connecter sur un réseau et de gérer toutes les ressources de votre système. De nos jours, dans nos maisons, il y a soit Windows, Mac ou Linux. J'ai inclus quelques-uns plus anciens juste pour donner une idée de la progression.

5.2 DOS

DOS est l'acronyme de "Disk Operating System". Il ne peut exécuter qu'un seul logiciel à la fois et si ce dernier s'arrête brutalement, il faut redémarrer l'ordinateur. Cet environnement est uniquement en ligne de commandes, mais cela n'empêche pas les jeux d'avoir leur propre environnement graphique. Tous les pilotes, en particulier la souris pour certains jeux, doivent être appelés en ouvrant un programme spécial avant d'exécuter l'application désirée. Normalement, cela est fait dans un script au langage Batch qui s'appelle "autoexec.bat" et qui se démarre à l'ouverture de DOS.

Ce système d'exploitation était utilisé entre 1981 et 1995. Ensuite, les gens se sont mis à utiliser Windows 95 qui lui-même démarrait comme logiciel principal dans DOS. Cela a été ainsi pour les versions de Windows 3.1, 95, 98 et ME (Milennium Edition) en 2000.

Il y a plusieurs saveurs de DOS puisque c'est un peu comme un standard qui a été implémenté par différents vendeurs. Par exemple, il y a le PC-DOS fait par IBM et MS-DOS que Microsoft avait acheté et continué de développer par la suite.

Pour les bidouilleurs et les nostalgiques, il y a une version de DOS qui existe encore et qui est à source ouverte. Son nom est FreeDOS[4].

4 https://r.foilen.com/freedos : Site officiel de FreeDOS

5.3 OS/2

En 1987, IBM et Microsoft se sont associés pour créer un système d'exploitation graphique et multitâches appelé OS/2. Cela n'a pas duré et IBM est resté le seul propriétaire de ce logiciel tandis que Microsoft a créé son propre système d'exploitation. La dernière version offerte était la 4.52 en décembre 2001. Le soutien technique offert par IBM s'est terminé en décembre 2006.

Il y a encore une petite communauté[5] qui développe des applications OS/2 pour le plaisir, tout comme il y a encore une communauté pour DOS avec FreeDOS.

5.4 Windows

Microsoft a un long passé avec ses versions de Windows. La version 1 est sortie en 1985, la version 2 en 1987 et la version 3 en 1990. C'est à partir de cette dernière version que ce système a gagné en popularité avec la 3.1. Par la suite, il y a eu Windows 95, Windows 98, Windows ME (Milennium Edition) en 2000. Ces trois versions continuaient d'utiliser DOS en dessous.

En parallèle, Microsoft faisait une version entreprise de son système d'exploitation du nom de Windows NT. C'est important de le mentionner étant donné que la version suivante pour particuliers est basée sur le même noyau NT plutôt que l'ancien noyau. C'est ainsi que Windows XP est sorti en 2001 et a depuis eu trois mises à jour majeures appelées "Service Pack".

Par la suite, il y a eu Windows Vista qui est sorti 6 ans plus tard en 2007 et qui a connu beaucoup de problèmes. Le système a reçu plusieurs changements visuels et de paradigme de sécurité d'un seul coup et pour plusieurs utilisateurs, cette version causait trop de problèmes lors de l'utilisation. Il y avait aussi des critiques sur les spécifications matérielles minimales requises qui étaient assez élevées. Il faillait pratiquement acheter un nouvel ordinateur pour le rouler sans anicroche.

Seulement deux ans plus tard, Vista a été délaissé pour la version Windows 7 sortie en 2009. L'expérience utilisateur a été grandement améliorée d'après les commentaires de tous et le matériel requis est moins important. C'est pourquoi il peut aussi fonctionner sur des NetBooks qui sont des minis ordinateurs portables avec très peu de ressources matérielles. Par contre, certaines fonctionnalités sont tronquées pour le permettre.

Dans la lignée du support de matériel moins performant et suivant la croissance fulgurante des tablettes comme les iPad, la nouvelle version 8 de Windows, sortie en octobre 2012 est faite pour les ordinateurs de bureau et pour les tablettes. L'interface du bureau virtuel a été modifiée pour ressembler plus à des chaines de télévision ce qui est parfait pour les tablettes, mais très peu convivial pour les ordinateurs. C'est encore un changement radical (tout comme pour Vista) et les utilisateurs sont un peu hésitants à faire le saut. Par contre, toute nouvelle machine achetée avec Windows va bien entendu avoir cette version. Ce n'est donc qu'une question de temps avant qu'une adoption de masse se produise. Un autre coup de foudre des utilisateurs avancés est que le système est plus barré, un peu à la Apple, avec un magasin virtuel contrôlé par Microsoft et la sécurité plus sévère. C'est pourquoi des acteurs importants comme Steam qui est un vendeur de jeu désirent se diriger plus sur Linux[6] qui est un système plus ouvert.

5 https://r.foilen.com/os2 : Communauté OS/2
6 https://r.foilen.com/steam-linux : Nouvelles sur la progression de Steam pour Linux

5.5 Mac OS

Depuis 1984, la compagnie Apple offre des ordinateurs personnels faciles à utiliser grâce à leur interface graphique et leur souris. De la première version à la 4.1 sortie en 1987, il n'était possible que de rouler une application à la fois comme pour DOS, mais entièrement en mode graphique. À partir de la version 5 en fin 1987, il était possible d'exécuter plusieurs logiciels en même temps.

Par la suite, il y a eu plusieurs incrémentations jusqu'à la version 9 et c'est en 2001 qu'un changement radical a été fait avec la version 10.0. Cette version est basée sur le système d'exploitation Unix. C'est un peu comme Microsoft qui utilisait DOS comme base; Apple utilise Unix. Depuis cette version, il y a eu 8 itérations dont la plus récente en 2012 qui est rendue à 10.8 et portant le petit nom "Mac OS X Montain Lion".

Ce système d'exploitation n'est utilisable officiellement qu'avec les ordinateurs d'Apple. Par contre, il existe plusieurs projets pour permettre d'installer Mac OS X sur un PC comme PC-EFI[7], mais dans la majorité des cas, la licence[8] d'utilisation rend ce procédé illégal.

5.6 Unix/Linux

Unix existe depuis 1969 et a commencé comme système d'exploitation vendu par AT&T. Par la suite, plusieurs autres vendeurs ont créé leur propre variante telle Solaris, HP-UX et AIX. Par contre, c'était toujours payant et le code source n'était pas disponible pour l'améliorer et c'est pourquoi plusieurs communautés se sont créées avec le désir de faire une implémentation de Unix ouverte. Il y a eu presque en même temps BSD, MINIX et Linux.

La première version de Linux est sortie en 1991. Il faut préciser que Linux en tant que tel n'est que le noyau du système d'exploitation et il n'est pas possible de faire quoi que ce soit seulement avec lui. Il sert à gérer les pilotes matériels, les formats de systèmes de fichiers et d'autres ressources matérielles. Pour le restant comme la ligne de commande, les éditeurs de fichiers, le mode graphique, etc., ce sont tous des logiciels à part.

Étant donné que seul, Linux ne sert à rien, il faut l'entourer de plusieurs logiciels utilitaires pour bien profiter de ce système d'exploitation. Puisque le choix de certains logiciels par rapport aux autres est difficile à faire, plusieurs entreprises et communautés se sont formées pour créer des distributions de Linux avec un choix plus ou moins grand d'applications. Il y a des centaines de distributions existantes, mais les plus connues sont entre autres RedHat[9], Gentoo[10], Debian[11] et Slackware[12]. Par la suite, d'autres communautés se sont formées par-dessus ces distributions pour incorporer des logiciels supplémentaires et simplifier l'installation. Présentement, la distribution la plus connue pour les néophytes et simple d'installation comme environnement de bureau est Ubuntu[13] qui est basé sur Debian. Une autre très active est Fedora[14] qui est basé sur RedHat.

L'utilisation de Linux est très repandue dans les entreprises pour fournir des services web et elle commence à pénétrer le marché des ordinateurs de bureau. Il y a déjà suffisamment de logiciels

7 https://r.foilen.com/pc-efi : PC-EFI pour installer Mac OS sur un PC
8 Voir chapitre 26 à la page 113
9 https://r.foilen.com/redhat : Le site de l'entreprise RedHat
10 https://r.foilen.com/gentoo : Le site de la communauté Gentoo
11 https://r.foilen.com/debian : Le site de la communauté Debian
12 https://r.foilen.com/slackware : Le site de la communauté Slackware
13 https://r.foilen.com/f-ubuntu : Vidéos sur l'installation et l'utilisation d'Ubuntu
14 https://r.foilen.com/fedora : Le site de la communauté Fedora

disponibles dont la majorité totalement gratuite pour pouvoir travailler efficacement sur une machine Linux. Par contre, il n'y a pas beaucoup de grands titres de jeux qui sont disponibles pour le moment et d'applications commerciales que les gens connaissent bien sur Windows. Il y a donc beaucoup d'apprentissages à faire, mais plusieurs personnes qui ne sont pas connaisseurs en informatique apprécient ce système d'exploitation pour leurs tâches quotidiennes. De plus cela pourrait bientôt changer puisque Steam (un magasin virtuel de jeux vidéos) est en train de créer une version Linux de ce logiciel. L'entreprise Valve qui est derrière cette application va aussi porter son moteur de jeux 3D (Source Engine). Ce qui fait que bientôt, il risque d'y avoir plus de jeux disponibles sur ce système d'exploitation.

6 Les systèmes de fichiers

6.1 Introduction et exemple

Tout support physique de données comme un disque dur, une disquette, une clé USB ou un DVD a besoin d'une structure logique pour emmagasiner ces données. Il faut voir ces mémoires comme des cases à remplir avec n'importe quoi comme données. Le but recherché est de structurer les données en fichiers et en répertoires. Pour se faire, il faut faire un lien entre la partie matérielle qui ne supporte qu'une série de bits de données emmagasinés et la partie logique qui définit les limites des fichiers (la position et la taille).

Allons-y par étapes en commençant avec un disque vide pour terminer avec un système de fichier que nous aurons créé de toutes pièces. Les vrais systèmes de fichiers sont souvent plus complexes puisqu'ils offrent plus d'options, mais l'idée de base est toujours la même. Nous verrons donc comment écrire un seul fichier sur un disque dur vierge, poursuivre avec l'ajout de plusieurs fichiers les uns à la suite des autres et terminer avec la gestion des répertoires.

6.1.1 Un seul fichier

Prenons le cas où nous avons un seul fichier à enregistrer et qu'il est petit par rapport au disque. Ce fichier va simplement contenir la phrase "Allo Monde". Avant même de l'enregistrer, le disque est vide, mais ce vide n'existe pas en informatique. C'est pourquoi nous allons choisir de mettre une valeur par défaut qui sera le nombre zéro. L'illustration suivante montre dans sa première ligne le contenu d'un disque vide.

Vide	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Contenu	A	l	l	o		M	o	n	d	e	0	0	0	0	0	0	0	0	0	0	0	0
Grosseur	10	A	l	l	o		M	o	n	d	e	0	0	0	0	0	0	0	0	0	0	0
Titre	2	Y	o	10	A	l	l	o		M	o	n	d	e	0	0	0	0	0	0	0	0

Illustration 8: Système de fichiers : Exemple d'un enregistrement d'un fichier

Ensuite, puisque nous n'avons qu'un seul fichier, nous allons mettre le contenu directement sur le disque et c'est ce que nous pouvons voir sur la seconde ligne qui contient le texte "Allo Monde" et qui est suivie de tous les autres zéros. Lorsque nous désirons le lire, nous commencerons à lire le début et nous arrêterons dès que nous atteindrons le nombre "0". Voyez-vous le problème?

Le problème est que si nous désirons utiliser le nombre zéro, par exemple avec la phrase "Allo, j'ai 20 enfants", cela fera une fin de fichier en plein milieu du fichier. De plus, ici nous avons un fichier texte, mais si nous voulons créer un fichier quelconque comme un programme, toutes les valeurs peuvent être utilisées. Dans cette situation, nous ne pouvons pas choisir une simple valeur pour spécifier la fin du fichier. Ce que nous pourrions faire, c'est ajouter la taille du fichier juste avant son contenu. La troisième ligne montre que nous avons 10 caractères à lire pour ce fichier.

Avant de passer à plusieurs fichiers, nous aurons certainement besoin de plus d'information que simplement un contenu et une taille. Une autre propriété importante est le nom du fichier que nous pourrions placer juste en avant, tout en spécifiant aussi sa grosseur. Ainsi sur la ligne quatre, nous pouvons mettre que le titre "Yo" est d'une grosseur de 2 et suivit de la grosseur du fichier et de son

contenu.

6.1.2 Plusieurs fichiers

Maintenant, prenons un bon souffle et montons d'un niveau. Ajoutons plusieurs fichiers avec le format décrit dans la section précédente. Cela donne un graphique comme suit:

Illustration 9: Système de fichiers : Disque dur avec 4 fichiers qui se suivent

Ici, il y a quatre fichiers avec la section des noms en gris pâle et le contenu en gris foncé. Notons que la partie de contenu varie de grosseur selon la taille du fichier. Rendu à cette étape, plusieurs problèmes apparaissent. Le premier est que si nous désirons savoir le nom de tous les fichiers présents sur ce disque, il faut le lire en entier pour trouver toutes les sections de noms. Le second est que nous ne pouvons pas ajouter du texte à notre fichier de tantôt puisqu'il n'y a plus d'espace de libre à sa suite. Le troisième est que si nous effaçons le second fichier, pour utiliser l'espace ainsi libre, il faut balayer le disque pour la trouver et vérifier que la taille du nouveau fichier n'est pas plus grosse. Qui a dit que les ordinateurs c'était simple?

Pour résoudre tous ces problèmes, la façon la plus simple est de séparer le disque en trois parties: un espace de contenu, une carte des espaces de contenus libres et un espace des informations des fichiers.

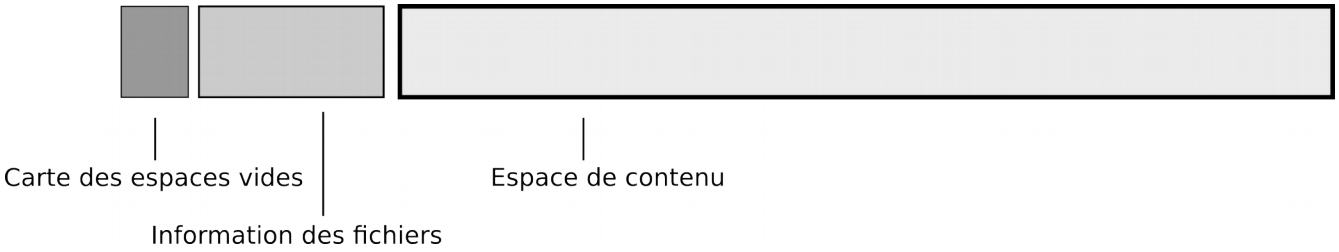

Illustration 10: Système de fichiers : Trois grandes sections d'un disque dur

Voici un exemple avec le seul fichier contenant "Allo Monde" suivit de la description de chaque section:

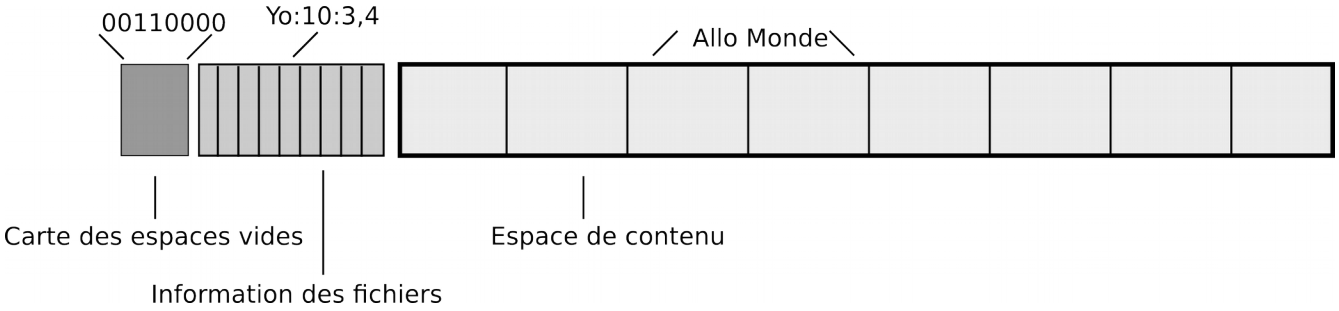

Illustration 11: Système de fichiers : Exemple des trois sections avec un "Allo Monde"

L'espace de contenu est séparé en plusieurs blocs d'une même taille. Ils doivent être assez gros pour ne pas être trop nombreux, mais assez petits pour ne pas gaspiller de l'espace avec des fichiers qui n'utiliseraient qu'une partie d'un bloc. Par exemple, nous pourrions faire des blocs de 5 caractères ce qui ferait que notre fichier "Allo Monde" de 10 caractères prendrait exactement 2 blocs. Par contre, le

fichier "Allo Monde!" (notez le point d'exclamation à la fin) utiliserait 2 blocs entiers plus 1 bloc final qui aurait 4 caractères de libres. La beauté de ce système est que les blocs peuvent être n'importe où. Ainsi, si nous ajoutons jusqu'à 4 caractères dans notre second fichier, nous n'aurons pas à utiliser d'autres blocs. Si nous ajoutons 6 caractères, alors le dernier bloc sera rempli et ensuite, nous pouvons prendre n'importe quel bloc libre qui pourrait être avant les autres, entre n'importe quel autre fichier, etc. Cela règle donc le 2e problème.

La carte des espaces de contenus libre est simplement composée d'une liste de "vrai/faux" qui dit quels blocs sont utilisés. Par exemple, si nous avons un total de 100 blocs, il y aura 100 "vrai/faux" qui se suivront. Il suffit de lire ces petites valeurs et prendre le premier disponible pour grossir un fichier. Cela règle donc le 3e problème.

Dans notre exemple, il y a huit blocs de contenu et huit bits qui composent la carte des espaces vides. Étant donné que seulement les blocs 3 et 4 sont utilisés, tous les bits sont à "0" sauf les deux à ces positions respectives.

L'espace des informations d'un fichier est l'endroit où il y a les entrées qui contiennent le nom, la taille et la liste des blocs. Dans notre exemple, le titre est "Yo", il y a 10 caractères de longs et le contenu est dans les blocs 3 et 4. En regroupant ses entrées ensemble, il est bien plus rapide de connaitre la liste des fichiers en ne balayant que cette petite partie du disque dur. Par contre, il faut qu'il y ait suffisamment d'espace pour les entrées pour contenir tous les noms de fichiers, mais pas trop pour laisser de l'espace aux blocs de contenu.

Comme vous pouvez le voir, les systèmes de fichiers doivent pouvoir balancer plusieurs items:

- il faut des blocs suffisamment gros pour ne pas qu'un même fichier en prenne trop, mais les blocs doivent être assez petits pour éviter le gaspillage des fins de fichiers;
- il faut que la zone de contenu soit la plus grosse possible, mais il faut assez d'espace pour les informations des fichiers, car s'il y a pénurie, il ne sera plus possible d'ajouter des fichiers malgré qu'il y ait beaucoup d'espace de contenu libre.

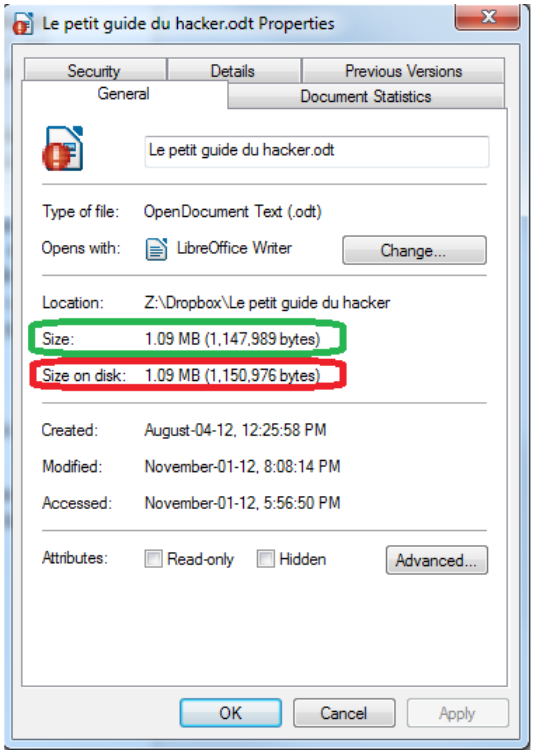

Illustration 12: Système de fichiers : Propriétés d'un fichier sous Windows

Vous pouvez d'ailleurs voir un exemple sur cette illustration des propriétés d'un fichier sous Windows. La grandeur réelle est ce qui est en vert, alors que la taille prise sur le disque dur est en rouge. Cette dernière est plus grande puisqu'elle dépend de l'espace pris par le dernier bloc.

6.1.3 Les répertoires

Pour terminer, il est crucial de pouvoir organiser tous les fichiers. C'est bien beau d'avoir un nom de fichier, il est aussi bien de pouvoir limiter la vue en ne les voyant pas tous d'un seul coup. C'est ici que vient le principe des répertoires.

Pour faire simple avec notre système de fichiers abstrait, nous pourrions simplement considérer un dossier comme étant un fichier qui possède le nom des fichiers et des sous-répertoires qu'il contient. Lors de la création de notre système de fichiers, le premier fichier pourrait être le répertoire racine et son contenu sera simplement une liste des positions des fichiers et autres répertoires qu'il contient.

6.2 Les fonctionnalités

Étant donné qu'il est possible de créer n'importe quelle structure au système de fichiers, plusieurs fabricants d'ordinateurs et de systèmes d'exploitation ont créé de multiples saveurs. Malgré les quelques décennies d'évolutions, il n'existe toujours pas un seul format standard que tous utilisent et ce ne sera certainement jamais le cas. La raison est fort simple : chaque structure a des côtés positifs et négatifs tout dépendant de l'utilisation de l'ordinateur. Par exemple, en ajoutant une fonctionnalité de gestions des permissions sur les fichiers (définir qui peut lire, écrire ou exécuter ces fichiers), cela ajoute des données à emmagasiner dans la mémoire, ce qui ralentit la lecture et qui enlève de l'espace libre pour

créer plus de contenu. Dans le cas d'un CD où tous les fichiers sont lisibles par tous, ce serait une fonctionnalité inutile.

Dans les fonctionnalités de base communes, il y a l'écriture, le déplacement et l'effacement de fichiers et répertoires. Selon le système de fichiers, certaines méthodes peuvent avoir des particularités différentes alors vous devrez vérifier selon celui que vous utilisez. Pour **l'écriture**, lorsque la grosseur d'un fichier est connue d'avance comme dans le cas d'une copie ou d'un téléchargement, il est parfois possible de réserver toute cette espace d'un seul coup pour s'assurer que les blocs de contenus ne soient pas éparpillés partout. Pour ce qui est du **déplacement** dans un autre répertoire, tant que celui-ci se fait sur une même partition[15], cela devrait être instantané puisque le contenu du fichier n'est pas copié ailleurs, mais seulement l'information sur le répertoire qui contient ce fichier est modifiée. Si ce déplacement est entre deux partitions, il y a d'abord une copie, suivie d'un effacement. Pour **l'effacement**, le contenu n'est pas effacé. Ce qui change, c'est que le lien entre le répertoire et le fichier est rompu et les blocs de contenus sont considérés comme libres. Ainsi, l'effacement est très rapide. Par contre, cela amène une fonctionnalité qui peut être très utile en cas d'effacement accidentel, mais très nuisible pour effacer des données sensibles : il est possible de **récupérer un fichier** dont ses blocs n'ont pas encore été écrasés par le contenu d'un nouveau fichier. Pour ce faire, vous pouvez utiliser un logiciel gratuit comme Recuva[1617] sous Windows.

Du côté des fonctionnalités plus avancées, certains systèmes de fichiers permettent de mettre des permissions sur les fichiers et dossiers et certains permettent aussi les journaux. Les **permissions** sont données par utilisateurs ou groupes d'utilisateurs et sont souvent pour décider qui peut écrire, lire et exécuter des fichiers. De plus pour les répertoires, il y a aussi la décision sur qui peut les lister et les traverser. Un point important à souligner sur les permissions est que celles-ci ne sont valables que durant l'exécution du système d'exploitation normal puisque c'est ce dernier qui bloque les accès. Par exemple, si vous démarrez l'ordinateur d'un ami avec un CD de Linux, vous pourrez voir tous les fichiers sur son disque dur sans tenir compte des permissions sauf si c'est d'un type comme EXT3. Malgré cela, puisque vous êtes administrateur sur ce CD de Linux, vous aurez la permission de tout voir même si le type était EXT3. La seule protection contre cela est le chiffrement des fichiers avec un logiciel qui crypte la partition en entier comme Truecrypt[18]. Pour ce qui est des **journaux**, ce système sert en cas de fermeture abrupte de l'ordinateur, comme lors d'une panne de courant. En gros, les modifications à un fichier sont enregistrées dans un journal avant d'être écrites dans la mémoire. Ainsi, en cas de coupure, les données partiellement mises dans le journal sont ignorées au lieu d'être en partie écrites en partie par-dessus l'ancien contenu, ce qui aurait rendu le fichier corrompu et souvent inutilisable.

6.3 Types

Un des premiers systèmes de fichiers utilisés est le **FAT** (File Allocation Table) qui était sur DOS et Windows 95. La limitation majeure de ce type est le nombre de caractères maximal dans le nom du fichier qui n'est que de huit, suivis d'une extension de maximum trois. De plus, il ne voit pas de distinction entre les majuscules et les minuscules. Par contre, son utilisation sur Windows était améliorée grâce à la **VFAT** (Virtual FAT) qui permet des noms de 255 caractères et de distinguer les minuscules.

15 Voir chapitre 6.4 à la page 30
16 https://r.foilen.com/recuva : pour télécharger Recuva
17 https://r.foilen.com/f-recuva : vidéo sur l'utilisation de Recuva
18 Voir chapitre 16.4.4 à la page 85

Pour la version 98 de Windows et aussi la version 95B, le **FAT32** a été introduit. Ce type donne les avantages de la VFAT en plus de permettre des fichiers d'une grandeur maximale de quatre Go.

Malgré que le FAT et le FAT32 semblent un peu vieillots, ils sont encore très utilisés sur les clés USB puisqu'ils sont simples et ne gaspillent pas trop d'espace avec les métadonnées.

En parallèle sur les serveurs Windows, la saveur NT, il y a eu le **NTFS** (NT File System) qui est sorti. C'est ce qui est utilisé par défaut sur Windows XP et les suivants et il permet de gérer les permissions, crypter les fichiers avec EFS (Encrypting File System), compresser les fichiers, établir des quotas par volume et avoir un journal.

Dans la zone Linux, les types les plus connus sont **EXT2**, **EXT3** et **EXT4**. La grande différence entre la version deux et trois est la fonction de journal et un point intéressant est qu'il est possible de passer de la version antérieure à la version supérieure sans avoir à migrer les données puisque l'EXT3 ne fait qu'ajouter des informations. Pour ce qui en est du EXT4, il ajoute de la vitesse par rapport au EXT3 en modifiant la structure. Par contre, il n'est pas possible de migrer sans souci à partir des versions deux ou trois (il faut copier les fichiers d'une partition à l'autre).

Un autre type que vous risquez de voir souvent est le **CDFS** (ISO 9660) pour les CD. Il possède trois implémentations qui s'encapsulent pour offrir des restrictions moindres. Par exemple, sur le niveau un, les noms de fichiers sont restreints à ceux sur le FAT (huit caractères et trois pour l'extension), tandis que le niveau 2 permet 180 caractères.

6.4 Les partitions

Sur un même disque dur, il est possible de mettre plusieurs formats de système de fichiers dans des parties différentes. Cela permet entre autres de démarrer sous Windows et Linux sans avoir à changer de disque dur. Une autre raison pour avoir de multiples parties est de séparer les données à des endroits différents. Ainsi, il est possible de mettre Windows et ses applications sur une partition et toutes les données des utilisateurs sur une autre. Cela rend plus facile la réinstallation de Windows en cas de problème avec ce dernier en effaçant tout sur la première partie sans toucher à la seconde. Chacune de ses parties est appelée une partition.

Par défaut, il ne peut y en avoir qu'un maximum de quatre à moins qu'il y en ait une d'un type spécial appelé : « partition étendue ». Cette dernière permet de créer autant de partitions désirées à l'intérieure de cette partition virtuelle.

6.5 Le master boot record

Lorsque l'ordinateur démarre, il ne connait rien des programmes, et plus particulièrement des systèmes d'exploitation, que vous avez installés. Ce qui a été décidé est qu'un programme en langage machine est directement lu au début du disque dur avant toute autre donnée. Par contre, il faut aussi un endroit bien précis où commencer à écrire la table des partitions. C'est pourquoi le master boot record ne fait que 512 octets. Le programme à cet endroit est très petit et ce qu'il fait normalement, c'est lancer un autre exécutable installé sur une partition. Par exemple, le MBR de Windows est très semblable à celui de DOS et va lire la table de partitions (qui commence au 512e octet), regarder laquelle a un bit spécial indiquant qu’il peut démarrer et exécuter un programme sur cette partition.

Le problème le plus courant avec le MBR est qu'à l'installation d'un système d'exploitation, ce dernier risque d'écraser ce secteur avec son propre code. Installer Windows va donc briser une installation

Linux puisque son programme de chargement n'est fait que pour détecter de multiples versions de Windows et pour les démarrer. Par contre, celui de Linux permet de démarrer plusieurs systèmes d'exploitation connus et pour les moins connus, il permet quand même une configuration spéciale pour les accepter.

À retenir, si vous désirez Windows et Linux sur une même machine, il vous faudra d'abord installer Windows et ensuite Linux pour ne pas devoir réinstaller le MBR. Si jamais vous désirez effacer Linux[19] et ravoir le MBR de Windows, il suffit de démarrer le CD d'installation de Windows en mode récupération. Puis, dans la ligne de commandes, écrire "fdisk /mbr" sur une version avant XP, "fixmbr" pour XP ou "bootrec /fixmbr" pour les suivantes. Par la suite, n'oubliez pas d'effacer la partition Linux et de grandir celle de Windows pour ne pas perdre ce précieux espace.

6.6 Outils et termes

Il y a deux mots que vous verrez souvent lorsque nous parlons de systèmes de fichiers et ce sont le formatage et la défragmentation.

Le **formatage** est l'action de choisir un type (un format) de système de fichiers à appliquer sur une partition. Il va en résulter une partition vierge de toute donnée et c'est une façon rapide de tout effacer le contenu d'une partition ou d'une clé USB. Il y a souvent les modes « complet » et « rapide » qui sont offerts. Le premier va tout effacer alors que le second ne va que réécrire l'entête. De manière visible les deux font la même chose (tout vider), mais avec le rapide, il est possible de récupérer des fichiers puisque les données sont encore sur le disque tant que d'autres fichiers ne les ont pas écrasés.

Un **défragmenteur** est un outil permettant de rechercher tous les blocs de contenus appartenant à un fichier et à les mettre les uns à la suite des autres. Cela accélère la lecture du fichier puisque tous les blocs seront contigus, mais l'opération est très longue à effectuer puisqu'il faut d'abord déplacer les blocs utilisés par d'autres fichiers plus loin et ensuite rapatrier les bons. Cette action était utile dans le passé avec les ordinateurs plus lents, mais de nos jours, le gain de vitesse n'est pas si appréciable, alors c'est un outil que nous pouvons mettre de côté. Il est aussi déconseillé de le faire sur une clé USB puisqu'elles sont faites de manière à lire n'importe quel bloc aussi rapidement, ce qui rend l'opération inutile, d'autant plus que leur durée de vie en sera gravement atteinte dû au trop grand nombre de réécritures durant les déplacements de blocs.

6.7 Récapitulatif

Pour ce chapitre, le tout a été exploré du plus petit au plus gros. Maintenant, récapitulons à l'envers:

- Un disque dur possède un mini master boot record de maximum 512 octets qui permet de démarrer un système d'exploitation.
- Après ce MBR, le disque est séparé en partitions.
- Chaque partition possède un système de fichier qui peut être de n'importe quel format.
- Le système de fichier choisi va décider comment gérer les fichiers, les répertoires et les permissions tout en ayant ses limitations propres.

19 https://r.foilen.com/f-linux-desinstallation : Vidéo sur comment effacer Linux

L'Internet

7 Le réseau Internet

7.1 Introduction

Internet est un réseau mondial d'ordinateurs qui existe depuis les années 1970, mais qui était surtout utilisé par l'État, les écoles et les scientifiques. L'adoption par le public s'est faite autour de 1996 et maintenant avoir un accès à ce réseau est rendu un service essentiel. Son utilisation est très simple, mais lorsque nous désirons creuser un peu pour voir comment il fonctionne, nous pouvons voir plusieurs couches qui sont superposées.

Pour commencer, il y a une carte réseau avec un numéro d'identification unique appelé MAC, ensuite cette carte va choisir ou recevoir une ou plusieurs adresses IP et avec la combinaison de cette adresse IP et d'un numéro de port, nous pourrons accéder aux services web qui nous intéressent. De plus, pour ne pas avoir à se souvenir par coeur des adresses IP des serveurs comme Google (d'autant plus qu'elles peuvent changer avec le temps), il y a le service DNS[20] qui permet de traduire une adresse comme "Google.com" en adresse IP.

20 https://r.foilen.com/f-dns : Vidéo sur le fonctionnement des serveurs DNS

7.2 Topologie du réseau

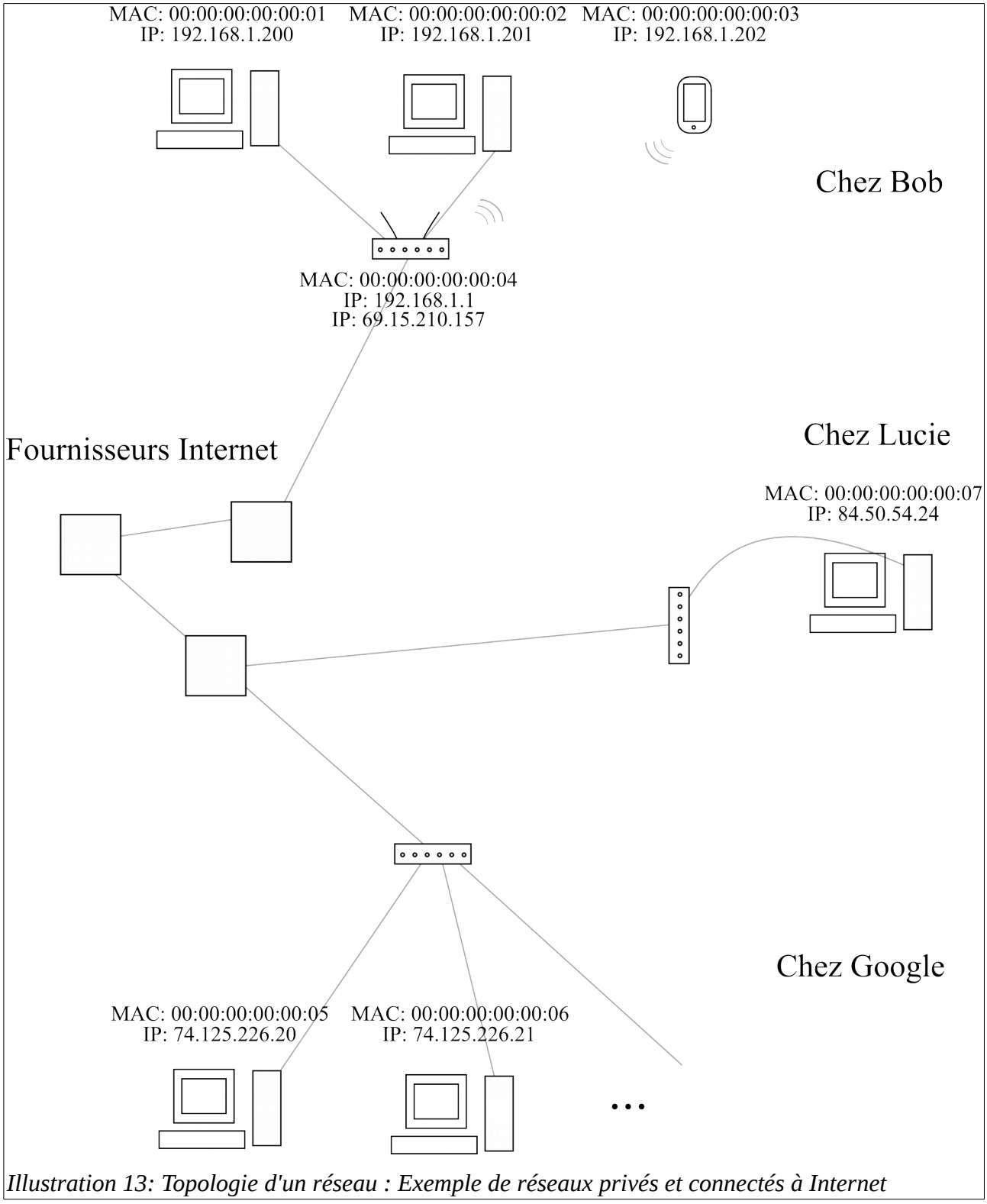

Illustration 13: Topologie d'un réseau : Exemple de réseaux privés et connectés à Internet

Sur ce diagramme, il y a la maison de Bob avec deux ordinateurs et une tablette. Pour son réseau local, il utilise un routeur avec fil et sans-fil qui lui est connecté à l'Internet. Chez Lucie, il n'y a qu'un seul ordinateur et il est connecté à Internet directement par l'entremise d'un modem. Pour Google, il possède des milliers d'ordinateurs qui sont en grande partie connectés directement à l'Internet, mais il pourrait aussi y avoir quelques réseaux locaux comme chez Bob.

Dans la demeure de Bob, chaque machine reçoit une adresse IP locale du routeur qui est du type 192.168.1.XXX. Les réseaux commençant par 192.168.XXX.XXX sont toujours locaux, ce qui signifie que ces adresses ne peuvent pas être rejointes par Internet. Si Lucie avait aussi plusieurs ordinateurs, elle aurait un réseau local avec les mêmes adresses IP que Bob. Par contre, elle ne pourrait pas se connecter directement chez Bob avec ces adresses locales et vice-versa. Ensuite, le routeur de Bob possède deux adresses IP: une locale et une pour le réseau Internet. Les gens qui veulent communiquer avec Bob devront utiliser son adresse IP d'Internet qui est le 69.15.210.157 et le routeur se débrouillera pour savoir à quel ordinateur sur le réseau local la communication est destinée. C'est l'utilisation du NAT (Network Address Translation) qui permet de transformer un IP d'Internet à un IP local et vice-versa.

Pour Lucie, c'est assez simple puisqu'elle est directement connectée à l'Internet. Par contre, elle n'a pas de routeur qui filtre les communications alors elle devra bien configurer son ordinateur pour ne pas partager des données non souhaitées avec l'utilisation d'un pare-feu par exemple.

Chez Google, étant donné que c'est une entreprise, il peut se permettre d'avoir de nombreuses connexions directes à l'Internet pour chacun de ses serveurs.

La dernière partie importante est les fournisseurs Internet. J'ai mis trois blocs, mais c'est bien plus immense que cela. Il faut penser que pour un même fournisseur, il faut pouvoir avoir des connexions entre leurs centres qui sont répartis dans une ville, dans des provinces et parfois même dans des pays. Ensuite, ces fournisseurs communiquent entre eux pour nous permettre de jouer à des jeux entre nous et de parler avec les Chinois.

7.3 Le matériel

Pour connecter toutes les machines (ordinateurs, serveurs, cellulaires ...) sur le grand réseau Internet, il faut utiliser du matériel spécialisé qui relie le tout ensemble. **L'ordinateur** lui-même est souvent assez limité puisqu'il ne permet que de se connecter sur un réseau Ethernet. Ce réseau est utilisé pour connecter des ordinateurs locaux ensemble, mais pour accéder à l'Internet, il faut souvent passer par d'autres types de réseaux comme la ligne téléphonique, le câble ou la fibre optique.

Pour se connecter avec la ligne téléphonique ou le câble à partir de notre ordinateur, il faut le brancher à un **modem** comme dans le cas de Lucie. Cet appareil a comme acronyme "modem" qui vient de modulation et démodulation. Ce matériel permet de faire le pont entre deux types de réseaux, par exemple, entre votre réseau local Ethernet et un autre type de réseau. Dans le cas de l'ADSL qui est utilisé avec la ligne téléphonique, le modem utilise souvent la technologie PPPOE qui signifie "Point to Point Protocol Over Ethernet". Il permet donc de passer de votre fil Ethernet à un réseau "Point to Point". C'est assez bas niveau, alors l'important ici, c'est qu'à la maison, vous allez toujours avoir besoin du modem de votre fournisseur Internet pour y accéder.

Lorsque vous désirez connecter plusieurs ordinateurs ensemble, vous pouvez utiliser un hub, une switch ou un routeur. Pour simplement expliquer leurs différences, commençons par ce qu'ils ont en communs: ce sont des appareils qui ont plusieurs ports (trous) pour brancher des ordinateurs entre eux.

Ensuite, le **hub** va prendre toutes les données transmises sur un port et les envoyer à tous les autres. Cela veut dire que les autres ordinateurs vont aussi recevoir les données qui ne leur sont pas destinées et que ce n'est pas très efficace lorsque tous les ports sont utilisés en même temps à plein régime puisque les connexions seront saturées avec des données inutiles. La **switch** est un peu plus intelligente et va se rappeler quels ordinateurs sont accessibles par quel port. Ainsi, les données ne vont pas partout, mais uniquement au port de destination. Les données sont alors transmises beaucoup plus efficacement et les ordinateurs connectés ne verront pas les données qui ne leur appartiennent pas. Le dernier, le **routeur**, est comme une switch, mais en plus, il possède un port déterminé pour y brancher un modem et ainsi connecter le réseau local à l'Internet. Parfois, il inclut un modem à l'intérieur, ce qui permet de sauver de l'espace avec un seul appareil qui est un deux dans un.

Pour terminer la section sur le matériel, chaque appareil qui permet de se connecter à un réseau: carte réseau sur l'ordinateur, modem et routeur possèdent toute une adresse MAC (Media Access Control) qui est unique. Cette adresse peut être modifiée temporairement sous Linux avec une commande spéciale, mais est plus difficile à modifier sur Windows. Certains routeurs permettent aussi de changer leur adresse MAC pour ainsi permettre le remplacement d'un ancien routeur défectueux sans avoir à changer les routes définies par des adresses MAC.

7.4 Protocoles de base

7.4.1 Modèle OSI

En informatique, il y a souvent plusieurs couches qui sont superposées pour englober les fonctionnalités de base et permettre de se concentrer sur le haut niveau d'une application. Par exemple, pour visiter un site web, le navigateur a suffisamment à gérer comme le contenu de la page web pour ne pas avoir à se soucier de savoir comment communiquer sur un réseau DSL. Pour faciliter la compréhension du réseau Internet, il y a un modèle qui existe qui s'appelle OSI (Open Systems Interconnection) et qui possède 7 niveaux.

1. Couche physique: C'est le niveau matériel. C'est relié au type de port physique qui est utilisé. Par exemple: un fil Ethernet, du sans-fil WiFi, du sans-fil Bluetooth, etc. C'est surtout au niveau de l'électricité sur comment envoyer et interpréter les données provenant des influx.
2. Couche de lien: Ce sont les protocoles utilisés pour transférer des données et pouvoir vérifier leur intégrité en cas de perte d'un signal électrique. Comme protocoles, il y a par exemple PPP (Point to Point Protocol) et Ethernet. Oui, Ethernet est aux couches physiques et liens puisque ce standard définit les deux couches primaires.
3. Couche de réseau: Là nous arrivons à la première couche logicielle. Cette couche permet d'envoyer des messages d'une grosseur variable qui peuvent être ensuite découpés en sous-messages par les couches inférieures. Pour l'Internet, il y a entre autres ICMP, IPv4 et IPv6.
4. Couche de transport: C'est la couche que les programmeurs d'applications touchent. Toutes celles avant sont peu utiles lors de la programmation d'une application, mais très utile pour développer un système d'exploitation. Parmi ces protocoles, il y a UDP et TCP qui sont majoritairement utilisés sur l'Internet.
5. Couche de session: Cette couche gère les dialogues entre les ordinateurs. Par contre, elle n'est pas normalement utilisée sur l'Internet, mais possiblement sur d'autres réseaux privés.

6. Couche de présentation: Cette couche permet d'encapsuler les données envoyées par l'application de manière transparente. Par exemple, une application pourrait crypter toutes les données avec le protocole SSL ou TLS sans avoir à changer quoi que ce soit dans son code (excepté d'ajouter un appel lors de la création de la connexion pour dire d'utiliser le cryptage).
7. Couche applicative: C'est la couche des protocoles utilisés dans les applications. Pour voir un site web, il y a HTTP qui est utilisé, pour envoyer des courriels c'est SMTP, etc. Vous verrez ces protocoles dans la section "Les services Internet"[21].

7.4.2 Détails pour l'Internet

Comme vu précédemment, ce ne sont pas toutes les couches qui sont utilisées par l'Internet. La première est le numéro 3 (réseau), ensuite la 4 (transport), parfois la 6 (présentation) et en dernier la 7 (applicative).

Au niveau du **réseau**, il y a **ICMP** (Internet Control Message Protocol) qui sert surtout à envoyer des pings et des messages d'erreurs. Les pings vérifient si une machine distante existe en lui envoyant un petit message "ping" et en attendant une réponse "pong". Le temps entre l'envoi et la réponse est le temps de latence qui indique le temps pris pour un message pour se rendre à destination et revenir. Ce temps est très important dans les jeux multijoueurs puisque s'il est de 5 secondes par exemple, cela veut dire que chaque action exécutée est reçue uniquement 2.5 secondes plus tard par l'autre joueur. C'est trop de temps pour lui tirer une balle, mais c'est correct pour jouer aux échecs. Pour ce qui est des messages d'erreurs, ce sont des messages qui indiquent qu'un service n'est pas disponible, qu'un hôte n'est pas accessible, etc. Au même niveau, il y a aussi **IPv4** et **IPv6** qui connaissent une adresse unique appelée adresse IP. La différence fondamentale entre la version 4 et 6 c'est le nombre d'octets utilisés pour définir une adresse. La version 4 est de quatre octets, donc une série de 4 nombres allant de 0 à 255 (ex.: 192.168.1.1). Pour le second protocole, c'est une série de 16 octets (ex.: 20a1:0db8:f6a4:85a3:0000:0000:ad5f:8021). Pour "faciliter" l'écriture, les octets pour l'IPv6 sont écris en hexadécimales[22]... C'est plus facile pour les programmeurs, mais pas pour les humains normaux. Un octet est représenté par 2 caractères alphanumériques lorsqu'il est en hexadécimal. Dans l'exemple, il y a 32 caractères, donc 16 groupes de 2 caractères et par conséquent 16 octets. Le but d'avoir IPv6 est d'avoir assez d'adresses possibles étant donné que l'IPv4 est en train d'être saturé. Pour terminer, il y a des espaces d'adresses de réservés pour des réseaux privés[23] comme dans le cas de Bob.

Au niveau du **transport**, il y a le protocole **UDP** qui permet d'envoyer des messages à un destinataire ou plusieurs destinataires par multicasting. Ce protocole ne vérifie pas qu'un message s'est bien rendu ou qu'il est reçu dans le bon ordre, il faut donc que l'application gère cette partie à moins que la perte de données ne soit pas importante. L'utilisation la plus courante est dans le streaming de vidéos ou de radios puisque ce n'est pas grave de perdre quelques millisecondes étant donné que cela ne fait qu'une petite coupure. En ne faisant pas de vérification de réception, la transmission est plus rapide puisqu'il n'y a pas de paquets renvoyés à la source. Pour une connexion avec un ordre de réception et avec une vérification de la réception des messages, le protocole **TCP** est utilisé. Ce dernier est utilisé pratiquement partout puisque sur l'Internet, nous transmettons des données qui ne doivent pas être modifiées tels des courriels, des sites web, des fichiers d'images, etc. Que ce soit en UDP ou TCP, il faut toujours spécifier un numéro de port qui va de 0 à 65535 et ce numéro est choisi selon type de

21 Voir chapitre 9 à la page 44
22 Voir chapitre 21 à la page 98
23 Voir chapitre 22 à la page 99

service désiré. Pour faire une analogie avec le téléphone, l'adresse IP serait le numéro de téléphone d'une entreprise et le port serait le numéro de poste téléphonique de la personne à rejoindre.

Pour le niveau de la **présentation**, celle-ci n'est pas souvent utilisée. C'est surtout les protocoles **SSL et TLS** dont nous nous servons pour crypter des données transigées lorsque nécessaires. Normalement, quand vous visitez un site web, vous utilisez HTTP et toutes les données envoyées sont en clair. Pour les crypter, HTTPS est utilisé et c'est simplement le même protocole, mais avec la présentation TLS d'activée.

Le dernier niveau est celui **applicatif** et c'est au choix des programmeurs. Si vous créez un logiciel de clavardage, vous allez créer votre propre façon de passer vos messages aux autres utilisateurs. Les messages incluent la liste des utilisateurs, leur statut et les mots qu'ils écrivent. Pour plusieurs exemples de protocoles, allez voir la section "Les services Internet"[24].

7.4.3 Exemple d'utilisation

Maintenant, mettons tout ensemble pour voir comment naviguer sur un site web se fait de façon simple et élégante. Si nous désirons aller sur le site "http://google.com", nous allons ouvrir notre navigateur web préféré (Internet Exploreur, Firefox, Chrome, Safari, etc.) et taper cette adresse. Voici ce qui se passe en arrière-plan:

1. Le navigateur demande au service DNS (Domain Name System) en UDP sur le port 53 (port DNS) qu'elle est l'adresse IP de "google.com". Le DNS répond "74.125.226.16".
2. Le navigateur se connecte à la machine "74.125.226.16" en TCP sur le port 80 (port HTTP) et lui demande la page d'accueil. Le serveur web envoie la page web.

Vous remarquez certainement que nous restons toujours dans les niveaux 3 et supérieurs qui sont tous du niveau logiciel et non matériel. C'est ainsi que tout est décrit sur Internet étant donné que le matériel utilisé est très diversifié et par conséquent caché en allant dans les niveaux plus hauts.

7.5 Network Address Translation (NAT)

Rapidement, le NAT est une technique utilisée par les routeurs (entre autres) pour permettre à un groupe d'ordinateurs d'accéder à un autre réseau qui est l'Internet dans notre cas. C'est très utile à la maison lorsque nous possédons plusieurs machines parce que lorsque nous contactons un service avec un fournisseur Internet, ce dernier ne nous donne qu'une seule connexion à l'Internet, ce qui se traduit par une seule adresse IP. Pour permettre à une adresse IP d'être partagée par plusieurs ordinateurs, comme dans le cas de Bob, ce routeur sert alors de passerelle en utilisant le NAT.

En plus de permettre à des petits réseaux locaux domestiques de partager une seule connexion, le NAT sert à éviter la saturation des adresses IPv4 puisque celles-ci sont en train d'être complètement épuisées. En attendant l'adoption de l'IPv6 à grande échelle, utiliser des NAT permet de réduire le nombre d'IP utilisés. Pour expliquer le fonctionnement, il faut distinguer deux modes que peuvent prendre les ordinateurs locaux: client de services Internet et serveur de services Internet.

Lorsque vous désirez vous connecter à un site web, à un serveur de jeux, au clavardage, etc., vous êtes en **mode client**. Dans ce mode, votre ordinateur est celui qui crée la connexion entre deux machines et cela permet au routeur qui implémente le NAT de savoir où les messages en provenance du serveur

24 Voir chapitre 9 à la page 44

doivent retourner. Prenons l'exemple d'accéder au site de Google:

1. Votre ordinateur local ouvre un port TCP local au hasard. Cela pourrait être le port 5000. Nous avons donc une combinaison adresse/port qui pourrait être 192.168.1.100:5000.
2. Votre ordinateur local utilise ce port et demande au routeur de se connecter à la machine de Google sur son service web en TCP (74.125.226.16:80).
3. Supposons que le router est connecté à l'Internet avec l'IP de Bob (69.15.210.157), il va ouvrir un port local qui n'est pas utilisé étant donné que deux machines locales pourraient avoir leur port 5000 ouvert et le routeur ne peut pas ouvrir deux fois le même port. Dans ce cas, le port pourrait être 6000 et la combinaison adresse/port serait 69.15.210.157:6000.
4. À partir de ce point, le routeur sait que tous les paquets provenant de 192.168.1.100:5000 doivent être traduits comme provenant de 69.15.210.157:6000 et sont destinés à 74.125.226.16:80. Ensuite, tous les messages provenant de 74.125.226.16:80 qui sont destinés à 69.15.210.157:6000 sont en réalité destinés à 192.168.1.100:5000. C'est cela la traduction des adresses qui est effectuée dans les deux sens.

Tournons maintenant la table de sens et prenons le cas où vous aimeriez héberger un site web sur l'un de vos ordinateurs domestiques ou être le serveur d'un jeu en ligne. Dans ce cas, vous êtes en **mode serveur**. Pour un serveur web (HTTP), votre ordinateur écoutera sur le port TCP 80, ce qui a comme combinaison 192.168.1.100:80. Par contre, les gens sur l'Internet vont tenter d'accéder à votre serveur avec votre IP Internet qui est 69.15.210.157:80. Rendu là, le pauvre routeur n'a aucune idée où envoyer ces messages: serait-ce la machine 192.168.1.100, 192.168.1.101, 192.168.1.102 ou une autre qui peut répondre à cette requête? Pour démêler tout cela, vous devez configurer le routeur pour lui dire que si une personne se connecte au port 80, celui-ci doit être redirigé vers l'ordinateur 192.168.1.100. C'est normalement dans la section "serveur virtuel". Avec cela, le routeur a toutes les informations désirées pour traduire les adresses dans les deux sens.

7.6 Les outils

7.6.1 Ping

Faire un "ping" sur un hôte, c'est envoyer un petit message et attendre son retour. C'est possible grâce à une des commandes du protocole ICMP qui est "echo". Une fois le message revenu, nous pouvons en déduire que l'hôte existe et qu'il est présent sur le réseau, en plus de pouvoir mesurer le temps d'envoi et de retour du message pour ainsi connaitre la vitesse de transmission. Par contre, ne pas recevoir une réponse ne signifie pas que l'hôte n'est pas présent puisqu'il pourrait bloquer les "pings" et aussi, il pourrait y avoir des messages de perdus puisque le protocole utilisé n'est pas TCP, mais ICMP. C'est pour éviter la perte des messages que les outils utilisés envoient par défaut au moins quatre messages à la fois. Ainsi, même si un des quatre paquets est perdu, les trois autres pourront nous donner des informations importantes.

Si vous êtes sous Windows, vous pouvez effectuer un ping simplement en allant dans le menu démarrer, en ouvrant le programme "cmd.exe" que vous pouvez chercher et ensuite taper "ping HÔTE/IP". Par exemple, le résultat pour "ping google.com" pourrait être comme suit:

```
Pinging google.com [74.125.226.18] with 32 bytes of data:
Reply from 74.125.226.18: bytes=32 time=20ms TTL=57
```

```
Reply from 74.125.226.18: bytes=32 time=21ms TTL=57
Reply from 74.125.226.18: bytes=32 time=22ms TTL=57
Reply from 74.125.226.18: bytes=32 time=22ms TTL=57

Ping statistics for 74.125.226.18:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 20ms, Maximum = 22ms, Average = 21ms
```

Ce que nous pouvons voir est que l'adresse IP de Google est 74.125.226.18 et que le temps de réponse des quatre paquets est de 20, 21, 22 et 22 millisecondes.

Si vous êtes sur Linux, vous pouvez exécuter la même commande, mais par défaut, elle enverra éternellement des paquets plutôt que seulement quatre. Vous devez faire CTRL-D (end of file) ou CTRL-C (terminer l'exécution) pour l'arrêter et voir un sommaire.

7.6.2 Trace Route

L'Internet est composé d'une grande quantité d'ordinateurs interreliés et l'envoi d'un paquet sur le réseau se transmet d'une machine à une autre en espérant utiliser la route la plus courte. Pour connaitre le chemin utilisé, nous pouvons faire une trace de la route empruntée.

La technique utilisée est d'envoyer des requêtes "echo" sur ICMP (tout comme pour les pings), mais en ajustant le champ TTL (Time To Live). Cette valeur permet de dire le nombre maximal de machines à traverser avant de considérer que nous n'atteindrons jamais le serveur final. Par exemple, en plaçant cette valeur à 1, dès que notre message traverse la première machine, celle-ci répondra que malheureusement, l'hôte désiré n'a toujours pas été atteint. Ainsi, nous pouvons savoir que la première machine est celle qui vient d'envoyer le message d'erreur. En changeant la valeur pour 2, nous obtiendrons le message d'erreur de la seconde machine. Nous pouvons continuer ainsi jusqu'au moment que nous ne recevons plus de messages d'erreur, mais la réponse au ping. C'est ainsi que nous trouvons la route employée.

Si vous êtes sous Windows, vous pouvez utiliser cette fonction simplement en allant dans le menu démarrer, en ouvrant le programme "cmd.exe" que vous pouvez chercher et ensuite taper "tracert HÔTE/IP". Par exemple, le résultat pour "tracert google.com" pour moi est comme suit:

```
Tracing route to google.com [74.125.226.50]
over a maximum of 30 hops:

  1    <1 ms     2 ms    <1 ms  192.168.1.1
  2    13 ms    12 ms    14 ms  ip216-239-72-166.vif.net [216.239.72.166]
  3    13 ms    12 ms    13 ms  ip216-239-72-165.vif.net [216.239.72.165]
  4    12 ms    12 ms    12 ms  ip216-239-72-161.vif.net [216.239.72.161]
  5    20 ms    20 ms    20 ms  gw-google.torontointernetxchange.net [198.32.245.6]
  6    20 ms    20 ms    21 ms  216.239.47.114
  7    21 ms    21 ms    20 ms  64.233.175.132
```

```
  8    21 ms    21 ms    21 ms  74.125.226.50


Trace complete.
```

Puisque mon fournisseur d'Internet est Vif Internet, nous pouvons voir que la première machine est mon routeur, que les trois suivantes sont celles de mon fournisseur Internet et que les quatre dernières sont chez Google.

Si vous êtes sur Linux, vous pouvez exécuter "traceroute".

7.6.3 Nmap

Nous avons vu que tous les services web utilisent des ports différents par défaut. Avec Nmap, il est possible de rapidement scanner un hôte ou un réseau en entier pour savoir quels ports sont ouverts. Ce logiciel permet aussi de tenter de détecter quel système d'exploitation est utilisé par chacun de ces hôtes en analysant les paquets reçus. Un point qui démarque beaucoup cette application est qu'elle peut scanner sans que le service distant détecte une connexion puisque celle-ci ne sera pas ouverte en entier. Par contre, les pare-feux pourront toujours le voir, puisqu'ils vérifient les paquets entrants peu importe l'état de la connexion.

7.6.4 GRC shields UP

Le site de Gibson Research Corporation[25] possède plusieurs outils, dont le Shields UP. Le but de ce service est qu'il va scanner votre hôte pour vous et vous dire quels services sont accessibles à partir du web. C'est très utile surtout si vous avez un routeur et que vous essayez de configurer certains ports entrant pour valider qu'ils soient bien ouverts. Si vous utilisez uniquement Nmap à partir d'un autre ordinateur du réseau local, il risque de découvrir beaucoup plus de services, mais qui ne sont pas accessibles par le réseau Internet.

25 https://r.foilen.com/grc : Gibson Research Corporation

8 Le WiFi

8.1 Introduction

Le WiFi ou le réseau sans fil est une technologie très agréable à utiliser autant à la maison que dans un café. Par contre, lorsque nous créons un tel réseau à la maison, il ne faut pas oublier de le sécuriser. De plus, lorsque nous utilisons une borne WiFi publique, il faut faire attention, car nous ne savons jamais quand une personne malintentionnée va être à l'autre bout en train de nous espionner.

8.2 SSID

Pour facilement distinguer notre réseau de celui du voisin, nous pouvons lui donner un nom. Ce nom est le SSID (Service Set Identifier). Il est important de noter que ce nom peut être utilisé par d'autres personnes aussi, alors si votre ordinateur est configuré pour se connecter à un nom particulier et qu'une autre personne crée une borne avec ce nom, il se pourrait que vous vous connectiez sur celle dernière. Cela n'est pas un problème si vous mettez un mot de passe puisque votre ordinateur vous dira que le mot de passe n'est pas le bon s'il n'est pas sur le bon réseau.

Par défaut, les routeurs WiFi vont envoyer des messages visibles à tous indiquant qu'une borne avec le nom choisi existe. C'est ainsi qu'il est possible de voir la liste des routeurs offerts dans le coin. Par contre, vous pouvez décider de le cacher et à ce moment, vous devrez entrer vous-même le nom dans votre ordinateur puisqu'il ne sera pas visible. Il est important de noter que cela ne rend votre réseau invisible que dans la liste des réseaux disponible, mais si une personne surveille les réseaux sans fil et que vous utilisez le vôtre, il pourra voir le SSID que vous utilisez dans les messages transmis entre votre ordinateur et votre routeur.

8.3 Sécurité

Pour sécuriser votre réseau, il y a deux choses que vous pouvez faire: filtrer les adresses MAC et mettre un mot de passe.

La première technique est utile pour bloquer les personnes non expérimentées. Elle consiste à faire une liste des adresses MAC des cartes sans fil des ordinateurs et autres appareils qui se connectent au réseau WiFi. Le routeur ne va accepter que les adresses qui sont dans cette liste. Pour les personnes expérimentées, il est possible de surveiller le réseau et de voir les adresses MAC de ceux qui utilisent votre réseau. Par la suite, elles pourraient copier cette adresse lorsque vous n'utilisez plus le réseau (pour éviter une collision et que vous perdiez tous les deux accès au réseau).

Copier l'adresse MAC d'un utilisateur reconnu par un routeur est aussi une façon de contrer certains systèmes d'authentification pour des bornes publiques qui demandent un paiement. Si cette borne ne fait que conserver une liste des MAC qui ont payé, alors changer son adresse pour une dans cette liste permettra de ne pas avoir à payer.

La seconde technique est meilleure tant que vous choisissez un mot de passe assez bon. Par contre, sur les routeurs récents, ils vous laisseront choisir un type de mode de cryptage, ce qui peut ne pas être trop clair. Par exemple, ne prenez jamais le mode "**WEP** (Wired Equivalent Privacy)". Ce mode est vieux et peut rapidement être brisé[26] (en une ou deux minutes) peu importe la qualité du mot de passe puisqu'il a

26 https://r.foilen.com/tut-crack-wep : Tutoriel pour trouver les mots de passes WEP

une faille de sécurité. Préférez le **WPA2** qui est plus solide si vous utilisez un bon mot de passe. Pour briser ce dernier, il faudra que l'attaquant essaye tous les mots de passe possibles. Pour terminer, si vous désirez essayer de briser votre réseau, vous pouvez utiliser la suite "Aircrack-ng" sous Linux.

9 Les services Internet

9.1 Introduction

Dans cette section, plusieurs applications courantes d'Internet sont décrites avec plus ou moins de détails. Vous verrez entre autres quel protocole de transport (TCP ou UDP) et quel port par défaut sont utilisés. Les ports ne sont que ceux officiellement choisis et peuvent être modifiés par les administrateurs. Par exemple, le port par défaut d'un site web est le port 80. Il est ainsi possible d'accéder au site avec l'URL suivant: "http://www.google.com". Si le port avait été changé pour 5656, il aurait fallu le spécifier dans l'URL comme suit "http://www.google.com:5656".

9.2 Dynamic Host Configuration Protocol (DHCP)

Port UDP 67 pour envoyer une requête au serveur DHCP; Port UDP 68 pour recevoir la réponse du côté du client

Un des premiers services Internet avec lequel votre ordinateur communique est inconnu de la plupart des gens. Pourtant, sans lui, il faudrait manuellement configurer plusieurs paramètres comme notre adresse IP. En utilisant le DHCP, notre ordinateur va pouvoir **obtenir une adresse IP** locale de notre routeur pour accéder au réseau local et notre routeur va obtenir une adresse IP de notre fournisseur. Ce n'est pas tout, ce protocole va aussi nous donner une liste des ordinateurs importants du réseau comme le **Gateway** et les **serveurs DNS**.

Le Gateway est l'adresse IP de la machine où envoyer les messages d'une destination inconnue. Par exemple, si sur mon réseau local j'essaie de communiquer avec un autre ordinateur local, je vais aller directement à lui. Par contre, si j'essaye de me connecter sur les serveurs de Google, mon ordinateur ne connait pas ce réseau et va simplement envoyer le message au Gateway qui serait dans ce cas-ci l'adresse IP de mon routeur puisque ce dernier est celui qui a accès à l'Internet. Le Gateway va transmettre notre message au destinataire s'il est sur le même réseau que lui ou le transmettre à son tour à son Gateway.

Normalement, sur tout réseau, il ne devrait y avoir qu'un seul serveur DHCP. Par contre, s'il y en a plusieurs qui répondent à une même requête, le client prendra le premier reçu et avertira tous les serveurs de celui choisi.

Pour les plus curieux, vous vous demandez sûrement comment le protocole UDP peut être utilisé pour recevoir une adresse IP sans même avoir d'adresse IP préalablement assignée. Dans ce cas-ci, le message envoyé est un broadcast sur le réseau local (donc tous le reçoivent) et la réponse pourra vous revenir en utilisant votre adresse MAC qui est incluse dans votre message.

9.3 Domain Name System (DNS)

Port UDP 53

(Vidéo d'explications de ce chapitre disponible[27])

Comme décrit précédemment, le DNS sert à transformer une adresse comme "google.ca" en adresse IP de la machine à qui elle appartient étant donné que sur l'Internet, le seul identifiant utile pour les

27 https://r.foilen.com/f-dns : Vidéo sur le fonctionnement des serveurs DNS

connexions est l'adresse IP. C'est par conséquent le service que vous utilisez le plus tous les jours sans même y penser.

Il y a plusieurs types d'entrées qui peuvent être utilisés et demandés:

A: C'est le défaut et c'est simplement pour définir une adresse IPv4. Il est possible de définir des sous-domaines spécifiques ou tous les domaines.

```
foo.exemple.com. A 96.96.0.1
bar.exemple.com. A 96.96.0.2
*.bar.exemple.com. A 96.96.0.3
```

AAAA: C'est la même chose que "A", mais pour des adresses IPv6.

```
ipv6.google.com. AAAA 2001:4860:800f::68
```

MX (Mail Exchange): C'est pour donner l'adresse du serveur de courriels. Ainsi, il est possible de séparer les pages web d'un serveur et son service de courriels.

```
gmail.com. A 74.125.226.21
gmail.com. MX 74.125.65.27
```

CNAME (Canonical Name): C'est un alias pour un nom de domaine. Ainsi, vous pouvez écrire un autre hôte à résoudre plutôt que de donner une adresse IP. Cela permet de simplement modifier une adresse IP pour plusieurs hôtes à la fois lorsque les services sont sur la même machine.

```
firstfloor.exemple.com. CNAME homeexemple.dyndns.org.
secondfloor.exemple.com. CNAME homeexemple.dyndns.org.
homeexemple.dyndns.org. A 69.15.210.157
```

DNAME (Delegate Name): Même chose que CNAME, mais pour tous les sous-domaines étant donné qu'il n'est pas possible de mettre un "*" pour spécifier tous les sous-domaines à un CNAME.

```
exemple.com. DNAME homeexemple.dyndns.org.
homeexemple.dyndns.org. A 69.15.210.157
```

NS (Name Server): Étant donné qu'il n'y a pas qu'un seul serveur central qui gère tous les noms de domaines, il est possible de déléguer tous les sous-domaines à un autre serveur DNS.

```
gmail.com. NS ns2.google.com.
```

Si vous désirez explorer un peu ce qu'il y a sur l'Internet, vous pouvez utiliser l'outil "nslookup" sur Windows ou "dig" sur Linux. Dans "nslookup", vous pouvez écrire l'hôte à observer et vous pouvez modifier le type en écrivant "q=MX" ou "q=AAAA" par exemple. Avec "dig", il suffit d'appeler la commande avec le type comme "dig NS gmail.com".

9.4 World Wide Web (HTTP et HTTPS)

Port TCP 80 ou 443 (crypté)

Lorsque vous utilisez un navigateur Internet tels Internet Explorer, Firefox, Chrome, Safari ou autres, vous êtes en train d'explorer le World Wide Web ou ce que nous pouvons aussi appeler "des sites web". Ce protocole est celui qui permet de demander des documents, d'envoyer des informations et des fichiers et d'échanger des cookies avec le serveur.

Le protocole HTTP n'est pas crypté et rien ne prouve que l'ordinateur distant est bien celui avec lequel nous désirons nous connecter. Il est déconseillé de l'utiliser pour faire des transactions importantes telles que bancaire puisque toutes les informations du compte pourraient être interceptées. C'est pourquoi le protocole HTTPS a été créé. Ce dernier est identique au niveau des commandes qui sont envoyées au serveur pour demander des ressources, mais lors de l'initialisation de la communication, il y a une étape supplémentaire pour vérifier si l'ordinateur cible est bien celui qu'il prétend être, grâce à un certificat, et une étape pour crypter la connexion en entier. Pour en savoir plus sur la sécurité fournie par les certificats et ses limitations, il y a un chapitre sur ce sujet[28] et il y en a un autre pour décortiquer le protocole[29].

9.5 Emails (SMTP , POP3, IMAP)

SMTP: Port TCP 25 ou 465 (crypté); POP3: Port TCP 110 ou 995 (crypté) ; IMAP TCP 143 ou 993 (crypté)

Pour pouvoir communiquer sur l'Internet avec des courriels, plusieurs protocoles différents sont utilisés. Le SMTP sert à envoyer des courriels tandis que les POP3 et IMAP servent à les lire.

Le **SMTP** (Simple Mail Transport Protocol) est utilisé par votre logiciel de courriels (si vous ne l'envoyez pas d'une application web telle Gmail ou Hotmail) pour envoyer vos messages à votre fournisseur de courriels. Par la suite, ce dernier va se connecter au serveur SMTP de la destination et lui transférer le message. Par exemple, si mon courriel est moi@supermail.com et que j'envoie un message à toi@megamail.com, mon logiciel va envoyer le message à mon serveur supermail.com et ce serveur va ensuite envoyer le message à megamail.com.

Lorsque vous envoyez un courriel, les informations envoyées sont l'adresse de source, les adresses de destination, le sujet, la date et le message. Tout ceci est envoyé en texte clair (non crypté) et tous les champs peuvent être modifiés. Cette vieille façon de faire amène deux problèmes importants: pas d'authentification et pas de confidentialité. Il n'est donc pas possible d'être certain que le message que vous recevez vient bel et bien de votre ami et tout ce que vous écrivez peut être lu par des employés qui contrôlent le serveur où le message est passé.

Pour parer à ces deux problèmes, vous pouvez utiliser un certificat qui permet de signer les messages que vous envoyez (authentification que c'est bien de vous que le message provient et qu'il n'a pas été modifié en court de route) et avec le certificat de votre destinataire, vous pouvez crypter le message pour que seulement lui puisse le lire. Pour utiliser ces techniques, vous pouvez télécharger l'application GnuPG[30] ou créer un certificat gratuitement avec StartSSL[31] et configurer votre logiciel de courriels comme Thunderbird pour qu'il l'utilise. Pour comprendre comment les signatures électroniques fonctionnent, aller voir le chapitre sur la cryptographie[32].

Si vous désirez voir le fonctionnement du protocole SMTP plus en détail un chapitre y est consacré[33].

Le **POP3** (Post Office Protocol de version 3) est utilisé pour télécharger les courriels de la boite de réception au logiciel de courriels et ensuite pour les effacer du serveur. Il est possible de les laisser sur le serveur, mais ce n'est pas le but premier.

28 Voir chapitre 12.2 à la page 66
29 Voir chapitre 25.1 à la page 109
30 https://r.foilen.com/gnupg : Site officiel de GnuPG
31 https://r.foilen.com/startssl : Site de StartSSL pour obtenir un certificat gratuit
32 Voir chapitre 16.4.3 à la page 84
33 Voir chapitre 25.2 à la page 111

Le **IMAP** (Internet Message Access Protocol) est utilisé pour visionner des courriels à partir du serveur. Il permet aussi de gérer des dossiers dans lesquels les courriels peuvent être archivés. Le but est de garder les messages sur le serveur pour pouvoir y accéder à partir de plusieurs ordinateurs et de plusieurs endroits. Ainsi, à la maison, vous pouvez regarder vos courriels avec un logiciel de lecture et du travail, vous pouvez utiliser une application web comme Roundcube[34] ou un autre logiciel de lecture pour y accéder.

9.6 File Transfert Protocol (FTP)

Port TCP 21

Comme son nom l'indique, le FTP sert à transmettre des fichiers. Il est souvent utilisé avec les hébergements de site web gratuit ou payant pour pouvoir y déposer les fichiers composants le site. Vous pouvez utiliser le logiciel FileZilla[35] et vous avez même la possibilité de créer un serveur FTP sous Windows avec FileZilla Server, le tout gratuitement. Si vous ne désirez pas installer un nouveau logiciel, les différents systèmes d'exploitation ainsi que les navigateurs web populaires permettent aussi d'accéder à un serveur FTP en plaçant comme chemin *ftp://HÔTE* dans la barre de localisation.

Le plus grave problème de ce protocole est qu'il n'est pas crypté et même le mot de passe est envoyé en clair à qui veut bien le regarder. Une version plus sécuritaire est le SFTP qui utilise un serveur SSH[36]. Avec cette version, tout est crypté, même les fichiers envoyés, et il est toujours possible d'utiliser FileZilla pour y accéder.

Un autre problème est une fonctionnalité qui est maintenant désactivée par défaut et qui permet d'envoyer des fichiers d'un serveur FTP à un autre serveur FTP sans passer par le client. Les serveurs qui font face à l'Internet ne devraient jamais permettre cela puisqu'il est possible d'envoyer un faux fichier à n'importe quel serveur. De cette façon, un serveur FTP pourrait attaquer un serveur quelconque en y envoyant des instructions mises dans un fichier texte. Par exemple, un faux fichier pourrait contenir les instructions pour envoyer un courriel à un serveur SMTP, ce qui ferait que le serveur FTP servirait comme plate-forme pour envoyer des pourriels en envoyant un fichier (qui contient les instructions SMTP) à un serveur SMTP au lieu d'un serveur FTP..

9.7 Partage de fichiers Windows ou Samba sous Unix (Samba)

Ports TCP et UDP 137-139

Le partage de fichiers et d'imprimantes avec Windows est servi par le service Netbios. Ce service est aussi accessible avec Linux grâce à l'application Samba[37].

9.8 Partage de fichiers sous Unix (NFS)

Port TCP et UDP 2049

NFS est un protocole de partage de fichiers tout comme Samba, mais qui tient compte de la sécurité des fichiers Unix comme les permissions d'accès selon les utilisateurs et les paramètres d'exécution, de lecture et d'écriture. Cet acronyme veut dire Network File System.

34 https://r.foilen.com/roundcube : Application web en PHP pour lire des courriels sur un serveur utilisant IMAP
35 https://r.foilen.com/filezilla : Logiciel de transfert de fichiers par FTP ou SFTP
36 Voir chapitre 9.13 à la page 54
37 https://r.foilen.com/samba : Logiciel pour accéder au partage de fichiers Windows

9.9 Internet Relay Chat (IRC)

Ports TCP 6660-6669 et 7000

Ce protocole est utilisé pour communiquer avec des gens en direct. Un serveur IRC est composé de plusieurs "channels" qui sont des salles de clavardage. Une fois dans une salle, il est possible de voir qui est présent, communiquer avec tout le monde en même temps et communiquer avec une personne en particulier. Chaque salle est gérée par des opérateurs qui peuvent choisir un sujet pour la salle, bannir les indésirables et faire sortir des gens. Il est aussi possible de configurer une salle pour qu'elle ait un mot de passe ou qu'elle soit modérée. La modération fait en sorte que seuls les opérateurs et les gens qui reçoivent une voix peuvent écrire.

Pour se connecter sur ce réseau, il faut utiliser un logiciel client comme mIRC qui est très populaire sur Windows. Ensuite, il faut se connecter à un réseau de serveurs. Une énorme liste est présente dans mIRC et les plus utilisés sont QuakeNet, IRCnet, Undernet, Efnet et DALnet. Chaque réseau est formé par plusieurs serveurs qui accueillent des utilisateurs et qui propagent leurs messages aux autres serveurs. Pour continuer, il faut choisir une salle comme #francais, #ubuntu, #canada, #france, etc. Il est souvent possible pour les serveurs de donner une liste des salles disponibles avec la quantité de personnes présentes à l'intérieur.

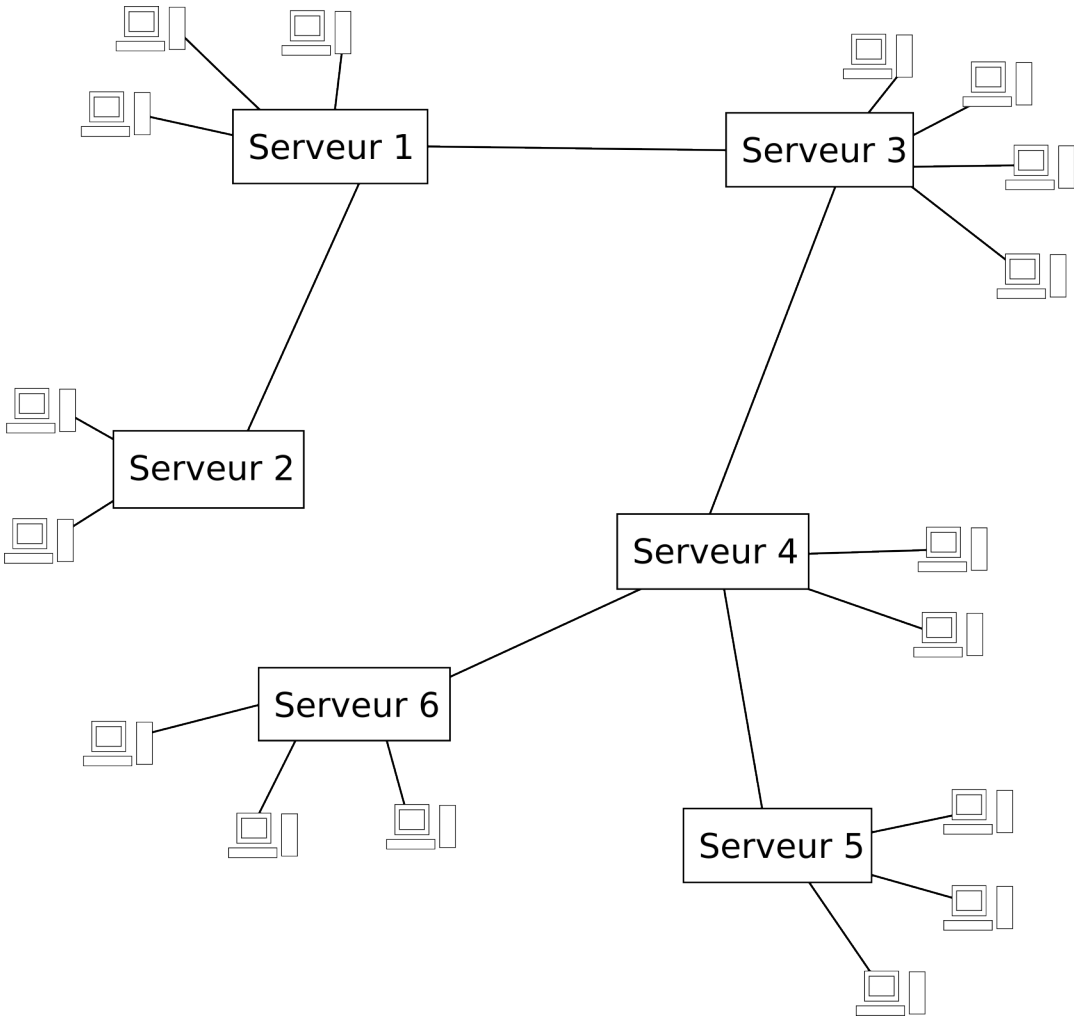

Illustration 14: Internet Relay Chat : Exemple d'un réseau IRC distribué

En plus de servir d'outil de clavardage, IRC possède d'autres fonctionnalités comme l'envoie de fichiers entre les utilisateurs. Étant donné qu'il y a plusieurs logiciels qui peuvent être programmés pour faire des tâches automatisées, il y a des robots qui permettent de lister plusieurs fichiers qu'ils rendent disponibles à tous et les gens n'ont qu'à écrire une phrase spécifiée pour entrer dans la file d'attente et recevoir le fichier quand leur tour est venu. Bien entendu, plusieurs robots peuvent aussi envoyer des virus à tous ceux qui joignent une salle, alors méfiez-vous. Vérifiez le nom d'extension[38] du fichier avant de l'accepter pour être certain qu'il n'est pas dangereux.

9.10 Usenet (NNTP)

Port TCP 119 ou 563 (crypté)

Usenet est un vieux service de forum de discussion qui existe encore, mais qui devient de moins en moins accessible. Pour le décrire simplement, des messages sont postés dans des groupes de discussion et les gens qui sont inscrits à ces groupes peuvent voir ces messages. C'est donc un peu comme les courriels sauf qu'il est possible de voir des messages plus anciens lorsque nous nous inscrivons.

Le réseau Usenet en tant que tel ne possède qu'un seul réseau qui est distribué sur plusieurs serveurs. C'est un peu comme pour IRC sur lequel il y a plusieurs serveurs sur plusieurs réseaux différents, mais à la différence qu'Usenet n'a qu'un seul réseau.

À la base, ce service a été conçu pour n'échanger que des messages au format texte, mais certaines personnes ont trouvé une façon d'outrepasser cette limitation en encodant des fichiers binaires (images, sons, vidéos, logiciels...) avec uniquement des caractères textes grâce à la Base 64[39]. Étant donné qu'il y a beaucoup moins de caractères disponibles que de valeurs binaires, les fichiers prennent ainsi plus de place. C'est pourquoi les fichiers binaires ne sont acceptés que dans des groupes spécifiques.

Pour ce qui est d'y accéder, auparavant, les fournisseurs de service Internet offraient l'accès à leurs clients. En possédant des serveurs d'Usenet, ces fournisseurs devaient disposer d'assez de ressources pour emmagasiner plusieurs journées de messages chez eux. C'est pourquoi ils décidaient souvent de ne pas donner l'accès aux groupes qui partageaient du binaire puisque c'était le plus demandant. Étant donné la faible utilisation du service dû à la croissance des forums sur le web et des services d'échanges de fichiers, de moins en moins de fournisseurs de service Internet hébergent ce type de serveurs. Pour ceux d'entre vous qui souhaitent encore y accéder, vous devrez passer par des services payants tels Easynews[40] et Giganews[41], mais ils offrent quelques jours gratuits si vous n'êtes que curieux.

9.11 Gnutella

Port TCP 6346

Ce protocole sert à partager, à rechercher et à télécharger des fichiers de manière décentralisée. Les fichiers les plus disponibles sont de la musique et des vidéos et bien entendu, certains sont légaux et d'autres illégaux. Les logiciels les plus utilisés qui permettent d'accéder à ce réseau sont FrostWire[42],

38 Voir chapitre 20 à la page 94
39 Voir chapitre 15 à la page 78
40 https://r.foilen.com/easynews : Le site d'Easynews pour accéder à Usernet
41 https://r.foilen.com/giganews : Le site de Giganews pour accéder à Usernet
42 https://r.foilen.com/frostwire : FrostWire, un logiciel de partage de fichiers

gtk-gnutella[43] et Shareaza[44] (attention, shareaza.com est un faux; vous devez prendre la version sur SourceForge).

Ce réseau a été créé après la fermeture de Napster. Ce dernier était un logiciel de partage de fichiers, mais qui était centralisé. Cela veut dire que toutes les recherches étaient faites au niveau du serveur de Napster, ce qui fait qu'il n'a fallu qu'une ordonnance de la cour pour fermer ce site et ainsi terminer le service en entier. Le protocole Gnutella est décentralisé, ainsi il faut que tous les ordinateurs connectés au réseau soient fermés pour qu'il n'existe plus. Les recherches sont envoyées aux machines sur lesquelles nous sommes connectés et celles-ci regardent dans leurs fichiers partagés si elles ont le fichier et demandent aussi aux machines sur lesquelles elles sont connectées de leur donner des résultats. Pour télécharger un fichier, la personne qui offre et celle qui reçoit se connectent directement entre elles, ce qui fait que ce n'est pas anonyme, mais plus rapide comme transfert.

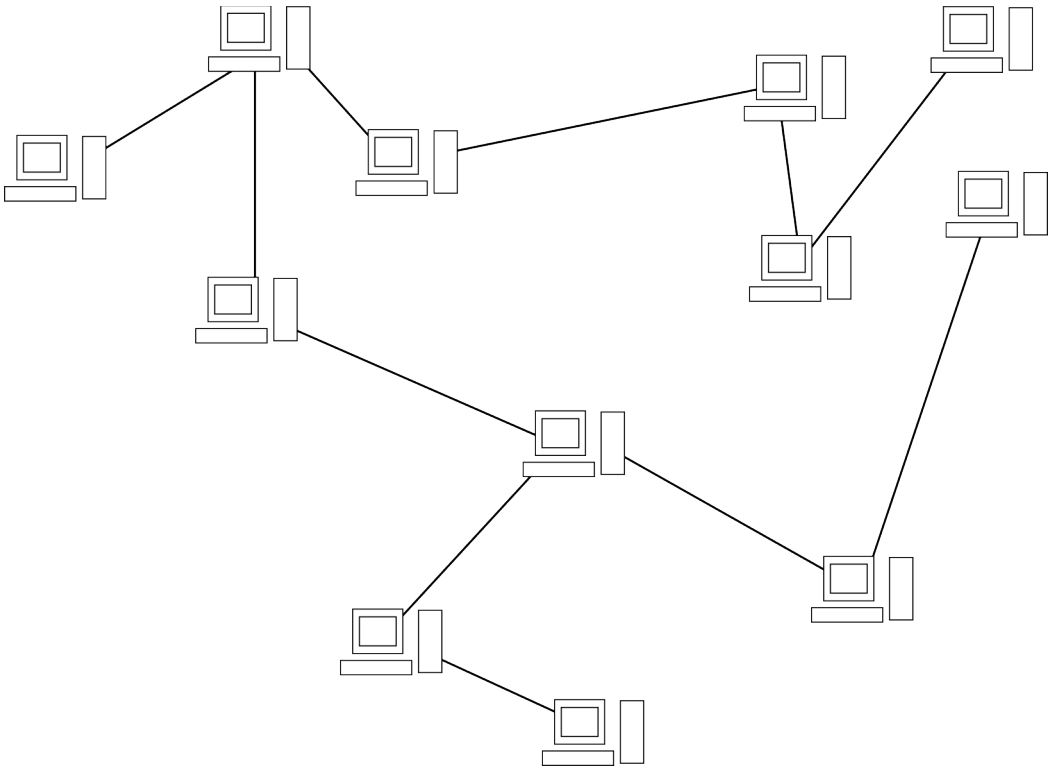

Illustration 15: Gnutella : Exemple de réseau Gnutella décentralisé

43 https://r.foilen.com/gtk-gnutella : gtk-gnutella, un logiciel de partage de fichiers
44 https://r.foilen.com/shareaza : Le véritable site officiel de Shareaza, un logiciel de partage de fichiers

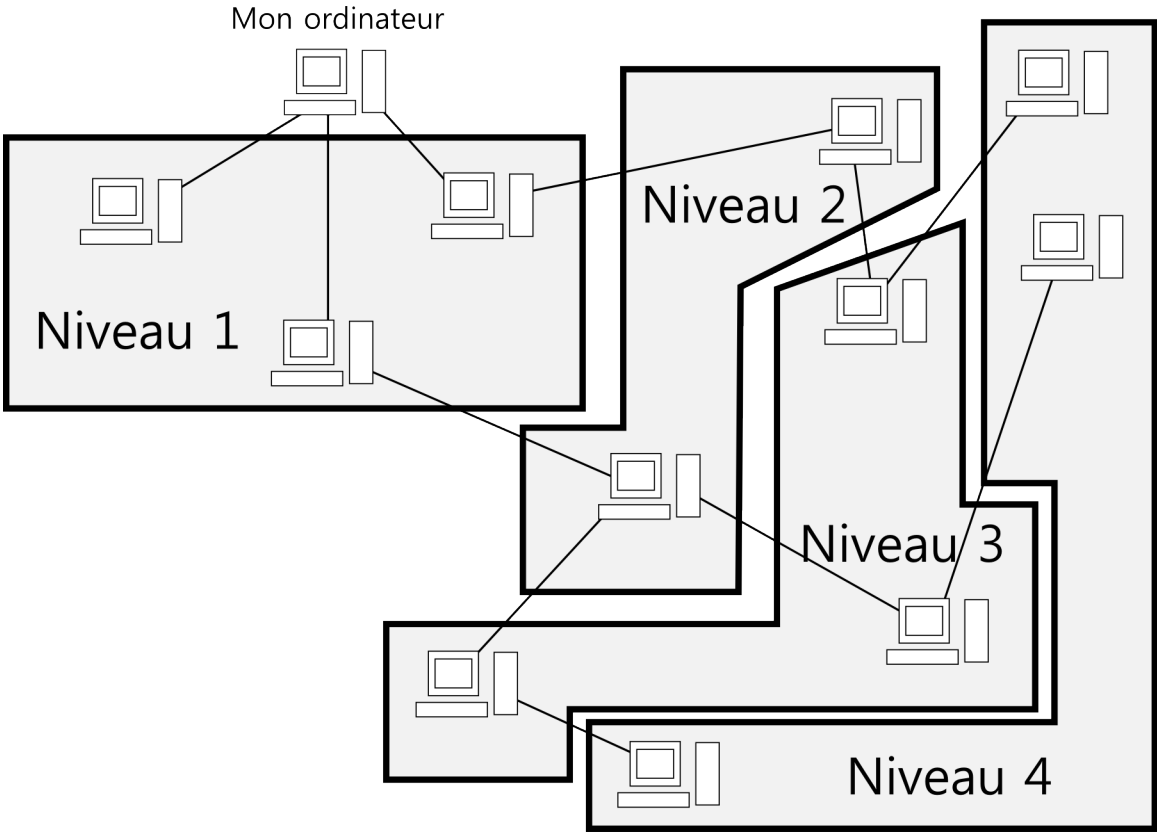

Illustration 16: Gnutella : Propagation des messages sur Gnutella

L'illustration précédente montre les niveaux d'envois de messages sur Gnutella. Par exemple, lorsque je désire rechercher des fichiers, mon ordinateur en haut à gauche envoi un message aux trois ordinateurs du niveau 1 (directement connectés à moi), ceux-ci envoient un message à ceux qui sont connectés à eux et ainsi de suite. De cette façon, avec seulement trois messages et quatre niveaux de profondeur, j'ai déjà joint onze ordinateurs. En vrai, il y a plus de quatre niveaux et il y a plus de deux ou trois ordinateurs de connectés entre eux.

Il faut faire attention avec les réseaux de partages de fichiers de bien choisir les fichiers à partager. Il y a plusieurs personnes qui partagent leur disque dur en entier et en faisant une recherche pour des Curriculums Vitae, il est possible de trouver ces documents confidentiels. Ne faites pas la même erreur.

9.12 BitTorrent

Ports TCP et UDP choisis au hasard

BitTorrent est un protocole qui permet d'échanger des fichiers, mais pas d'en rechercher. Le but est de distribuer rapide des données en permettant à tous ceux qui téléchargent un fichier, de le télécharger des autres. Cela permet de ne pas surcharger des ressources web. Un exemple d'utilisation est dans la distribution de Linux: étant donné que l'installateur prend souvent quelques Go, auparavant, il y avait plusieurs serveurs, dits miroirs, qui permettaient à tous de le télécharger. Grâce à BitTorrent, ces miroirs sont moins utiles puisque plus la quantité de personnes qui téléchargent augmente, plus de personnes peuvent télécharger. D'une certaine façon, tout le monde qui désire télécharger un fichier devient un miroir qui distribue ce fichier par la suite. Voici quelques illustrations montrant la différence entre un transfert normal et un transfert par BitTorrent.

Pour cette démonstration, supposons que tous les ordinateurs (miroirs et clients) peuvent envoyer des données à une vitesse de 1 Mo par seconde et que la réception de données ne ralentit pas l'envoi. Cette hypothèse est réaliste puisque normalement l'envoi et la réception sont indépendants et la réception est beaucoup plus rapide.

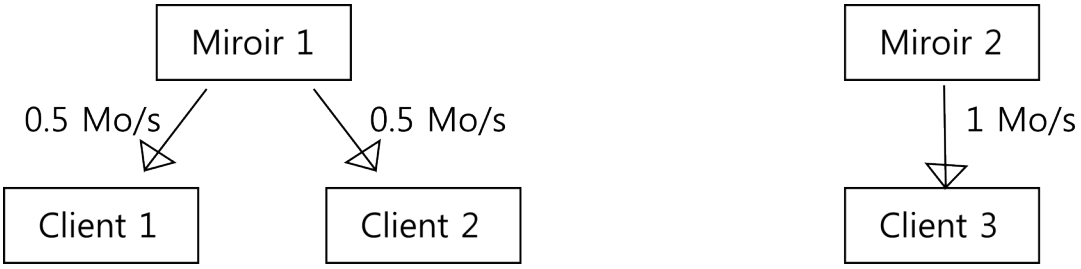

Illustration 17: BitTorrent : Transfert de fichier normal avec l'utilisation de miroirs

Dans ce premier cas, chaque miroir doit diviser sa vitesse d'envoi par le nombre de clients. Ainsi, pour servir les deux premiers clients, la bande passante est divisée par deux et est de 0.5 Mo/s tandis que le dernier client reçoit 1 Mo/s d'un second miroir puisqu'il est le seul dessus.

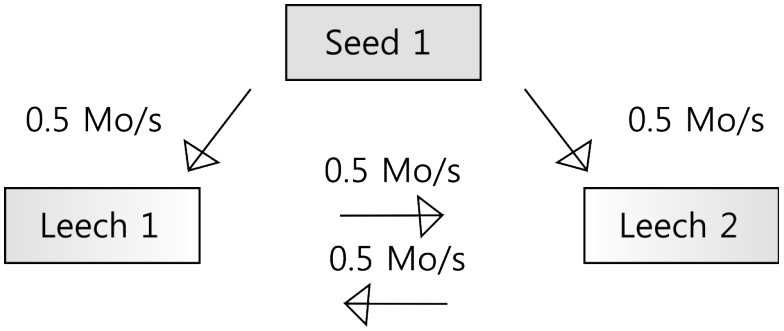

Illustration 18: BitTorrent : Transfert de fichier avec BitTorrent (1 seeder et 2 leechers)

Dans ce second cas, il n'est pas nécessaire d'avoir un second miroir pour avoir une vitesse de réception de 1 Mo/s par tous les clients plutôt que seulement le troisième. Pour ce faire, le serveur (appelé Seeder puisqu'il possède le fichier en entier) envoie la moitié des données au client 1 (appelé Leecher puisqu'il télécharge le fichier) et il envoie l'autre moitié des données au client 2. Les données ne doivent pas être les mêmes, comme cela, il est possible au client 1 d'envoyer les données reçues au client 2 et vice-versa. Ici, le client 1 reçoit le début du fichier et le client 2 reçoit la fin du fichier. Ainsi, au fur et à mesure que le client 1 reçoit des données, il peut aussi les envoyer au client 2 et vice-versa. Si nous calculons les flèches entrantes dans le client 1 ou dans le client 2, nous pouvons voir qu'il y a 0.5 parvenant du serveur et 0.5 parvenant de l'autre client. La somme donne 1 Mo/s. Ensuite, si nous calculons les flèches sortantes, le serveur envoi un total de 1 Mo/s aux deux clients tandis que chacun des clients n'envoie que 0.5 Mo/s à l'autre client. Ces derniers ne peuvent pas envoyer plus rapidement étant donné que le serveur n'envoie pas plus vite, mais cela n'empêche pas à tous les clients de recevoir à 1 Mo/s qui est la vitesse maximale du serveur. C'est donc un très grand gain. Voyons maintenant ce qui se passe si nous ajoutons un troisième client qui ne touchera pas au serveur puisqu'il est déjà pas mal occupé, mais plutôt aux deux clients.

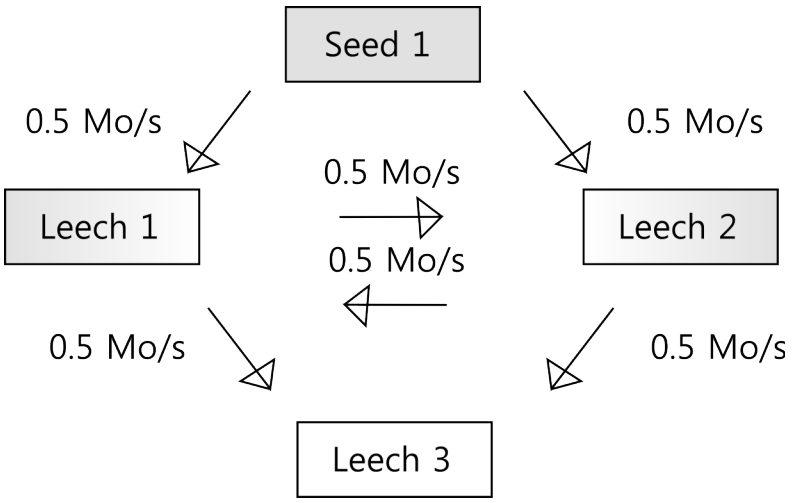

Illustration 19: BitTorrent : Transfert de fichier avec BitTorrent (1 seeder et 3 leechers)

Étant donné que les deux clients ne transféraient que la moitié de leur vitesse maximale, ils peuvent utiliser l'autre moitié pour envoyer des données à un troisième client. Le serveur envoie le début du fichier au client 1 qui l'envoie aux clients 2 et 3. De la même façon, le serveur envoie la fin du fichier au client 2 qui l'envoie aux clients 1 et 3. En calculant les flèches entrantes sur tous les clients, il y a toujours une vitesse de 1 Mo/s qui est la vitesse maximale de transfert du serveur. À titre de comparaison, avec la méthode normale, un seul serveur qui enverrait à 1 Mo/s à trois clients, ces derniers ne recevraient le fichier qu'à 0.3 Mo/s, ce qui prendrait trois fois plus de temps. Nous pouvons donc en déduire que la vitesse de réception du fichier ne dépend plus du nombre de clients sur un serveur, mais simplement de la vitesse d'envoi de ce serveur.

Pour continuer, lorsqu'un client (leecher) a reçu le fichier en entier, il n'est pas obligé de se déconnecter. À la place, il peut devenir un seeder et ceci est fait automatiquement. Dans ce cas-ci, nous obtenons un partage comme suit.

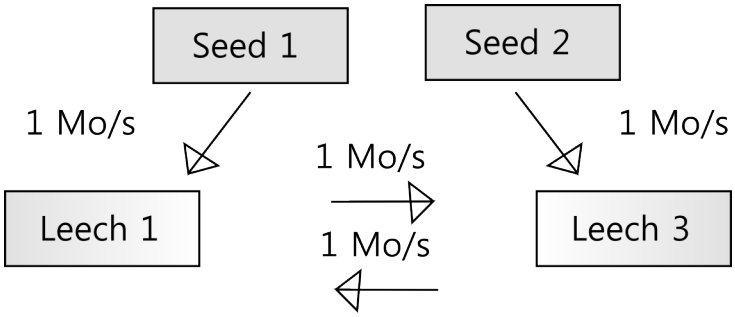

Illustration 20: BitTorrent : Transfert de fichier avec BitTorrent (2 seeder et 2 leechers)

Il y a deux seeders et deux leechers. La somme des flèches entrantes pour les clients 1 et 3 est maintenant passée à 2 Mo/s. Nous pouvons généraliser que la vitesse maximale théorique de téléchargement par chaque client est de la somme de la vitesse d'envoi des seeders. Côté pratique, c'est normalement lent à démarré, mais la vitesse de téléchargement s'améliore beaucoup après quelques minutes. De plus, la vitesse d'envoi est souvent plus lente que la vitesse de réception, alors le système peut être ralenti par cela, mais puisque les données d'un fichier sont éparpillées sur beaucoup

d'ordinateurs différents, il est possible d'additionner la vitesse d'envoi d'une dizaine de machines juste pour nous et il y a souvent des milliers de seeders pour compenser.

Pour permettre ce genre de service, il faut créer un fichier Torrent dans lequel il y a toute l'information essentielle pour créer les fichiers, les valider et les télécharger. À l'intérieur, il y a le nom de chaque fichier avec sa taille et un hash[45] de son contenu pour le valider. Il y a aussi la liste des Trackers. Ces derniers sont des serveurs qui savent qui partagent le fichier en entier (les "Seeders") et qui le téléchargent (les "Leechers") présentement. C'est grâce à eux que les clients savent où ils peuvent se connecter pour obtenir les fichiers. De plus, parfois il y a des "Web seeds" dans le fichier Torrent. Cela indique où le fichier en entier est disponible sur un site web plutôt que par BitTorrent. Ainsi, s'il n'y a pas de personnes avec tous les fichiers, il est possible d'aller les chercher à cet endroit et de les partager par la suite.

Bien entendu, utiliser un Tracker crée un point singulier de défaillance. C'est pourquoi il y a un système qui a été ajouté par la suite qui permet à tous d'être un Tracker sans savoir pour quel Torrent en particulier. Ce système s'appelle DHT pour Distributed Hashing Table. Le but ici est que chaque client BitTorrent qui supporte le DHT va Tracker quelques Torrents et lorsqu'il se déconnecte, une autre personne va prendre automatiquement sa place (avec un certain délai).

Malgré toutes ces technologies mises en place, il n'y a pas une façon de trouver tous les fichiers disponibles par ce protocole. Plusieurs sites indexent des fichiers Torrents comme Mininova[46] et il y en a un autre qui indexait le DHT, mais ce dernier est fermé depuis décembre 2014.

Pour voir en vidéos comment cela fonctionne[47].

Pour un logiciel très connu comme client, vous pouvez utiliser uTorrent[48] et voir comment il fonctionne en vidéos[49].

9.13 Telnet et SSH

Telnet: Port TCP 23 ; SSH: Port TCP 22 (crypté)

Ces deux protocoles servent à se connecter en mode console à un ordinateur distant. Le mode console est simplement une interface texte par laquelle l'utilisateur envoie des commandes écrites et reçoit des réponses aussi écrites. Avec cette console, il est possible d'administrer un ordinateur et d'exécuter tous les logiciels installés.

Telnet est une version non sécuritaire de cet utilitaire. Étant donné que dans le passé les réseaux étaient surtout internes et non publics comme l'Internet, l'envoi du nom d'utilisateur et du mot de passe en clair sur le réseau n'était pas vraiment un problème. Maintenant, pour administrer un ordinateur via l'Internet, mieux vaut crypter la connexion avec le protocole SSH. De plus, il n'y a rien qui nous dit que l'ordinateur distant est bien celui sur lequel nous désirons vraiment nous connecter. C'est pourquoi le protocole SSH utilise des certificats pour confirmer l'identité du serveur sur lequel nous nous connectons, tout comme le protocole HTTPS le fait pour les sites web.

En plus du mode console, SSH permet aussi d'envoyer des documents au serveur en les cryptant. C'est le mode SFTP qui est un dérivé sécurisé du protocole FTP (File Transfert Protocol) et accessible avec

45 Voir chapitre 14 à la page 76
46 https://r.foilen.com/mininova Site pour chercher des fichiers Torrents
47 https://r.foilen.com/f-bittorrent : Vidéos sur le fonctionnement de BitTorrent
48 https://r.foilen.com/utorrent : uTorrent, un locigiel pour BitTorrent
49 https://r.foilen.com/f-utorrent : Vidéos sur le logiciel uTorrent

le logiciel FileZilla. Aussi, SSH a d'autres fonctionnalités comme la redirection de ports. Cela permet à un utilisateur de rediriger un service Internet par son serveur SSH de manière sécurisée. Concrètement, si vous êtes dans un café Internet et vous désirez vous connecter sur un site web qui n'est pas sécurisé par du HTTPS et que vous ne faites pas confiance qu'il n'y a personne qui écoute ce qui se passe sur le réseau au café Internet, vous pouvez rediriger ce site web par votre ordinateur chez vous avec une connexion cryptée via SSH. En plus de ces nouvelles fonctionnalités, le protocole SSH en permet d'autres alors c'est un outil essentiel à apprendre.

9.14 OneSwarm

Ports TCP et UDP choisis au hasard

OneSwarm se veut un logiciel de partage de fichiers qui est anonyme entre amis. Pour se faire, les recherches de fichiers passent par les amis, tout comme pour le réseau Gnutella, mais le message lui-même ne permet pas de savoir qui a fait la demande initiale et qui a le fichier de disponible. Ensuite, pour le téléchargement, le chemin par où la recherche est passée est utilisé pour transmettre les données. Ainsi, il n'est pas possible pour celui qui télécharge de connaitre la source et pour celui qui envoie de connaitre la destination. De plus, le tout est crypté pour que les amis intermédiaires ne sachent pas se qu'ils redirigent. Malgré tout cela, ce réseau n'est pas à l'abri d'attaques sophistiquées à l'anonymat et un chapitre complet porte ce sur ce sujet[50].

En plus de fournir ce système anonyme, ce logiciel permet aussi de télécharger des Torrents. Par contre, ce n'est pas fait de manière anonyme, mais cela permet d'ajouter des fichiers à la liste locale qui peuvent ensuite être partagés sur le réseau d'amis.

50 Voir chapitre 13 à la page 72

Les problèmes de sécurités et leurs solutions

10 L'insécurité d'Internet pour les particuliers

10.1 Dû à la topologie

Le plus grand problème d'Internet est dû à la manière dont le réseau est implémenté à cause de son histoire. Au tout début, le réseau était local à une université, un laboratoire de recherche ou une entreprise. Il n'y avait par conséquent aucun problème à ce qu'il y ait d'autres personnes qui puissent observer ce qui se passe sur le réseau ou qui tenteraient d'endommager les systèmes puisqu'elles faisaient toutes parties de la même équipe. De plus, les ordinateurs du temps étant lents, alors il ne fallait pas les forcer à faire du cryptage partout, sinon la transmission des données aurait été très laborieuse. C'est ce qui a ouvert plusieurs portes à des failles qui sont exploitables sur le réseau public d'Internet et ce qui a amené des outils pour contrer les attaques qui en résultent. Toutes les techniques qui seront discutées sont possibles dues à l'envoi de messages en clair et sont contrées simplement en utilisant un protocole de communication crypté comme HTTPS, SFTP, SSH, etc. Les techniques sont: le Sniffing et l'attaque de l'homme au milieu (man in the middle).

La première technique, le **Sniffing**, est passive. Cela veut dire qu'il suffit de lancer un programme spécial et il va simplement écouter et afficher tout ce qui passe sur la carte réseau. C'est utile uniquement quand le trafic qui nous intéresse transige par notre carte réseau comme dans le cas de l'utilisation d'un hub. Pour mémoire, un hub est un équipement informatique qui permet de connecter plusieurs ordinateurs en réseau local et qui va transmettre tous les messages envoyés sur un de ses ports à tous les autres ports. C'est pourquoi nous pouvons capter toutes les communications.

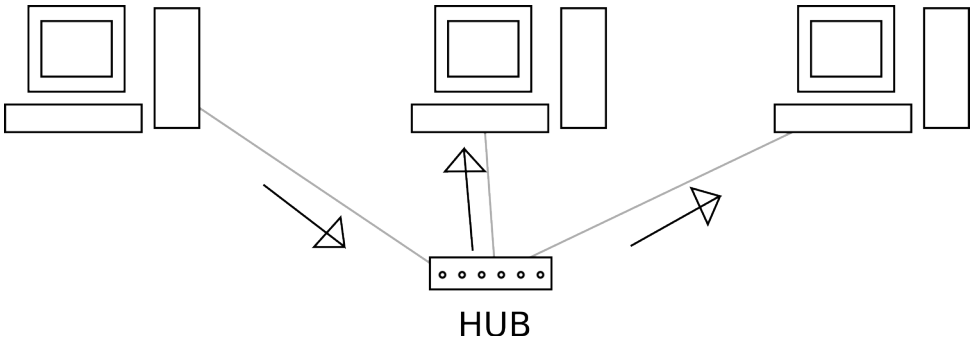

Illustration 21: Sniffing : Un hub envoyant les données reçues sur son premier port à tous les autres ports

La seconde technique, **l'attaque de l'homme au milieu**, va aussi utiliser le Sniffing pour lire les données, mais il y a un peu de préparation à effectuer pour obtenir le trafic désiré. Le plus simple, c'est de transformer la carte réseau sans fil en une borne d'accès. C'est normalement fait en étant connecté par fil à un routeur qui a accès à l'Internet et en spécifiant de faire un pont entre la carte sans fil et la carte avec fil pour que les gens aient réellement accès à l'Internet en se connectant à votre borne. Il suffit ensuite d'écouter ce qui se passe sur la carte réseau. Pour une façon un peu plus complexe, si vous êtes connectés sur un réseau et que vous voulez rester sur ce réseau, mais faire transiger les messages d'un ordinateur par le vôtre et que vous le redirigez par la suite sur le routeur local, vous aurez besoin de plus d'outils. Les techniques sont nombreuses pour réussir ce coup, mais elles dépendent du réseau et de la machine à détourner. Le but de tous ces outils est de faire croire à la cible que vous êtes le

routeur et c'est pourquoi les messages passeront par vous.

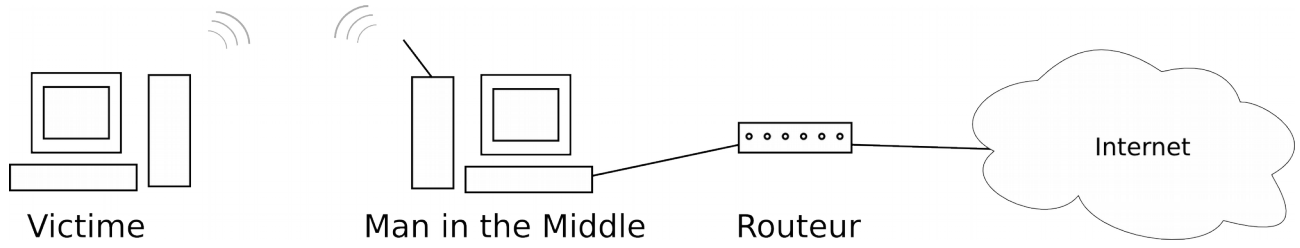

Illustration 22: Insécurité d'Internet : Attaque de l'homme au milieu à l'aide d'une borne WiFi

Lorsque l'attaque de l'homme au milieu est exécutée, il est possible d'écouter ce qui se passe comme précédemment expliqué pour le sniffing, mais il est aussi possible de modifier le trafic. Prenons l'exemple où vous êtes la personne piégée, si vous visitez un site de nouvelles connu, l'attaquant peut le voir et étant donné que le contenu du site web passe par lui avant d'arriver à votre ordinateur, il pourrait ajouter des nouvelles ou modifier des images. Aussi, si vous téléchargez un installateur d'un logiciel, il pourrait y insérer un virus en même temps de le relayer. Plus grave encore, si vous allez sur le site de votre banque qui est normalement sécurisé par le protocole HTTPS, le malfaiteur pourrait vous dire que le site web est sur son ordinateur en modifiant la réponse du serveur DNS. Ainsi, vous vous connecterez sur son ordinateur en voyant dans la barre d'adresse la bonne URL de la banque. Ensuite, puisque le site est crypté par HTTPS, son ordinateur se connectera sur le vrai serveur de la banque et puisque c'est lui qui se connecte, il peut décrypter les données et ensuite les crypter pour vous. Ainsi, il pourra voir toutes les données qui s'y échangent.

Par contre, le protocole HTTPS ne fait pas que donner une connexion cryptée, il permet aussi de certifier la destination et ça, l'attaquant ne peut rien y faire. La non-sécurité du site est expressément affichée par les navigateurs (tel que vu dans l'illustration ci dessous). L'attaquant ne peut qu'espérer que vous ne remarqueriez pas que le site n'est pas certifié. Vous pouvez en apprendre plus sur les certificats d'authenticité dans un chapitre à venir[51]. L'image qui suit montre ce que vous devez voir (à gauche) en tout temps sur votre navigateur pour être protégé contre les attaques de l'homme au milieu (notez l'état du cadenas) et la seconde image montre le fonctionnement précédemment expliqué pour l'attaque plus complexe de se faire passer pour un serveur sécurisé.

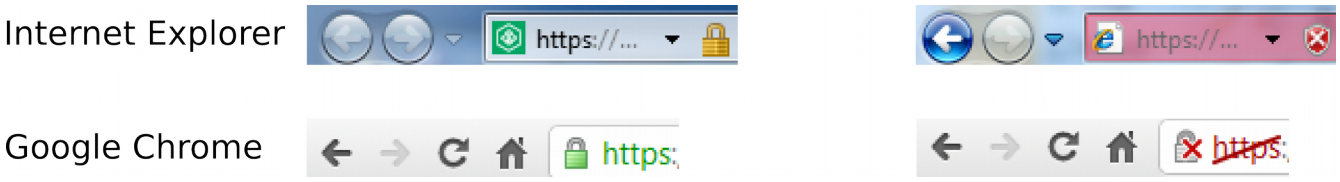

Illustration 23: Insécurité d'Internet : Les navigateurs en mode sécurisé et non sécurisé

51 Voir chapitre 12.2 à la page 66

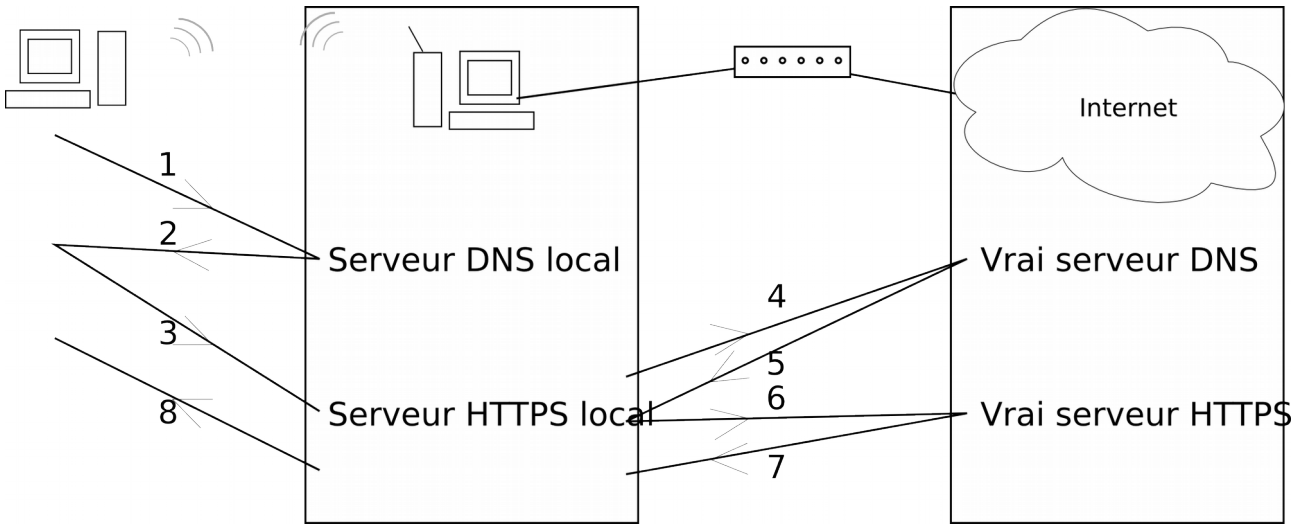

Illustration 24: Insécurité d'Internet : Attaque de l'homme au milieu qui voit le HTTPS

1. Le navigateur de la victime demande l'IP du site "https://mabanque.com"au serveur DNS
2. Le serveur DNS de l'attaquant répond que l'IP est son ordinateur
3. Le navigateur de la victime demande d'afficher la page "https://mabanque.com" au serveur HTTPS sur l'ordinateur de l'attaquant
4. Le programme de l'attaquant demande l'IP du site "https://mabanque.com"au vrai serveur DNS
5. Le vrai serveur DNS répond l'IP de la banque
6. Le programme de l'attaquant demande la page "https://mabanque.com" au vrai serveur HTTPS de la banque
7. La page est envoyée à l'attaquant et il peut la voir puisque c'est lui qui s'est connecté sur le serveur de la banque
8. Le serveur HTTPS de l'attaquant envoie la page au navigateur de la victime

Par la suite, toutes les actions de la victime passent par l'attaquant et il peut tout voir. Par contre, le navigateur de la victime affiche que ce n'est pas la page légitime de la banque en montrant un cadenas ouvert et une barre d'adresse rouge.

10.2 Dû aux cookies

Les cookies ont été rapidement mentionnés dans la section sur le protocole web HTTP et HTTPS. Je vais donc commencer par expliquer plus longuement ce qu'est cette technologie, la faille qui peut être exploitée et comment s'en protéger.

Lorsque vous accédez à un site web, à la base, il n'est pas possible pour le serveur de vous distinguer d'un autre utilisateur. Vous ne pouvez pas sauvegarder vos préférences, votre panier d'achats, etc. C'est pour cela que les cookies existent. Ces derniers sont des valeurs qui sont enregistrées dans votre navigateur et qui sont envoyées au serveur web à chaque requête. Ainsi, il est possible d'enregistrer des paramètres tels: *langue = français*, *couleur_thème = bleu*, *articles_panier = chaussettes*, etc. Par contre, en temps normal, la majorité des informations vous concernant sont enregistrées dans la base de données du service que vous utilisez. C'est pourquoi qu'en pratique, lorsque vous entrez votre nom

d'utilisateur et votre mot de passe, un seul cookie sera créé dans votre navigateur et il ressemblera à *session = 847f664873*. En envoyant cet identifiant de session à chacune de vos requêtes, le serveur sait qui vous êtes et tout ce qui vous appartient. De plus, les cookies ne sont envoyés qu'au serveur qui les a émis. Il n'est donc pas possible pour Google de connaitre vos cookies Facebook et vice-versa.

Le problème ici, c'est que la valeur est envoyée à chaque requête au serveur et que si la connexion n'est pas cryptée par HTTPS et que votre réseau n'est pas sécuritaire (tel que vu dans la section précédente), alors cette valeur de session peut être copiée sur le navigateur de l'attaquant et il aura accès au site en tant que vous sans avoir à connaitre votre nom d'utilisateur et mot de passe. C'est autant plus déconcertant dans le cas de sites très connus qui laissent les utilisateurs s'authentifier via HTTPS, mais qu'ensuite sont renvoyés sur HTTP avec un simple cookie de session. Si vous désirez voir à quel point il peut être simple de voler ces cookies, allez voir l'extension Firefox du nom de Firesheep[52] qui permet de capturer tous les cookies vus sur le réseau.

La seule méthode pour se protéger ne dépend malheureusement pas seulement des utilisateurs, mais plutôt des serveurs web puisqu'ils doivent activer le protocole HTTPS pas seulement lors de la connexion, mais durant toute la durée de l'utilisation. D'ailleurs, récemment, plusieurs gros acteurs tels Facebook et Twitter ont modifié leur service pour être utilisés entièrement sur HTTPS. C'est donc un bon pas en avant pour contrer ce type d'attaque.

10.3 Dû aux virus

Que nous les appelons virus, trojans ou vers, ces logiciels malveillants font du dommage aux ordinateurs et il faut s'en protéger le mieux possible. Je vais d'abord donner les différentes définitions, lister les problèmes qu'ils peuvent créer, lister les endroits où ils peuvent se trouver et comment s'en prémunir.

Un **virus** est un logiciel qui va s'attacher à un autre logiciel exécutable. Il se reproduit donc en se copiant sur un disque amovible, un CD/DVD, un envoi par courriel, etc. Un **trojan** est un logiciel qui va s'installer une fois exécuté, mais qui est caché dans un autre logiciel légitime comme un petit jeu ou un générateur de clés pour des logiciels piratés. Il va donc s'installer sans que vous ne le sachiez en même temps que vous utilisez un autre logiciel. Un **ver** est un logiciel qui se propage tout seul. Il va souvent viser des services Internet comme des sites web et va exploiter une faille connue pour entrer dans le système et l'infecter. Une fois infecté, ce nouveau système va chercher d'autres systèmes vulnérables et les infecter.

Peu importe les trois types de propagations précédemment décrits, ces logiciels malveillants peuvent faire plusieurs choses une fois installés. Ils peuvent effacer des fichiers importants et rendre l'ordinateur inutilisable; permettre un contrôle de la machine à distance pour faire des attaques de masse ou envoyer des pourriels; enregistrer les mots de passe entrés; afficher de faux messages pour vendre des services inutiles; n'importe quoi d'autre qui passe par la tête du créateur.

Ces logiciels peuvent se trouver un peu n'importe où, mais ils sont en particulier sur les réseaux de partage de fichiers, sur IRC et sur des sites qui permettent de télécharger des fichiers illégaux. Il faut donc toujours être méfiant surtout lorsque nous visitons des sites qui sont à la limite de la légalité.

Pour s'en prémunir, il faut absolument avoir un antivirus et être alerte. Pour l'antivirus, vous pouvez en

52 https://r.foilen.com/firesheep : Extension Firefox pour voir toutes les sessions ouvertes et non cryptés sur un réseau local

télécharger un qui est gratuit comme Microsoft Security Essentials[53] ou Avast[54]. Pour être alerte, il faut toujours au minimum vérifier que le fichier que vous ouvrez est bien du type qui vous intéresse. Par exemple, si l'extension du fichier est “.exe”, c'est un exécutable, alors si vous vous attendez à de la musique, c'est clairement un virus puisque vous devriez plutôt avoir une extension comme “.mp3”, “.ogg” ou “.flac”. Pour en savoir plus sur les extensions, allez voir une prochaine section[55].

10.4 Dû aux injections de scripts

Plusieurs sites web permettent aux utilisateurs de laisser des commentaires ou des messages qui sont visibles par tous. Par exemple, un forum, un guestbook ou un commentaire sur un billet de blogue sont des endroits où n'importe qui peut écrire. Si le site est bien fait, il ne permet pas d'inclure des balises HTML ou JavaScript dans le message, mais s'il est mal fait, il pourrait le permettre. Dans ce cas, il est possible de faire afficher des images d'un autre site, d'appeler des fonctions d'un autre site ou de faire planter le navigateur web du visiteur.

Prenons un forum de discussion qui n'est pas sécurisé. Dans un message qui est beaucoup consulté, un attaquant pourrait écrire une réponse qui inclut du code JavaScript qui ferait écrire un nouveau message à tout le monde qui le lit. Ainsi, s'il y a 1000 personnes qui lisent la réponse, il y aura 1000 nouveaux messages sur le forum, ce qui le rendrait illisible. Au lieu d'ajouter un message, l'attaquant pourrait plutôt appeler la fonction d'effacement d'un message et s'il y avait un modérateur qui lisait cette réponse, il effacerait automatiquement le message visé par l'attaque.

Ce dernier exemple était pour un forum non sécurisé et les fonctionnalités demandées étaient pour ce même forum. Par contre, il est aussi possible de faire appeler des fonctionnalités d'un autre site web. C'est ce qui s'appelle du "cross scripting". Cela permet de faire créer un message sur Facebook si l'utilisateur est connecté à son compte simplement en lisant une réponse d'un forum mal codé. Par contre, les navigateurs ont quelques protections contre ces attaques, mais de nouvelles failles peuvent toujours être trouvées.

La seule façon de s'en prémunir, c'est en ne visitant que des sites web qui sont bien faits, ce qui n'est pas possible de savoir à l'avance en tant qu'utilisateur.

10.5 Dû aux pourriels et hameçonnages

Le **pourriel**, c'est recevoir des courriels publicitaires non sollicités. Vous pourriez recevoir des messages sur des produits pharmaceutiques, des logiciels à rabais (souvent piratés) et autres d'entreprises que vous ne connaissez pas.

L'**hameçonnage**, c'est d'essayer de vous soutirer des informations confidentielles en se faisant passer pour une entreprise que vous connaissez ou en vous suppliant d'aider un mourant à transférer une somme substantielle d'argent à une fondation en vous offrant une partie de ce montant. La première situation s'apparente souvent aux pourriels: vous recevez un courriel de votre banque vous disant que votre compte va être fermé à moins que vous ne vous connectiez en utilisant le lien inclus dans le message. En cliquant dessus, vous atterrissez sur un site qui ressemble au panneau de connexion de votre banque, mais si vous regardez l’URL du site, vous verrez que ce n'est pas tout à fait le même. En entrant votre numéro de compte et votre mot de passe, ceux-ci seront envoyés à l'attaquant. Dans le

53 https://r.foilen.com/mse : Anti-virus gratuit de Microsoft
54 https://r.foilen.com/avast : Anti-virus gratuit
55 Voir chapitre 20 à la page 94

second cas, vous avez une personne qui vous contacte en disant qu'elle sollicite votre bonté en l'aidant à transférer 1 million d'euros du Niger à un organisme charitable en France. Elle vous promet 10% de cette somme pour votre aide. Si vous y répondez, elle va vous demander d'envoyer au Niger mille euros pour le notaire, ensuite mille euros pour la paperasse du gouvernement, etc. En gros, tant que vous payez, vous recevrez de nouvelles raisons d'envoyer plus d'argent.

Pour éviter de recevoir ce genre de courriels, il ne faut pas que votre adresse soit connue de ces malfaiteurs. Évitez donc d'afficher votre courriel sur des sites web puisque des programmes ballaient le web pour trouver toutes les adresses possibles. Ensuite, évitez de vous inscrire sur tous les sites web puisque certains pourraient transmettre votre courriel à des malfaiteurs. Si vous désirez vraiment vous inscrire à un site qui vous y force, mais dont vous n'avez pas confiance, utilisez une adresse jetable gratuite de Mailinator[56] ou Jetable[57].

56 https://r.foilen.com/mailinator : Utiliser une adresse courriel visible à partir de ce site
57 https://r.foilen.com/jetable : Obtenir une adresse courriel temporaire qui redirige à la vôtre

11 Protection logicielle

11.1 Antivirus

Comme spécifié dans le chapitre précédent, les virus peuvent être détectés par un antivirus et ce logiciel peut empêcher son exécution et l'effacer. Par contre, ce n'est pas une méthode infaillible due à leur mode de fonctionnement. Je vais commencer par expliquer le fonctionnement de ces applications, les mesures importantes à tenir compte dans le choix d'un antivirus et ensuite vous donner une liste de logiciels gratuits et efficaces.

La majorité des antivirus fonctionnent de la même façon. Lorsqu'un nouveau virus émerge, les spécialistes vont l'analyser et vont tenter de trouver comment il peut être détecté. S'il ne change pas, il est très facile de simplement le comparer avec une signature qui représente le contenu du virus. Ainsi, si l'antivirus fonctionne en arrière-plan et que vous accédez à un répertoire, à une archive ou un fichier infecté, l'action sera arrêtée et il vous sera demandé de choisir quoi faire avec le virus. L'effacer est en règle générale la bonne solution, mais si vous n'êtes pas certain, vous pouvez toujours le mettre en quarantaine. Cette option déplace le virus dans un endroit où il ne sera jamais exécuté, mais d'où vous pourrez le restaurer si vous le désirez.

Comme spécifié, seulement les nouveaux virus qui se propagent beaucoup vont être remarqués par les experts et ensuite analysés pour vous protéger. Vous pouvez donc être victime d'un virus si vous êtes une des premières personnes infectées ou si le virus est prévu pour un nombre restreint de victimes. De plus, les anciens virus sont souvent enlevés de la liste des signatures lorsqu'ils ne sont plus dans la nature pour rétrécir la grosseur de la base de données de signatures. Les développeurs font ce ménage parce que la taille de celle-ci doit être assez grande pour protéger un maximum d'utilisateurs, mais être assez petite pour ne pas ralentir la détection puisqu'il faut comparer tous les fichiers utilisés à ces signatures. Il est donc possible à un ancien virus de revenir infecter quelques personnes.

Pour pouvoir choisir un bon antivirus, il y a plusieurs critères qui peuvent avoir des priorités différentes selon les gens. Entre autres, il y a le taux d'échecs, les faux positifs, la facilité d'utilisation, l'utilisation de ressources et la vitesse de balayage. Le taux d'échecs est le nombre de virus qui ne sont pas détectés et qui vont donc infecter l'ordinateur s'il est exécuté. Les faux positifs sont des fichiers normaux qui sont détectés comme des virus et que vous ne devriez pas effacer. C'est pourquoi si vous n'êtes pas certain, mieux vaut les mettre en quarantaine que les effacer. La facilité d'utilisation est très subjective, mais pour les particuliers, c'est souvent ce qu'ils vont regarder en premier puisque le restant est trop obscur pour eux. L'utilisation de ressources c'est surtout pour le programme qui roule en arrière plan: s'il est gourmand en mémoire, il ralentira votre machine et s'il est lent à vérifier les fichiers que vous ouvrez, l'ouverture de ceux-ci sera aussi ralentie. Pour terminer, la vitesse de balayage n'est pas pour l'exécution en arrière-plan, mais durant un balayage complet que vous effectuez avec le logiciel principal. Il est de plus en plus rare d'exécuter ce type de tâche puisque la protection en temps réel (arrière-plan) est très efficace.

Plusieurs logiciels sont offerts gratuitement aux particuliers. Cela ne signifie pas qu'ils ne sont pas bons, c'est simplement parce que la facture est donnée uniquement aux clients commerciaux. C'est une belle façon pour les éditeurs de se faire connaitre des particuliers pour ensuite que ceux-ci les recommandent à leurs employeurs. Parfois, il y a aussi des versions "Pro" qui offrent plus d'options comme un pare-feu.

Pour les utilisateurs de Windows, Microsoft offre gratuitement "Security Essentials[58]" depuis quelque temps. Il est peu gourmand et très discret. D'autres logiciels gratuits et populaires sont Avast[59] et AVG[60].

11.2 Anti logiciels malvaillants (malwares)

Les logiciels malveillants ne sont pas tout à fait des virus, mais sont quand même nuisibles à votre ordinateur. Ces logiciels vont souvent s'installer avec des installateurs de logiciels gratuits que vous téléchargez sur l'Internet. Par exemple, en installant un logiciel d'échange de fichiers, il pourrait vous demander si vous désirez installer une "barre d'outils" dans votre navigateur. Parfois, ce n'est pas demandé, mais c'est écrit dans la licence[61] d'utilisation que cette barre d'outils ou autre logiciel malveillant s'installe en même temps. Souvent sur ce type de barre, il y une case pour faire des recherches sur le web et en l'utilisant, cela leur fait de l'argent. Ce type de logiciels n'est pas dans les pires, mais il fait quand même ralentir l'ordinateur en plus de prendre de la place. D'autres, moins gentils, vous espionnent et envoient une liste des sites web visités à leur centre de commandes pour analyser votre profil et vous afficher de la publicité ciblée.

La plupart du temps, ces logiciels ne sont pas détectés par les antivirus puisqu'ils sont "légitimes". Ils le sont dans le sens que vous avez accepté (souvent sans vous en rendre compte) de les installé puisque c'est écrit dans la licence d'utilisation que vous n'avez pas lu ou parce que vous avez laissé cochés les cases pour l'installer en cliquant trop vite sur "suivant". Par contre, certains un peu moins "légitimes" peuvent être détectés par les antivirus cités précédemment.

Tout comme pour les antivirus, vous avez des logiciels gratuits pour les particuliers avec des options payantes si désirées qui peuvent se charger d'effacer rapidement tous ces intrus. Les plus connus sont Lavasoft Ad-Aware[62], Spybot Search&Destroy[63] et Malwarebytes Anti-malware[64].

11.3 Pare-feu (firewall)

Un pare-feu est un logiciel qui permet de restreindre l'accès à certains services sur Internet en acceptant ou en bloquant les messages envoyés sur le réseau. En d'autres mots, il va filtrer les données qui proviennent et qui vont vers l'Internet. Ces règles de filtrages peuvent être configurées à plusieurs niveaux selon les besoins des utilisateurs.

Au plus bas niveau, il est possible de filtrer les adresses MAC. C'est une pratique très répandue sur les routeurs sans-fils domestiques pour choisir quels appareils ont accès au réseau local. Par contre, étant donné que cette information est facilement disponible et modifiable, pour quiconque s'y connait, il est aisé de passer outre cette protection. C'est pourquoi il ne faut pas négliger de sécuriser un routeur avec un mot de passe.

Au premier niveau logiciel, les adresses IP et des ports peuvent être bloqués. Par exemple, si le port 80 (HTTP) entrant est toujours bloqué, cela veut dire que notre ordinateur ne peut pas servir des pages web. De l'autre côté, si c'est le port 80 sortant qui est bloqué, cela veut dire que nous ne pouvons plus accéder à aucun site web. C'est ainsi que dans certaines entreprises, des services comme ceux de

58 https://r.foilen.com/mse : Microsoft Security Essentials
59 https://r.foilen.com/avast : Avast
60 https://r.foilen.com/avg : AVG
61 Voir chapitre 26 à la page 113
62 https://r.foilen.com/adaware : Ad-Aware
63 https://r.foilen.com/spybot : Spybot Search&Destroy
64 https://r.foilen.com/malwarebytes : Malwarebytes Anti-malware

clavardage peuvent être bloqués. Aussi, puisqu'il est possible de bloquer selon les adresses IP, s'il y a une adresse qui est reconnue pour envoyer du pourriel et que nous avons un service de courriel sur notre ordinateur (port 25 entrant), il peut être bon de bloquer cette combinaison IP/port pour bloquer cet utilisateur indésirable.

Au niveau supérieur, l'accès est restreint selon les applications et c'est ce qui est normalement utilisé chez les particuliers. Avec ce genre de contrôle, il est possible de dire que Firefox peut accéder à Internet, mais que le partage de fichiers de Windows (port 139) ne peut pas être visible d'Internet. Souvent, nous pouvons ajouter des règles selon le type de réseau. Par exemple, en utilisant le pare-feu qui est fourni avec Windows, nous pouvons choisir que le partage de fichiers ne soit accessible que lorsque nous sommes sur un réseau à la maison, mais pas pour un réseau public ou au travail. Avec d'autres logiciels comme ZoneAlarm[65], chaque fois qu'un nouveau programme désire accéder à l'Internet, une boite s'ouvre nous demandant si ce programme peut y accéder ou y fournir un service. Cela permet de n'autoriser que les programmes connus. Ainsi, un logiciel malveillant, décris dans la section précédente, ne pourrait pas envoyer des données sur nous à leur centre de commande si nous ne lui donnons pas accès. C'est alors une protection supplémentaire, mais uniquement si nous faisons bien le tri de quelles applications a le droit de communiquer au réseau.

La question importante est: "avez-vous besoin d'un pare-feu?" et la réponse est "cela dépend". Si vous êtes dans votre maison sur un réseau local créé avec un routeur, vous avez déjà une bonne protection de base puisque le routeur bloque tous les services offerts par votre ordinateur sauf ceux que vous avez spécifiés comme serveur virtuel dans la configuration de votre routeur. Dans ce cas, si vous faites confiance à tous les utilisateurs de votre réseau local, le pare-feu de Windows est tout à fait suffisant. Par contre, si vous êtes connectés directement à l'Internet ou si vous désirez dicter quelles applications peuvent y accéder pour bloquer les logiciels malveillants, il vous faut absolument un pare-feu ne serait-ce que pour bloquer le partage de fichier de Windows et pour être certain qu'aucun service n'est accessible sur votre ordinateur à moins de votre approbation explicite.

Comme anecdote, j'ai déjà fait un balayage du port 139 (partage de fichiers Windows) sur l'Internet pour voir s'il y avait des personnes qui partageaient des fichiers. J'ai rapidement trouvé des personnes qui ne faisaient pas que partager, mais qui permettaient aussi de leur envoyer des fichiers n'importe où sur leur disque dur. Dans ces cas particuliers, j'ai laissé un fichier texte dans leur répertoire "Démarrage" pour que la prochaine fois qu'ils ouvriront leur ordinateur, qu'ils soient mis au courant qu'au lieu d'être un document texte, cela ait pu être un virus qui aurait été ouvert automatiquement. De plus, avoir un antivirus n'est pas suffisant dans cet exemple, puisqu'en ayant accès à tout le disque, il est possible de l'effacer avant d'infecter la machine.

65 https://r.foilen.com/zonealarm : Pare-feu ZoneAlarm

12 Protection sur Internet

12.1 HTTPS

Comme spécifiées dans la section sur le réseau Internet, toutes les données transférées entre deux ordinateurs passent par plusieurs autres ordinateurs et ces intermédiaires peuvent voir et conserver une copie de ces informations. Lorsque votre navigation se limite à regarder des sites publics comme ceux de nouvelles ou d'apprentissage, il n'y a aucun problème à ce que ce contenu soit connu de tous puisqu'il est déjà public. Par contre, si vous désirez voir votre compte bancaire en ligne ou même faire des achats avec votre carte de crédit, vous voulez que seulement le destinataire et vous puissiez connaitre ces informations et vous voulez être certains que votre destinataire est bien votre banque ou le magasin en question. Ce sont les deux points que HTTPS vient ajouter au simple protocole HTTP.

Lorsque vous vous connectez à un site web en mode sécurisé, il y a plusieurs choses qui se passent entre votre navigateur et le serveur. Lors de la connexion, votre navigateur va demander le certificat et la clé de cryptage à l'autre ordinateur. Le certificat sert à confirmer que le site est bien celui qu'il prétend être. Ainsi, une personne malveillante ne peut pas faire d'attaques d'homme au milieu sans que le navigateur ne vous avertisse. Puis, la clé de cryptage sert à crypter tous les échangent qui se feront entre vous et le service web.

12.2 Les certificats d'authenticité

12.2.1 L'utilité et l'utilisation

Un des grands problèmes d'Internet (et parfois un avantage) est que tout le monde est anonyme. Dans ce contexte, il n'est pas aisé de faire confiance à un site web pour faire du commerce ou des transactions bancaires étant donné qu'il pourrait y avoir un usurpateur entre nous et le service que nous désirons accéder.

Une solution a ce problème est d'utiliser la cryptographie pour prouver notre identité. Grâce aux clés privées et publiques, il est possible de s'annoncer au monde et prouver que nos messages sont bien de nous en les signant. Un site web va utiliser un certificat qui contient la clé publique du serveur. Comme ce site est le seul à avoir la clé privée, il est le seul à pouvoir écrire des messages qui seront par la suite lisibles avec la clé publique. Avec ce système, il est possible de confirmer qu'un serveur est bien celui à accéder, mais puisque c'est lui qui nous fournit le certificat, un usurpateur n'a qu'à créer son propre certificat et nous envoyer ce dernier. C'est ce qu'on appelle un certificat signé par soi-même puisqu'il est signé par le site lui-même. Lorsque nous accédons à ce type de site, le navigateur nous avertira que la communication est cryptée, mais que le site n'est pas authentifié. C'est suffisant si c'est un site personnel, mais pour prouver que c'est bien votre banque, un organisme tiers doit signer le certificat.

Tous les navigateurs ont une liste de certificats racines de certaines entreprises et de certains organismes en lesquels ils ont confiance que leur processus de signature est suffisant. Par exemple, les navigateurs font confiance à Verizon pour que ce dernier vérifie bien l'identité des gens derrière le site web qu'il va approuver avec une signature. Par exemple, il va vérifier que ce sont bien des employés mandatés par la banque qui demande un certificat. Puisque Verizon peut signer des certificats, son certificat dans le navigateur en est un appelé "intermédiaire".

Une fois que la banque ou le service a bien été authentifié par Verizon ou un autre organisme, il va créer un hash du certificat et le crypter avec sa clé privée. C'est cela qui est la signature puisque seule une personne avec la clé privée de Verizon pourra le faire. C'est ce certificat qui est ensuite donné à la banque et elle peut l'installer sur son site web.

Le navigateur va prendre ce certificat et tous ceux intermédiaires pour vérifier leurs signatures. Si la chaine se rend jusqu'à un certificat racine, alors tout est beau, sinon, vous verrez un avertissement dans votre navigateur.

12.2.2 Le problème

Le problème important est que n'importe quelle autorité de certification peut créer un certificat valide pour n'importe quel domaine même si un autre certificat est déjà en circulation. C'est correct tant que le processus de validation est bien effectué, mais comme l'actualité[66] nous l'a montré à plusieurs reprises, certaines personnes malveillantes piratent ces organismes et créent des certificats illégitimes. Dans ce cas, il faut attendre que l'attaque soit découverte pour révoquer ces certificats et ce délai permet aux pirates de faire leurs mauvaises actions.

Une solution avait vu le jour pour le navigateur Firefox : l'extension Convergence. Le but de cette extension était de demander à plusieurs ordinateurs connectés à l'Internet, appelés notaire, de télécharger le certificat directement sur le serveur web et de nous le transmettre. Ensuite, l'extension vérifiait que tous renvoient le même certificat que celui obtenu par le navigateur et cela prouvait que c'est bel et bien le site désiré puisqu'il n'est pas possible à un malfaiteur de faire une attaque d'homme au milieu à tous ces ordinateurs sur des réseaux différents. Un point de plus pour cette solution est qu'elle permet aux sites de signer leur propre certificat plutôt que de l'acheter auprès d'une autorité de certification puisque la chaîne n'a pas à être validée. Par contre, n’étant pas suffisamment populaire, cette extension est morte quelques années plus tard.

12.3 Navigation incognito

Lorsque vous naviguez sur le net, presque tous les sites vont ajouter des cookies pour vous identifier ou vous suivre à la trace. Vous pouvez toujours les effacer par la suite, mais c'est laborieux et vous perdrez aussi ceux que vous voulez garder, par exemple, ceux pour vous connecter à vos sites préférés sans avoir à entrer votre mot de passe. C'est pourquoi plusieurs navigateurs comme Firefox et Chrome ont un mode privé ou incognito. Ce mode vous crée une session qui n'a pas de cookies au départ et qui ne les sauvegarderont pas à la fermeture du navigateur.

En plus de ce petit confort, ils ne garderont pas non plus l'historique des sites visités. Par contre, cela ne vous rend pas anonyme puisque votre adresse IP reste connue à moins que vous n'utilisiez Tor en même temps (voir la prochaine section).

Personnellement, j'utilise beaucoup cette fonctionnalité quand je développe des sites web pour pouvoir m'authentifier avec deux comptes différents en même temps grâce à une session normale et une autre incognito. Je peux donc être dans un compte utilisateur et administrateur tout à la fois pour voir les modifications en temps réel.

Faites tout de même attention avec ce mode, car malgré que le navigateur ne gardera pas vos cookies, rien n'empêche les extensions comme Flash ou Java de les garder. Ces extensions peuvent donc vous

66 https://r.foilen.com/n-comodo : Comodo piraté

compromettre et c'est pourquoi le navigateur Google Chrome désactive toutes les extensions. Il y a d'ailleurs des extensions qui permettent de désactiver toutes ces fonctionnalités aisément.

12.4 Tor

(Vidéo d'explications de ce chapitre disponible[67])

Lorsque nous naviguons sur l'Internet, il est possible pour le serveur web auquel nous accédons (tel Google) de savoir quelle machine se connecte à lui (en l'occurrence, notre machine) puisque notre fournisseur Internet nous assigne une adresse IP qui est unique tant que nous restons connectés sur notre compte Internet. Certains fournisseurs vont changer l'adresse à la prochaine connexion, mais en connaissant la date, l'heure et l'IP d'une connexion, il est possible de savoir qui est derrière en demandant au fournisseur.

Tor[68] permet de cacher notre IP en traversant un réseau d'ordinateurs et en laissant le dernier communiquer avec le serveur web que nous désirons. Cela permet deux choses:

- Rester anonyme en cachant notre véritable adresse IP
- Communiquer avec des services web bloqués par un fournisseur Internet (par exemple, la Chine bloque certains sites, mais en passant par un réseau qui sort de la Chine, il est possible de communiquer avec ces serveurs)

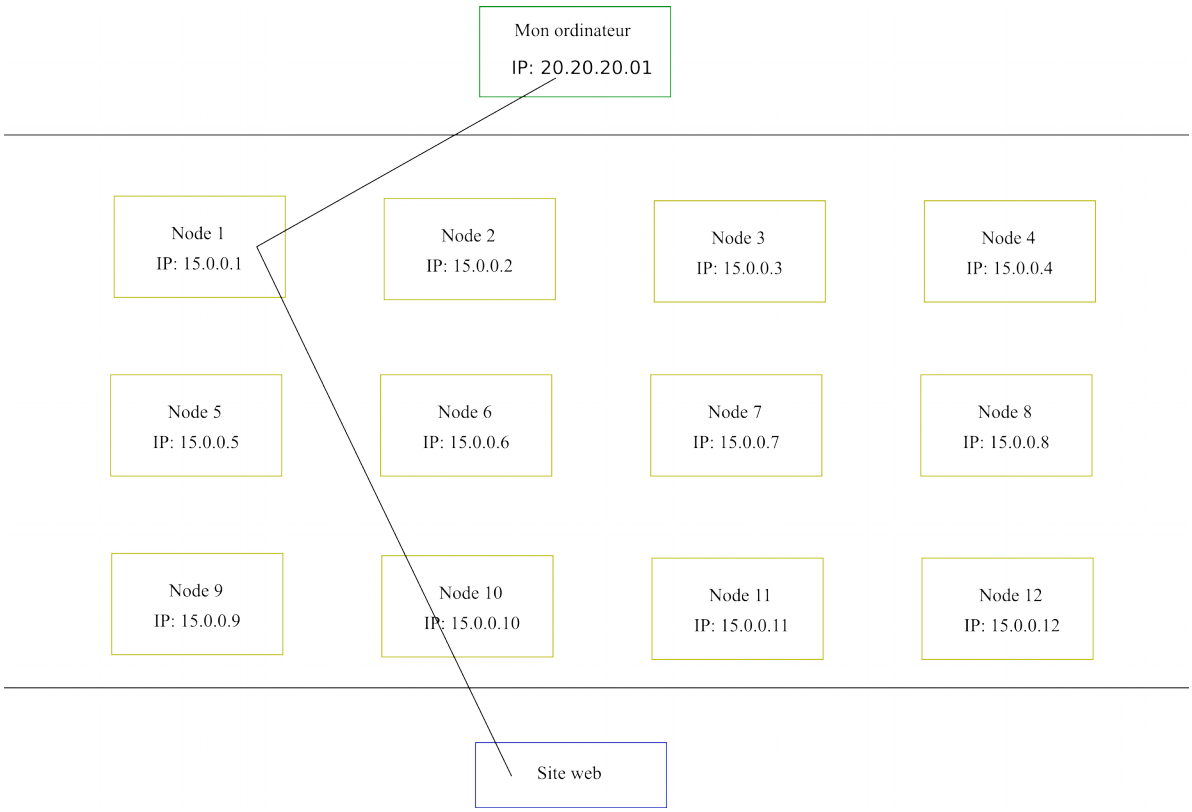

Illustration 25: Tor : Route sur Tor

Par contre, il y a aussi certaines pratiques à suivre pour éviter de laisser des traces qui peuvent nous

67 https://r.foilen.com/f-tor : Vidéo sur l'utilisation de Tor
68 https://r.foilen.com/tor : Site officiel de Tor

identifier. Il faut utiliser le mode privé (incognito) des navigateurs pour ne pas qu'un des cookies présents permette de nous identifier. Il faut aussi utiliser des sites sécurisés (HTTPS) pour toute information sensible à transmettre étant donné que sinon, la communication entre le noeud sortant de Tor et le site web ne sera pas cryptée et le noeud sortant aurait donc potentiellement accès à tout ce qui se transige. Un dernier point pour les navigateurs, c'est de désactiver toutes les extensions installées ainsi que Java et Flash. Cela semble somme tout complexe et fastidieux, mais depuis plus d'un an, une version de Firefox configuré comme il se doit est téléchargeable avec Tor tout d'un coup. Ce paquet est le téléchargement par défaut sur le site officiel du projet et ne fonctionne qu'en mode client (vous pouvez accéder aux sites au travers le réseau Tor, mais vous ne pouvez pas être un Node qui fait le relai ou une sortie. Il faut la version complète pour le faire).

La version classique permet, en plus du mode client, d'être un relai ou un noeud sortant. Un relai est un noeud comme le "Node 1" dans l'illustration. Ce mode ne fait que transmettre des messages entre les clients et autres noeuds de Tor. Tout cela est fait de manière cryptée et personne ne sait ce qui s'y échange. Si vous désirez participer au projet sans avoir de problèmes légaux, c'est la meilleure façon. Si vous êtes plus aventureux, gérer un noeud de sorti (comme "Node 10") peut être un peu plus demandant, car votre ordinateur pourra être identifié comme transférant des fichiers avec des droits d'auteurs ou comme exécutant des attaques sur des sites web. Ceci peut arriver puisque vous transféreriez toutes les données que n'importe qui utilisant Tor veut bien envoyer.

Une dernière fonctionnalité intéressante est de permettre d'avoir un site web caché et uniquement accessible sur le réseau de Tor. Pour ce faire, vous devez avoir un serveur web local tel Apache[69] ou Tomcat[70] et configurer Tor pour qu'il se connecter à ce serveur. Un nom de domaine tel "fjvudsfjkge.onion" vous sera automatiquement assigné et n'importe qui pourra accéder à cette page en utilisant Tor. Ce qui est intéressant, c'est qu'il n'y a pas de méthode pour savoir qui fournit quel service puisque tout est relayé entre plusieurs chemins internes à Tor.

12.5 Freenet

(Vidéo d'explications de ce chapitre disponible[71])

12.5.1 Introduction

Freenet[72] est un logiciel qui roule sur l’ordinateur en arrière-plan et qui possède une interface web. Celle-ci permet de naviguer sur un web parallèle où les créateurs et lecteurs de contenus sont anonymes. Cela permet d’empêcher la censure par certains pays et les représailles aux auteurs.

12.5.2 Son fonctionnement

Chaque utilisateur possède un espace de cache sur son ordinateur dont il choisit la grosseur maximale dès le départ. Celui-ci est rempli de plusieurs manières dont l’insertion, la lecture et le relai.

Lorsqu’une personne désire mettre du contenu sur le réseau, elle **insère** le contenu en l’envoyant à ses voisins qui vont par la suite propager l’information. Il est ainsi possible à l’auteur de fermer son programme et laisser les autres voir ses réalisations.

69 https://r.foilen.com/apache-httpd : Serveur web Apache
70 https://r.foilen.com/tomcat : Serveur d'applications web Java
71 https://r.foilen.com/f-freenet : Vidéos sur l'utilisation de Freenet
72 https://r.foilen.com/freenet : Site officiel de Freenet

Lorsque nous désirons **lire** une page, la demande est envoyée aux voisins qui nous enverront cette page. Elle doit nécessairement être téléchargée localement et par la même occasion est emmagasinée dans le cache local pour future lecture ou propagation.

Puis, lorsqu'un voisin demande de voir une page que nous n'avons pas, il faut la demander à nos voisins et elle transige donc entre tous les utilisateurs, du premier qui l'a à celui qui veut la voir. C'est ce segment de **relai** qui permet le plus d'anonymat étant donné que le fait de transmettre une page n'implique pas que ce transmetteur a vu cette page d'autant plus que le cache est crypté et que sans l'URL, il n'est pas possible de savoir ce qui est présent dans notre cache.

12.5.3 La beauté de la chose

Prenons par exemple le partage d'une chanson sur un réseau d'échange de fichiers ordinaire et sur le réseau Freenet.

Dans le premier cas, la demande de recherche se transige entre les utilisateurs et ce n'est pas parce qu'une recherche est demandée qu'elle est pour cet utilisateur. Par contre, lors du téléchargement, la connexion est directe entre ceux qui possèdent le fichier et ceux qui veulent le télécharger. Il est donc simple de dénoncer tous ces gens. Le fait d'être en connexion directe amène une grande vitesse de téléchargement, mais aucun anonymat.

Dans le second cas, lors d'une demande d'un fichier, ce dernier est envoyé de voisins à voisins jusqu'au destinataire. Personne ne sait qui demande et ce n'est plus seulement ceux qui utilisaient ce fichier qui le possèdent maintenant, mais tous les intermédiaires aussi. Cela amène que plus une information est demandée, plus de gens vont l'avoir en cache et elle va être plus facilement accessible.

12.5.4 Ce qu'on y retrouve

Sur ce réseau, il est possible d'y trouver des documents dans plusieurs langues, dont l'anglais et le français. Au premier abord, j'ai été très surpris de la quantité d'articles en français. Le contenu qui s'y trouve est très diversifié tel :

- Textes normaux: des pages personnelles, des blogues, etc. de contenu normal. C'est simplement pour mettre de la vie dans ce réseau.
- Textes marginaux: du piratage, des histoires de viol, comment faire des bombes, etc. Donc tout pour fantasmer!
- Miroir: certains sites sont bloqués par exemple en Chine, alors ces sites sont mis sur le réseau Freenet pour être accessibles de partout.
- Forums: les gens peuvent discuter de manière anonyme sans surnom sur n'importe quel sujet controversé.
- Contenu illégal: des chansons piratées, de la pornographie (même juvénile), des logiciels, etc. De quoi satisfaire la déviance de chacun!

12.5.5 Les outils

Il n'est pas toujours aisé de trouver l'information souhaitée et c'est pourquoi il y a plusieurs manières de chercher de l'information. Pour commencer, il y a les portails qui sont affichés sur la page

principale. Ensuite, il est possible de faire des recherches pas très poussées avec l'aide d'un plug-in déjà présent. Puis, il y a Frost qui est un programme à part qui utilise le réseau pour les forums et du téléchargement de fichiers.

12.5.6 Les failles

La lenteur due au relai de l'information est le problème majeur. Ce type de réseau n'est pas très utile pour transmettre de gros fichiers rapidement, mais au moins les fichiers sont mis à la disposition de tous. Les gros fichiers récents écrasent des pages qui risquent de disparaître si elles sont peu populaires. Par contre, la vitesse a quand même augmenté depuis 2008. Étant donné que le contenu est distribué, il fonctionne un peu comme un torrent: plus il y a de monde qui a le contenu (plus il y a de seeders), plus c'est rapide. Puisque beaucoup de gens se sont ajoutés au projet, la vitesse est beaucoup plus intéressante.

13 Les attaques contre l'anonymat

13.1 Introduction

Les logiciels comme Freenet veulent offrir de l'anonymat aux utilisateurs. Ils aident beaucoup, mais il y a quand même certains risques s'il y a des gens qui sont très déterminés à nous prendre. Toutes les attaques sont de l'ordre des statistiques ce qui signifie qu'elles ne permettent jamais de confirmer un usage illicite à 100%, et de ce fait ne sont pas nécessairement utilisables en cour, mais elles peuvent offrir un assez gros soupçon pour commencer à investiguer plus en profondeur sur une personne.

Pour obtenir ces statistiques, les attaquants doivent être connectés sur nous avec plusieurs machines différentes ou avoir accès à nos paquets transmis, et ce pendant un grand laps de temps. Si nous n'avons que des amis sur qui nous nous connectons de manière cryptée, nous sommes saufs tant qu'ils ne sont pas compromis. Si nous avons des inconnus, ils pourraient être des attaquants.

Ceux qui veulent nous espionner, une fois connectés sur nous, leur but est de regarder notre trafic entrant et sortant et d'analyser tout ce qu'ils reçoivent. Ce qui permet de distinguer ce que nous regardons de ce que nous ne faisons que transiger (relayer aux autres personnes) est les deux points suivants. Premièrement, ce que nous relayons va entrer et sortir, tandis que ce que nous consommons va entrer et en grande partie ne pas sortir puisque nous n'avons pas de raison de partager ce que nous consommons avec ceux autour de nous qui ne l'ont pas demandé. Deuxièmement, un relai ne copie pas un fichier en entier puisqu'un consommateur va recevoir des parties de fichier de plusieurs relais, tandis qu'un consommateur se doit d'avoir le fichier en entier pour le consommer.

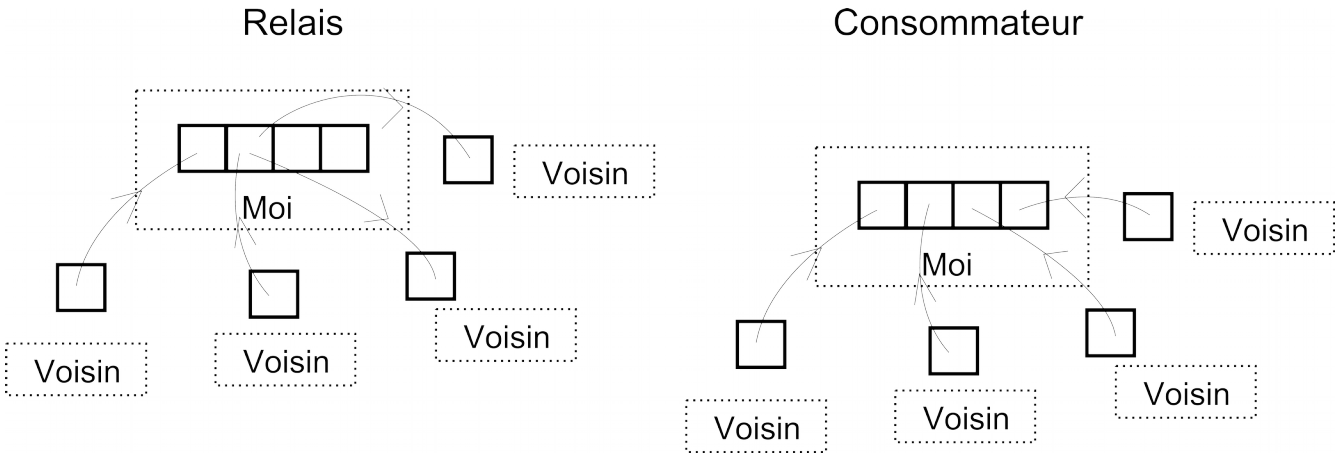

Illustration 26: Anonymat : Relayer versus consommer

C'est sur ces points que ces attaques se basent pour tout ce qui est du partage de fichiers et les recherches.

13.2 Analyse de trafic

Pour l'analyse de trafic, il y a deux facettes à regarder : les recherches et le partage de fichiers. Dans le premier cas, le but est de savoir si une personne recherche des fichiers illégaux et dans le second, si elle télécharge ces fichiers.

Pour les recherches, normalement il y a une valeur qui est le TTL (time to live) ou le temps de vie d'une requête. Elle pourrait commencer à 10, ensuite tous les ordinateurs qui reçoivent la demande décrémentent cette valeur à 9 avant de l'envoyer aux autres ordinateurs qui sont connectés sur nos voisins. Il y aura dix bonds avant que le message soit considéré comme étant rendu à la fin de sa vie s'il n'a toujours pas de résultats à retourner. Avec cet algorithme, un attaquant qui est connecté sur une victime et qui reçoit des recherches avec un TTL de 10 peut être certain qu'elles sont émises de la victime et c'est pourquoi les logiciels vont choisir un temps de départ au hasard et ne vont pas décrémenter de 1, mais d'un nombre aléatoire comme de 0.5 à 1.5. Avec ces nouveaux paramètres de hasard (le départ et la décrémentation), il devient très difficile de savoir qui est le premier à avoir émis et par conséquent, qui est l'émetteur de cette requête. Il est aussi difficile de savoir combien de bonds ont déjà été faits et donc de savoir la distance entre la cible et la requête.

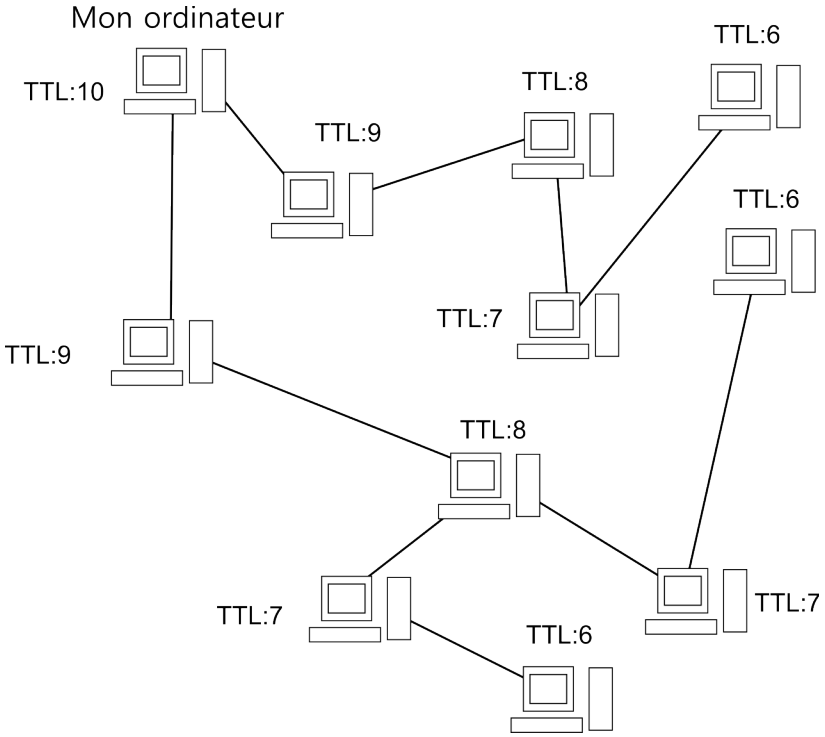

Illustration 27: Anonymat : Time To Live

Nous venons de voir la partie qui est de faire une requête, mais une autre analyse possible est sur les réponses à ces requêtes. Lorsqu'un ordinateur a la réponse, il va la renvoyer au voisin qui lui a demandé. Ce qui veut dire que l'émetteur de la requête, lors de la réception de la réponse, ne va que conserver le résultat et ne pas la renvoyer à un voisin puisque c'est lui qui était intéressé par cette requête. Ainsi, toute réponse qui entre dans l'ordinateur de la victime sans en ressortir est considérée comme lui étant destinée. Pour mélanger le jeu, le logiciel peut tout simplement garder la réponse et aussi la transmettre à un de ses voisins simplement pour ne pas avoir l'air coupable.

Pour le téléchargement de fichier, l'analyse de trafic se base sur plusieurs hypothèses. La première est que pour visionner un fichier, il faut le recevoir en entier, et ce rapidement. La seconde est qu'un relai n'envoie qu'une partie d'un fichier puisque le destinataire va recevoir plusieurs parties de plusieurs personnes différentes. Par contre, cela n'empêche pas un relai d'avoir un fichier en entier éventuellement, il faut donc qu'ils possèdent plusieurs fichiers volumineux illicites ou plusieurs fichiers

du même type pour réduire la probabilité que ce ne soit qu'une insertion ou relai qui a passé par lui.

Une façon pour prévenir ce genre d'attaque est de recevoir lentement les fichiers, dans un ordre aléatoire, avec beaucoup de données d'autres fichiers et de retransmettre ces parties pour simuler un relai. Une seconde façon est d'avoir une fonction d'insertion, ce qui veut dire qu'un logiciel va transmettre des fichiers en entier aux voisins pour créer de la redondance et pour qu'il ne soit pas possible de considérer que tous fichiers reçus en entier est une demande explicite. Cela fonctionne bien pour des petits fichiers, mais est plus problématique avec les gros.

13.3 Attaque de collusion

L'analyse de trafic peut se faire surtout par une personne qui contrôle un grand réseau comme le fournisseur d'accès à Internet tant que les messages entre voisins ne sont pas cryptés. Sinon, en étant un nombre limité de voisins à une victime, il n'est jamais possible d'être certains à 100% de quoi que ce soit puisque ce n'est pas tout le trafic qui est disponible pour l'analyse. Dans ce temps-là, la victime est connectée à d'autres utilisateurs et cela amène beaucoup de bruits sur la ligne puisque les logiciels agissent avec toutes leurs fonctions de sécurité. Le but de l'attaque de collusion est de se connecter comme voisin à la victime, pas seulement avec une connexion, mais avec le maximum pour ainsi être le seul voisin que la victime utilise. Comme l'illustration suivante le montre, l'ordinateur de la victime est entouré uniquement par des attaquants. Ainsi, il est possible de modifier le logiciel pour enlever beaucoup de bruits comme les messages qui seraient normalement relayés par la victime.

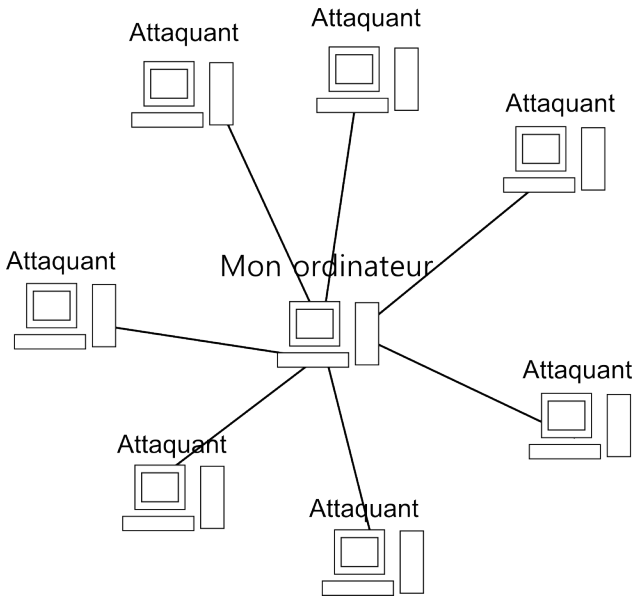

Illustration 28: Anonymat : Attaque de collusion

En contrôlant le trafic entrant et sortant dans la victime, cette dernière ne peut plus être considérée comme un relai (puisque l'attaquant ne lui fait plus de requête), mais uniquement comme un consommateur. De cette façon, toutes les demandes de recherches sont émises par la victime et tous les morceaux de fichiers demandés sont consommés par la victime. Bien entendu, les logiciels modifiés doivent au moins servir de relai pour la victime, car si elle ne reçoit aucune réponse à ses requêtes et aucun fichier, elle se déconnectera rapidement du réseau.

Pour réussir cette attaque, il faut tout de même beaucoup de préparations. Premièrement, il faut un logiciel modifié. Deuxièmement, il faut avoir plusieurs machines pour être des voisins. Dans le cas de Freenet, les utilisateurs se connectent environ à 20 voisins, donc il faut au minimum ce nombre de machines (réelles ou virtuelles). Troisièmement, il faut que la victime se connecte uniquement aux machines qui ont été préparées pour l'occasion. Il faut donc réussir à modifier la liste des hôtes disponibles reçue par la victime.

14 Hashing

14.1 Ce que c'est

Le hachage, c'est prendre un contenu quelconque comme du texte ou un fichier et de faire une fonction mathématique dessus pour obtenir une petite valeur. Pour bien comprendre, le plus simple c'est de prendre l'exemple d'un fichier téléchargé sur Internet. Ce fichier peut être petit ou gros et si nous désirons vérifier que le transfert s'est bien effectué, nous pouvons passer son contenu dans une fonction de hachage qui retournera toujours un seul caractère donc 1 octet. Nous pourrions par exemple prendre une fonction qui additionne tous les octets d'un fichier, mais qui boucle lorsque la somme dépasse 1 octet de grosseur donc 256. Ce genre de fonction donne uniquement une idée de si le fichier est intacte puisqu'il y a beaucoup de collisions possibles. Par exemple, un fichier contenant "abc" ou "bac" donnerait tous les deux la même réponse. Pour éviter ce genre de collision, les fonctions de hachage peuvent aussi tenir compte de la position dans le fichier ou du résultat précédent.

Pour valider des fichiers, il y a le MD5SUM ou le SHA1 qui sont très utilisés. Le premier étant plus rapide à calculer pour de gros fichiers, mais ayant plus de collisions.

Voici quelques exemples de Hash:

Fonction de hachage	**Résultat**
Aucune	Le petit livre du Hacker
md5sum	99e7559e76636bac20b84cf0b34daceb
sha1	9b5e99ac994adad6cf2410991537e09ee57e4520
sha256	07e95fa2d699fdc7b6892c6edb6a08241a6f73576a079e3dbc614c5430a569e7
sha512	82ef419f1d29d3f4b9782edc8815f670311e488617708ec41eae2f717cefd2c3a 1120bf4a0b9cea6b2cdfa035454627212250fbf9517266a04a98411923c10da

Dans ce cas-ci, le fichier contient la phrase: “Le petit livre du Hacker” et si nous y ajoutons un point d'exclamation à la fin, nous verrons que les résultats sont très différents :

Fonction de hachage	**Résultat**
Aucune	Le petit livre du Hacker!
md5sum	cf96ba123d81d28ebdf37efe9c9ac9da
sha1	5858694f9457b3ea39803501b5096cf81b4dc163
sha256	7d6aeaec5eca507836a7bd583071246e7389908bbb5a7a72ef88d5af87a161cd
sha512	71e8c2e81590514842f41a687251f740cf48bcb7c5a6c47630f51f7d333365dd 9b0e9328c2edf44ed9d92b15867698eda61941c0664799bd15a87bf1f0855475

14.2 Utilités

Comme dit plus haut, le hachage est utilisé pour vérifier la cohérence d'un fichier, mais ce n'est pas sa seule utilité. Il est aussi présent dans l'entreposage des mots de passe. Ainsi, puisqu’il n'est pas possible de prendre un hash et de retrouver le message original (due aux collisions et aussi au fait qu'on ne

connait pas la grosseur initiale du message) il est plus sécuritaire de toujours hacher un mot de passe avant de le mettre dans une base de données ou de le comparer à celui dans la base de données.

Une autre utilisation est en conjonction avec la cryptographie. En utilisant une fonction de hachage cryptographique, il est possible de signer un document. Premièrement, le hash est fait sur le message et il est crypté avec la clé privée de l'émetteur. Ensuite, lors de la réception, le message est rehaché et la signature décryptée avec la clé publique de l'émetteur. Les deux hashs sont alors comparés et s'ils sont égaux, cela prouve que le message n'a pas été altéré et provient bien de l'expéditeur.

14.3 Protection avec sel

Tel que dit dans la section précédente, le hachage peut être utilisé pour conserver des mots de passe. Par contre, c'est tellement commun que des gens ont créé d'énormes fichiers avec énormément de mots de passe et leur hash. Ainsi, ils peuvent obtenir le mot de passe directement en cherchant le hash dans ce fichier.

Pour éviter cela, nous pouvons utiliser une méthode d'ajout de sel. En gros, si le nom d'utilisateur est "bob" et le mot de passe "god" au lieu de hacher uniquement "god", on va hacher "bobgod" ou même ajouter un autre mot uniquement connu du serveur. Comme cela, le fichier est inutile et il faut encore essayer toutes les possibilités.

15 Base 64

(Vidéo d'explications de ce chapitre disponible[73])

Savez-vous qu'au début de l'informatique, la majorité des tâches s'effectuaient en mode texte avec une console texte? Même envoyer un courriel était fait ainsi. C'est pourquoi le protocole de communication utilisé pour transmettre des messages électroniques est en format texte. Il n'est donc pas possible d'envoyer des données binaires comme des images par courriel.

Je vous entends déjà commencer à vous demander ce que je peux bien raconter étant donné qu'il n'est pas rare de nos jours de recevoir des pièces jointes comme des images et des archives ZIP. Si c'est possible, c'est grâce à une technique spéciale qui permet d'encoder un fichier binaire en texte. Avant d'aller plus loin dans l'explication de la méthode employée, prenons un exemple concret.

J'ai créé pour l'occasion une petite image au format JPG qui n'est que de 5 par 5 pixels et que j'ai nommé "image.jpg". La voici:

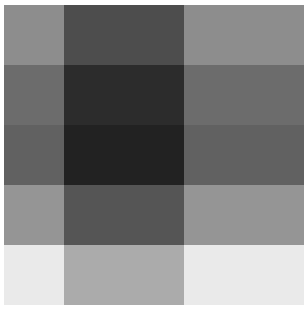

Illustration 29: Base 64: Fichier image.jpg

Ce que vous voyez, c'est l'affichage des données binaires par un logiciel d'imagerie qui comprend ce qu'elles veulent dire. Par contre, si vous ouvrez ce même fichier dans un éditeur texte tel Notepad, vous verrez ceci:

73 https://r.foilen.com/f-base64 : Vidéo explicative sur la base 64

Illustration 30: Base 64: Fichier binaire image.jpg lu avec un éditeur texte

N'est-ce pas magnifique? Essayez d'écrire cela avec votre clavier dans un logiciel d'envoi de courriels dans une console texte. Bonne chance pour trouver les touches! Si vous avez bien regardé le contenu de ce fichier, vous aurez remarqué qu'il y a des lettres un peu partout et même certaines forment des phrases telles que "Paint.NET v3.5.10". Qu'est-ce que représente ce que nous voyons?

En gros, toutes les données sur les ordinateurs sont conservées avec seulement des 1 et des 0. C'est du binaire étant donné que c'est deux valeurs possibles. En plaçant deux nombres binaires ensemble (aussi appelés deux bits), nous pouvons faire quatre possibilités: 00, 01, 10 et 11. En ajoutant un troisième bit, nous doublons pour huit possibilités. Si nous continuons jusqu'à huit valeurs consécutives (un octet), cela donne 256 possibilités d'agencements. En utilisant ces 256 possibilités, nous pouvons leur associer chacune un symbole, comme les chiffres, les lettres minuscules, les lettres majuscules, les accents et bien d'autres. En ouvrant un fichier en mode texte, chaque groupe de 8 bits (1 octet) est lu et sa correspondance en symbole graphique est affichée sur l'écran.

Si vous voulez voir la représentation de chaque symbole de 0 à 255, vous trouverez toutes les correspondances en regardant une table ASCII[74]. En la regardant, vous verrez que de 0 à 31, ce sont des caractères de contrôles comme des retours de ligne. Ensuite, la valeur 32 représente un espace, s'en suit quelques symboles de ponctuations, puis les chiffres (48 à 57), les lettres majuscules (65 à 90) et pour terminer, les lettres minuscules (97 à 122). Tout le restant est du charabia pour le commun des mortels.

Pour revenir à ce que nous voyons dans l'éditeur de texte, il faut comprendre qu'il y a deux formats de fichiers possibles : binaires et textes. Les fichiers textes ne vont avoir à l'intérieur d'eux que des

74 https://r.foilen.com/asciitable : Tableau de toutes les possibilités de la norme ASCII

données que votre clavier peut écrire tandis que les fichiers binaires utilisent toute la gamme des symboles puisque ce ne sont pas des humains qui vont lire le contenu avec un fichier texte, mais plutôt la machine qui va le lire et le traduire dans une autre représentation, comme dans notre cas, une image.

Maintenant, comment pouvons-nous envoyer un fichier quelconque par courriel alors que seulement du texte peut être envoyé? Pour ce faire, il faut simplement changer de base : en lisant un octet à la fois (8 bits), cela nous donne 256 possibilités de symboles. Pour réduire ce nombre à une quantité qui permet de se limiter aux caractères, aux nombres et aux ponctuations puisqu'ils sont du texte, il suffit de lire moins de bits. En n'en lisant que 6 à la fois, cela donne 64 symboles et c'est un nombre acceptable pour transformer le fichier binaire en format texte. Voici comment la lecture d'une même séquence se fait normalement et avec ce procédé :

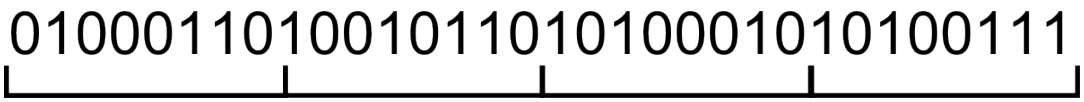

Illustration 31: Base 64: Fichier normalement lu par groupes de 8 bits

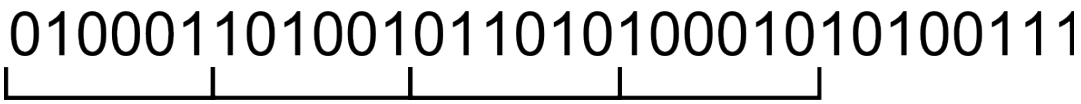

Illustration 32: Base 64: Fichier lu par groupes de 6 bits

Après avoir lu 6 bits, un symbole texte y est associé et ces bits peuvent être transmis dans n'importe quel protocole texte. Reprenons l'image du début et encodons-la en base64 avec cette ligne de commande Linux :

```
base64 image.jpg > image.txt
```

Nous obtenons un fichier texte légèrement plus volumineux qui une fois ouvert dans un éditeur de texte affiche ceci :

```
/9j/4AAQSkZJRgABAQEAYABgAAD/4QBoRXhpZgAATU0AKgAAAAgABAEaAAUAAAABAAAAPgEbAAUA
AAABAAAARgEoAAMAAAABAAIAAAExAAIAAAASAAAATgAAAAAAAABgAAAAAQAAAGAAAAABUGFpbnQu
TkVUIHYzLjUuMTAA/9sAQwD/////////////////////////////////////////////////////
////////////////////////////////9sAQwH/////////////////////////////////////////
////////////////////////////////////////////////8AAEQgABQAFAwEiAAIRAQMRAf/E
AB8AAAEFAQEBAQEBAAAAAAAAAAABAgMEBQYHCAkKC//EALUQAAIBAwMCBAMFBQQEAAABfQECAwAE
EQUSITFBBhNRYQcicRQygZGhCCNCscEVUtHwJDNicoIJChYXGBkaJSYnKCkqNDU2Nzg5OkNERUZH
SElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6g4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1
tre4ubrCw8TFxsfIycrS09TV1tfY2drh4uPk5ebn6Onq8fLz9PX29/j5+v/EAB8BAAMBAQEBAQEB
AQEAAAAAAAABAgMEBQYHCAkKC//EALURAAIBAgQEAwQHBQQEAAECdwABAgMRBAUhMQYSQVEHYXET
IjKBCBRCkaGxwQkjM1LwFWJy0QoWJDThJfEXGBkaJicoKSo1Njc4OTpDREVGR0hJSlNUVVZXWF1a
Y2RlZmdoaWpzdHV2d3h5eoKDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXG
x8jJytLT1NXW19jZ2uLj5OXm5+jp6vLz9PX29/j5+v/aAAwDAQACEQMRAD8AXb796KKAP/Z
```

Illustration 33: Base 64: Fichier binaire encodé en Base 64 (image.txt)

Il est tout de suite évident que ce fichier est bien plus lisible pour un humain que le fichier binaire, malgré qu'il ne nous dit pas grand-chose au premier coup d'oeil. Par contre, ce fichier peut maintenant

être transmis par courriel et rendu au destinataire, il sera reconverti en fichier binaire. Si nous copions ce texte dans un fichier et exécutons la commande suivante:

```
base64 -d image.txt > image.jpg
```

Nous obtiendrons l'image originale.

16 La cryptographie

16.1 Introduction

Sur le réseau Internet tout comme dans le monde réel, tout message envoyé peut être vu par des personnes à qui il n'est pas destiné. Dans le cas de message contenant des données confidentielles, il est important que seule la destination puisse comprendre le message. Étant donné qu'Internet est composé de beaucoup de relais qui transmettent les messages (un peu comme la poste par laquelle un message passe entre les mains d'un facteur et d'autres préposés), la cryptographie est tout indiquée pour résoudre ce problème. Cette technique permet d'encoder des messages d'une manière que seul l'interlocuteur puisse décoder. Pour ce faire, il existe plusieurs méthodes différentes avec des points positifs et d'autres, négatifs.

16.2 Clé symétrique

Les méthodes de cryptage les plus simples sont d'utiliser une fonction mathématique qui est réversible. Par exemple, vous pouvez ajouter le nombre 10 à n'importe quel nombre pour ensuite le soustraire pour retrouver le message initial. Ce sont des méthodes dites symétriques puisqu'il suffit de faire l'opération inverse sur le message. Dans ce cas-ci, la clé serait 10 et l'encodage serait l'addition. Bien entendu, des techniques plus évoluées vont utiliser des opérations plus complexes et avec une clé à plusieurs valeurs.

La simplicité de cette méthode permet des vitesses d'exécution plus rapide et aussi une plus grande facilité aux développeurs à créer ce genre d'algorithme. Par contre, le plus gros problème avec cette méthode est la transmission de la clé. Si je vous envoie un courriel en vous disant que la clé est « 10 » et que vous me transmettez un courriel crypté avec cette clé, si mon compte courriel est surveillé, l'espion possède la clé et le message et il peut donc le décrypter très aisément. Il faudrait plutôt que je vous donne en personne la clé pour m'assurer que vous être le seul à la connaitre.

16.3 Clés publique et privée

Le problème avec les clés symétriques, c'est que pour les transmettre à une autre personne, il faut passer par un canal sécurisé ou en personne, sinon n'importe qui qui l'intercepte pourra décrypter les messages cryptés. Pour utiliser des services bancaires en ligne, il faudrait que la banque vous donne en main propre la clé pour s'assurer que vous êtes le seul à la connaitre. S'il fallait obtenir des clés ainsi, ce serait vraiment pénible et problématique.

Le but des méthodes de cryptage avec une clé privée et une clé publique est de pouvoir s'échanger une clé sur un réseau non sécurisé comme Internet. Pour se faire, les deux clés doivent être différentes et seulement la clé publique est donnée à n'importe qui (même les personnes mal intentionnées). Ainsi, n'importe qui peut crypter des messages avec la clé publique, mais seule la personne avec la clé privée pourra les décrypter. Les mots de passe transmis à la banque peuvent alors être cryptés et seulement lus par la banque elle-même.

Les algorithmes utilisés pour ce genre de cryptage sont bien plus complexes que ceux pour les clés symétriques étant donné que l'opération à effectuer n'est pas une simple addition. Il n'est d'ailleurs pas possible de le faire avec cette opération mathématique. La technique présentement utilisée se sert

d'opérations sur les nombres premiers comme la factorisation et le modulo[75].

16.4 Applications

16.4.1 Sécurité web

J'ai souvent parlé dans ce livre de l'importance du protocole HTTPS pour sécuriser les sites web. Par contre, je ne suis pas encore allé dans les détails d'implémentation de ce protocole. Vous savez déjà qu'en l'utilisant, vous obtenez un certificat d'authenticité et une connexion cryptée. Maintenant, examinons plus en détail la phase d'obtention de la connexion cryptée.

Premièrement, étant donné que nous ne possédons pas une clé de cryptage partagée entre nous et le site web, il faut pouvoir transmettre une clé de manière sécurisée sur le réseau Internet qui lui à la base n'est pas sécurisé. Pour se faire, la méthode des clés publique et privée est utilisée. Puisque le serveur envoie un certificat d'authenticité, ce dernier inclut une clé publique avec laquelle il est possible d'envoyer un message crypté au serveur. Par contre, il n'est pas possible au serveur de nous envoyer un message crypté puisque nous ne lui avons pas envoyé notre clé publique, car nous n'en possédons pas. Il ne peut pas non plus utiliser sa clé privée puisque n'importe qui avec sa clé publique pourrait alors décrypter le message, ce qui rend la procédure inutile.

Deuxièmement, puisque le système de cryptage asymétrique est plus gourmand en ressources et qu'un serveur veut pouvoir servir beaucoup d'utilisateurs à la fois, pour le restant de la connexion, une clé symétrique est utilisée. Nous allons donc créer une clé symétrique temporaire et l'envoyer au serveur en la cryptant avec sa clé publique. Ainsi, uniquement le serveur pourra décrypter notre message contenant la clé symétrique et nous pourrons ensuite échanger des données avec la même clé symétrique durant tout le restant de l'échange.

En résumé, il suffit de prendre la clé publique du serveur qui est fournie dans son certificat et ensuite lui envoyer un message crypté contenant une clé symétrique temporaire. C'est une façon économique d'utiliser la cryptographie et c'est très sécuritaire.

16.4.2 Transmettre des courriels

L'envoi de courriels est certainement le service sur Internet le moins confidentiel. En plus des sauts normaux entre les relais d'Internet qui peuvent intercepter des morceaux de messages, il y a les relais de serveurs de courriels qui eux peuvent aisément conserver une copie de nos messages au complet sans le moindre effort.

Pour bien comprendre le monde des courriels, voici ce qui se passe typiquement lorsque nous transmettons un message à une autre personne. Si nous utilisons le service de courriels de notre fournisseur Internet, appelons-le « Vidéotron », nous allons envoyer un courriel sur le serveur de Vidéotron. Ce dernier va regarder l'adresse de destination, disons « bob@hotmail.com » et se connecter sur le serveur courriel d'Hotmail pour lui transmettre le message. Ensuite, le destinataire va lire le message et l'effacer. Par contre, une copie peut facilement être conservée sur les serveurs de Vidéotron et d'Hotmail.

En plus de ces relais, il y a aussi des gens qui peuvent essayer de pirater notre compte ou les autorités qui peuvent en demander l'accès. Pour débouter tous ces gens, rien de mieux que crypter les messages.

75 Voir chapitre 24 à la page 105

Pour ce faire, nous pouvons utiliser un logiciel comme Enigmail[76] (extension à Thunderbird[77]) qui utilise GPG[78] en arrière-plan ou utiliser la technologie S/MIME. Pour se servir des deux types, il faut bien comprendre les étapes à franchir. Pour envoyer des courriels cryptés, il faut que notre destinataire nous ait fourni une clé publique et qu'il possède sa clé privée. Cela veut dire que tous nos contacts doivent individuellement posséder un certificat. C'est d'ailleurs une des raisons pour lesquelles si peu de gens utilisent cette technologie, car il faut quand même prendre le temps de comprendre le mécanisme et de créer les certificats.

Avec GPG, pour recevoir des courriels cryptés, il faut créer une clé publique à mettre sur notre site web ou l'envoyer par courriel à tous ceux qui utilisent GPG. Enigmail permet d'automatiser ceci en publiant notre clé publique sur son serveur et en laissant les gens le télécharger au moment d'envoyer le message. En même temps de créer une clé publique, nous recevrons une clé privée qu'il faut sécuriser à tout prix, car quiconque possède la clé privée pourra lire les messages cryptés et aussi signer des messages comme s'ils parvenaient de nous. L'explication des signatures est discutée dans la prochaine section.

Avec S/MIME, il faut demander à une autorité émettrice de certificats (les mêmes que ceux qui en émettent pour le HTTPS) de créer un certificat pour nous. Par exemple, StartSSL[79] peut le faire gratuitement et nous donner un certificat (incluant la clé publique) et une clé privée que nous pouvons importer dans Thunderbird ou un autre client de courriels le supportant. Le destinataire pourra obtenir notre clé publique en demandant à l'autorité de lui envoyer plutôt qu'aux serveurs de clés d'Enigmail.

16.4.3 Signature électronique

Une signature électronique, tout comme une signature sur un papier, sert à authentifier que le message ou document provient de la bonne personne. Que ce soit pour signer un courriel avec GPG (voir section précédente), un document Word ou un document PDF, le mécanisme reste le même.

Pour bien comprendre la théorie derrière la signature électronique, il faut revenir à ce qu'est le cryptage asymétrique. Dans ces algorithmes, nous avons deux clés distinctes dont l'une est privée et l'autre public. Le choix de laquelle est laquelle n'a pas d'importance, car tout ce qui est crypté par l'une d'elles n'est déchiffrable que par l'autre. Lorsqu'une est choisie comme clé privée, il est ainsi clairement affiché que cette dernière n'est disponible à personne d'autre que l'auteur original. Ce qui veut dire que tout ce qui est crypté par l'autre clé (dite publique) n'est lisible que par la personne qui a la clé privée. Dans l'autre sens, tout message crypté par la clé privée ne peut être déchiffré que par la clé publique. En soit, ce n'est pas une méthode utile pour crypter puisque tout le monde peut le déchiffrer, mais cela amène les fondations de la signature.

Continuons l'exploration en reprenant le dernier bout : « tout message crypté par la clé privée ne peut être déchiffré que par la clé publique ». Nous pouvons donc dire que s'il est possible de lire un message avec une clé publique en particulier, l'auteur n'a pas d'autre choix que d'être celui qui possède la clé privée. Puis, comme cette clé est censée n'appartenir qu'à un seul auteur, nous pouvons être certains de l'authenticité du message. Ainsi, un message crypté avec une clé privée prouve son auteur parce qu'il peut être lu avec la clé publique de cet auteur.

Dans la pratique, le message ou le document n'est jamais crypté en entier pour plusieurs raisons :

76 https://r.foilen.com/enigmail : Site de l'extension Thunderbird
77 https://r.foilen.com/thunderbird : Site officiel du client de courriels Thunderbird
78 https://r.foilen.com/gnupg : Site officiel de GPG
79 https://r.foilen.com/startssl : Autorité émettrice de certificats gratuits

- ce n'est pas tout le monde qui a un lecteur de courriel qui peut décrypter les messages;
- ce n'est pas possible d'avoir deux personnes ou plus qui authentifient le même message sans avoir à dédoubler l'information en entier.

Pour contourner ce problème, le message ou document n'est pas crypté du tout et le signataire ne fait qu'ajouter un Hash[80] et il crypte ce Hash avec sa clé privée. Ainsi, plusieurs personnes peuvent ajouter leur Hash crypté du message ou document original.

16.4.4 Crypter des fichiers

Parfois il est utile de crypter des fichiers importants que ce soit pour les transporter sur une clé USB sans avoir peur en cas de perte que quiconque les ouvre ou pour sauvegarder ces fichiers chez un fournisseur de service comme Dropbox[81] sans s'inquiéter des mauvais yeux des employés.

Une solution simple et rapide pour crypter ponctuellement des fichiers est de les compresser au format Zip, 7z ou rar en ajoutant un mot de passe dessus. Le mot de passe doit être assez long pour permettre une bonne sécurité, car il y a beaucoup de logiciels qui existent pour essayer toutes les possibilités de mots de passe. Plus il est long avec des symboles spéciaux, moins il sera vulnérable à ce type d'attaque.

Si vous désirez pouvoir ajouter, éditer et enlever des fichiers fréquemment, mieux vaut aller du côté de TrueCrypt[82]. Ce logiciel permet de crypter un disque dur ou une clé USB en entier ou encore de simplement créer un gros fichier qui peut être présent dans n'importe quel répertoire. C'est cette dernière option que j'utilise pour crypter mes fichiers que je sauvegarde sur Dropbox. Lorsque nous voulons accéder à nos fichiers, il suffit d'ouvrir Truecrypt et de choisir le lecteur virtuel qui sera créé. Ainsi, il est possible d'accéder à tous les fichiers comme s'ils étaient sur un média amovible (clé USB, disquette) avec n'importe quelle application; contrairement aux Zip qu'il faut extraire avant de pouvoir les utiliser.

Une autre fonctionnalité intéressante de TrueCrypt est de pouvoir posséder deux mots de passe distincts pour le même entrepôt et ces deux mots de passe donnent accès à des fichiers différents. Ainsi, si nous avons un secret à cacher même des autorités, nous les mettons sous le second mot de passe et nous plaçons des fichiers peu importants dans le premier. Ensuite si nous nous faisons demander le mot de passe, nous pouvons donner le premier et dire qu'il n'y en a pas un second. Puisqu'il n'y a pas de façon de savoir si un deuxième mot de passe est utilisé sans avoir ce mot de passe, c'est ce qui permet de dénier en avoir un second.

16.5 Cryptanalyse

La cryptanalyse c'est l'étude d'un message crypté pour tenter de le décrypter en devinant la clé utilisée avec des méthodes statistiques.

Par exemple, si le message initial est un texte en français, il est possible de regarder statistiquement parlant les lettres et les mots qui se répètent le plus. Si l'algorithme de cryptage est simple, alors cette répétition risque de se produire dans le message codé. Ensuite, les associations les plus probables vont permettre d'essayer de former des mots avec les lettres restantes.

Pour contrer ce genre d'analyse, quelques méthodes existent comme de compresser le message avant de

80 Voir chapitre 14 à la page 76
81 https://r.foilen.com/f-dropbox : Vidéos sur l'utilisation de Dropbox
82 https://r.foilen.com/f-truecrypt : Vidéos sur l'utilisation de TrueCrypt

le crypter. Ainsi il n'est pas possible de savoir la véritable taille du message et il n'y a plus les répétitions linguistiques.

16.6 Stéganographie

La stéganographie est une technique pour dissimuler un message parmi des données ordinaires. Par exemple, en partant d'une photo, il est possible de cacher du texte dans quelques pixels bien précis. En regardant l'image, les pixels modifiés ne sont pas aisément visibles, mais un programme qui sait où chercher peut extraire le message ainsi caché.

Les utilités sont diverses comme la transmission d'un message qui semble anodin pour quiconque espionne, mais qui ne l'est pas pour qui le message est destiné. Ainsi, contrairement à l'envoi d'un fichier crypté, il n'est pas évident à l'observateur qu'une information importante est en train de s'échanger. Une autre utilité est d'ajouter une trace à une photo, une musique ou toute autre oeuvre qui pourrait se faire pirater. La trace pourrait contenir un identifiant de la personne qui a acheté le bien et qui l'a distribué à tout le monde. En téléchargeant la version piratée, l'auteur pourra ainsi remonter au fautif qui a volé son produit.

17 Les failles à exploiter sur des services Internet ou des logiciels

17.1 Introduction

Nous entendons souvent que des pirates ont pénétré dans un système en utilisant une faille de sécurité. Comment se fait-il qu'il ne semble pas y avoir un logiciel de fiable à 100% ? D'où proviennent ces failles et comment les trouvons-nous?

17.2 L'oublie

Pour beaucoup d'utilisateurs, l'ordinateur semble tout faire par magie, mais dans la réalité quotidienne des développeurs, cette machine est stupide et ne fait qu'exactement ce qui est demandé. Ce qui fait que si un programmeur oublie de faire une vérification des entrées des utilisateurs ou s'il oublie de changer l'état du système, cela peut ouvrir des brèches de sécurité. D'ailleurs voici une anecdote qui m'est arrivée sur IRC dans le temps où j'y étais activement.

Dans un canal que j'allais souvent, une personne qui était opérateur a voulu faire un jeu de trivia automatisé en script pour mIRC[83] (un client IRC pour Windows). Le but du jeu était de répondre à une question et le premier qui a la bonne réponse peut écrire en privé au robot la commande "kick NICK" pour faire sortir une personne du canal. De plus, après la 5e bonne réponse donnée par un même joueur, celui-ci peut entrer la commande "ban NICK". À la toute fin, le gagnant est bien entendu le dernier qui reste sur le canal.

Je me suis dit que puisqu'il était en train de développer son script en même temps que de solliciter notre participation, peut-être avait-il oublié de faire en sorte qu'après avoir entré une commande (kick ou ban), de désactiver les prochaines commandes de cet utilisateur. En gros, je me suis dit que sûrement le script ne faisait que se souvenir de la dernière personne qui a bien répondu à une question et ne vérifie pas si elle a aussi banni ou fait sortir une personne du canal depuis sa bonne réponse puisque cela demande plus de lignes de code.

J'ai donc préparé un fichier dans Notepad avec une série de "kick NICK", un par ligne et avec le nick de chaque personne du canal. Ensuite, j'allais copier-coller cette liste dans un message privé au robot après avoir bien répondu à une question. C'est ce que je fis et tout le monde a été sorti du canal. Le développeur m'a supplié d'arrêter alors que je n'avais collé qu'une fois le message pour tester. Ce fût quand même cocasse, mais l'effet aurait été encore mieux si j'avais bien répondu à 5 questions de suite pour bannir tout le monde, mais je n'avais pas le goût d'attendre plus longtemps.

Ce qu'il faut en retenir c'est qu'il y a beaucoup de validations à faire sur l'état de l'application à un temps donné et qu'un oubli humain arrive très souvent. Un logiciel récent a plus de chances de posséder ce genre de failles qu'un logiciel mature.

17.3 La simplicité

Apprendre un langage de programmation n'est pas si complexe pour réussir à faire de petits logiciels. Par contre, il est important de ne pas s'arrêter au langage, mais il faut s'attarder sur les limitations des

83 https://r.foilen.com/mirc : Site officiel du client mIRC

outils qu'il nous offre. En ne se concentrant que sur le problème que nous désirons résoudre, il est possible d'oublier des problèmes plus généraux tels ceux de sécurités. Les failles se concrétisent surtout lorsqu'il y a des interactions entre plusieurs composantes comme un site web et une base de données. Par exemple, tout va bien tant que toutes les instructions envoyées à la base de données sont entièrement écrites par un programmeur, mais si une requête permet à un utilisateur du site web d'y entrer certains paramètres, il faut toujours les valider pour ne pas qu'il injecte des commandes SQL.

Prenons le cas d'un site web avec un champ de nom d'utilisateur et de son mot de passe. Cela existe partout et c'est très simple de coder une vérification d'utilisateur. La logique utilisée est de demander à la base de données, combien d'utilisateurs ont le nom d'utilisateur demandé et ce mot de passe. Si la réponse est 1, alors nous avons un bon utilisateur et s'il y en a 0, ce n'est pas le bon mot de passe. En SQL, cela donnerait par exemple:

```
select count(*) from User where login='$login' and password='$pass'
```

J'y ai inséré deux variables directement à l'intérieur, ce qui est très dangereux si l'utilisateur peut entrer n'importe quoi. Dans ce cas, il pourrait dire qu'il s'appelle:

```
joe'--
```

Cela ferait la commande:

```
select count(*) from User where login='joe'--' and password=''
```

Étant donné que le – en SQL indique que le restant est un commentaire cela devient un équivalent de:

```
select count(*) from User where login='joe'
```

S'il y a un utilisateur joe qui existe, cela retournera 1, alors nous pouvons nous connecter en tant que n'importe qui simplement en connaissant son nom d'utilisateur et en donnant n'importe quel mot de passe. Ceci est appelé une attaque par injection, puisqu'il y a injection de code à exécuter. Dès qu'il est possible de faire exécuter du code, il est possible de le faire pour n'importe quel code. Par exemple, pourquoi se limiter à se connecter en tant qu'un autre utilisateur? Pourquoi ne pas terminer la commande courante et demander un effacement complet de la table utilisateur pour semer la zizanie? Cette faille est donc plus grave qu'elle n'y paraît au premier abord.

En conclusion, il faut toujours penser aux interfaces entre les technologies et entre l'utilisateur et le système. La meilleure pratique étant de tout bloquer par défaut et de débloquer au fur et à mesure ce qui est minimalement nécessaire. Même s'il est très simple de programmer, connaitre toutes les subtilités vient avec l'expérience et l'étude.

17.4 Les humains

Étrangement, la plus grande faille de sécurité en informatique n'est pas la technologie elle-même, mais les humains. Pourquoi tenter de trouver une faille où pénétrer dans un système pour y soutirer de l'information lorsqu'il est plus simple de demander à une personne qui a déjà les accès de vous fournir les données désirées? Ceci s'appelle le « social engineering ». Un autre facteur humain est dans la façon dont certains vont interagir avec leur ordinateur et créer des failles non intentionnellement.

Pour le « social engineering », le but est de se faire passer pour quelqu’un de légitime pour obtenir des informations confidentielles. Par exemple, il est possible d'appeler chez une personne en se faisant passer pour une agence de recouvrement et lorsque la victime dit qu'elle n'a pas cette dette, tenter d'obtenir son numéro d'assurance sociale sous prétexte de s'assurer que c'est le bon dossier que nous sommes en train de consulter. Une victime crédule pourrait donner son numéro et c'est toujours le but

du « social engineering » : créer une histoire crédible pour tenter de prendre des gens au piège. C'était un exemple avec un particulier, mais le plus intéressant, c'est les entreprises. En se faisant passer pour un autre employé qui a un problème pour se connecter au réseau de l'entreprise et qui aimerait avoir de l'aide en se faisant envoyer des documents par courriel est un autre exemple courant de ce problème.

Pour prévenir ce genre d'attaque, il faut que les gens soient mis au courant des stratagèmes et des réponses à donner. Pour le particulier, il n'a jamais à donner son numéro d'assurance sociale à qui que ce soit, alors simplement ne pas le dire est suffisant. S'il désire quand même le faire, il doit s'assurer que la personne au bout du fil est légitime en lui demandant un numéro où il peut rappeler et investiguer sur ce numéro en plus de rappeler avant de donner les renseignements. Pour l'entreprise, toujours demander le numéro de l'employé qui demande de l'aide et le rappeler à un numéro de téléphone qui est dans le bottin de l'entreprise est la base même d'une bonne procédure de sécurité.

Pour la seconde faille humaine, la sécurité des systèmes dépend de la bonne volonté de ses utilisateurs. Laisser un mot de passe traîner proche de l'ordinateur ou en utiliser un faible sont des portes d'entrée aux malfaiteurs. Parfois, un code qui semble fort ne l'est pas toujours. Par exemple, dans ma jeunesse, j'ai remarqué que mes deux parents et ceux d'un ami avaient tous utilisé leur numéro d'adresse comme mot de passe à leur messagerie vocale téléphonique. En soit, cela semble bien puisqu'une personne qui ne connait que le numéro de téléphone ne pouvait pas savoir l'adresse civile (maintenant avec le site Canada411[84], il est possible de trouver une adresse avec un téléphone). Je m'étais alors demandé s'il y avait beaucoup d'autres personnes qui avaient eu cette idée et en prenant le bottin téléphonique, j'ai cherché qui avait le service de boite vocale et sur une dizaine, il y en avait huit qui utilisaient leur adresse comme code. À partir de là, j'aurais aisément pu envoyer des messages de leur part à n'importe qui et écouter leurs messages confidentiels.

17.5 Déni de service (DDOS)

Un DOS (Denial of Service) est une attaque qui consiste à utiliser toutes les ressources d'un service pour qu'il ne soit plus accessible. Ainsi, personne ne peut plus utiliser un site web ou tout autre service disponible. Le plus gros problème de cette attaque est qu'elle ne peut pas être contrée, sauf en fournissant plus de ressources.

Souvent, ce type d'attaque est faite en utilisant plusieurs ordinateurs, ce qui fait un DDOS (Distributed Denial of Service). Pour l'exécuter, c'est très simple puisqu'il suffit de créer plusieurs connexions à un serveur et de les laisser ouvertes le plus longtemps possible. Plusieurs scripts et logiciels qui automatisent ce processus existent et sont facilement disponibles.

Pour bien comprendre, imaginons un site web nommé supermeteo.com . Si ce site peut supporter 100 utilisateurs par secondes, en utilisant 200 ordinateurs qui créent 10 connexions en même temps sur le site, cela fait 2000 faux utilisateurs par seconde. Ainsi, une personne qui voudrait accéder à ce site recevrait un message d'erreur disant qu'il n'est pas disponible. La seule façon de contrer cette attaque serait d'ajouter des serveurs pour supporter plus d'utilisateurs en même temps. Par contre, ce n'est pas tous les services qui sont prêts à grandir sur demande. Dans le passé, il y a déjà eu une attaque faite contre Amazon[85], mais étant donné que cette compagnie est toujours prête à recevoir plus d'affluence comme dans le temps des fêtes, elle est passée inaperçue.

84 https://r.foilen.com/can-411 : Bottin téléphonique en ligne
85 https://r.foilen.com/amazon : Site d'un magasin en ligne

18 Ingénierie inverse

18.1 Introduction

Pour bien comprendre ce qu'est l'ingénierie inverse, il faut d'abord connaitre le processus de création d'un logiciel puisque le but de l'ingénierie inverse est de faire le processus à l'envers pour analyser une application.
Un programme exécutable est simplement une liste d'instructions (de commandes) qui sont exécutées les unes après les autres. Ces instructions doivent être écrites de façon à ce que l'ordinateur (plus particulièrement le processeur) puisse les comprendre. C'est donc du code machine en format binaire. Créer un logiciel ne se fait pas directement en langage machine, sinon ce serait bien trop pénible. Par exemple, pour sauter à une série d'instructions (qui est une fonction), il faut dire de combien d'instructions sauter. Cela veut dire que si nous désirons ajouter une instruction entre le saut et la destination, il faudrait modifier l'instruction de saut pour qu'il saute une ligne supplémentaire. Le but des compilateurs est de faciliter la vie des développeurs en faisant eux-mêmes les calculs des positions et en transformant des instructions de ligne de texte en format binaire.

Le langage de programmation le plus proche du langage machine est l'assembleur. Il permet d'avoir des étiquettes pour calculer les sauts des fonctions et des boucles en plus de permettre d'écrire des mots plutôt que de connaitre leur équivalent en binaire.

D'autres langages de plus haut niveau comme le C et C++ permettent de faciliter la programmation en générant plusieurs lignes d'instructions en en écrivant que quelques-unes par le programmeur. C'est aussi des langages où il est plus facile de comprendre ce qui se passe en regardant le code source puisqu'un concept est exprimé en une seule ligne de ce langage alors qu'il pourrait en être des dizaines en assembleur.

Le langage Java est quelque peu différent. Il ne crée pas du code machine puisque ce dernier est dépendant du système d'exploitation et du processeur utilisé. Au lieu, il va créer du Bytecode qui est une version comprimée en binaire du code source et une fois qu'il est exécuté dans une machine virtuelle Java, cette dernière va le traduire en code machine pour l'exécution courante.

18.2 Ce que c'est

Lorsque des développeurs de logiciels créent des applications, ces dernières sont normalement compilées. Ce que cela signifie est que les programmeurs écrivent du code dans un certain langage comme Java, C++, assembleur, etc. et qu'ensuite, ce code est converti en langage machine compréhensible par l'ordinateur uniquement. Si nous prenons comme exemple un programme fait en C++ pour Windows, le code source est composé de centaines de fichiers ".cpp" et ".h". Une fois compilées, ces sources vont être transformées en ".exe" et ".dll". Une fois que c'est terminé, le résultat n'est pas facile à comprendre pour les personnes qui ne reçoivent que les exécutables. L'ingénierie inverse, c'est l'action de partir des fichiers compilés et de tenter de les analyser pour retrouver un code assez semblable au code source original. Ce n'est pas aisé, surtout qu'avec toute la bonne volonté de ceux qui créent des outils, il y a certaines informations qui sont perdues et impossibles à retrouver comme les commentaires des développeurs et parfois les noms des variables et des fonctions. C'est donc très demandant pour une personne d'analyser le code produit par un de ces outils, mais c'est toujours mieux que de devoir lire des fichiers binaires à l'oeil nu.

18.3 Les outils

Pour les fichiers ".exe" et ".dll", plusieurs outils existent en fonction du langage initial. Par exemple, il y a des outils pour du code en C++ et d'autres en .NET[86].

En dernier recours, il est toujours possible de désassembler l'exécutable en code assembleur. C'est possible puisque comme mentionné dans la section précédente, le C++ passe par l'assembleur avant d'être mis en code machine. Par contre, ce n'est pas très utile pour comprendre le fonctionnement et les algorithmes utilisés dans le logiciel en entier, car c'est bien trop long comme code, mais c'est parfait lorsque nous cherchons un point précis pour par exemple cracker le logiciel. Le but de cracker une application est d'enlever une protection en modifiant un code qui fait un test de validité pour qu'il soit toujours valide peu importe la situation. Le cas le plus courant étant de sauter les protections d'un numéro de série ou d'une activation du logiciel par Internet.

Pour les fichiers Java en ".class", le décompilateur de Bytecode Jad[87] est très efficace. Mis à part les commentaires, tout le reste est pratiquement une copie conforme des sources originales. C'est grâce à toute l'information présente dans chaque fichier ".class" que ce miracle est possible. Le Bytecode possède beaucoup d'information pour pouvoir aisément changer un fichier ".class" et le remplacer pour un autre qui possède les mêmes signatures de fonctions puisque les noms de fonctions ne sont pas modifiés. Le but étant de pouvoir utiliser des librairies différentes qui offrent les mêmes fonctionnalités, mais optimisées différemment. C'est une fonctionnalité qui est souvent utilisée par les ingénieurs qui utilisent Java. Ainsi, ils peuvent aisément prendre une librairie ou une autre qui sont appelées de la même façon sans avoir à recompiler le code source à chaque fois.

86 https://r.foilen.com/reflector : Site officiel de Reflector
87 https://r.foilen.com/jad : Liste de miroirs de Java JAD

Les bases de la théorie de l’informatique

19 Les différents préfixes de grandeurs

Les préfixes de grandeurs n'ont rien de sorciers puisque nous les utilisons tous les jours dans d'autres domaines. Par exemple, 10 km veut dire 10 000 mètres tout comme 10 Ko veut dire 10 000 octets.

L'espace mémoire est en octet et la vitesse d'un processeur est en hertz.

Aucun préfixe : 1 octet / 1 hertz

Kilo : 1 Ko = 1 000 octets (10^3)

Méga : 1 Mo / 1 MHz = 1 000 000 octets / hertz (10^6)

Giga : 1 Go / 1 GHz = 1 000 000 000 octets / hertz (10^9)

Téra : 1 To = 1 000 000 000 000 octets (10^{12})

Ceci est bien entendu une simplification pour facilement s'en rappeler puisque c'est un ordre de grandeur reconnu. Par contre, les valeurs réelles sont en puissances de 2 étant donné que nous sommes en informatique. Au lieu d'être 1000, le multiplicateur est 1024. Les équivalents sont comme suit:

1 Ko = 1024 octets (2^{10})

1 Mo = 1024 Ko = 1 048 576 octets (2^{20})

1 Go = 1024 Mo = 1 073 741 824 octets (2^{30})

1 To = 1024 Go = 1 099 511 627 776 octets (2^{40})

Si nous comparons l'approximation du haut avec la réalité, les marges d'erreurs sont très petites lorsqu'il est question de Ko et de Mo (erreur de 2% à 5%). Elles sont plus significatives dans les Go et les To (erreur de 5% à 9%).

20 Les types de fichiers (extension)

Il est important de connaitre les types de fichiers les plus communs pour éviter de se faire avoir. Par exemple, si une personne vous dit qu'elle vous transmet une photo d'elle et que le fichier est un exécutable, il pourrait très bien être un virus. Pouvoir distinguer les différents fichiers permet de ne pas laisser un fichier ouvrir une application non désirable.

L'extension est ce qui suit le dernier « . » dans un nom de fichier. Au tout début, sous DOS, elle était limitée à trois caractères, mais maintenant cela peut être autant que désiré. Voyons quelques exemples pour se réchauffer :

Nom du fichier	Extension	Type de fichier
Photo.jpg	jpg	Une image
Gugus.exe	exe	Exécutable (logiciel, application, …)
Photo.jpg.exe	exe	C'est à s'y méprendre, mais c'est ce qui est après le dernier point seulement qui est l'extension, alors ceci est un exécutable et non une image
Bobby – Love Reggea.mp3	mp3	Musique

Ce chapitre va donner les extensions les plus répandues, mais il y en a tellement que si vous en voyez une que vous ne connaissez pas, faites une petite recherche sur Internet pour vous protéger et apprendre.

20.1 Exécutable

Un exécutable est un programme quelconque qui va s'exécuter sur votre ordinateur. Tout jeu et toute application sont des exécutables. Si vous ne vous attendez pas à recevoir un logiciel, ne l'ouvrez pas, car il pourrait être un virus.

Extension	Type de fichier
com, exe	Ce sont des programmes compilés.
bat, cmd	Ce sont des scripts qui appellent d'autres programmes qui peuvent effacer le disque dur, télécharger d'autres logiciels, etc.
vbs	C'est un script Visual Basic. C'est un peu comme les « bat » et « cmd », mais il permet aussi d'afficher des boites de texte et toucher à des parties plus importantes d'un système.

20.2 Image

Les images sont normalement sécuritaires à regarder tant que le logiciel qui les ouvre n'a pas de failles

de sécurités. Si vous utilisez des visualiseurs populaires, il ne devrait pas y avoir de danger.

Extension	Type de fichier
bmp	C'est une image « bitmap » qui n'est pas compressée. Il est normal que ce type de fichier devienne rapidement très gros, mais il n'a aucune perte de qualité.
gif, jpg, jpeg	Ce sont des images compressées et avec une perte de qualité pour réduire encore plus la taille des fichiers. Le « gif » supporte aussi la transparence.
png	Ce type d'image est très intéressant puisque tout comme les « bmp » il n'y a pas de perte de qualité, mais contrairement à eux, il peut compresser les données. De plus, il supporte la transparence, ce qui le rend très intéressant pour des interfaces de site web.
svg	C'est une image « vectorielle ». Puisque ce type d'image contient des instructions de dessin (ligne, cercle, …), normalement sa taille est très petite et sa qualité est parfaite.
psd	Fichier d'Adobe Photoshop. Ce logiciel est très utilisé dans le milieu professionnel pour créer des images, mais normalement les images finales sont exportées en « bmp », « jpg » ou « png ».

20.3 Musique

Tout comme pour les images, il est sécuritaire d'ouvrir de la musique tant que le visualiseur n'a pas de failles de sécurité.

Extension	Type de fichier
wav	Ce fichier sonore n'a aucune compression et aucune perte de qualité. Une musique de 3 minutes prend souvent 30 Mo d'espace.
mp3, ogg	Ces deux formats enlèvent les trop hautes et trop basses fréquences que l'oreille n'entend pas toujours, ce qui amène une baisse de qualité et ils ajoutent de la compression. Une musique de 3 minutes prend environ 3 Mo. La différence notable entre ces deux types est que le premier est commercial avec un brevet alors que le second est un standard libre et ouvert.
flac	Ce type de musique est sans perte de qualité, mais avec de la compression. Une musique de 3 minutes est d'environ 15 Mo ce qui est la moitié de l'autre format sans perte, mais trois fois plus que ceux qui ont une perte. Certains audiophiles préfèrent ce format pour conserver toutes les fréquences d'une œuvre.

20.4 Vidéo

Pour les vidéos, il y a deux notions importantes à distinguer: le codec et le contenant. L'extension du fichier va dire quel est le contenant, mais pas toujours quel est le codec utilisé. Ce dernier est ce qui

choisit le type de compression vidéo et audio. L'important pour cette section est simplement de distinguer ce qui peut être lu dans un logiciel vidéo.

De plus, j'aimerais préciser qu'il est rare qu'un codec utilisé soit sans perte de qualité puisque la taille des fichiers serait bien trop imposante pour permettre un transfert sur Internet. Par contre, pour les professionnels qui font des montages, eux prennent normalement un format sans perte directement de leur caméra pour ensuite produire un fichier qui sera diffusé.

Extension	Type de fichier
avi, mpg, ogv, wmv, mov, flv	Des fichiers vidéos très communs pour toute sorte d'utilisations.
mkv, mp4	Des fichiers vidéos qui normalement, mais pas obligatoirement, contiennent un codec pour des vidéos en hautes résolutions. La qualité est très haute et la taille des fichiers est plus grosse.
vob	Le format d'un fichier DVD. Il a une perte de qualité pour entrer 2 heures de vidéo sur un DVD et il est parfois crypté. Beaucoup de logiciels gratuits existent pour les décrypter.

20.5 Texte

Pour cette section, le format texte est plutôt à prendre comme un document généré par un logiciel de traitement de texte. Cela signifie qu'il peut aussi contenir des images et autres fioritures. C'est donc à ne pas confondre entre un fichier texte et binaire.

Extension	Type de fichier
txt	Il n'y a pas plus texte que ce fichier. Il ne possède aucune image, aucun formatage (aucun choix de police, grosseur, gras, souligné, etc.). C'est du texte brut.
doc, docx, odt	Du traitement de texte complet qui peut être ouvert et édité dans Microsoft Word[88] ou LibreOffice Writer[89].
pdf	À la base, ce type de fichier était un document texte avec uniquement des images. Maintenant, il permet d'aller jusqu'à mettre des scripts Javascripts ainsi que des animations Flash. Plusieurs failles de sécurités dans le lecteur le plus connu, Adobe Reader, ont été découvertes et continuent de l'être. Vous pouvez toujours prendre un autre lecteur comme Foxit[90] ou plusieurs autres à sources ouvertes.
ppt, pptx, odp	Ce sont des fichiers de présentation comme Microsoft PowerPoint et Libreoffice Impress. Le contenu est semblable à du traitement de texte complet.

88 https://r.foilen.com/f-word-2010 : Vidéos sur l'utilisation de Microsoft Word 2010
89 https://r.foilen.com/f-libreoffice-writer : Vidéos sur l'utilisation de LibreOffice Writer
90 https://r.foilen.com/foxit-reader : Site officiel de Foxit qui est un lecteur de fichiers PDF

20.6 Archives

Les archives sont des fichiers qui contiennent d'autres fichiers. Souvent ces paquets de fichiers sont compressés et peuvent aussi être cryptés. À la base, ce type de fichier n'est pas dangereux, mais les fichiers à l'intérieur peuvent l'être. Dans le passé, beaucoup de service de courriels ne permettait pas l'envoi de fichiers exécutables par courriel, mais permettait les archives. Ainsi, beaucoup de virus ont été transférés en étant « cachés » dans l'archive. Ainsi, les victimes ouvraient l'archive et ensuite l'exécutable sans regarder que le fichier compressé qu'elles ouvraient était un exécutable. Maintenant, les archives sont scannées par des antivirus pour éviter ce type de problème. Par contre, un antivirus n'est pas toujours à l'affût des nouveaux virus.

Les extensions sont nombreuses pour les archives et en voici quelques-unes: 7z, zip, rar, arc, tar, gz, gzip, bz2, bzip2, tbz, tbz2, etc.

21 Les nombres binaires et hexadécimaux

(Vidéo d'explications de ce chapitre disponible[91])

Pour une machine électrique, tout ce qui existe, c'est deux états: Courant et absence de courant. Cela peut être représenté par 1 et 0 et c'est la base du système binaire.

La base que tout le monde connait particulièrement bien est la base décimale: chaque position va de 0 à 9 et ensuite, pour avoir des nombres plus gros, il suffit d'ajouter une position en face de l'autre. Pour le binaire, chaque position n'a que deux valeurs de 0 à 1. Dès que nous désirons représenter le nombre 2, il faut ajouter une nouvelle position, ce qui donne 10.

En informatique, chaque position est appelée un bit et comme il ne représente pas beaucoup d'information, l'unité de base utilisée est l'octet. Cette dernière est composée de 8 bits pour ainsi avoir des valeurs de 00000000 à 11111111, ce qui en base dix équivaut à 0 à 255 pour un total de 256 possibilités.

256, c'est encore un très petit nombre comparé aux utilisations de l'ordinateur que nous faisons tous les jours. Par exemple, cela ne peut même pas représenter le montant d'une paie. C'est pourquoi les ordinateurs sont plutôt basés sur 32 ou 64 bits d'adresses. Ainsi, nous pouvons atteindre des nombres plus faramineux. Par exemple, avec juste 16 bits, nous pouvons atteindre 65536 valeurs différentes (256*256). Faites 65536*65536 sur une calculatrice pour voir a quel point le 32 bits vous offre de possibilités. Réponse: cela fait plus de quatre millions et pour le 64 bits, je n'ai pas de mots pour le décrire.

Maintenant, c'est bien beau de pouvoir représenter des nombres en binaires, mais si nous voulons lire dans la mémoire d'une machine aisément, cela n'est pas possible en voyant des groupes de 32 ou 64 "0" et "1" de suite. C'est sûr qu'il est possible de le transformer en base dix, mais ce n'est pas naturel pour bien comprendre la mémoire de l'ordinateur. Par exemple, si je vous dis que 255+1 donne 0, ce n'est pas évident de voir la raison de ce problème, tandis que si vous voyez 1111 1111 + 1 = 0000 0000, vous pouvez facilement comprendre que nous étions a la limite de la mémoire et qu'il manquait de l'espace pour faire 1 0000 0000.

Il nous faudrait donc une base qui se découpe aux mêmes endroits que ces nombres binaires. Si nous n'en prenons qu'un seul, nous avons déjà statué que c'était trop peu. Puisque l'unité de base est de 8 bits, ce serait une bonne séparation. Par contre, il faudrait 256 symboles pour ce faire. Nous pourrions utiliser les dix chiffres et les 26 lettres, mais cela ne nous mène qu'à 36 symboles. Ce n'est donc pas viable. Si nous coupons au demi-octet, cela fait 4 bits pour 16 symboles. Là, c'est intéressant! Les symboles sont alors 0123456789ABCDEF avec A qui vaut dix et F qui vaut quinze. C'est la base hexadécimale qui est une base de seize.

Avec cette nouvelle base, pour représenter un nombre de 32 bits, nous n'avons de besoin que de 8 caractères au lieu de 32. Le problème de tantôt revient à dire FF + 1 = 00. C'est beaucoup plus court, surtout si nous décidons d'ajouter un octet pour résoudre notre problème en ayant 00FF + 1 = 0100.

91 https://r.foilen.com/f-bin-hex : Vidéo explicative sur comment faire la conversion en binaire et hexadécimale

22 Les IP réservés

Lorsque nous sommes connectés à Internet, une adresse IP publique nous est assignée. Par contre, sur le réseau local dans notre maison, il ne faut pas utiliser des adresses qui pourraient entrer en collision avec les autres. C'est pourquoi il y a certaines plages qui sont réservées pour toute sorte de réseaux non publics.

Les IP suivants pointent toujours sur l'ordinateur local et ne sont pas accessibles par aucun autre ordinateur.

127.0.0.0 à 127.255.255.255

Les IP suivants sont réservés pour les réseaux locaux et ne sont pas accessibles à partir de l'Internet.

10.0.0.0 à 10.255.255.255

169.254.0.0 à 169.254.255.255 (utilisées par des configurations automatiques sur un réseau sans DHCP)

172.16.0.0 à 172.31.255.255

192.168.0.0 à 192.168.255.255

Plus en profondeur

23 Cryptographie avancée

23.1 Introduction

Après avoir lu le chapitre 16 (La cryptographie) qui est un bon tour d'horizon à la cryptographie et ce qui est possible de faire avec, il y a quand même des détails plus techniques et intéressants à explorer si vous désirez vous servir correctement de cette technologie en tant qu'utilisateur ou que développeur de logiciels. Nous verrons donc les différents algorithmes, leurs forces et leurs limites en passant par les attaques que nous désirons prévenir ou non.

23.2 Les types d'algorithmes de cryptographie

23.2.1 Introduction

Il y a plusieurs méthodes pour crypter des messages et c'est cette diversité qui en fait la richesse et qui peut être une source d'ennuis, car il faut choisir le bon algorithme selon la tâche à accomplir. Il faut tenir compte de la vitesse d'exécution, des attaques que nous désirons contrer et de la complexité de chaque solution. Par exemple, si le but est de décoder un visionnement vidéo en temps réel et que le décryptage prend une journée pour une vidéo d'une heure, c'est loin d'être du temps réel. C'est pourquoi il y a toujours des compromis à faire.

Chaque algorithme s'appelle un algorithme de cryptographie et il détaille la méthode pour crypter et décrypter ainsi que la gestion des clés (symétrique ou non, quelle grandeur, etc.). Tous les algorithmes ont un point en commun et c'est qu'ils utilisent l'opération XOR (un « ou » exclusif). Je vais vous expliquer ce qu'est cette opération, mais tout d'abord, en tant qu'humain, l'opération la plus simple à utiliser pourrait être l'addition ainsi que sa contrepartie, la soustraction. Alors si je vous donne comme message « 12345 » et que je vous dis de le crypter avec le texte « 69184 », vous devrez additionner non pas les deux nombres ensemble, mais uniquement les chiffres un à un et s'ils dépassent 10, en soustraire 10. Donc 1+6=7; 2+9=11=1; 3+1=4; 4+8=12=2; 5+4=9 ce qui donne « 71429 ». Puis pour décrypter vous faites l'inverse : 7-6=1;1-9=-8=2; 4-1=3; 2-8=-6=4; 9-4=5 et cela revient au « 12345 » du début.

Comparaison OR

Bit #1	Bit #2	Résultat
0	0	0
0	1	1
1	0	1
1	1	1

Comparaison XOR

Bit #1	Bit #2	Bit #3
0	0	0
0	1	1
1	0	1
1	1	0

Maintenant, pour un ordinateur qui utilise des bits (valeur 0 ou 1), une opération très simple est le XOR. Un simple OR (un « ou » en français) est de dire que si l'un des deux bits à comparer est un 1, alors la réponse est 1 (voir le tableau à droite). Le problème est que trois des quatre possibilités donnent 1 et une seule des quatre donne 0. C'est un problème étant donné que si nous avons comme résultat un 1 et comme texte de cryptage un 1, le message initial pourrait être 0 ou 1, mais nous voulons une seule valeur puisque le message décrypté est soit une valeur ou l'autre et non les deux. Le XOR (« ou » exclusif) stipule que si les deux bits comparés sont différents, alors la réponse est 1; sinon elle est 0 (voir le tableau à droite). Avec cette opération, il y a deux possibilités sur quatre pour

chaque résultat et cela nous donne une seule réponse possible. Il est donc faisable de cacher un message en faisant un XOR entre le message à crypter et le texte de cryptage.

Avant de continuer plus loin, voici un exemple. Prenons le message « 11010 » et cryptons-le (faisons un XOR) avec le texte « 01111 » : 1 XOR 0 = 1; 1 XOR 1 = 0; 0 XOR 1 = 1; 1 XOR 1 = 0; 0 XOR 1 = 1. Le tout « 10101 ». Comme dit précédemment, cette opération est très rapide pour une machine, mais il y a aussi une autre propriété importante et c'est que contrairement à l'addition dont son inverse est la soustraction; pour le XOR, son inverse est encore XOR. Donc en prenant le message crypté et en faisant un XOR avec, nous obtenons le message initial : « 10101 » XOR « 01111 » = « 11010 ».

23.2.2 Sécurité parfaite avec le masque jetable (One Time Pad)

Pour crypter un message de façon sécuritaire, il faut que le résultat final ait l'air aléatoire. Si tel n'est pas le cas, par exemple si notre algorithme ne fait qu'additionner 1 sur toutes les positions, il sera assez aisé de le découvrir en analysant le résultat. L'analyse est souvent possible, car nous pouvons nous attendre à un certain type de message initial particulier comme un texte. Si le texte est en français, tous les espaces seront cryptés avec la même valeur et comme les espaces sont fréquents, la valeur cryptée qui se répétera le plus pourra être considérée comme étant l'espace. En faisant une analyse des lettres les plus fréquentes dans cette langue, la seconde valeur la plus fréquente pourra être découverte et ainsi de suite. Bien sûr, chaque texte étant différent, il y aura de mauvaises valeurs trouvées, mais ensuite, ce sera le jeu du pendu pour découvrir le message.

Revenons à la première phrase : « pour crypter un message de façon sécuritaire, il faut que le résultat final ait l'air aléatoire». La meilleure façon d'être aléatoire est l'utilisation d'un masque jetable. Le principe est simple : **créer une clé réellement aléatoire aussi grande que le message à crypter** et faire un XOR entre les deux. Étant donné que le texte de cryptage est totalement aléatoire, peu importe le message initial, le résultat sera aléatoire. **Ceci est le seul moyen d'avoir une sécurité parfaite** et tous les autres algorithmes qui suivront vont tenter d'être le plus sécuritaires possible sans jamais atteindre ce but.

La difficulté ici, c'est d'avoir une clé réellement aléatoire. Tous les programmeurs connaissent une méthode pour générer des nombres aléatoires dans leurs langages de programmation préférés, mais celle fournie n'est pas complètement aléatoire, mais bien **pseudo-aléatoire** (voir l'encadré plus bas). Ce terme signifie que la méthode de génération tente d'avoir une distribution uniforme (ne pas préférer un nombre plus qu'un autre) et d'avoir l'air aléatoire (pas simplement des nombres qui se suivent). Pourquoi ne pas avoir une méthode de génération réellement aléatoire plutôt que de faire des calculs mathématiques pour avoir juste l'air aléatoire? Parce qu'un ordinateur, c'est par définition déterministe, ce qui signifie qu'il va toujours donner le même résultat avec les mêmes paramètres donnés en entrée.

La **génération de nombres aléatoires est quand même possible** avec un ordinateur et tous les systèmes d'exploitation fournissent ce genre de méthodes. Pour réussir cet exploit, plusieurs données différentes vont être utilisées pour ajouter de **l'entropie** aux valeurs générées. Cette entropie est une foule de données qui doivent changer d'une utilisation à l'autre de l'ordinateur. Par exemple, l'heure courante peut être additionnée aux positions du curseur de la souris dans l'écran, aux positions des fichiers temporaires écris sur le disque dur et au niveau d'utilisation du CPU courant. Faire un XOR entre cette valeur et une valeur pseudo-aléatoire donnera un résultat qui n'est pas possible de déduire par une suite mathématique puisqu'elle n'est plus qu'une simple suite, mais devient dépendante de l'état de l'ordinateur au moment de la création de la clé.

Le masque jetable est dit « jetable », car **il ne faut jamais utiliser la clé générée pour crypter plus d'un seul message**, sinon il sera possible de faire de l'analyse position par position. Par exemple, si la même position sur plusieurs messages est la même valeur cryptée, cela donne comme indice que la valeur initiale à cette position est la même pour tous ces messages. Il peut sembler pratiquement impossible de faire ce genre d'analyse, mais il ne faut pas oublier que les ordinateurs sont capables de faire énormément de calculs rapidement. Il faut voir ce travail d'analyse comme un Sudoku. Ceux qui ont essayé ce jeu savent qu'avec très peu d'information, il est possible de remplir la grille par plusieurs techniques de déductions. C'est la même chose avec la cryptanalyse où chaque indice rapproche de la solution.

Pour terminer, puisque le masque jetable est le seul moyen de crypter un message avec une sécurité parfaite, pourquoi ne pas arrêter ici et ne pas se bourrer le cerveau de toutes les autres méthodes qui ne veulent que s'approcher de ce résultat? En revenant à la génération de la clé, la réponse saute aux yeux : « elle doit être aussi grande que le message à crypter ». Donc un DVD crypté aurait besoin d'un second DVD pour décrypter le premier ou utiliser moitié moins d'espace en enregistrement vidéo pour à la place avoir la moitié de l'espace gobée par la clé. Crypter un disque dur en entier lui ferait aussi perdre la moitié de sa taille utile. C'est pourquoi les autres algorithmes vont utiliser des trucs pour réussir à

Générateur de nombres pseudo-aléatoires

Il existe plusieurs algorithmes pour créer des nombres qui ont l'air aléatoires. Certains sont très rapides et simples tandis que d'autres sont plus complexes. Pour des applications normales, comme des jeux vidéos, où les choix aléatoires n'ont pas besoin d'être parfais, mais uniquement différent d'une fois à l'autre, tel le cas d'un générateur de labyrinthe, il n'est pas grave d'avoir une simple suite mathématique qui mimique un choix au hasard. Par contre, quand l'application a besoin de **sécurité** en cryptant des données, **il ne faut pas que les résultats du générateur puissent être distingués statistiquement d'une véritable séquence de nombres aléatoires**.

Qu'il soit sécurisé ou pas, le générateur va toujours prendre une **valeur initiale (seed)** où commencer à choisir ses nombres dans sa suite. Sans ce nombre, à chaque démarrage de l'application, ce serait toujours le même nombre qui serait choisi en premier et les suivants seraient aussi toujours les mêmes dans le même ordre. C'est pourquoi dans une application, la valeur initiale donnée est souvent l'heure courante à la seconde ou milliseconde près. Utiliser cette valeur dans un logiciel de cryptographie serait insensé puisque ce serait trop simple de découvrir cette valeur selon l'heure que le message a été crypté. Par contre, **pour les générateurs sécuritaires**, le fait que la même valeur initiale redonne la même suite de nombres aléatoires est important puisque **la valeur initiale sera la clé**. Ainsi il sera possible avec une clé précise de générer la même suite de nombres autant lors du cryptage et du décryptage.

avoir la plus petite clé possible avec la meilleure sécurité possible.

23.2.3 xxx Un semblant de OTP avec les chiffrements de flux (Stream Ciphers)

Puisque la solution parfaite pour crypter est d'avoir un bloc de nombres réellement aléatoires aussi longs que le message à crypter, la solution presque parfaite serait de trouver une façon de générer un bloc qui a l'air d'un bloc réellement aléatoire lors de du cryptage et de pouvoir le regénérer lors du décryptage. Le but ici est de partir d'une petite valeur (la clé) qui est utilisée comme valeur initiale (seed) et que les générateurs de nombres aléatoires sécuritaires (tel qu'expliqué dans l'encadré plus haut) vont utiliser pour créer le bloc de nombres.

L'algorithme ici s'appelle un chiffrement de flux parce que les nombres ne sont pas tous générés avant de crypter ou décrypter puisque cela pourrait utiliser trop de mémoire. À la place, ils sont créés au fur et à mesure qu'il avance dans le message. Ainsi, si le message fait 10 caractères de long, voici les étapes qui sont faites :

1. Initialiser le générateur sécuritaire de nombres aléatoires avec la clé;
2. Obtenir la première valeur aléatoire et l'utiliser (en XOR) sur le premier caractère du message;
3. Obtenir la valeur aléatoire suivante et l'utiliser (en XOR) sur le second caractère du message;
4. Continuer ainsi jusqu'au 10^{e} caractère.

Les points importants à considérer dans le choix d'un des algorithmes au lieu d'un autre sont :

- La taille de la clé : Nous désirons la plus petite possible pour sauver de l'espace, mais qu'il y en ait suffisamment pour que l’essaie de toutes les clés possibles ne se fasse pas trop rapidement;

Ce qui fait qu'un générateur de nombres aléatoires est considéré comme sécuritaire :

- La sortie ne peut pas être distinguée d'une sortie réellement aléatoire;
- La sortie ne peut pas être prédite : en connaissant plusieurs nombres de la suite, il n'est pas possible de prédire les prochains nombres de cette même suite.

Voici quelques algorithmes de chiffrement de flux :

- RC4 (HTTPS et WEP)
- CSS (DVD, GSM, Bluetooth)
- eStream : Salsa 20 (uses a nonce)

Limitations :

- Une seule fois par clé xxxx
- Pas d'intégrité

23.2.4 xxx Les block ciphers

23.3 xxx Les modes d'opérations des block ciphers

23.4 xxx L'intégrité des messages avec MAC

23.4.1 xxx Introduction

23.5 xxx Authenticated Encryption

23.5.1 xxx Introduction

23.6 xxx Les attaques

23.6.1 xxx Introduction

Brute force

Two times pad

No integrity

Meet in the middle

On the implementation : side channel attacks, linear and differential attaks, Quantum attacks

Chosen-plaintext attack (CPA)

Predict IV

24 Cryptographie avec RSA - Son fonctionnement

24.1 Introduction

La cryptographie RSA est une méthode asymétrique avec une clé privée et une autre publique. Ces clés sont basées sur des choix de gros nombres premiers. Ce qui fait de cet algorithme un qui est sécuritaire est la difficulté à factoriser de gros nombres et c'est ce processus qui permettrait de décrypter un message codé. Plus les ordinateurs deviennent rapides, plus il faut choisir de gros nombres premiers.

24.2 La méthode RSA

Pour réussir à faire une clé asymétrique, il faut trouver une fonction qui n'est pas réversible et dont la seule façon de la craquer est par force brute. Cette façon consiste à essayer toutes les possibilités et elle doit être très longue à arriver au résultat. Par exemple, si ça prend 15 ans à décrypter votre mot de passe de compte bancaire, il se peut que vous ailliez déjà fermé ce compte ou changé le mot de passe entre-temps. Cette fonction doit donner plusieurs résultats identiques, ainsi, il n'est pas possible de partir du résultat pour obtenir la valeur initiale. Par exemple, pour une multiplication par 2, la fonction inverse

est la division par 2. Pour RSA, la fonction est le restant d'une division que nous appelons le modulo. Par exemple, si nous divisons 5 par 3, cela donne 1.6667, ce qui est 1 avec un restant de 2 ($1 \cdot 3 + 2 = 5$). Regardons les graphiques des deux fonctions :

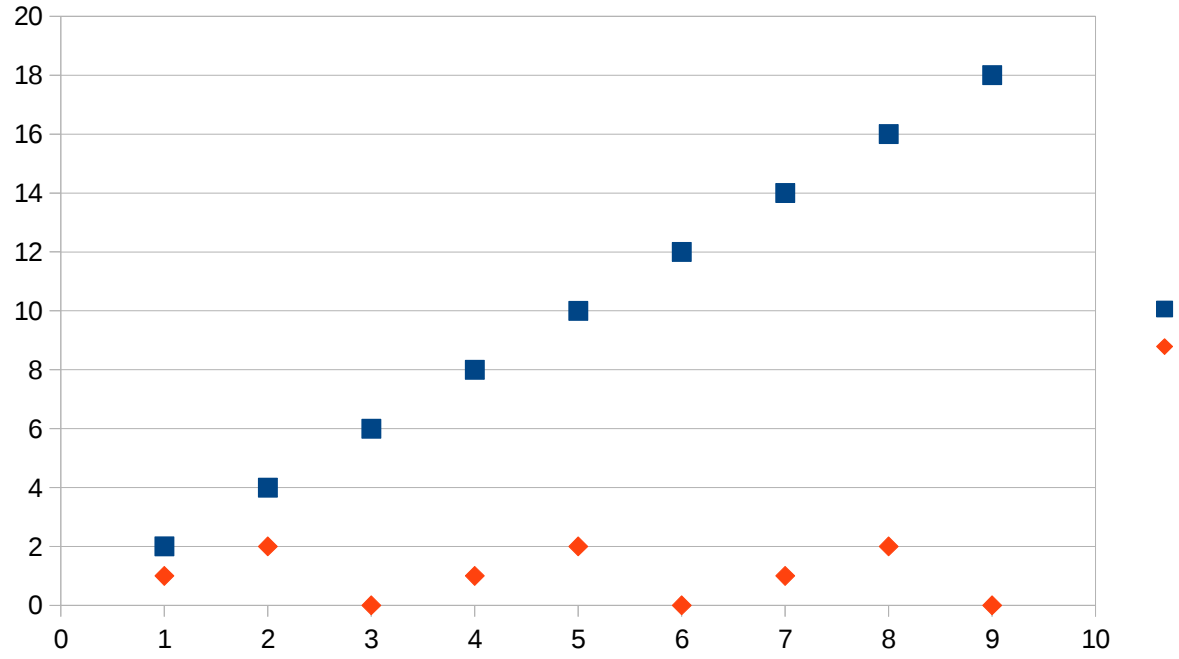

Illustration 34: RSA : Graphique d'une fonction modulo

Nous pouvons voir que la multiplication par 2 donne toujours une seule réponse : si nous partons de la valeur 4, le nombre initial se doit d'être 2.

Nous pouvons aussi voir que le modulo 3 va de 0 à 2 et recommence plusieurs fois. Ainsi, pour la valeur 2, nous avons une infinité de nombres initiaux possibles tels 2, 5, 8, etc. Cette fonction est donc irréversible.

Par contre, il ne suffit pas d'utiliser un modulo pour avoir une fonction cryptographique. La méthode RSA prend le message, lui ajoute une puissance définie dans la clé et à ce résultat, le modulo d'une valeur définie dans les deux clés est appliqué. De plus, cette méthode est faite pour que la fonction de cryptage soit la même que celle de décryptage. Ce qui change, c'est simplement de mettre le message crypté au lieu du message et la puissance définie dans la seconde clé (vous verrez ces équations dans la prochaine partie). Pour faire fonctionner les deux exposants avec le même modulo, le choix de toutes ces valeurs est fait en fonction de deux énormes nombres premiers avec lesquels toutes les valeurs sont calculées.

24.3 Utilisation

24.3.1 Crypter un message

$$Crypté = (Message)^{e} \bmod N$$

où

- *Crypté* est le message crypté

- *Message* est le message initial à crypter
- *e* est la puissance de cryptage
- *mod* est le restant de la division par *N*
- *N* est la multiplication de deux nombres premiers

La clé publique contient donc les valeurs *e* et *N*.

24.3.2 Décrypter un message

$$Message = (Crypté)^{d} \bmod N$$

où

- *Message* est le message crypté une fois décrypté
- *Crypté* est le message crypté
- *d* est la puissance de décryptage
- *mod* est le restant de la division par *N*
- *N* est la multiplication de deux nombres premiers

La clé privée contient donc les valeurs *d* et *N*.

24.4 Le fonctionnement mathématique

La démarche mathématique est assez ardue pour expliquer comment trouver les valeurs de *e*, *d* et *N*, alors cette partie n'est pas montrée dans ce document. Ce que vous verrez est comment trouver les différentes valeurs utiles.

24.4.1 Trouver le modulo N

Pour commencer, il faut choisir deux énormes nombres premiers qui sont appelés *p* et *q*. Plus ils sont gros, plus ils seront longs à trouver et à essayer lors des essais pour craquer les clés.

$$N = p \cdot q$$

L'important ici est que la grosseur maximale du message doit être inférieure à *N* puisqu'après ce nombre, il y a une boucle à cause du modulo. Par exemple, si *p* et *q* sont 3 et 5, *N* sera 15. Puis si nous voulons crypter les lettres de l'alphabet, il y a 26 lettres. Nous ne pouvons pas utiliser un *N* de 15 puisqu'il est plus petit que 26. Par contre, 5 et 7 peuvent être utilisés puisque le *N* est alors de 35.

24.4.2 Trouver les puissances e et d

Trouver le modulo N était assez simple à expliquer (malgré qu'il ne soit pas simple à programmer), mais pour trouver les puissances, il faut faire quelques calculs préalables.

Pour commencer, nous avons besoin d'une valeur intermédiaire *r* qui est définie comme suit :

$$r = (p-1) \cdot (q-1)$$

Ensuite, la formule qui lie *e* et *d* au modulo *N* (et en même temps aux deux nombres premiers *p* et *q*)

est :

$$(e \cdot d) \bmod (r) = 1$$

Ce qui nous intéresse ici est de trouver la multiplication de *e* et *d* et puisqu'elle est gouvernée par un modulo, il y a plusieurs possibilités de *e* et *d*. Pour trouver toutes ces valeurs, il faut retourner à la base de ce qu'un modulo représente :

Si nous prenons un nombre *a* que nous désirons diviser par *b*, nous obtenons un quotient *q* et un restant *t*.

$$a = b \cdot q + t$$

Par exemple, le nombre 5 divisé par 3 donne un quotient de 1 et un restant de 2 :

$$5 = 3 \cdot 1 + 2$$

Si nous voulons 5 modulo 3, nous voulons savoir le restant, ce qui revient à dire :

$$a \bmod b = (b \cdot q + t) \bmod b = t$$

La partie *bq* est simplement enlevée du nombre et cela nous donne le restant. En termes plus clairs, pour trouver les valeurs de *a*, il suffit de prendre le diviseur *b*, de le multiplier par un entier *q* et d'ajouter le restant désiré. Dans notre cas,

$$(e \cdot d) \bmod (r) = 1$$

où

$$a = e \cdot d \; ; b = r \; ; t = 1$$

Nous avons donc

$$\begin{aligned} a &= b \cdot q + t \\ (e \cdot d) &= r \cdot q + 1 \end{aligned}$$

Et il suffit d'essayer plusieurs combinaisons de *q*. Il faut en essayer plusieurs puisque le nombre a d'autres contraintes :

- Le nombre ne doit pas être un nombre premier puisqu'il ne sera pas factorisable en *e* et *d*
- Les deux facteurs ne doivent pas être identiques puisque sinon la clé privée sera la même que la clé publique

Pour terminer, vous trouvez les valeurs de *e* et *d* en les factorisant.

25 Protocoles

25.1 HTTP et HTTPS

25.1.1 Introduction

HTTP et HTTPS sont des protocoles qui sont utilisés en grande majorité pour afficher des pages web. Quand vous écrivez dans votre navigateur « http://etc… », vous demandez justement à votre logiciel d'utiliser ce mode de communication avec le serveur distant. Le même langage est utilisé pour obtenir des fichiers sur le réseau peer-to-peer Gnutella et aussi pour demander des informations sur les personnes qui possèdent un certain fichier Torrent sur un tracker Bittorrent.

25.1.2 Les requêtes HTTP et leurs paramètres

Voici un exemple de demande par un client Firefox :

```
GET / HTTP/1.1
Host: www.google.ca
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.5) Gecko/2008120122 Firefox/3.0.5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Cookie: PREF=ID=fkgiu…
Cache-Control: max-age=0
```

Ce n'est sûrement pas visible, mais une requête se termine toujours par une ligne vide pour dire qu'il n'y a plus d'options supplémentaires à ajouter. La première ligne est la plus importante puisqu'elle définit le document à prendre sur le serveur. Dans ce cas-ci, c'est tout simplement la racine « / », mais ce pourrait être « /index.php ».De plus, vous pouvez très bien demander un document avec seulement cette ligne et vous obtiendrez une réponse positive du serveur.

La seconde ligne est optionnelle telle que mentionnée dans le dernier paragraphe, mais lorsque nous parlons de sites web, elle est en pratique indispensable. Cela peut sembler bizarre de dire à google.ca qu'en plus de nous être connectés directement sur lui, nous voulons qu'il sache qui il est. C'est parce qu'un serveur web peut héberger plusieurs sites web avec des noms de domaines différents. Dans la configuration d'Apache[92] (un serveur web), cela s'appelle des hôtes virtuels. Il est donc impératif sur ce genre de serveur de préciser le domaine désiré.

Le restant est une liste de plusieurs paramètres spécifiés par le client. Tous ces champs sont optionnels,

92 https://r.foilen.com/apache-httpd : Site officiel du serveur web Apache

mais certains serveurs peuvent désirer connaitre beaucoup d'informations sur le client pour être certains que ce ne soit pas un robot, mais bien un utilisateur. Nous reviendrons plus tard sur certains points.

Voici un exemple de réponse à cette requête :

```
HTTP/1.1 200 OK
Cache-Control: private, max-age=0
Date: Sun, 11 Jan 2009 18:29:39 GMT
Expires: -1
Content-Type: text/html; charset=UTF-8
Content-Encoding: gzip
Server: gws
Content-Length: 2708
```

La première ligne donne le code 200 avec la signification « OK ». Les autres codes peuvent être 404 pour une page non trouvée par exemple ou encore 403 pour une non-permission d'accès.

Le restant est une description de la ressource. Actuellement, c'est un fichier texte qui est un fichier html. Par contre, il n'est pas envoyé directement, mais est compressé en gzip pour prendre moins de bande passante. En temps normal, cela n'est fait que si le client spécifie qu'il peut recevoir ce type de fichiers avec l'option « Accept-Encoding: gzip » tel que dans notre exemple.

Après ce message, le fichier est envoyé et est d'une longueur de 2.7 Ko.

25.1.3 La gestion de la connexion

Dans les premières versions du protocole, pour chaque demande d'une ressource sur un serveur, une connexion devait être faite, suivie de la requête. À la fin de la transmission, le lien était coupé. Ainsi, lorsque nous demandions une page web qui contenait 5 images, nous devions faire 6 connexions (1 pour la page, 5 pour les images). Ce n'est pas énorme, mais regardez maintenant la grosseur des sites: la page html, le fichier css, les scripts JavaScript, les nombreuses images pour les menus, les fichiers flash, etc. C'est énorme et c'est demandant de faire plusieurs connexions.

C'est pourquoi le client peut spécifier qu'il désire faire plusieurs demandes sur une même connexion en écrivant

```
Connection: keep-alive
```

et en donnant un nombre de secondes

```
Keep-Alive: 300
```

Si le serveur ne supporte pas cela, il peut ajouter dans sa réponse

```
Connection: close
```

pour annuler.

Maintenant que cette nouvelle fonctionnalité est présente, il faut pouvoir savoir quand un fichier est terminé d'envoyé pour ensuite faire d'autres demandes. Il est aussi utile de connaitre à l'avance la taille d'un gros fichier pour pouvoir dire à l'utilisateur un temps d'attente estimé.

25.1.4 La gestion de la taille

Il y a trois façons de connaitre la fin d'un fichier. La première est la déconnexion. La seconde est d'envoyer des parties. La dernière est de donner la grosseur dès le départ.

La première méthode est celle de la paresse suprême et de tout simplement dire que lorsque la connexion se coupe, alors c'est fini. Il y a bien entendu deux problèmes à cela. Le premier est que le keep-alive ne peut pas fonctionner et la seconde est que le client ne peut pas savoir si ce n'est pas tout simplement la connexion qui s'est brisée. Dans ce dernier cas, il se dit que c'est la fin du fichier bien qu'il soit incomplet.

La deuxième technique est celle d'envoyer en plusieurs parties. C'est utile lorsque la grosseur finale du fichier n'est pas connue d'avance comme dans le cas des scripts qui génèrent du contenu dynamiquement. Il suffit au serveur de répondre

```
Transfer-Encoding: chunked
```

et d'ensuite toujours dire la grosseur en hexadécimal du morceau qui suit. Une fois le transfert terminé, le serveur renverra une grosseur de 0 suivi d'un retour de ligne.

La troisième façon est de simplement écrire dans la réponse du serveur une ligne contenant la taille totale des données comme dans l'exemple ci-dessus

```
Content-Length: 2708
```

Dès que le client a lu 2708 octets, il considère le fichier reçu et il est prêt à faire une autre demande.

25.2 SMTP

25.2.1 Introduction

Le transfert des courriels se fait grâce à un protocole d'envoi des messages entre les serveurs et ensuite par deux protocoles différents pour lire les messages. Cette section est uniquement pour expliquer le premier qui est SMTP (Simple Mail Transfert Protocol). Il est ainsi possible d'envoyer un message soit à votre fournisseur de service de courriels ou encore directement au serveur possédant la boite de l'utilisateur.

25.2.2 Les requêtes

En temps normal, si vous utilisez Thunderbird ou Outlook, vous devez choisir le serveur SMTP auquel vous enverrez tous vos messages. Par la suite, ce serveur vérifiera la destination qui est l'hôte suivant l'arrobas. Le serveur peut être découvert en faisant un appel à un serveur DNS (Domain Name Server), mais en demandant le champ « MX » pour Mail Exchange. Pour voir un exemple, allez dans « Démarrer », « Exécuter » et entrer le nom du programme « nslookup ». Une fois dedans, écrivez « set q=MX » et appuyez sur « entrer ». Ensuite vous écrivez le nom d'hôte tel « hotmail.com » et faites « entrer ». Vous verrez ainsi l'adresse des serveurs sur lesquels se connecter avec le protocole SMTP.

Pour vous connecter sur la machine, vous pouvez utilisez la commande « telnet hôte 25 » (le nombre 25 étant le port par défaut sur lequel se connecter). Une fois la communication établie, il suffit d'attendre le message d'accueil. Voici un exemple de message de base envoyé. Le texte commençant par C est pour désigner le client et par S le serveur.

```
S: 220 mx4.hotmail.com Postfix
C: HELO 192.168.1.1
S: 250 OK
C: MAIL FROM:<ti-guy@mavie.com>
S: 250 OK
C: RCPT TO:<ti-guy@hotmail.com>
S: 250 OK
C: DATA
S: 354 Enter mail, end with « . » on a line by itself
C: From: “Guillaume” <ti-guy@mavie.com>
C: To: <ti-guy@hotmail.com>
C: Date: Sat, 24 Jan 2009 20:58:26 -0500
C: Subject: Ceci est un test
C:
C: Héhé!
C: .
S: 250 OK Message accepted for delivery
C: QUIT
S: 221 Bye
```

La première partie en vert sert à se présenter. Ainsi le serveur commence et ensuite c’est le client avec la commande HELO. Cette étape est obligatoire, malgré qu’elle est totalement inutile puisque les serveurs ne font rien avec ce que nous écrivons après le HELO. Vous pouvez donc écrire ce qui vous chante.

Ensuite il y a la partie bleue qui dit de qui provient le message et à qui envoyer le message. Le courriel de provenance n’est jamais vérifié alors il est très facile de se faire passer pour n’importe qui. Vous pouvez donc écrire « bill.gates@microsoft.com » si cela vous chante (je ne crois pas que ce soit véritablement son adresse). Ensuite pour la destination, vous pouvez appeler plusieurs fois de suite la fonction « RCPT TO » pour envoyer le message à plusieurs destinataires. Si vous vous connectez directement sur un serveur comme Hotmail, vous ne pourrez normalement écrire qu'à des adresses appartenant à Hotmail parce que ce ne sont pas des relais comme celui votre service de courriel.

En orange, nous avons le fameux message. Il commence par un « DATA » et fini par une ligne avec seulement un « . ». Tout ce qui est écrit là-dedans est envoyé tel quel. Vous n’avez pas à mettre autant de choses que j’ai mis tel « From », « To », « Date » et « Subject » puisque ceux-ci ne font pas partie de SMTP en tant que tel. Par contre, tous les lecteurs de courriels s’attendent à voir quelques champs pour donner le plus d’informations possible sur le message avant même que celui-ci soit ouvert.

Et pour terminer, il faut dire « QUIT » au serveur et attendre qu’il réponde par « Bye » avant de couper la ligne.

25.2.3 Le problème du protocole

Le plus grave problème est le manque de mécanisme de vérification de l’émetteur. N’importe quelle

machine peut envoyer des courriels à n’importe quel autre serveur. C’est pour cette raison que le SPAM prolifère si facilement sur le net. Il suffit d’écrire d’avance le petit texte complet que nous voulons envoyer et le lancer dans le serveur.

Il y a quand même quelques façons de contrer les SPAM à cette étape. Par exemple, le serveur peut mettre des délais avant chaque réponse de sa part et regarder si le client va attendre avant d’envoyer sa prochaine commande. Si tel n’est pas le cas, le client a été fabriqué rapidement et c’est donc louche. Une autre technique utilisée est de mettre un délai avant de dire « Bye » et voir si la connexion va être coupée avant. C’est encore pour vérifier si le client suit bien le protocole.

26 Les licences d'utilisation

26.1 Introduction

Lorsque nous installons n'importe quel logiciel, celui vient avec une licence appelée EULA qui défini les permissions d'utilisation. Normalement, il faut le lire et l'accepter, mais étant donné une longueur qui peut être assez débile avec des dizaines de pages, la majorité de la population ne fait qu'accepter sans lire. Pour ceux qui décident d'offrir leur applications sous une licence à source libre doivent bien prendre le temps de choisir celle qui répond le mieu à leur besoin parmis les nombreuses possibilités telles Apache, GPL, LGPL, MIT, etc. De plus, si un développeur veut utiliser des librairies libres, il est important qu'il sâche s'il peut l'utiliser commercialement sans donner son propre code source ou s'il veut lui-même faire une librairie libre en en utilisant d'autres, si les licences de ces dernières sont compatibles.

Note importante, je ne suis pas avocat et ce chapitre va montrer les lignes directrices telles qu'expliquées sur les sites qui offrent ces licences. Il pourrait donc y avoir certaines subtilitées ou failles non découvertes dans ces licences. Prenez le temps des les lire vous-même et de lire toute l'information pertinante sur chacun des sites les offrants puisqu'eux ont normalement fait affaire avec des avocats pour les écrire ou les confirmer.

26.2 Sections importantes

Une licence contient plusieurs sections qui définissent différentes utilisations et différentes parties de l'application. Par exemple, ce n'est pas parce que le code source (partie) se veut disponible à tous (utilisation) que les images (autre partie) sont disponibles en dehors de l'utilisation du logiciel courant. C'est souvent le cas avec des remake de moteurs de jeux vidéos populaire comme Doom qui utilise les images du jeu original. Dans ce cas, le moteur réécrit est à source libre, mais comme les images appartiennent à l'éditeur du titre original et qu'il n'est pas associé au projet, celles-ci ne peuvent pas être distribuées avec ce jeu.

Les utilisations sont nombreuses comme les limitations sur les motifs d'usage (interdir l'usage commercial), la distribution (ce n'est pas parce que l'installateur à besoin d'une clée que vous pouvez donner celui-ci à une autre personne, même si elle a sa propre clée), la copie, le transfert à une personne tierce, la garanti offerte, l'attribution du travail, etc.

Les différentes parties peuvent aussi avoir différentes limitations. Les parties les plus communes sont le code source, les binaires (le code source compilé), les images, les librairies, les textes, les traductions, les services en ligne, etc.

Il y a aussi des auteurs malhonnêtes qui vont mettre dans leur licence qu'en acceptant cette licence, vous acceptez qu'un logiciel espion soit installé sur votre ordinateur. Certains diront qu'ils ne sont pas si malhonnêtes puisqu'ils l'écrivent, mais comme c'est souvent caché dans des dizaines de pages et qu'ils savent pertinament que personne ne les lit, c'est tout simplement une pratique croche.

Propriétaire: par utilisateur, par ordinateur, par processeur, limité dans le temps.

Open source: doit fournir la source quand dans le programme, quand une librairie, quand sur le réseau. Logiciels dérivés

Domaine public

Contourner les licences

Compatibilité des licences libres

TODO