

Olivier Ricou

# Les coulisses de l'Internet

Version 2.3 $\alpha$  du 29 avril 2014



# Table des matières

<b>1</b>	<b>La mécanique d'Internet</b>	<b>5</b>
1.1	La topologie . . . . .	5
	Une langue commune . . . . .	6
	Des machines connectées en réseau, des réseaux interconnectés . . . . .	7
	Internet : des milliers de réseaux . . . . .	9
1.1.1	L'information relayée de réseaux en réseaux . . . . .	9
	Un exemple : la route entre deux universités . . . . .	11
	Le calcul du débit . . . . .	13
1.1.2	Des domaines et des noms . . . . .	15
	Trouver l'adresse IP d'un nom, le fonctionnement du DNS . . . . .	19
1.2	La sécurité des communications . . . . .	20
1.2.1	Les faiblesses d'Internet . . . . .	20
	Les communications en clair . . . . .	20
	Des mails trop visibles . . . . .	21
	Le Web sécurisé . . . . .	23
	L'erreur humaine . . . . .	23
1.2.2	La cryptographie . . . . .	24
	Les clés symétriques ou secrètes . . . . .	26
	Les clés asymétriques ou publiques . . . . .	27
	Différents algorithmes de cryptographie . . . . .	28
	Les mathématiques de RSA . . . . .	30
1.2.3	Utilisation de la cryptographie . . . . .	31

---

	Protéger son courrier avec GPG . . . . .	31
	Surfer protégé . . . . .	34
1.2.4	L'authentification . . . . .	34
	Les autorités de certification commerciales . . . . .	35
	Les autorités de certification gouvernementales . . . . .	36
	L'auto certification . . . . .	38
	Les réseaux de confiance . . . . .	38
1.2.5	La sûreté de la cryptographie . . . . .	39
	La force brute . . . . .	39
	L'intelligence contre la cryptographie . . . . .	42
	La bêtise contre la cryptographie . . . . .	44
1.3	Plus . . . . .	45
	À propos de l'architecture d'Internet . . . . .	45
	À propos de la sécurité . . . . .	45
	À propos de la cryptographie . . . . .	46

# Chapitre 1

## La mécanique d'Internet

Ce premier chapitre est le chapitre technique du livre. Il est divisé en deux parties, la première présente les bases du fonctionnement d'Internet qu'il est nécessaire de lire pour comprendre où est le pouvoir et la spécificité technique de ce réseau. La seconde partie introduit la sécurité avec quelques équations mathématiques afin de démystifier la cryptographie. Cette seconde partie n'est pas nécessaire pour la suite du livre mais étant donné l'importance qu'à Internet aujourd'hui dans nos vies, il est vivement recommandé d'avoir quelques notions de sécurité pour ne pas être victime de méfaits<sup>1</sup>

### 1.1 La topologie

La grande force d'Internet est de permettre aux machines de communiquer entre elles. Historiquement d'autres systèmes ont permis la même chose, mais avec Internet la sauce a pris.

Aujourd'hui l'internaute sait qu'il peut communiquer avec 1 milliard de personnes grâce à Internet et, a priori, peu lui importe de savoir comment. Mais pour celui qui désire comprendre les enjeux liés à Internet, tant en interne lorsqu'on parle de gouvernance qu'en externe lorsqu'on compare la téléphonie sur IP à la téléphonie classique par exemple, il est nécessaire de connaître les bases techniques sur lesquelles Internet est construit.

Dans son principe, la mécanique d'Internet est simple. Elle est basée sur deux notions :

1. un empilement de protocoles de communication avec au milieu une *langue* commune composée des protocoles TCP et IP<sup>2</sup>, voir l'encart page 6,
2. la connexion de machines en réseaux et l'interconnexion des réseaux.

---

<sup>1</sup>Les responsables de la sécurité des systèmes d'information en France, l'ANSSI, qui connaissent les menaces réelles et voient le manque de protection des internautes, vivent un cauchemar quotidien.

<sup>2</sup>pour simplifier, il existe aussi UDP sur IP utilisé pour la vidéo par exemple et d'autres nettement moins utilisés.

## Une langue commune

Le premier point souligne le fait que toutes les machines connectées à Internet parlent la langue informatique commune qu'est TCP/IP<sup>3</sup>. Outre l'aspect d'une langue commune, l'utilisation de TCP/IP impose une numérotation unique des machines, comme il existe une numérotation des téléphones. Cette numérotation est appelée l'adresse IP<sup>4</sup> et se présente sous la forme de 4 nombres inférieurs à 256 séparés par des points comme 134.157.1.12.

Ainsi deux ordinateurs respectant le protocole TCP/IP peuvent se contacter et communiquer si il existe une liaison entre eux.

### TCP/IP, le protocole d'Internet

Les informaticiens ont découpé les communications, entre deux machines ou entre deux programmes, en couches avec le principe que chaque couche communique seulement avec les deux couches l'encadrant. Le modèle de référence des informaticiens fait intervenir 7 couches allant de la couche physique, comment transmettre des 0 et des 1 avec du courant électrique ou des photons, à la couche applicative sur qui définit le protocole de communication d'un programme.

Internet réduit le nombre de couches mais le principe reste le même. Il impose seulement d'utiliser le tronc commun que sont la couche de transport, TCP ou UDP, et la couche réseau qu'est IP. C'est la raison pour laquelle on associe Internet au protocole TCP/IP.

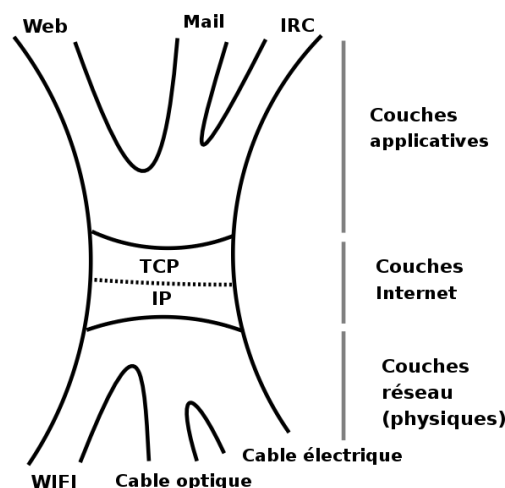


FIG. 1.1 : TCP/IP au cœur du protocole de communication d'Internet

Ainsi les supports physiques et leur protocole peuvent varier sans avoir d'impact sur la compatibilité Internet. Ce modèle permet aussi de définir tous les protocoles applicatifs désirés tant qu'*in fine* leur couches applicatives peuvent se raccorder à la couche de transport. D'où la possibilité de créer toutes les applications imaginables.

<sup>3</sup>en informatique on parle de *protocole*.

<sup>4</sup>Ip version 4, la version encore la plus répandue. Pour une brève description de la version suivante, IP version 6, voir l'encart page 8.

## Des machines connectées en réseau, des réseaux interconnectés

Le second point souligne la structure d'Internet : Internet est une interconnexion de réseaux indépendants, cf figure 1.2. L'internaute à la maison est sur le réseau de son fournisseur d'accès, réseau Vert, et au travail, il est sur le réseau de son entreprise, réseau Bleu. S'il peut se connecter de chez lui au travail, c'est qu'il existe une connexion directe ou indirecte entre ces deux réseaux.

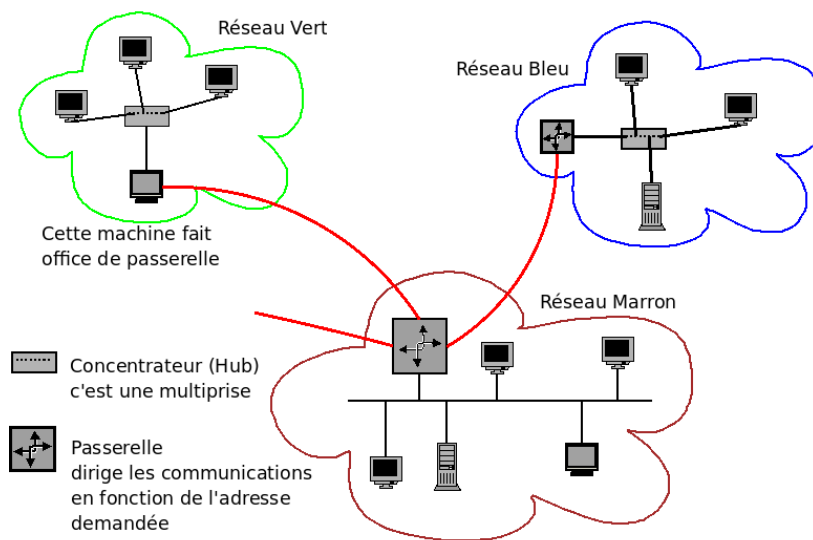


FIG. 1.2 : Des réseaux interconnectés.

*La connexion entre les réseaux passe par des machines spéciales très importantes puisque permettant l'accès aux autres réseaux donc à Internet. Il s'agit des passerelles.*

Parfois une machine fait partie d'un réseau qui, lui-même, fait partie d'un gros réseau auquel est relié le reste d'Internet. Ainsi avec le même dessin, les réseaux Vert et Bleu peuvent faire partie du réseau Marron, le réseau Marron étant alors relié au reste d'Internet. C'est souvent le cas dans les réseaux d'entreprise où chaque département a son réseau (Bleu ou Vert) mais doit passer par celui du service informatique (Marron) pour sortir sur Internet. On retrouve cette notion de sous-réseaux au niveau de l'adresse IP dans l'ordre des 4 nombres. Le premier nombre indique une zone, le second une sous-zone... comme 33 1 42 37 xx xx indique que ce téléphone est en France, dans la région parisienne, à côté de la Croix de Berny (237 étant BER). Mais la comparaison se limite là car l'adressage IP est plus souple, les sous-réseaux n'ayant pas obligatoirement le même préfixe que le réseau auquel ils appartiennent et surtout l'adresse IP n'est pas géographique. D'ailleurs l'attribution des numéros de téléphone a aussi évolué et n'est plus basée sur la position géographique.

### Le danger de l'analogie avec le téléphone

```

Blgron : jve te tracé ac ton ip
Nonoeil : Cool.
Blgron : tu va voir
Nonoeil : Oui. Je vais voir, comme tu dis.
Blgron : putl sa marche pa!!! ta 1 brouyeur????
Nonoeil : Mais qu'est-ce qu'il dit, l'autre? Qu'est-ce qui ne marche pas?
Blgron : sa sonne mm pa che toi
Nonoeil : ça sonne? Je suis au boulot là, tu vas tomber sur le Central,
          si t'appelles, mon rigolo
Blgron : ok alors le central a 1 brouilleur
Nonoeil : le central a ce qu'il veut en même temps
Blgron : jvé toser cher a coze de ses coneries
Nonoeil : Ouais ouais. Si tu le dis!
Myrdène : Mais attends... T'as composé son numéro IP sur ton portable,
          Blgron?
Blgron : ui pk???

```

source : Les perles d'IRC, [www.danstonchat.com](http://www.danstonchat.com)

Ainsi une entreprise connue possède les adresses IP qui commencent par 129.42. Il est probable qu'elle a distribué à ses départements des sous-zones comme 129.42.2.xxx pour le département Vert, 129.42.3.xxx pour le Bleu etc...

Si le département Bleu s'achète une connexion directe vers Internet qui ne passe pas par le réseau Marron, alors cela lui offre deux façons de se connecter à Internet. Il est fort probable que les responsables du réseau n'apprécient guère car ils ne pourront plus filtrer toutes les communications entre l'entreprise et Internet, ce qui rendra d'autant plus difficile la protection du réseau.

### IPv6

La nouvelle version d'IP est la version 6, déjà en activité même si l'ancienne version, la version 4, reste la plus courante. La version 6 a été créée afin principalement de répondre au manque d'adresse IPv4 pour tout le monde. Avec 128 bits par adresse, la version 6 offre  $2^{128} = 3,4 \cdot 10^{38}$  adresses ce qui fait 670 millions de milliards d'adresses par millimètre carré sur la Terre.

48 bits	16 bits	64 bits
Préfix	Sous-réseau	Interface

TAB. 1.1 : Format des adresses d'IPv6

Une adresse s'écrit en hexadécimal (contrairement à IPv4). Ainsi par exemple on a 2001:0db8:0000:85a3:0000:0000:ac1f:8001 ce qui peut aussi être écrit en supprimant les zéros non significatifs 2001:db8:0:85a3:0:0:ac1f:8001 voire 2001:db8:0:85a3::ac1f:8001. Pour cet exemple, le préfix est 2001:0db8:0000, le sous-réseau 85a3 et l'interface ::ac1f:8001.



### Des adresses à usage privé

Comment a-t-on une adresse IP ? En la demandant à celui qui vous fournit la connexion à son réseau. Il vous donnera une adresse parmi celles qui lui ont été attribuées.

Vous pouvez aussi utiliser, sans rien demander, des adresses réservées à usage interne et donc interdites sur Internet. Il s'agit pour la version 4 d'IP de :

- 10.xx.xx.xx pour se faire un très gros réseau local (16 millions de machines),
- 172.16 à 31.xx.xx pour un gros réseau (1 million de machines)
- 192.168.xx.xx pour un réseau moyen (65 000 machines quand même),
- 127.0.0.1 pour désigner votre machine (chaque machine a au moins 2 adresses IP : celle ci qui ne sert qu'à usage interne, l'autre pour communiquer avec l'extérieur.)

Pour IP version 6, les adresses privées appartiennent à l'espace `fc00::/7` (cf RFC4193). En pratique cela revient à choisir comme préfixe `fd` puis à choisir de façon aléatoire l'identifiant global et l'identifiant de sous-réseaux. On a ainsi  $2^{64}$  adresses pour soi et très peu de chances qu'une autre personne ait le même réseau privé.

## Internet : des milliers de réseaux

D'un point de vue topologique, Internet n'est que la duplication en millions d'exemplaires de la figure 1.2. Pour que l'image globale soit correcte, il faut détecter quels réseaux sont reliés à quels réseaux, ce qu'a fait sur une partie d'Internet CAIDA en 2001 en analysant 535 000 nœuds d'Internet et plus de 600 000 connexions, cf figure 1.3.

On retrouve le schéma de l'entreprise à une plus grande échelle. Ainsi un point, une machine, est rattachée à un paquet qui est son réseau local. Ce paquet est le plus souvent rattaché à un groupe qui est celui de son fournisseur d'accès. Ce groupe appartient à un plus grand groupe qui est le réseau du cablo-opérateur<sup>5</sup>. Enfin les cablo-opérateurs ont des interconnexions entre eux.

On voit qu'un tel réseau a des nœuds centraux qui sont ceux qui relient un groupe à son réseau père ou les nœuds d'interconnexion entre les cablo-opérateurs. Ceux qui contrôlent ces nœuds peuvent limiter les communications, les bloquer ou les espionner. Bien sûr un État peut faire de même avec les nœuds qui sont sur son territoire.

### 1.1.1 L'information relayée de réseaux en réseaux

Que l'on envoie un mail à une machine distante ou que l'on récupère une page web, le principe est le même : l'information est découpée en paquets de données et relayée de réseaux en réseaux.

<sup>5</sup>les cablo-opérateurs sont les entreprises qui posent les câbles d'Internet. Les grandes entreprises des télécommunications sont souvent des cablo-opérateurs, cf section [?].

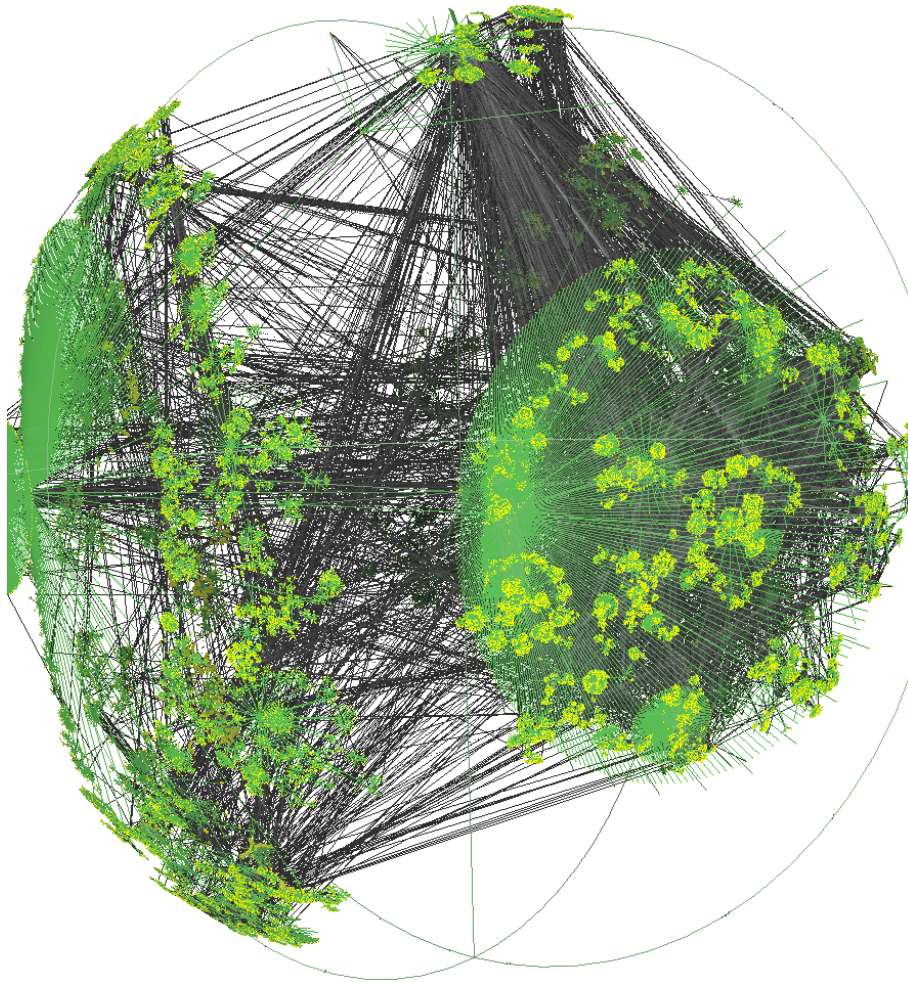


FIG. 1.3 : Une partie d'Internet vue par le logiciel Walrus

*source : CAIDA<sup>6</sup>, mars 2001*

### Traçons la route suivie par les paquets de données

La détection des réseaux et de leur interconnexion peut se faire simplement à l'aide de la commande `traceroute`, mais aussi par le Web à partir de machines qui offrent ce service, cf <http://www.traceroute.org/>. Ce programme permet de suivre la route d'un chemin entre deux machines d'Internet. Si l'affichage produit peut sembler abscons au premier abord, il est en fait relativement simple : chaque ligne représente une machine par laquelle passe le message.

Outre le joli dessin page 10, ce type de commande permet surtout de connaître son environnement et de connaître la qualité de sa connexion à Internet ou au moins aux nœuds d'Internet que l'on considère le plus important.

m.à.j. sur <http://www.ricou.eu.org/e-politique.html>

## Un exemple : la route entre deux universités

Dans l'exemple qui suit, la connexion est établie depuis Jussieu vers le MIT utilise les réseaux universitaires :

```
(mendel) ../home/ricou>tracertoute www.mit.edu
tracertoute to www.mit.edu (18.7.22.83), 30 hops max, 40 byte packets
 1  134.157.204.126 (134.157.204.126)  1.089 ms  1.159 ms  0.971 ms
 2  cr-jussieu.rap.prd.fr (195.221.126.49)  2.728 ms  11.658 ms  2.525 ms
 3  gw-rap.rap.prd.fr (195.221.126.78)  2.317 ms  3.238 ms  2.225 ms
 4  jussieu-g0-1-165.cssi.renater.fr (193.51.181.102)  1.981 ms  2.159 ms  2.119 ms
 5  nri-c-pos2-0.cssi.renater.fr (193.51.180.158)  2.535 ms  2.55 ms  3.054 ms
 6  nri-d-g6-0-0.cssi.renater.fr (193.51.179.37)  1.977 ms  3.465 ms  2.224 ms
 7  renater-10G.frl.fr.geant.net (62.40.103.161)  2.454 ms  3.003 ms  2.814 ms
 8  fr.uk1.uk.geant.net (62.40.96.90)  10.002 ms  10.814 ms  9.274 ms
 9  uk.nyl.ny.geant.net (62.40.96.169)  78.124 ms  78.424 ms  78.166 ms
10  esnet-gw.nyl.ny.geant.net (62.40.105.26)  116.806 ms  88.891 ms  78.344 ms
11  198.124.216.158 (198.124.216.158)  78.261 ms  86.282 ms  95.795 ms
12  nox230gw1-PO-9-1-NoX-NOX.nox.org (192.5.89.9)  83.425 ms  83.591 ms  83.412 ms
13  nox230gw1-PEER-NoX-MIT-192-5-89-90.nox.org (192.5.89.90)  83.675 ms  83.745 ms  83.75 ms
14  B24-RTR-3-BACKBONE.MIT.EDU (18.168.0.26)  83.767 ms  86.358 ms  83.427 ms
15  WWW.MIT.EDU (18.7.22.83)  85.26 ms  85.565 ms  85.304 ms
```

Un essai fait d'une machine chez un fournisseur d'accès commercial français vers une université française fera apparaître la machine passerelle `renater.sfinx.tm.fr` qui sert de passerelle entre Renater et les réseaux commerciaux. Elle est située dans le GIX<sup>7</sup> parisien nommé SFINX qui permet à tous les opérateurs Internet de se relier entre eux suivant leurs accords, dit accords de peering.

Essayons de comprendre le chemin suivi par notre paquet IP entre Jussieu et le MIT. Le premier intermédiaire que notre message va rencontrer est la passerelle de notre réseau. Son adresse IP est 134.157.204.126 comme on le voit sur la ligne numérotée 1. De là on rejoint l'interconnexion entre Jussieu et le RAP, réseau académique parisien, en 2, pour entrer sur le réseau universitaire français, Renater, en 4, cf figure 1.4.

On passe de Renater à Géant, le réseau universitaire européen, en 7, qui nous envoie en Angleterre, en 8, d'où on va à New-York rejoindre le réseau académique d'Amérique du Nord, Internet 2, en 9 et 10, cf figures 1.5 et 1.6.

De là on passe sur NOX, le réseau de la Nouvelle Angleterre, en 12 et 13, pour atteindre le réseau du MIT, en 14, et enfin le serveur web `www.mit.edu`, en 15, cf figure 1.7.

---

<sup>7</sup>Global Internet eXchange point ou IXP, Internet eXchange Point.



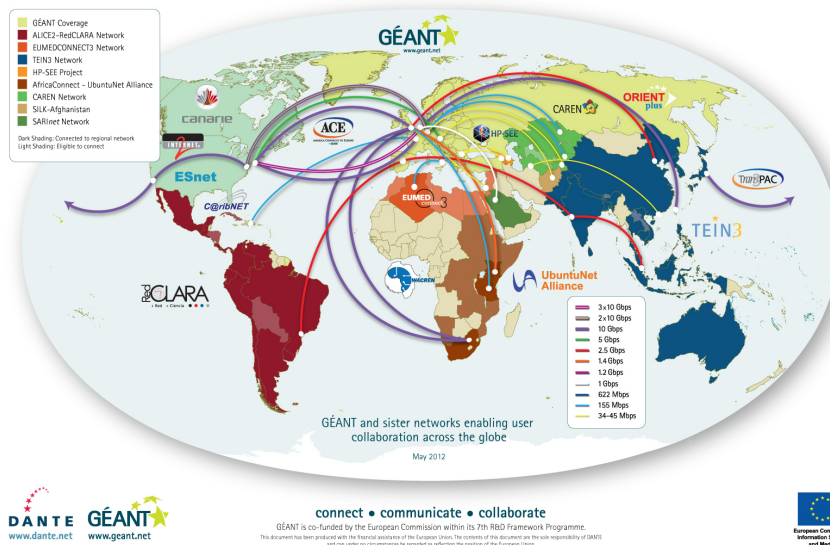


FIG. 1.6 : Interconnection entre Géant et les autres réseaux académiques  
source : Géant, 2012

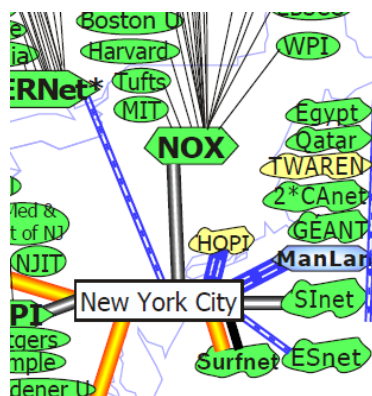


FIG. 1.7 : Internet 2 et NOX pour arriver au MIT  
source : Internet 2<sup>9</sup>, 2005

## Le calcul du débit

Sachant que le débit entre deux machines est celui du nœud le plus faible, si un réseau a un goulot d'étranglement en un point, cela se ressent directement. Aussi il est toujours bon de savoir quels seront vos partenaires principaux et de savoir par quels cablo-opérateurs vous devrez passer. En pratique il faut savoir quels accords<sup>9</sup> a votre hébergeur, avec quels cablo-opérateurs et quelle est l'occupation moyenne du réseau.

Les cablo-opérateurs les plus sérieux proposent de pouvoir suivre en direct la *météo* de leur réseau, malheureusement cette information est rare en France.

<sup>9</sup>en jargon Internet on appelle ça le peering.



Concernant les offres pour les particuliers, ce service est tout aussi rare. Heureusement un site web, la grenouille, publie des données récupérées auprès d'abonnés aux différentes offres mais l'information reste limitée.

- La grenouille, <http://www.grenouille.com/>, est un site indépendant qui donne en temps réel l'état des connexions à Internet des différentes offres commerciales en France,
- L'état du réseau chez Free : <http://www.free-reseau.fr/>
- L'état du réseau chez Nerim : <http://stats.nerim.net/nav/map/>

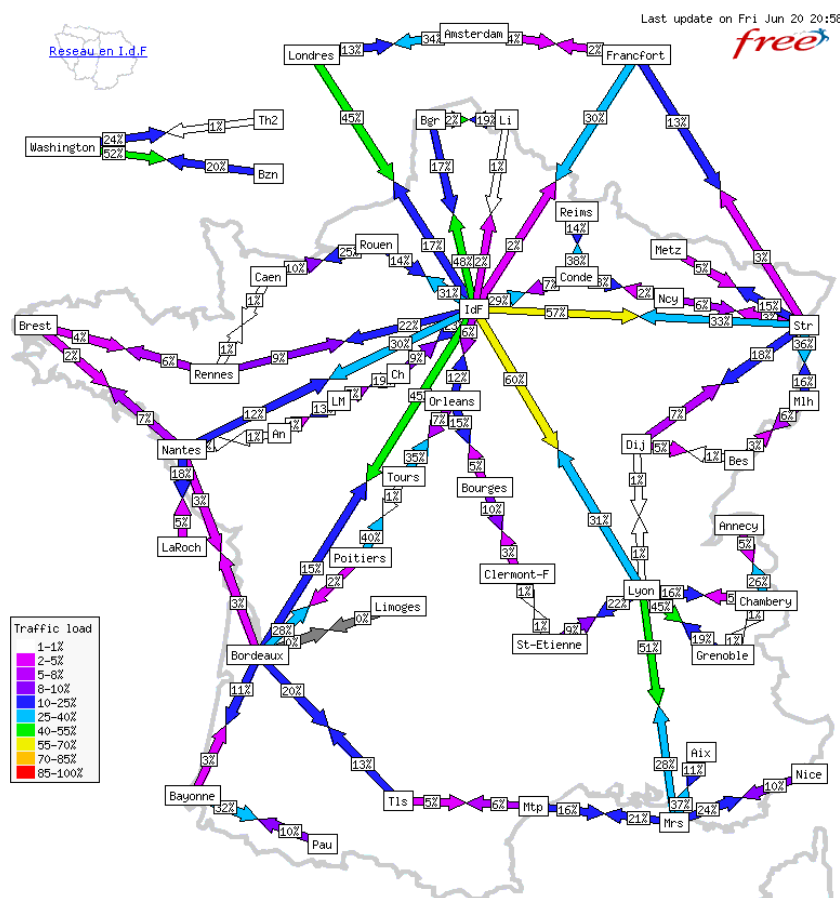


FIG. 1.8 : L'état du réseau national chez Free un vendredi soir

Il est aussi possible de faire le travail à la main avec des outils comme `bing` qui mesurent le débit entre deux machines. Dans l'exemple suivant on teste la connexion entre notre machine à la maison<sup>10</sup> et le site web de Yahoo :

```
# bing 127.0.0.1 www.yahoo.fr
...
```

<sup>10</sup>rappel, 127.0.0.1 est l'adresse de la machine sur laquelle on est.

```
--- estimated link characteristics ---
host                               bandwidth      ms
www.euro.yahoo.akadns.net         718.596Kbps   58.753
```

Si on regarde maintenant le débit entre la machine de notre fournisseur<sup>11</sup> d'accès et le site de Yahoo, on a :

```
# ping 193.252.103.97 www.yahoo.fr
...
--- estimated link characteristics ---
host                               bandwidth      ms
www.euro.yahoo.akadns.net         4.031Mbps     10.116
```

On constate que le débit passe de 718 Kbits par seconde à 4 031 Kbits par seconde ce qui montre que le goulot d'étranglement est la connexion entre la machine à la maison et la machine de notre fournisseur d'accès.

### 1.1.2 Des domaines et des noms

**Historiquement** les machines n'avaient que des numéros, puis devant la croissance du nombre de machines, on a créé un système pour nommer les machines. Cela consistait à avoir sur sa machine un fichier avec les noms et adresses IP de toutes les machines d'Internet. Puis le nombre de machine est devenu trop important et variait trop vite pour garder ce fichier à jour sur toutes les machines. Aussi on a créé un service appelé Domain Name System, DNS, qu'on interroge pour connaître l'adresse IP d'une machine dont on connaît le nom.

La gestion de ce service, et plus précisément de la racine de son arborescence, est à l'origine de nombreuses controverses touchant le contrôle des noms de domaine et leur vente. Elle a mobilisé et mobilise toujours l'Europe et les États-Unis, chacun cherchant à défendre ses intérêts sans pour autant casser Internet ce qui aurait lieu si plusieurs espaces de nommage se faisaient concurrence<sup>12</sup>. Actuellement l'espace de nommage, et donc le DNS, est géré au niveau mondial par l'ICANN et aux niveaux nationaux par les pays concernés.

**En pratique** Internet étant un ensemble de réseaux, il faut une méthode pour nommer les sous-réseaux et y trouver une machine. Pour cela les réseaux et les machines ont des adresses classées dans un système d'arborescence.

---

<sup>11</sup>dont on trouve l'adresse avec `tracert`.

<sup>12</sup>en fait il existe des personnes/groupes ayant créé leurs propres espaces avec leurs propres terminaisons de nom de domaine, TLD, mais c'est marginal.

## L'archéologie des noms de domaines

Un ami qui aime consulter les textes de loi de l'Internet que sont les RFC, Request For Comments, a fait cette constatation :

- RFC 1, avril 1969 : aucune mention des noms des machines, juste les adresses (sur 5 bits)
- RFC 33, février 1970 (remplace RFC 1) : toujours pas de noms, mais les adresses passent à 8 bits
- RFC 229, septembre 1971 : première mention des noms. Aucun mécanisme de résolution n'est envisagé (même pas un simple fichier de correspondances) mais il y a une table des noms officiels et de l'adresse correspondante.
- RFC 606, décembre 1973 : première mention d'un mécanisme de résolution, un fichier, avec une syntaxe formelle, placé à un endroit bien connu, le futur HOSTS.TXT

Le DNS tel qu'il est aujourd'hui arrivera seulement en 1984.

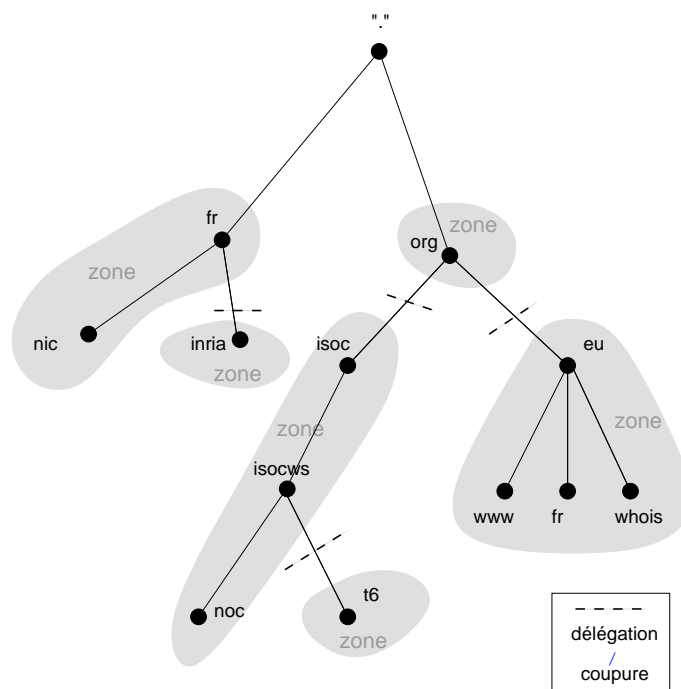


FIG. 1.9 : Une toute petite partie de l'arborescence des noms de domaines

La terminaison la plus à droite sur ce dessin est la machine `whois.eu.org`. Logiquement le nom devrait se lire de droite à gauche avec au début la racine que l'on nomme `."`<sup>13</sup> :

Cette machine appartient au domaine `eu.org`, domaine qui lui même appartient au domaine `.org`.

On imagine bien qu'il ne serait pas gérable que chaque machine soit nommée par une seule

<sup>13</sup>.org est un raccourci accepté pour .org, où le point final est la racine.



## Confessions d'un voleur ou l'argent des noms de domaines

par Laurent Chemla, co-fondateur de la société Gandi

Je vends des noms de domaines sur Internet.

Un peu d'histoire et de technique sont nécessaires pour comprendre à quel point je suis un voleur.

Un nom de domaine, c'est ce qui sert à identifier un ordinateur sur Internet. Quand on vous propose d'aller visiter [www.machinchose.org](http://www.machinchose.org) on vous indique un nom d'ordinateur (www) qui se trouve dans le domaine « machinchose.org » et qui contient ces informations que vous pouvez consulter sur le Web.

Sans un nom de ce genre, un ordinateur ne peut être consulté qu'en utilisant un numéro, tel que par exemple 212.73.209.251. C'est nettement moins parlant et beaucoup plus difficile à mémoriser. Alors pour simplifier on donne des noms aux ordinateurs qui contiennent de l'information publique. Ce qui nécessite, bien sûr, une base de données qui soit capable de retrouver un numéro à partir d'un nom. Et que cette base soit unique et accessible de n'importe où.

Pendant des années, ce système a fonctionné grâce à un organisme de droit public financé par le gouvernement américain. L'Internic (c'était le nom de cet organisme) se chargeait de faire fonctionner la base de donnée, et chacun pouvait y ajouter le nom de domaine de son choix, gratuitement, selon la règle du « 1er arrivé 1er servi ».

Puis vint le temps de l'ouverture d'Internet au grand public (1994), et la fin des subventions gouvernementales au profit du seul marché. Et là, surprise : une agence publique (qui gérait gratuitement ce qu'il faut bien appeler une ressource mondiale unique) fut transformée en entreprise commerciale (Network Solutions Inc, ou NSI), sans que quiconque s'en émeuve particulièrement, et se mit à vendre 50\$ par an (puis 35\$ par an dans un fantastique élan de générosité) ce qui était totalement gratuit peu de temps avant. Et pour son seul profit.

Je dois vous livrer un chiffre qui, s'il n'est pas confidentiel, mérite cependant le détour : le coût réel de l'enregistrement d'un nom dans la base de données mondiale, y compris le coût de fonctionnement d'une telle base, a été évalué il y a deux ans à 0,30\$.

Des chiffres comme ça, je pourrais en donner beaucoup. Je pourrais dire par exemple qu'en estimant le nombre de domaines enregistrés par NSI à une moyenne mensuelle de 40.000, son bénéfice sur les 5 dernières années tourne autour des 80 millions de dollars. Et encore ce chiffre est-il une estimation basse, quand on sait que NSI vient d'être racheté par une autre Net-Entreprise pour la modique somme de 21 milliards de dollars.

Et pourtant, NSI vend du vent, tout comme moi. En fait, nous vendons le même vent.

*source : Extrait d'un article publié dans le journal Le Monde en avril 2000 et disponible dans son intégralité sur <http://www.chemla.org/textes/voleur.html>.*

autorité ne serait-ce que pour des raisons de réactivité et de contrôle. Aussi le nommage d'Internet se base sur un système de délégation de zone. Ainsi `.org` a délégué la gestion de `.eu.org` ce qui fait que `.eu.org` est une zone indépendante de la zone `.org`.

La figure 1.10 montre comment le domaine `eu.org` délègue les sous domaines `gr.eu.org` ou `dk.eu.org` mais conserve la gestion du sous domaine `fr.eu.org` et des machines `www.eu.org` et `whois.eu.org`.

Il existe donc :

- les *domaines* qui comprennent tout ce qui finit par le nom de domaine,
- les *zones* formées de l'ensemble des machines et sous-réseaux contrôlés par le propriétaire du nom de domaine.

On comprend ainsi pourquoi le propriétaire d'un domaine, comme `.fr`, ne peut être tenu responsable de ce qu'on trouve sur un serveur web hors de sa zone, comme `www.tfl.fr` par exemple.

Par contre, techniquement parlant, il peut toujours retirer la délégation de zone et donc fermer le domaine `tfl.fr`. De même le gestionnaire du point final, les États-Unis, peut fermer `.com` ou `.fr`.

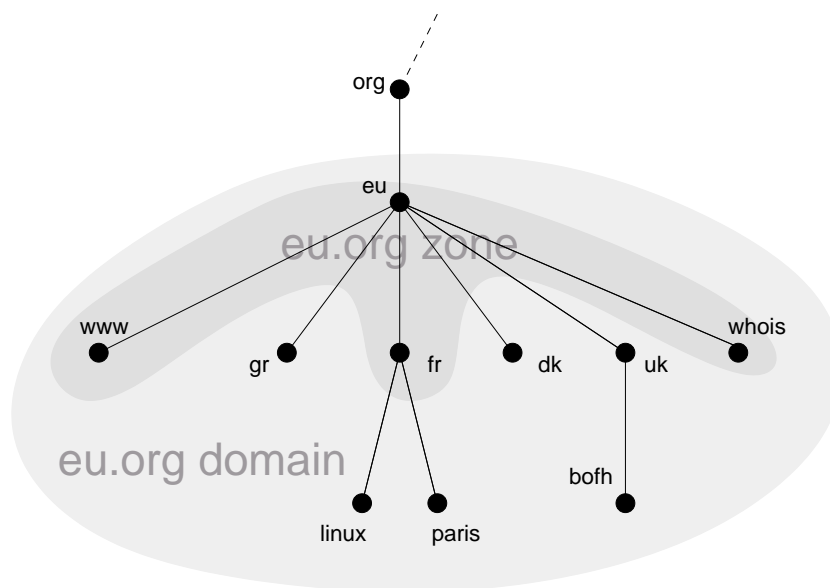


FIG. 1.10 : Délégation de zone

#### eu.org, des domaines gratuits

L'exemple `eu.org` est d'autant plus intéressant que ce domaine délègue gratuitement des sous domaines c.à.d. que si vous désirez avoir un sous domaine comme `ricou.eu.org`<sup>a</sup>, il suffit de le demander sur le site web `www.eu.org`. Cela demande bien sûr de savoir gérer un sous domaine.

<sup>a</sup>`ricou.eu.org` est déjà pris et pour longtemps puisque c'est gratuit.

## Trouver l'adresse IP d'un nom, le fonctionnement du DNS

La recherche d'une adresse IP est l'opération initiale pour chaque connexion dès lors que l'on initie la connexion avec le nom de la machine et non son adresse IP. Pour faire la correspondance nom/adresse IP, vous devez avoir indiqué à votre machine l'adresse d'un "Serveur de nom", ou serveur DNS. Si tel n'est pas le cas vous ne pourrez plus vous connecter aux autres machines d'Internet, sauf en donnant directement leur adresse IP bien sûr.

Pour trouver l'adresse IP d'une machine à partir de son nom, votre serveur de nom va lire le nom de la machine de droite à gauche pour savoir à quel autre serveur de nom il pourra demander l'adresse IP s'il ne la connaît pas.

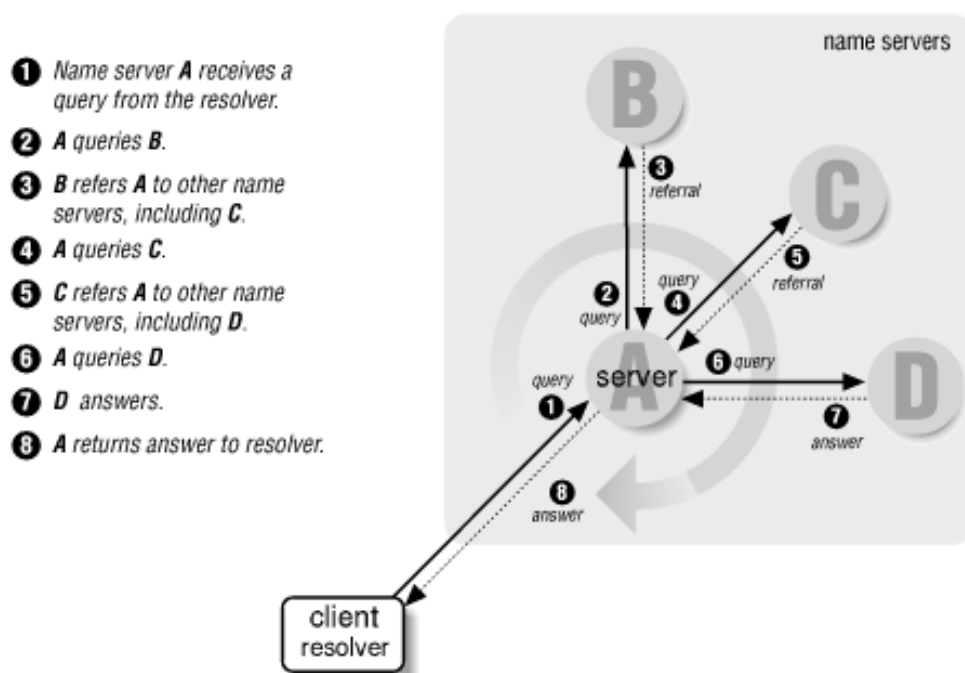


FIG. 1.11 : Fonctionnement récursif du DNS  
(illustration extraite du livre DNS & Bind chez O'Reilly)

Supposons que l'on cherche à se connecter sur le serveur web d'un laboratoire de Jussieu : `www.ann.jussieu.fr`. Notre serveur de nom, A sur la figure 1.11, n'ayant pas en mémoire l'adresse IP<sup>14</sup> de cette machine, va devoir la demander au serveur B, qui gère la racine d'Internet<sup>15</sup>, lequel renverra sur le serveur C qui gère `.fr.`, qui renverra au serveur D qui gère `jussieu.fr.` et qui donnera l'adresse IP recherchée `134.157.2.68`.

<sup>14</sup>il la conservera un certain temps une fois la demande faite ce qui évite de réitérer le processus à chaque connexion.

<sup>15</sup>ce serveur racine est tellement important qu'il est dupliqué en 13 exemplaires. Si ces 13 serveurs (plus en fait, cf la section ??) tombent tous en panne, Internet s'arrêtera doucement, le temps que les mémoires des serveurs de nom de la planète s'effacent.

## 1.2 La sécurité des communications

L'importance d'Internet dans notre société ne cesse de croître. Un nombre de plus en plus important de données circulent sur Internet et de plus en plus d'entreprises sont reliées au réseau. En même temps, de plus en plus de machines infectées par des virus servent malgré elles de relais aux attaques informatiques. Les pirates n'ont jamais eu autant de puissance ni autant de victimes à leur disposition.

### 1.2.1 Les faiblesses d'Internet

D'un point de vue technique, Internet actuel a deux vulnérabilités fondamentales : les messages sont transmis sans protection sur le réseau et l'identification de l'interlocuteur qu'il soit individu ou machine est peu fiable.

Ces deux faiblesses peuvent permettre à un pirate d'intercepter des informations qui ne lui sont pas destinées.

À ces deux vulnérabilités il est important d'ajouter l'erreur humaine qui est responsable de bien des mésaventures.

#### Les communications en clair

Le protocole de transport des données sur Internet, TCP/IP v.4, ne prévoit pas de protéger les données transportées. Tous les paquets sont transmis en clair. Ainsi toute personne qui contrôle un des ordinateurs par lequel passent les données peut les lire. Par exemple au niveau d'un réseau local, tous les paquets sortant vers Internet doivent passer par une passerelle. Le contrôle de cette machine permet la lecture de tout ce qui va et vient. Toujours sur un réseau local une personne qui est physiquement sur le même fil Ethernet qu'une autre<sup>16</sup> peut y détecter le courant qui y passe et donc lire les données.

Ce manque de sécurité devient important lorsqu'on transmet des informations confidentielles comme des mots de passe. Ainsi pour rapatrier son mail depuis un serveur distant, il est usuel d'envoyer son mot de passe qui passe en clair sur le réseau. Sur le web, autant les sites importants comme celui de votre banque protègent sérieusement la phase d'identification, autant cette protection n'est pas appliquée dans de nombreux sites de forum d'où l'importance d'utiliser différents mots de passe suivant le contexte.

**Espionner les communications** Lorsqu'on utilise Internet, on communique toujours avec une machine distante. Si la connexion est établie avec un programme qui ne cache pas les données, comme un navigateur, le flux de données est lisible avec un renifleur de paquets IP comme le programme `tcpdump`. Voici ce que l'on peut voir passer si on est sur le chemin pour écouter :

```
18 :12 :23.983918 IP (tos 0x0, ttl 64, id 24337, offset 0, flags [DF], proto :
```

<sup>16</sup>Toutes les personnes branchées sur un même *hub* sont sur le même fil Ethernet. Avec un *switch* il est plus difficile d'intercepter les communications mais cela reste possible (voir l'ARP Spoofing).

```
TCP (6), length : 1019) portable18.pmmh.espci.fr.3192 > mg-in-f147.google.com.www :
P 1 :968(967) ack 1 win 1460 <nop,nop,timestamp 7090623 2265318920>
E..._.@.@.i..6Q..U...x.P.Yn....B....r.....
.l1.....GET /search?hl=fr&q=piratage+i
```

où l'on voit que je recherche sur Google une information sur "piratage+i", le "i" étant le début du mot "internet" qui est dans le paquet suivant.

Bien sûr des logiciels existent pour améliorer la présentation voire pour ne retenir que l'information désirée. Ainsi le programme `dsniff` peut récupérer les mots de passe :

```
[root@diane dsniff-2.2]# ./dsniff -i eth0
dsniff : listening on eth0
-----
09/25/00 18 :34 :25 tcp hermes.devinci.fr.1415 -> aldebaran.devinci.fr.23 (telnet)
test
pass_compte
-----
09/25/00 18 :34 :39 tcp hermes.devinci.fr.1416 -> aldebaran.devinci.fr.110 (pop)
user test
pass pass_pop
```

Le premier couple login/mot de passe intercepté provient d'une connexion à un ordinateur distant en clair (plus personne ne fait ça maintenant, enfin normalement...), le second d'une connexion à un serveur de mail POP pour y rapatrier son courrier (beaucoup de personnes font toujours ça).

Il est donc important de garder à l'esprit que les données ne sont pas protégées par le réseau et que le travail de protection doit être fait au niveau des applications<sup>17</sup> afin que les données ne quittent votre machine que chiffrées. Cela étant il existe une exception lorsque vous utilisez un canal sécurisé comme un tunnel ou un VPN. Dans ce cas tout ce qui passe sur le réseau est protégé mais cela ne marche que pour les machines qui sont reliées par ce canal, cf le transport du mail ci-dessous.

### Des mails trop visibles

Le protocole utilisé pour diffuser le courrier électronique, SMTP, ne contient pas plus de système de protection des données que les protocoles IP ou TCP (il s'agit de l'empilement de couches informatiques présentées figure 1.1).

Sachant que tout n'est que connexion de machine en machine, le courrier électronique comme les autres services est ouvert à tous si l'on n'utilise pas de moyen de protection. Ainsi, par défaut, le mail

1. peut être lu lors de son transport

<sup>17</sup>Cela demande à ce qu'il existe un protocole chiffré pour les applications concernées.

- du logiciel de mail de l'émetteur à son relai (dit SMTP),
  - de machine en machine jusqu'à la destination,
  - de chez votre hébergeur de mail à votre logiciel de Mail via POP ou IMAP.
2. peut être intercepté par un pirate (qui prend l'identité de votre serveur par exemple)
  3. peut être lu par le responsable système de votre site.

Pour se protéger on peut appliquer diverses solutions qui fonctionneront à différents niveaux :

1. lors du transport du mail on peut
  - établir un tunnel chiffré entre sa machine et son serveur pour rendre impossible toute écoute<sup>18</sup>
  - utiliser un réseau protégé virtuel VPN (Virtual Private Network) qui comprend votre machine et le serveur de mail,
2. pour se protéger contre un pirate il faut utiliser un certificat qui permet d'identifier chaque machine comprise dans le chemin (le protocole TLS le permet), cf section 1.2.4,
3. pour se protéger du responsable système de votre site il n'existe qu'une seule méthode : que votre mail soit illisible.

#### Le FBI lit les mails du patron de la CIA

Même chez les espions on ne comprend pas toujours très bien que le mail n'est pas protégé. L'affaire Petraeus en a été la preuve.

Paula, la maîtresse caché du chef de la CIA, David Petraeus, est jalouse de Jill, une copine de ce dernier. Elle lui envoie donc des mails malveillants mais en se protégeant (faux compte Gmail, mails envoyés que depuis des lieux publics et hôtels avec wifi gratuit). Jill porte plainte contre X.

Le FBI récupère auprès de Google les adresses IP des machines qui ont envoyé les mails puis en regardant les registres des hôtels, il s'avère qu'une seule personne était dans ces différents hôtels à ces différents moments : la maîtresse secrète. Une fois Paula identifiée, le FBI obtient de Google l'accès à son compte Gmail officiel. Il y découvre la correspondance avec le chef de la CIA. Le FBI en profite pour regarder aussi le compte Gmail de Jill et découvre une relation avec un général.

Résultat, le patron de la CIA démissionne et le général quitte l'armée alors que le poste de chef de l'OTAN lui tendait les bras.

**GPG, la protection absolue** La solution la plus simple pour éviter qu'un mail ne soit lu par une autre personne que le destinataire est de le chiffrer. Ainsi le mail quitte la machine de l'émetteur illisible et ne sera déchiffré qu'une fois arrivé sur la machine du destinataire. Cette protection est absolue mais demande le plus souvent l'installation d'un greffon<sup>19</sup> ainsi que la compréhension du mode d'emploi (cf section 1.2.3). Le jeu en vaut la chandelle, d'autant plus que c'est ce même système qui permet la signature électronique.

<sup>18</sup>Sous Unix ainsi : `slogin -L2110 :localhost :110 serveur -f sleep 999d` et dire à son logiciel de mail que le serveur POP est sur localhost, port 2110.

<sup>19</sup>plugin pour les anglophones

## Le Web sécurisé

Le Web est protégé par l'algorithme de chiffrement SSL<sup>20</sup> qui est présent sur les navigateurs les plus courants. Par contre, comme pour tout algorithme de chiffrement, son efficacité est directement liée à son utilisation et à la taille de la clé de codage utilisée.

Ainsi la majorité des pages web ne sont pas chiffrées et donc passent en clair sur le réseau avant d'arriver sur votre ordinateur. Cela veut dire que toute personne qui contrôle les machines intermédiaires peut savoir quelles sont les pages web que vous regardez.

Lorsque vous arrivez sur une page sécurisée, ce qui est visible par une icône en forme de clé ou de cadenas ainsi que dans l'URL qui commence par `https`, personne ne peut intercepter le contenu si la clé de chiffrement utilisée est dite forte, à savoir contient 128 bits<sup>21</sup>. Si par contre la clé est plus courte, alors la sécurité est illusoire car trop faible pour résister aux attaques brutales, type d'attaques qui essayent toutes les clés possibles.

En mai 2001, une étude faite par ProjeWeb, cf tableau 1.2, montrait qu'une majorité de serveurs Web français continuait à utiliser des clés de 40 bits.

	Java	0	40 bits	56 bits	128 bits	total < 128
Banque - Bourse	1,9%	1,9%	65,4%	4,8%	26,0%	74,0%
Biens de consommation	0,0%	5,6%	36,1%	2,8%	55,6%	44,4%
Culture - Loisirs	0,0%	5,5%	34,1%	3,3%	57,1%	42,9%
Maison - Électro-ménager Hifi	1,9%	1,9%	28,8%	7,7%	59,6%	40,4%
Total	0,9%	3,8%	43,9%	4,4%	47,0%	53,0%

TAB. 1.2 : Niveau de sécurité des serveurs français (étude de ProjeWeb, mai 2001)

Cela n'est heureusement plus le cas aujourd'hui mais n'hésitez pas à cliquer sur le petit cadenas pour vérifier si la protection utilisée est bien SSL 128 bits. Il existe néanmoins un autre risque qui est celui de ne pas être connecté au véritable serveur (cf la section 1.2.4 sur l'authentification).

## L'erreur humaine

Quels que soient les outils de sécurité mis en place, il est difficile voire impossible de protéger un système si l'utilisateur autorisé aide le pirate. Cette aide peut aller du mot de passe offerts au pirate à l'installation sur sa machine de programme comprenant des virus. Dans la première catégorie on retrouve les mots de passe trop simples, ceux écrits sur un papier bien visible et enfin ceux donnés innocemment directement au pirate suite à un faux mail d'alerte (méthode dite du phishing, cf l'encart à ce sujet). La seconde méthode va des programmes téléchargés et installés aux greffons et autres extensions ajoutées à ses programmes ou à son système d'exploitation en passant par les pièces attachées des mails qu'on lance sans s'en rendre compte.

Si certaines méthodes techniques, comme la vérification de la sûreté d'un mot de passe, peuvent aider à limiter ce risque humain, il est important de savoir trouver le bon équilibre entre un

<sup>20</sup>renommé TLS depuis 2001

<sup>21</sup>un bit est 0 ou 1, 1001 est un nombre binaire à 4 bits qui vaut 9 en décimal.

système trop ouvert, où la moindre faille de l'utilisateur provoque un risque et un système trop coercitif qui sera contournées par les utilisateurs refusant que le système soit un obstacle à leur travail. On notera concernant ce dernier point qu'une étude (voir []) justifie d'un point de vue économique le comportement de ceux qui contournent les mesures de sécurité, l'idée étant qu'un risque dangereux mais rare coûte moins cher à l'entreprise qu'une mesure de protection peu coûteuse en temps mais quotidienne.

### 1.2.2 La cryptographie

La cryptographie est le remède au mal. Elle protège les communications dès lors que votre machine n'est pas infectée, que votre logiciel n'a pas de bug, que vous ne donnez pas vos clés ou mot de passe au pirate...

La cryptographie permet de chiffrer les données, d'en garantir l'intégrité et de les signer.

Le premier point permet

- la confidentialité des communications (transactions bancaires, connexions à distance, téléphone, mail...),
- la protection de données informatique stockées (secrets militaires, industriels, commerciaux, médicaux, personnels...)

Le second point, la garantie de l'intégrité, permet item d'avoir la certitude qu'un document est complet (contrat, mail, logiciel...) et que personne n'a pu le modifier.

Enfin le dernier point, la signature, permet

- de savoir avec certitude qui est l'origine d'un document
- d'apposer sa signature à un document
- la non-répudiation,
- de protéger des systèmes informatiques contre les intrusions en vérifiant l'identité des machines et utilisateurs,
- de vérifier l'authenticité d'un site Web

Avec la combinaison des trois, je peux envoyer un mail en étant certain que personne d'autre que mon destinataire ne pourra le lire (chiffre). Mon destinataire aura la certitude que mail vient bien de moi (signature) et qu'il n'a pas été modifié (intégrité). Ainsi je ne pourrai pas contester que je suis à l'origine du message (non-répudiation).



## Le phishing

Le phishing consiste à aller à la pêche au login/mot de passe, numéro de carte bleue... en envoyant un mail alarmant demandant au destinataire de suivre un lien pour se protéger. L'exemple qui suit demande aux propriétaires d'un nom de domaine chez Enom de se connecter sur leur compte pour valider les informations les concernant sous peine de perdre leur nom de domaine.

Date : Sat, 1 Nov 2008 10 :56 :39 +0100  
From : eNomCentral Team <support@enom.com>  
To : olivier@ricou.eu.org  
Subject : Inaccurate whois information.

Dear user,

On Sat, 1 Nov 2008 10 :56 :39 +0100 we received a third party complaint of invalid domain contact information in the Whois database for this domain. Whenever we receive a complaint, we are required by ICANN regulations to initiate an investigation as to whether the contact data displaying in the Whois database is valid data or not. If we find that there is invalid or missing data, we contact both the registrant and the account holder and inform them to update the information.

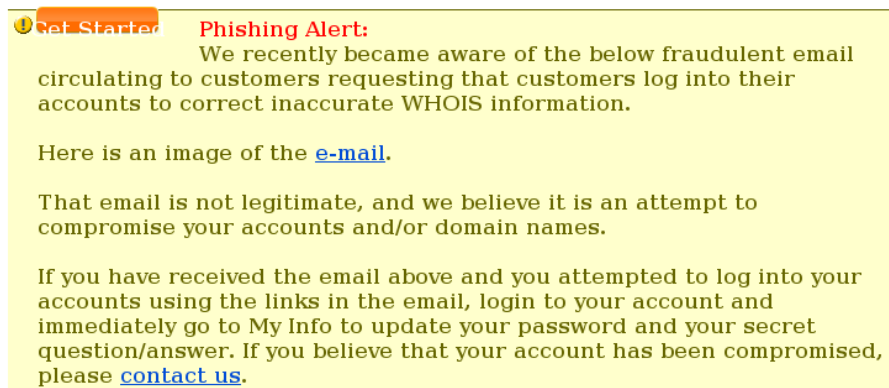
...

PLEASE VERIFY YOUR CONTACT INFORMATION - <http://www.enom.com.ssl48.mobi>  
LINK TO CHANGE INFORMATION - <http://www.enom.com.ssl42.mobi>

Thank you,  
Domain Services

Bien sûr, le lien donné est un faux qui ne renvoie pas chez Enom, [www.enom.com](http://www.enom.com), mais sur [www.enom.com.ssl48.mobi](http://www.enom.com.ssl48.mobi), site qui appartient à celui qui contrôle [ssl48.info](http://ssl48.info). Si l'on suit ce faux lien, on tombe sur une page identique d'aspect à la page d'authentification du site d'Enom et si l'on entre son login/mot de passe, on s'est fait avoir...

Lorsqu'on craint d'être la victime d'une telle attaque, il est conseillé de contacter directement et par la voie usuelle l'entreprise concernée. Ainsi, dans notre cas, en allant sur la page d'accueil d'Enom, la véritable : [www.enom.com](http://www.enom.com), on sait immédiatement à quoi s'en tenir, le message suivant confirmant l'arnaque :



**Get Started Phishing Alert:**  
We recently became aware of the below fraudulent email circulating to customers requesting that customers log into their accounts to correct inaccurate WHOIS information.

Here is an image of the [e-mail](#).

That email is not legitimate, and we believe it is an attempt to compromise your accounts and/or domain names.

If you have received the email above and you attempted to log into your accounts using the links in the email, login to your account and immediately go to My Info to update your password and your secret question/answer. If you believe that your account has been compromised, please [contact us](#).

FIG. 1.12 : Message d'avertissement d'Enom contre une arnaque

m.à.j. sur <http://www.ricou.eu.org/e-politique.html>

## Les clés symétriques ou secrètes

La façon la plus simple de chiffrer un message est de lui appliquer une fonction mathématique. Ainsi Jules César chiffrait ses messages en décalant les lettres de  $N$ , ainsi avec  $N=3$ , le A devient D. Pour le déchiffrer il suffit d'appliquer la fonction inverse avec la même clé. Bien sûr un bon système de cryptographie propose une fonction inverse qui ne donne rien sans la clé ( $N$  dans le cas de Jules César). Ce système est celui de la clé symétrique.

ATTAQUEZ GERGOVIE  
↓  
DWWDTXHC JHJRYLH

FIG. 1.13 : Un message secret de César

Ainsi pour communiquer entre 2 ou 3 personnes il suffit d'avoir une clé commune pour pouvoir communiquer de façon protégée par la suite.

À plus grande échelle, le système dit de tiers de confiance, TDC, (Trusted Third Party en anglais, ou TTP) propose de définir une clé  $K_i$  pour chaque utilisateur (voir figure 1.14). Ainsi pour chaque communication, le TDC donne aux 2 utilisateurs une clé de session  $k$  pour chiffrer leur communication. Bien sûr cette clé de session est transmise chiffrée avec la clé secrète de l'utilisateur.

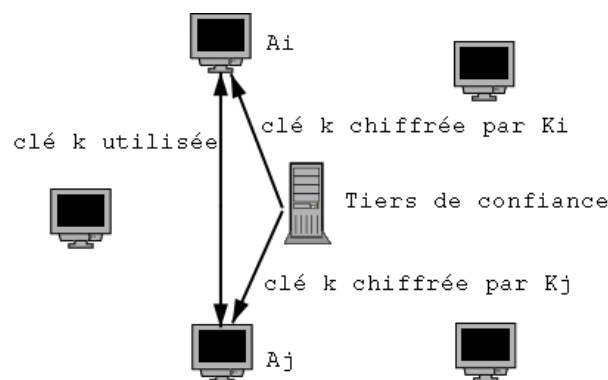


FIG. 1.14 : Tiers de confiance pour chiffrement symétrique

Ce système de clés symétriques a les avantages suivants :

- il est facile d'ajouter un nouvel entrant dans le réseau,
- chaque individu ne stocke que sa clé de communication avec le TDC,
- le chiffrement et déchiffrement sont rapides.

Les inconvénients sont :

m.à.j. sur <http://www.ricou.eu.org/e-politique.html>

- le besoin du TDC pour initier toute communication,
- le TDC peut lire tous les messages.

On comprend que la présence du tiers de confiance peut être jugée problématique.

### Les clés asymétriques ou publiques

Le système de cryptographie par clé symétrique a été le seul disponible jusqu'après la seconde guerre mondiale, ce qui veut dire que durant la seconde guerre mondiale les clés utilisées devaient être transmises physiquement à travers les théâtres d'opération avec tous les risques d'interception possibles lorsqu'on doit traverser les lignes ennemies. Lorsqu'on veut renouveler les clés régulièrement au cas où l'ennemi aurait réussi à les avoir, on en veut à la technologie qui impose cet exercice délicat.

La clé asymétrique corrige ce défaut en permettant de transmettre une clé publiquement tout en gardant une clé secrète pour déchiffrer les messages qu'on reçoit et qui ont été chiffrés avec la clé diffusée publiquement.

Ainsi une clé  $i$  est composée d'une clé publique et d'une clé privée ( $e_i/d_i$ ), chaque utilisateur ayant la sienne. Le message est chiffré avec la clé publique  $e_i$  et ne peut être déchiffré qu'avec la clé privée correspondante  $d_i$ .

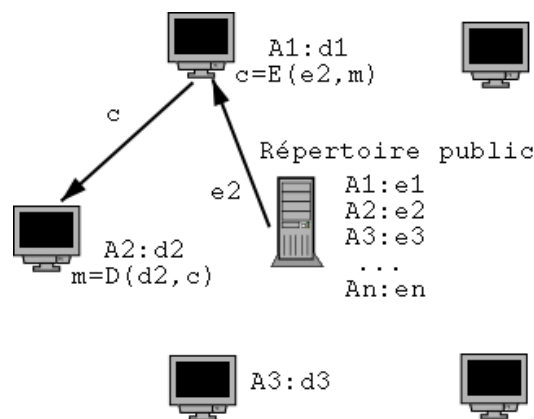


FIG. 1.15 : Utilisation de clés asymétriques

Les avantages de la méthode sont

- l'absence d'intermédiaire, pas de TDC,
- le fichier des clés publiques peut être largement diffusé.

Les inconvénients sont :

- un pirate peut diffuser une fausse clé publique (cf ci-dessous),

- le chiffrement est plus lent qu'avec une clé symétrique.

**L'attaque de l'homme au milieu** L'attaque la plus simple contre ce système est de substituer la clé publique d'un utilisateur par celle du pirate et d'intercepter tous les messages. Une fois le message intercepté, le pirate, l'homme au milieu, le décode, le note, puis le recode avec la véritable clé publique du destinataire pour lui envoyer afin qu'il ne détecte pas l'interception.

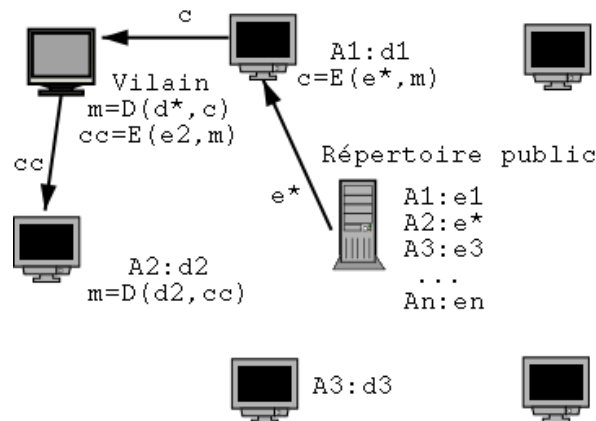


FIG. 1.16 : Attaque de l'homme au milieu

La parade, pour ne pas voir son message intercepté, réside dans la fiabilité de la clé publique de son destinataire. Une clé publique est sûre, soit parce que le destinataire vous l'a remise en main propre, soit parce qu'une personne en qui vous avez entièrement confiance vous garantit cette clé publique. Cette personne de confiance peut être un tiers de confiance institutionnel ou une personne dont vous êtes sûr car elle est dans votre liste des personnes de confiance. Dans ce dernier cas on parle de votre réseau de confiance ou *Web of trust* (cf section 1.2.4).

## Différents algorithmes de cryptographie

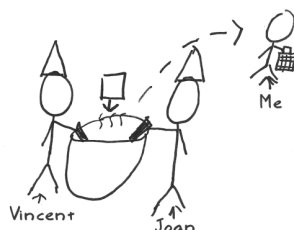
Sans remonter jusqu'à Jules César, il existe de nombreux algorithmes de cryptographie. Certains sont plus connus que d'autres et leur célébrité est la garantie de leur fiabilité. En effet il est difficile de créer un algorithme de cryptographie solide et seule sa vérification par le plus grand nombre possible de mathématiciens et d'utilisateurs peut offrir une garantie de sécurité.

**DES, Triple DES et AES** Historiquement DES, Data Encryption Standard, est le premier standard officiel des États-Unis à destination des entreprises. Il s'agit d'un algorithme de chiffrement à clé symétrique développé par IBM dans les années 70. DES utilise une clé de 56 bits qui, de nos jours, est bien trop faible pour résister aux attaques. Aussi DES ne doit plus être utilisé.

Son premier remplaçant a été Triple DES qui n'est que l'application de DES trois fois avec des clés différentes. Cela permet en effet d'amener la sécurité à un niveau correct mais pour un coût élevé en temps de calcul.

Aussi à la fin des années 90, le gouvernement américain a lancé un concours pour trouver le remplaçant idéal, sûr et peu gourmand en CPU afin de pouvoir l'exécuter sur le processeur d'une carte à puce. Le 2 octobre 2000 le gouvernement américain a annoncé<sup>22</sup> que l'Advanced Encryption Standard, AES, est l'algorithme belge Rijndael<sup>23</sup>.

My creators, Vincent Rijmen and Joan Daemen, were among these crypto wizards. They combined their last names to give me my birth name: Rijndael.\*



\* That's pronounced 'Rhine Dahl' for the non-Belgians out there.

FIG. 1.17 : AES expliqué en BD

cf <http://www.moserware.com/2009/09/stick-figure-guide-to-advanced.html>

**Les Rivest Cipher et RSA** Ronald Rivest est un cryptologue qui a conçu de nombreux algorithmes de chiffrements symétriques dit à la volée ("stream cipher" – RC4) et par bloc ("block cipher" – RC2 / RC5 / RC6). Parmi ces algorithmes, RC4 est le seul de la famille à être propriétaire ("trade secret") mais son code a été largement diffusé. RC6 était un des 5 candidats retenus à AES.

Mais l'heure de gloire<sup>24</sup> est arrivée avec RSA<sup>25</sup>. Cet algorithme conçu en 1977 avec Adi Shamir et Len Adleman est le premier algorithme publié<sup>26</sup> à clé publique/clé privée (ou asymétrique). Il est toujours très utilisé. Son principe mathématique est expliqué page 30.

**Les condensats MD5 et SHA-1 et suite** Les condensats permettent de garantir l'intégrité d'un document. Il s'agit d'une fonction à sens unique, dite de hachage, qui résume un document en une ligne, le condensat. Cette fonction est telle que si l'on modifie quoi que ce soit dans le document, alors le condensat devient totalement différent. Les condensats les plus connus sont MD5 et SHA-1. Malheureusement ils ont tous les deux été cassés ce qui rend possible la génération d'un autre document qui produit le même condensat. Ces failles sont à l'origine du concours SHA-3, qui à l'instar de l'AES, a choisi en octobre 2012 l'algorithme Keccak comme la nouvelle norme.

<sup>22</sup>[http://www.esat.kuleuven.ac.be/cosic/press/pr\\_aes\\_english.html](http://www.esat.kuleuven.ac.be/cosic/press/pr_aes_english.html)

<sup>23</sup><http://www.esat.kuleuven.ac.be/~rijmen/rijndael/>

<sup>24</sup>Prix Turing 2002, le Nobel des informaticiens

<sup>25</sup>Les initiales de ses inventeurs, Rivest, Shamir et Adleman.

<sup>26</sup>L'armée anglaise avait trouvé quelques années auparavant un algorithme asymétrique mais bien sûr, elle s'était bien gardée de l'annoncer

```
md5("Le condensat garantit l'intégrité") = 9fb6e5c02fd664892271ca02e0266457
md5("Le condensat garantit l'intégrite") = d80c680cf92d64cb7830c86fbb2350f7
```

Seul le é final a changé mais le condensat est totalement différent.

FIG. 1.18 : Utilisation d'un condensat

## Les mathématiques de RSA

L'algorithme RSA est un algorithme de chiffrement à clé publique/clé privée. Il est asymétrique et ne nécessite pas la transmission de la clé permettant le décodage.

L'idée d'un algorithme asymétrique a été proposée par Whitfield Diffie et Martin Hellman dans un article en 1975 et mise en pratique en 1977 par Ronald Rivest, Adi Shamir et Leonard Adleman. James Ellis et Clifford Cocks des services de communication de l'armée anglaise, avaient trouvé cet algorithme quelques années plus tôt mais ne purent le dévoiler pour cause de secret militaire (cf [?] et l'histoire présentée par Ellis<sup>27</sup>).

Son principe est relativement simple mais totalement révolutionnaire. On n'imaginait pas jusque là qu'il puisse être possible de décoder un message sans avoir la clé ayant permis de l'encoder. Pour cela chaque utilisateur a

- une clé publique  $(n||e)$
- une clé privée  $(n||d)$

où  $n$ ,  $d$  et  $e$  sont des entiers avec les propriétés suivantes :

1.  $n$ , le module, est le produit de 2 nombres premiers grands  $p$  et  $q$ ,
2.  $e < n$  est premier avec  $(p-1)(q-1)$ ,
3.  $d$  est tel que  $ed - 1$  soit divisible par  $(p-1)(q-1)$ .

Ces choix impliquent que  $ed \bmod J(n) = 1$  où  $J(n)$  est la fonction d'Euler sachant que  $J(n) = (p-1)(q-1)$  lorsque  $p$  et  $q$  sont premiers.

**Chiffrer un message pour un destinataire précis** En chiffrant un message  $M$  à l'aide de sa clé publique  $(n||e)$  on a :

$$M' = M^e \bmod n \quad (1.1)$$

ce qu'il peut déchiffrer avec sa clé privée  $(n||d)$  car

$$M'^d \bmod n = (M^e \bmod n)^d \bmod n = M^{ed} \bmod n = M^{ed \bmod J(n)} = M$$

<sup>27</sup><http://web.archive.org/web/19980507105259/www.cesg.gov.uk/ellisint.htm>

**Prouver son identité** En envoyant un message chiffré avec sa clé privée ( $n||d$ ) on a

$$M' = M^d \bmod n$$

que le destinataire peut lire avec la clé publique ( $n||e$ ) de l'émetteur en calculant  $M'^e \bmod n$ .

### 1.2.3 Utilisation de la cryptographie

#### Protéger son courrier avec GPG

Comme on l'a vu, le courrier est particulièrement vulnérable et la seule façon de le protéger nécessite l'usage de la cryptographie. Actuellement il existe deux principaux logiciels pour chiffrer les mails : PGP pour Pretty Good Privacy remplacé aujourd'hui par GPG, GNU Privacy Guard, et S/Mime. Tous les deux utilisent différents algorithmes de cryptographie pour remplir toutes les conditions nécessaires à la protection du courrier :

- un algorithme de chiffrement symétrique de type IDEA, CAST ou Triple-DES pour chiffrer la session,
- un algorithme de chiffrement asymétrique de type RSA, DH (Diffie-Hellman), or DSA (Digital Signature Alg) pour chiffrer la clé de session et signer,
- un condensat comme MD5, SHA-1, RIPEMD160 ou Tiger pour vérifier l'intégrité.

Pour des raisons de performance, les messages sont donc chiffrés à l'aide d'un système à clé symétrique dite clé de session. Cette clé est elle-même chiffrée avec la clé publique du destinataire, ainsi lui seul pourra la récupérer avec sa clé privée et donc lire le message.

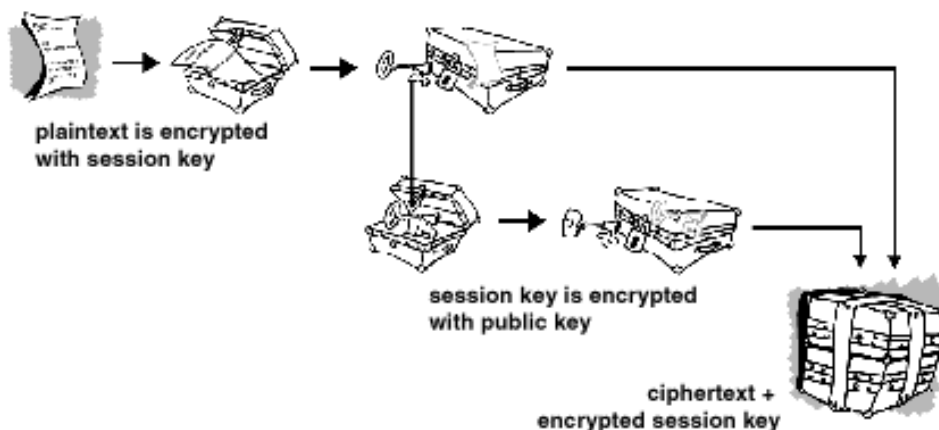


FIG. 1.19 : Encodage d'un message à l'aide de PGP

## La carte bleue cassée

Le 4 mars 2000, le texte suivant tombait dans le forum `fr.misc.cryptologie` :

Petite feuille Maple

```
> pub := 2^320 + convert('90b8aaa8de358e7782e81c7723653be644f7dcc6f816daf46e532b91e84f', decimal, hex);
pub := 2135987035920910082395022704999628797051095341826417406442524165008583957746445088405009...

> facteur1 := convert('c31f7084b75c502caa4d19eb137482aa4cd57aab', decimal, hex);
facteur1 := 1113954325148827987925490175477024844070922844843

> facteur2 := convert('14fdeda70ce801d9a43289fb8b2e3b447fa4e08ed', decimal, hex);
facteur2 := 1917481702524504439375786268230862180696934189293

> produit := facteur1 * facteur2;
produit := 213598703592091008239502270499962879705109534182641740644252416500858395774644508840...

> exposant_public := 3;
exposant_public := 3

> modulo_div_eucl := (facteur1 - 1) * (facteur2 - 1);
modulo_div_eucl := 2135987035920910082395022704999628797051095341823385970414850832581282681302...

> essai_rate_exposant_privé := expand((1 + modulo_div_eucl) / 3);
essai_rate_exposant_privé := 213598703592091008239502270499962879705109534182338597041485083258...

> exposant_privé := expand((1 + 2 * modulo_div_eucl) / 3);
exposant_privé := 14239913572806067215966818033330858647007302278822573136099005550541884542018...

> testnb := 1234;
testnb := 1234

> testsignnb := testnb &^ exposant_privé mod produit;
testsignnb := 2235938147775183775641042325450404557899532144626481715236694290974806919234121583...

> testverifsignnb := testsignnb &^ exposant_public mod produit;
testverifsignnb := 1234
```

On y trouve les nombres premiers  $p$  et  $q$ , ici `facteur 1` et `facteur 2` qui permettent de connaître le module  $n$ , ici `produit`. On voit que l'exposant publique,  $e$ , est 3 et après un premier test raté on trouve l'exposant privé  $d$ . Pour être sûr que tous ces chiffres sont bons, on chiffre 1234 et on le déchiffre. Ça marche.

Ce jour là le grand public voyait en clair la clé RSA à 320 bits qui permet de vérifier l'authenticité d'une carte bleue (voir l'article de Louis Guillou<sup>a</sup>). Cela indique seulement qu'une carte est authentique et non que l'on connaît le code secret de l'utilisateur, mais cela permet de faire des fausses cartes<sup>b</sup> qui tromperont un lecteur non relié aux banques comme celui qu'on présente souvent dans les restaurants (cf le reportage de LCI<sup>c</sup>).

C'est cette faiblesse connue des milieux de la cryptographie qu'a utilisé Serge Humpich<sup>d</sup>. La trouvaille n'est pas extraordinaire car casser une clé de 320 bits n'est plus un exploit depuis le début des années 90. L'exploit réside surtout dans la légèreté du groupement des cartes bleues.

<sup>a</sup>[http://parodie.com/humpich/LCguillou\\_AnnTelecom\\_No43\\_1988.jpg](http://parodie.com/humpich/LCguillou_AnnTelecom_No43_1988.jpg)

<sup>b</sup>faire une fausse carte bleue est assimilé à faire de la fausse monnaie. Le tarif est 30 ans de prison.

<sup>c</sup>[http://www.parodie.com/monetique/brevelci\\_demo\\_20072001.htm](http://www.parodie.com/monetique/brevelci_demo_20072001.htm)

<sup>d</sup>cf <http://www.parodie.com/monetique/>



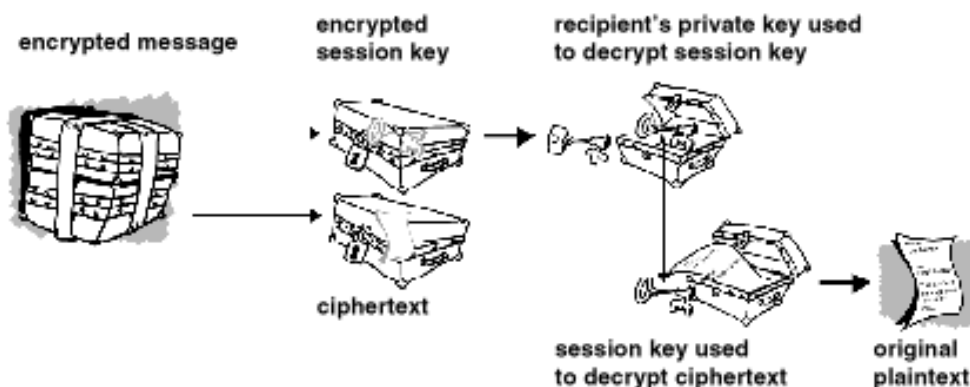


FIG. 1.20 : Décodage d'un message à l'aide de PGP

Pour signer et vérifier l'intégrité du courrier, l'émetteur fait un condensat du courrier et le chiffre avec sa clé publique. Ainsi le destinataire peut générer le condensat du courrier déchiffré et le comparer avec le condensat que lui a envoyé l'émetteur après l'avoir déchiffré avec la clé publique de l'émetteur.

Lorsque GPG est inclu dans votre logiciel de mail, son utilisation est transparente. Son initialisation peut faire peur pour celui qui ne connaît rien à la cryptographie puisque qu'on va lui demander de protéger sa clé privée avec un mot de passe et de publier sa clé publique. La publication de la clé publique est la partie la plus sensible puisque mal faite, elle peut permettre l'attaque de l'"homme au milieu", cf page 28. Il est donc soit nécessaire de la transmettre main dans la main<sup>28</sup>, soit de la faire signer par une autorité de certification, voir page 34.



FIG. 1.21 : Carte de visite avec condensat de la clé publique

**En pratique** S/MIME ou GPG sont de plus en plus intégrés dans les lecteurs de courrier. Pour les webmails<sup>29</sup> c'est plus rare mais il existe un greffon pour Gmail appelé FireGPG<sup>30</sup>.

<sup>28</sup>transmettre le condensat de la clé publique est souvent plus simple et permet ensuite de récupérer la clé sur Internet puis de vérifier qu'elle est la bonne.

<sup>29</sup>Le mail qu'on lit avec son navigateur.

<sup>30</sup><http://fr.getfirepg.org/s/home>

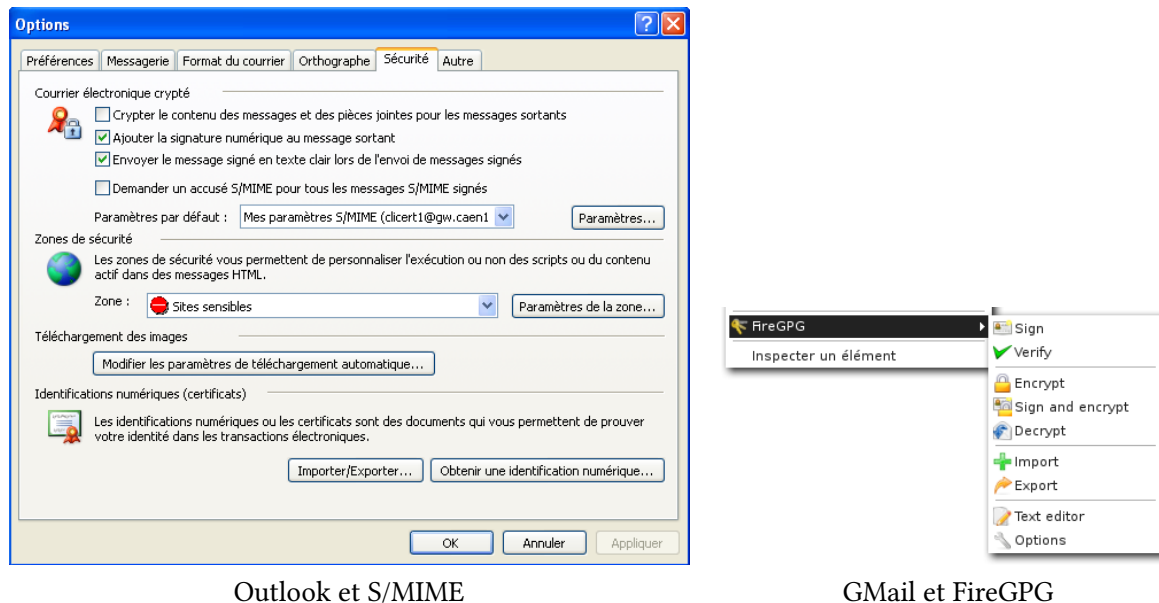


FIG. 1.22 : Lecteurs de mail et leur outil de cryptographie

## Surfer protégé

Le Web est probablement l'application la plus sensible pour la majorité des internautes. Il est utilisé pour faire ses achats, pour aller gérer son compte bancaire voire pour accéder à des données confidentielles. Aussi il est important qu'il comble les vulnérabilités inhérentes à Internet. Cela est fait avec le protocole TLS (anciennement SSL) qui chiffre toute donnée qui sort de votre navigateur avec une clé de session que ne connaît que le serveur avec qui vous communiquez. Ainsi une personne qui intercepte la communication ne comprendra rien.

TLS est automatiquement activé dès qu'on arrive sur une adresse qui commence par `https`. Les navigateurs soulignent l'aspect sécurisé de la communication avec une icône représentant un cadenas ou une clé.

Si cet aspect est sûr, il reste néanmoins des failles. La plus courante consiste à vous rediriger vers un autre site web qui ressemble trait pour trait à votre site web habituel. Pour éviter cela, un site web sécurisé doit vous envoyer son certificat qui prouve qu'il est bien celui qu'il prétend être. Votre navigateur accepte ce certificat dès lors qu'il est signé par une autorité de certification qu'il connaît, tout comme on accepte la clé publique d'un utilisateur dès lors qu'elle est signée par une personne en qui on a confiance. Là encore l'autorité de certification est à la base de la sécurité.

### 1.2.4 L'authentification

L'authentification consiste à vérifier l'identité du correspondant. Lorsqu'on regarde le champ `From` d'un mail on identifie le correspondant (peut-être en se trompant). S'il a signé le mail avec sa clé privée, et qu'on a confiance en sa clé publique, alors on peut vérifier qu'il est bien celui

m.à.j. sur <http://www.ricou.eu.org/e-politique.html>

qu'il prétend être et donc l'authentifier.

Pour le mail comme pour le web, l'authentification passe par la garantie que la clé publique que l'on possède est la bonne. Puisqu'il n'est pas toujours aisé de donner main dans la main cette clé ou son condensat, un autre système a été conçu : la certification.

La certification consiste à demander à un organisme reconnu d'offrir la garantie que le document<sup>31</sup> récupéré sur Internet est bien celui de notre correspondant. Pour cela l'organisme ajoute au document sa signature à l'aide de sa clé privée. Ainsi toute personne qui a la clé publique de l'organisme peut vérifier que la signature est bonne. On voit qu'on a seulement repoussé le problème puisque maintenant pour savoir si un document est le bon, il faut récupérer la clé publique de l'organisme.

Les autorités de certification sont ces organismes reconnus et leur clé publique sont intégrées par défaut sur tous les ordinateurs. Ainsi la personne qui désire falsifier une clé publique doit maintenant commencer par trafiquer les systèmes d'exploitation des ordinateurs pour y mettre de fausses clés publiques d'autorité de certification. La tâche est nettement plus ardue.

### Les autorités de certification commerciales

De nombreuses entreprises sont des autorités de certification<sup>32</sup>. Elles bénéficient d'un marché très lucratif puisque signer la clé d'une personne est une opération dont le seul coût est la vérification de son identité. On retrouve l'une des nombreuses "poules aux œufs d'or" qui se promènent sur Internet<sup>33</sup>.

L'autorité de certification la plus importante a longtemps été Verisign, la même entreprise que celle qui gère les .com et .net. Elle a acheté de nombreux concurrents comme Thawte et GeoTrust avant de céder en 2010 sa partie autorité de certification à Symantec<sup>34</sup> pour 1,28 milliards de dollars. Même avec ses rachats, Verisign ne dominait plus le marché comme avant.

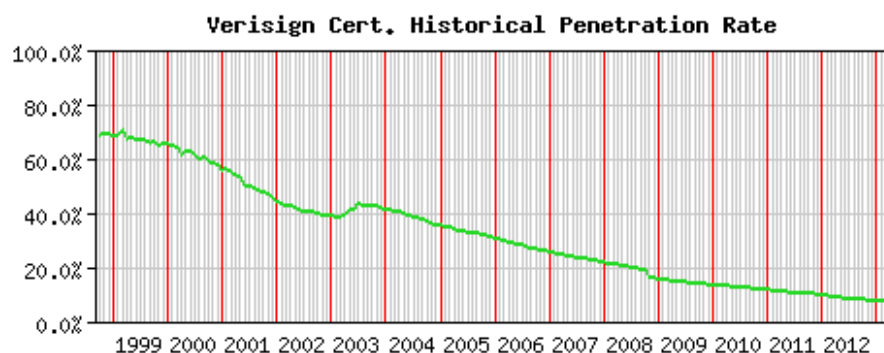


FIG. 1.23 : Part de marché de Verisign en tant qu'autorité de certification

source : Security Space, 2013

<sup>31</sup>clé publique, certificats SSL ou autre

<sup>32</sup>Pour se déclarer autorité de certification, il suffit d'avoir une clé publique et de se faire connaître

<sup>33</sup>Dans la même veine que la gestion des noms de domaine.

<sup>34</sup>L'entreprise d'anti-virus

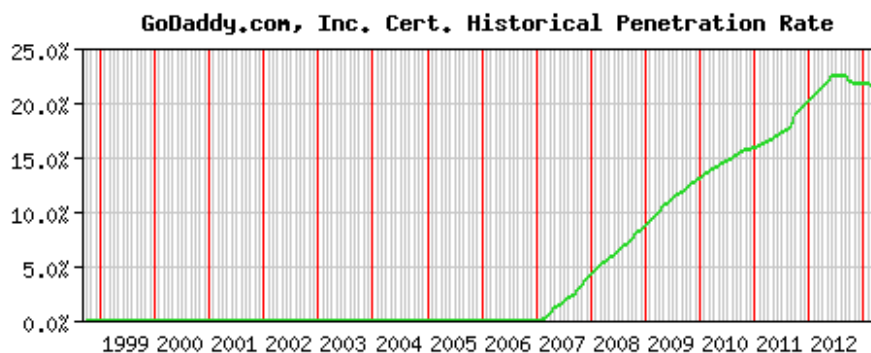


FIG. 1.24 : Part de marché de Go Daddy en tant qu'autorité de certification

source : Security Space, 2013

D'autres autorités de certifications ont pris la place dont l'hébergeur et bureau d'enregistrement de nom de domaine Go Daddy qui est devenu le numéro 1 (cf le site de Security Space<sup>35</sup> pour la liste des autorités et leurs parts de marché).

### Les autorités de certification gouvernementales

Étant donné que la signature électronique est reconnue par la loi française, il semblerait normal que l'État certifie les signatures des citoyens après les avoir dûment vérifiées comme il le fait pour les cartes d'identité. Et bien non, l'État délaisse l'identité numérique au secteur privé. Il n'est pas possible d'aller au commissariat de police avec sa clé publique et demander qu'elle soit certifiée.

L'État demande aux entreprises de payer la TVA par Internet. Pour cela il leur demande de justifier leur identité en présentant leur certificat numérique certifié par une autorité de certification. Et pour être bien clair, le ministère des finances indiquait dans sa FAQ sur la TéléTVA<sup>36</sup> que

*Les autorités de certification font autorité pour certifier les identités et principales caractéristiques des personnes à qui elles délivrent des certificats numériques. Elles jouent un peu le même rôle que les mairies lorsque vous faites une demande de passeport.*

et ajoute

*(le) Ministère de l'Economie, des Finances et de l'Industrie qui en les référençant, reconnaît la qualité des procédures mises en œuvre dans l'identification des demandeurs, l'enregistrement et la délivrance des certificats. C'est la raison pour laquelle elles sont amenées à vous demander de nombreux justificatifs.*

On voit que dans ces deux cas où l'internaute veut ou doit justifier son identité, il ne peut le faire qu'en passant par une entreprise privée qui sera amenée à demander de nombreux justificatifs,

<sup>35</sup>[http://www.securityspace.com/s\\_survey/data/man.201111/casurvey.html](http://www.securityspace.com/s_survey/data/man.201111/casurvey.html)

<sup>36</sup><http://tva.dgi.minefi.gouv.fr/pagesHTML/faq/faqcertificats.htm>

### Qui certifie ce site web ?

L'équivalent du champs `From` : pour identifier les sites web est leur adresse ou URL. On imagine que `www.lcl.fr` appartient à au Crédit Lyonnais (presque vrai, à sa maison mère) mais là encore il s'agit d'une information qui peut être trompeuse. Ainsi que penser de `www.lcl.net` ou `particuliers.secure-lcl.fr` ? Aussi le web dispose avec SSL d'un outil qui permet de certifier qui est derrière un site web.

This certificate has been verified for the following uses:	
SSL Server Certificate	
<b>Issued To</b>	
Common Name (CN)	particuliers.secure.lcl.fr
Organization (O)	Credit Agricole SA
Organizational Unit (OU)	SILCA
Serial Number	57:51:3C:B6:7B:29:7F:94:7D:C8:DE:66:25:C2:32:43
<b>Issued By</b>	
Common Name (CN)	VeriSign Class 3 Secure Server CA - G2
Organization (O)	VeriSign, Inc.
Organizational Unit (OU)	VeriSign Trust Network
<b>Validity</b>	
Issued On	11/19/2009
Expires On	12/10/2010
<b>Fingerprints</b>	
SHA1 Fingerprint	85:B3:A0:FC:52:A8:78:EE:0C:FA:44:63:22:92:4C:53:DA:FF:88:96
MD5 Fingerprint	68:77:A5:49:F4:6F:A8:04:C8:90:CF:20:6E:33:BA:3E

FIG. 1.25 : Le certificat de `particuliers.secure.lcl.fr`

Là encore on se base sur la signature de la clé publique par une autorité de certification supérieure. Ainsi on peut voir dans le certificat du site du Crédit Lyonnais (lcl) qu'il est certifié par le Crédit Agricole, sa maison mère, qui elle-même est certifiée par VeriSign *Class 3 Secure Server* laquelle est certifiée par VeriSign *Class 3 Primary*. Enfin cette dernière est certifiée par elle-même, il faut bien s'arrêter quelque part.

```
% openssl s_client -connect particuliers.secure.lcl.fr :443
CONNECTED(00000003)
depth=2 /C=US/O=VeriSign, Inc./OU=Class 3 Public Primary Certification Authority
verify error:num=19:self signed certificate in certificate chain
verify return:0
---
Certificate chain
 0 s:/C=FR/ST=Hauts de Seine/L=La Defense/O=Credit Agricole SA/OU=SILCA/\
   CN=particuliers.secure.lcl.fr
   i:/C=US/O=VeriSign, Inc./OU=VeriSign Trust Network/OU=Terms of use at\
     https://www.verisign.com/rpa (c)05/CN=VeriSign Class 3 Secure Server CA
 1 s:/C=US/O=VeriSign, Inc./OU=VeriSign Trust Network/OU=Terms of use at\
   https://www.verisign.com/rpa (c)05/CN=VeriSign Class 3 Secure Server CA
   i:/C=US/O=VeriSign, Inc./OU=Class 3 Public Primary Certification Authority
 2 s:/C=US/O=VeriSign, Inc./OU=Class 3 Public Primary Certification Authority
   i:/C=US/O=VeriSign, Inc./OU=Class 3 Public Primary Certification Authority
---
```

justificatifs qu'un citoyen n'a peut-être pas envie de donner à une entreprise privée. Ce point est d'autant plus triste que la carte d'identité électronique nationale serait très utile sur Internet pour réduire les risques d'arnaques, diminuer le nombre de spam, communiquer avec l'administration, vérifier l'âge des Internauts et peut-être un jour faire de la démocratie électronique à visage ouvert.

## L'auto certification

Puisque la certification est nécessaire dans certains cas, comme pour avoir un site Web sécurisé, et que les seules autorités de certification sont payantes, de nombreuses personnes s'auto-certifient à savoir qu'elles signent leur propre clé privée ou créent leur autorité de certification pour l'occasion. Dans tous les cas la clé ne sera pas reconnue puisque l'autorité n'est pas référencée mais cela répond à l'obligation initiale. Le résultat est que le navigateur qui arrive sur une page web chiffrée par une clé auto-certifiée va bloquer tant que l'utilisateur ne lui indique pas de passer outre. Une démarche que fait l'utilisateur sans plus rien vérifier, à juste titre, mais qui finalement affaiblit l'ensemble des protections.

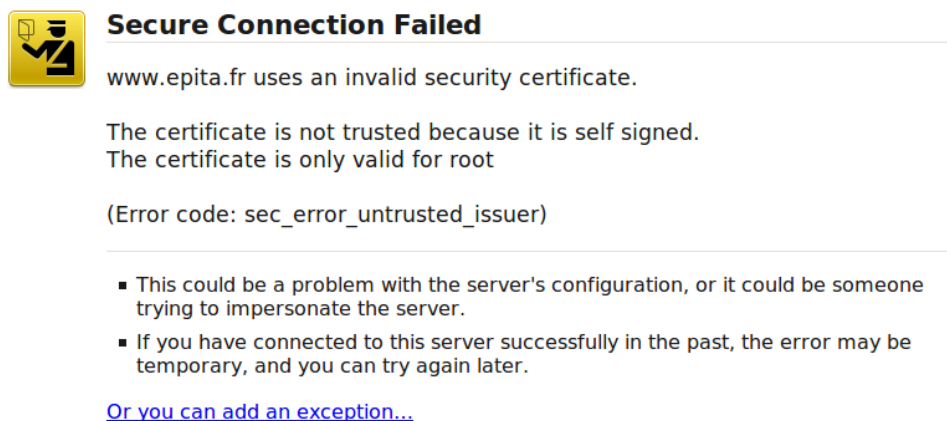


FIG. 1.26 : Firefox bloque un site web auto-certifiée

## Les réseaux de confiance

La dernière solution pour certifier sa clé privée revient à la faire signer par le plus de personnes qu'on connaît directement. Si ces personnes sont elles-mêmes connues d'autres personnes (cas probable), alors rapidement toute personne pourra vérifier votre clé sachant qu'elle fait confiance aux intermédiaires. Bien sûr cela demande que chacun fasse son travail de certification sérieusement et refuse de signer une clé sans une vérification physique (une clé envoyée par mail n'est pas assez sûre pour qu'on accepte de la certifier). Il est aussi possible de signer en indiquant un certain niveau de confiance (dans le dessin ci-dessous les flèches en noir offrent moins de confiance que celles en vert).

m.à.j. sur <http://www.ricou.eu.org/e-politique.html>

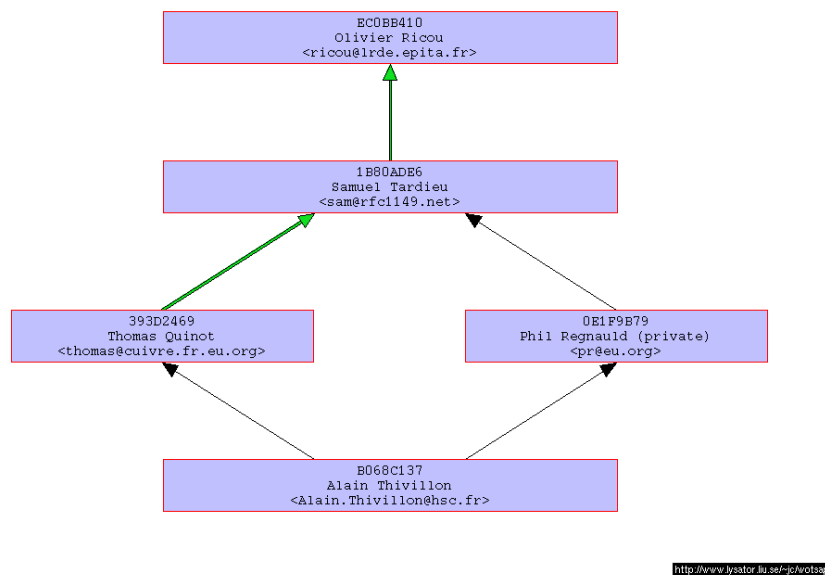


FIG. 1.27 : Chaîne de signatures de clés  
source : Wotsap<sup>37</sup>

Ce système est appelé le réseau de confiance (Web of trust en anglais).

### 1.2.5 La sûreté de la cryptographie

Pour finir ce chapitre, regardons comment on casse un algorithme de cryptographie :

- il existe une faille mathématique ou une découverte mathématique casse l'algorithme,
- il existe une faille de programmation<sup>38</sup>,
- la clé est trop courte et il est possible de tester toutes les clés possibles en un temps raisonnable,
- une faille ou découverte mathématique permet d'éliminer assez de clés pour que l'on puisse tester toutes les autres en un temps raisonnable.

Il y a donc deux catégories : les failles et la force brute qui teste toutes les clés possibles.

#### La force brute

La longueur d'une clé est la seule protection contre cette attaque. Ainsi suivant les caractères que vous utilisez, l'alphabet, et la longueur de votre mot de passe, tester toutes les clés possibles est

<sup>38</sup>Il est très difficile de programmer un logiciel de cryptographie même si l'algorithme est simple. Il est plus prudent d'utiliser une bibliothèque qui comprend les algorithmes dont on a besoin.

raisonnable ou non. Le tableau ci-dessous en donne une idée :

Alphabet	4 caractères	8 caractères	12 caractères
Lettres minuscules	$26^4 = 456\,976$	$208 \times 10^9$	$954 \times 10^{15}$
Lettres minuscules et chiffres	$36^4 = 1,6 \times 10^6$	$2 \times 10^{12}$	$4 \times 10^{18}$
Minuscules, majuscules et chiffres	$62^4 = 14 \times 10^6$	$218 \times 10^{12}$	$3 \times 10^{21}$

TAB. 1.3 : Nombre de clés possibles suivant l'alphabet et la longueur

Si on suppose qu'on a un ou des ordinateurs qui peuvent tester un million de clés par seconde (chiffre très raisonnable) alors on voit qu'une clé de 4 caractères résiste au mieux 14 secondes. Par contre la clé de 12 caractères avec minuscules, majuscules et chiffres résistera un million de siècles...

**La destruction de DES** DES a une clé de 56 bits<sup>39</sup>. Il a été l'une des premières victimes cassées par la force brute :

- **En juin 97** Rocke Verser de Loveland, Colorado, le casse avec des machines d'autres internautes en 90 jours.
- **En janv 98** `distributed.net` le casse en 39 jours avec 10 000 ordinateurs et une moyenne de 28.1 milliards de clés testées par jour<sup>40</sup>.
- **En juillet 98** Electronic Frontier Foundation, EFF le casse avec une machine à 250 000 \$ fabriquée pour en 3 jours<sup>41</sup>,
- **En janvier 99** DES est cassé en 22 heures par la machine de l'EFF couplée aux 100 000 machines réunies par le `distributed.net`.

Dans le dernier cas, près de mille milliards de clés étaient testées par secondes. A ce rythme, la clé de 12 caractères avec minuscules, majuscules et chiffres n'aurait tenu qu'un siècle. Sachant que la puissance des ordinateurs double tous les deux ans<sup>42</sup>, cela veut dire que dix ans plus tard, la même clé ne résisterait plus que 3 ans.

Cela étant, tester une clé de type DES peut prendre moins de temps que de tester une clé de taille égale d'un autre algorithme, aussi il est important de faire attention aux comparaisons.

**Le calcul distribué** La force brute est une méthode qui se répartie très bien sur un ensemble d'ordinateurs, chacun testant une partie des clés. Aussi des internautes ont créé une organisation chargée de répartir le travail parmi les ordinateurs mis à leur disposition, c'est [distributed.net](http://www.distributed.net).

Avec cette méthode, le RC5 a été régulièrement cassé avec des clés de plus en plus longues :

<sup>39</sup>il faut 6 bits pour stocker un caractère qui soit une minuscule ou une majuscule ou un chiffre, donc par rapport au tableau ci-dessus, 56 bits représente moins de 10 caractères.

<sup>40</sup><http://www.distributed.net/history.html>

<sup>41</sup>[http://www.eff.org/pub/Privacy/Crypto\\_misc/DESCracker/HTML/19980716\\_eff\\_descracker\\_pressrel.html](http://www.eff.org/pub/Privacy/Crypto_misc/DESCracker/HTML/19980716_eff_descracker_pressrel.html)

<sup>42</sup>Loi de Moore interprétée assez librement



- **En octobre 1997, RC5-56** est cassé en 212 jours de travail. Le pourcentage de clés vérifiées est de 47,03%, vitesse moyenne : 5,3 G clés/s. Au rythme final, il aurait fallu 83 jours pour vérifier l'ensemble des clés restantes.

- **En juillet 2002, RC5-64<sup>43</sup>** trouvé en

1 757 jours de calcul, environ 4 ans et 10 mois

331 252 participants

15 769 938 165 961 326 592, 15 milliards de milliards de clés testées  
soit 81% des clés possibles

vitesse maximale : 270 147 024 000 clés/seconde

- soit 32 000 de Apple PowerBook G4 800MHz ou  
46 000 PC AMD Athlon XP 2Ghz travaillant en parallèle
- à cette vitesse il suffirait de 790 jours pour tester  
l'ensemble des clés

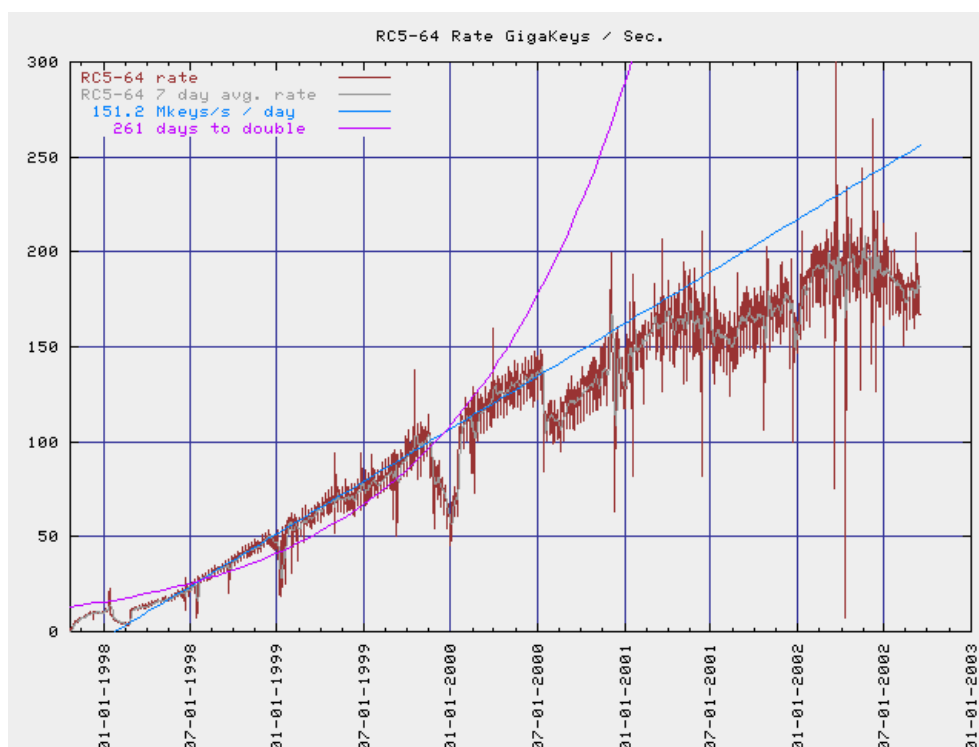


FIG. 1.28 : Vitesse de test des clés RC5-64 durant le calcul

**Casser le Web** L'algorithme de cryptographie du Web est SSL. Dans les années 90 et encore au début des années 2000, cf tableau 1.2, il se conjugait en 3 variantes de longueur de clés différentes : SSL 40 bits, SSL 56 bits et SSL 128 bits. Et pourtant dès l'été 1995, SSL 40 bits a été cassé en 32 heures à l'INRIA<sup>44</sup> et en 3h30 durant l'été 1997 à Berkeley<sup>45</sup>.

<sup>43</sup><http://www.distributed.net/pressroom/news-20020926.html>

<sup>44</sup><http://pauillac.inria.fr/~doligez/ssl/>

<sup>45</sup><http://catless.ncl.ac.uk/Risks/18.80.html#subj1>

Aujourd'hui quelques minutes voire quelques secondes suffiraient aussi il est indispensable d'utiliser la méthode SSL 128 bits. Heureusement il ne doit plus exister de sites qui utilisent encore SSL 40 ou 56 bits.

## L'intelligence contre la cryptographie

Il existe peu de cas où des avancées mathématiques cassent des algorithmes de cryptographie, en voici néanmoins deux exemples.

**RSA mis à l'épreuve** Afin d'avoir une estimation de la sécurité de RSA, l'entreprise RSA Security organise un concours ouvert dont le but est de casser un message chiffré avec l'algorithme RSA d'une longueur de clef déterminée. Le but est de trouver les deux nombres premiers  $p$  et  $q$  qui génèrent le module de l'algorithme de RSA ce qui permet d'avoir la clé privée. Pour venir à bout de ce défi, la méthode mathématique utilisée est celle du "crible algébrique" qui permet de ramener le problème à un calcul matriciel dont la résolution nécessite un super ordinateur<sup>46</sup>. Ainsi

- **En février 1999, RSA-140 chiffres** a été cassé. Le crible a nécessité environ 125 stations SGI et Sun à 175 MHz et environ 60 PCs à 300 MHz pendant 1 mois. Le système matriciel a demandé 100 heures CPU et 810 MO de mémoire vive sur un Cray C916.
- **En août 1999, RSA-155 (512 bits)** tombe. Le crible a nécessité 160 stations SGI et Sun à 175-400 MHz, 8 SGI Origin 2000 processeurs à 250MHz, 120 Pentium II PCs à 300-450 MHz et 4 500 Digital 500 Mhz pendant 3.7 mois. La matrice à résoudre avait 6 699 191 lignes et 6 711 336 colonnes pleines à 62.27%. Il a fallu 224 heures CPU et 3.2 GO de mémoire vive sur le même Cray pour résoudre le système.
- **En novembre 2005, RSA 640 bits** est tombé après 5 mois de calcul.
- **En décembre 2009, RSA 768 bits** est le dernier défi tombé.

On prédit la chute de RSA 1024 bits pour bientôt, aussi il est temps d'utiliser des clés plus longues.

**MD5 cassé affaiblit le Web** Depuis 2004 on sait qu'il est possible de faire deux messages qui ont le même condensat MD5. En 2008, une équipe de chercheurs<sup>47</sup> a appliqué cette possibilité théorique à un cas bien pratique : la génération de faux certificats Web.

En temps normal un site web sécurisé envoie au navigateur un certificat qui prouve qu'il est bien le site web qu'il prétend être, cf figure 1.29. Le navigateur vérifie l'identité du site Web en vérifiant que le certificat qu'on lui envoie est bien signé par une autorité de certification connue (c.a.d. dont la clé publique est dans le navigateur). Si c'est le cas, il ne reste plus qu'à vérifier que

<sup>46</sup> Une présentation sur la factorisation et donc sur la façon de casser RSA est présentée sur ce site : <http://pauillac.inria.fr/algo/banderier/Facto/>

<sup>47</sup> Alexander Sotirov, Marc Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik, Benne de Weger, cf <http://www.win.tue.nl/hashclash/rogue-ca/>

les données écrites sur le certificat, comme l'URL, correspondent à celles du site web qu'on est en train de visiter. Tout ce travail est invisible pour l'utilisateur si tout se passe bien.

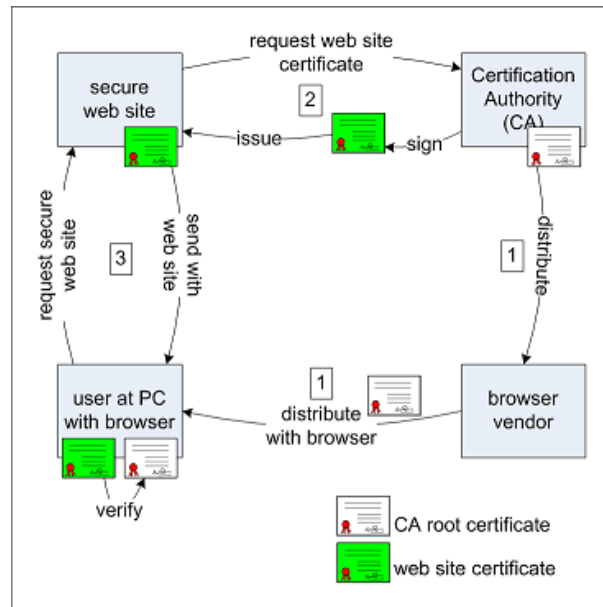


FIG. 1.29 : Signature et utilisation normale des certificats SSL

Dans le cas normal, l'utilisateur (en bas à gauche) vérifie le certificat du site web (en haut à gauche) avec la clé publique de l'autorité de certification (en haut à droite) qui lui a été fournie avec son navigateur (en bas à droite).

L'attaque, cf figure 1.30, consiste à demander à l'autorité de certification de nous signer un certificat (le bleu). La signature étant faite sur le condensat MD5 du certificat, elle sera aussi valable si elle est attachée à un autre document qui a le même condensat que le certificat qu'on a envoyé. Cet autre est ici la clé publique de notre fausse autorité de certification (la noire). Avec cette fausse autorité, on peut signer le certificat de notre faux site web (le rouge). Maintenant il ne reste plus qu'à intercepter les requêtes vers le site web d'origine (en haut à gauche) et à lui présenter le certificat rouge du faux site accompagné de celui de la fausse autorité de certification (le noir). Ainsi le navigateur constate que le site a un certificat (le rouge), que ce certificat est signé par le noir lequel est signé par le certificat officiel de l'autorité de certification (puisque le noir a le même condensat que le bleu). Donc tout va bien et aucun avertissement ne sera envoyé à l'utilisateur qui se connectera au faux site web en toute confiance puisque la connexion est sûre grâce à SSL.

Depuis l'annonce de cette faille, les autorités de certification sérieuses n'utilisent plus le condensat MD5. Cela peut être vérifié en regardant l'algorithme de signature utilisé dans la description du certificat.

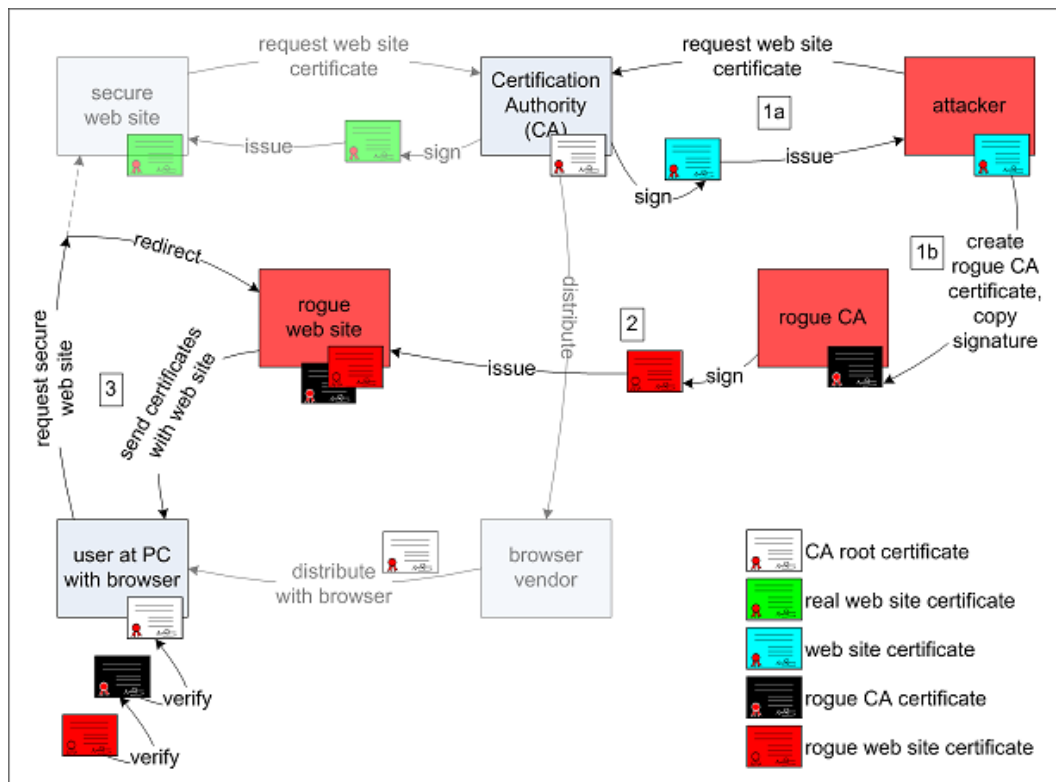


FIG. 1.30 : Introduction d'un faux certificat SSL

## La bêtise contre la cryptographie

**SSH cassé par ignorance** En mai 2008 la distribution Debian de Linux doit annoncer que toute la sécurité basée sur OpenSSL est compromise. Quelques années auparavant, une personne en charge de faire marcher le logiciel OpenSSL sur Debian a retiré du code source des lignes qui semblaient ne servir à rien. Et si le fait de retirer ces lignes n'a rien modifié au fonctionnement du logiciel, cela a détruit la fonction aléatoire en charge de fournir les nombres de base pour générer les clés. Or si on peut deviner ces nombres de base, on peut aussi deviner les clés, donc toute la sécurité s'effondre.

```
int getRandomNumber()
{
    return 4; // chosen by fair dice roll.
              // guaranteed to be random.
}
```

FIG. 1.31 : XKCD se moque

Moralité : la programmation de la cryptographie est réservée aux spécialistes. Il est illusoire d'espérer programmer un algorithme de cryptographie sans générer des failles de sécurité si on n'a pas une longue expérience dans le domaine.

m.à.j. sur <http://www.ricou.eu.org/e-politique.html>

## 1.3 Plus

### À propos de l'architecture d'Internet

- Quelques articles de l'encyclopédie Wikipédia :
  - le protocole d'Internet : [http://fr.wikipedia.org/wiki/Internet\\_Protocol](http://fr.wikipedia.org/wiki/Internet_Protocol),
  - le DNS : <http://fr.wikipedia.org/wiki/DNS>
- Architectural Principles of the Internet, RFC 1958 par B. Carpenter, IAB, Juin 1996, cf <http://asg.web.cmu.edu/rfc/rfc1958.html>.
- the DNS Resources Directory, cf <http://www.dns.net/dnsrd/>
- La page sur les logiciels graphiques de Traceroute, <http://www.nicolas-guillard.com/cybergeography-fr/atlas/routes.html>

Les passionnés pourront consulter en anglais CircleID, <http://www.circleid.com/>.

### À propos de la sécurité

La faille humaine est la grande faiblesse de la sécurité informatique contre laquelle chacun peut lutter :

- HoaxBuster, <http://www.hoaxbuster.com/>, le site à regarder avant de faire suivre un mail ou de suivre ce que demande un mail d'un inconnu ou d'un crédule,
- INFO ESCROQUERIES au 0811 02 02 17 ou <https://www.internet-signalement.gouv.fr/> pour tout renseignement ou pour signaler un mail ou site qui semble être une tentative d'escroquerie (je suppose que pour les renseignements la police vérifie sur HoaxBuster étant donné que le site est vide).

Pour les informaticiens ou passionnés, quelques sources hétéroclites :

- l'observatoire de la sécurité des systèmes, <http://www.ossir.org/>,
- Security vibes, <http://www.securityvibes.com/>, le blog commun de d'experts et responsables de sécurité,
- le blog de Bruce Schneier, <http://www.schneier.com/>, qui regarde aussi en dehors d'Internet,
- Black Hat, <https://www.blackhat.com/>, l'une des plus grande conférence sur la sécurité,
- le portail du gouvernement sur la sécurité informatique, <http://www.securite-informatique.gouv.fr/>

m.à.j. sur <http://www.ricou.eu.org/e-politique.html>

## À propos de la cryptographie

En ce qui concerne la cryptographie, on pourra aussi consulter les ouvrages suivants : *The Codebreakers* de David Kahn, cf [?], et *l'Histoire des codes secrets* de Simon Sing, cf [?].

Certains manuels (livres) sont disponibles en ligne dont *The Handbook of Applied Cryptography*, cf [?], les *Frequently Asked Questions About Today's Cryptography* des RSA Labs, cf [?].

Enfin, les sites suivants contiennent des informations intéressantes :

- Des centaines de transparents sur la cryptographie, cf <http://www.cs.auckland.ac.nz/~pgut001/tutorial/>,
- Une page sur les PKI, *Public Key Infrastructure*, cf <http://www.pki-page.org/>,
- Le site de TBS, « Abandonnons les protocoles non-chiffrés » présent, différentes méthodes pour se protéger, cf <https://www.tbs-internet.com/ssl/>
- Le site de Parodie, <http://www.parodie.com/monetique/>, présente l'affaire des cartes bleues.
- How PGP works, the Basis of Cryptography, cf <http://www.pgpi.org/doc/pgpintro/>,
- L'Agence nationale de la sécurité des systèmes d'information, ANSSI, cf <http://www.ssi.gouv.fr/>