

THEORIE DE L'INFORMATION ET CRYPTOGRAPHIE : Principes Fondamentaux

Prof.Dr.Patrick Mukala

Avant-Propos

Ce document est une compilation des notes sur la sécurité informatique et la cryptographie. Emanant de plusieurs sources méticuleusement choisies, il a été conçu pour servir de support du séminaire et mettant un accent sur principaux algorithmes de cryptographie qui sont introduits de façon succincte, avec tous les éléments constructeurs mais laissant d'espace pour stimuler la recherche dans l'esprit du lecteur. L'esprit de ce support est de vous accompagner au maximum afin que vous compreniez à quoi correspond chaque notion, quand utiliser telle technique de cryptage plutôt que telle autre, comment s'imbriquent tous les différents éléments de sécurité pour qu'à la fin du cours, vous soyez totalement autonomes et sachiez résoudre des problématiques relativement complexes.

Combinant soubassement théorique et exercices pratiques, chaque module est succinctement organisé pour permettre surtout un apprentissage autonome pour les étudiants.

Contenu

La sécurité constitue depuis belle urète un sujet de grande importance dans le sphère informatique. Depuis des décennies, le monde numérique est confronté à des défis de sécurisation de ses ressources sur tous les plans.

Particulièrement, les données (stockées et générées) sur un réseau informatique, représentent l'une des ressources les plus vulnérables et les plus sensibles et par ricochet, requièrent une attention soutenue pour sa sécurité. La sauvegarde des données est essentielle pour une entreprise, quelle que soit sa taille, à partir du moment où toute information importante est stockée sur l'infrastructure de son réseau à travers ces données. De ce fait, il est vivement indispensable de prendre des dispositions sécuritaires tenant compte de toutes les éventualités pouvant conduire à la perte partielle ou totale, temporaire ou définitive, des données. L'une de ces mesures, est l'application des mesures cryptographiques.

Dans ce cours, nous posons les bases de la sécurité informatique en tenant compte de risques réels qui sévissent les données ou informations dans le monde numérique. Nous passons en revue les types de risques et attaques les plus communs et discutons des principes fondamentaux de la cryptographie.

Table des matières

Avant-Propos.....	1
Contenu	2
Chapitre 1 : Sécurité Informatique - Généralités.....	7
1. Contexte	7
2. Sécurité.....	9
3. Sécurisation des Organisations	10
3.1. Services.....	11
3.2. Mécanismes	12
3.3. Attaques	12
3.3.1. Quelques exemples d'attaques (ou scenarios d'attaques) génériques	12
3.3.2. Attaques de systèmes d'information	13
3.3.3. Attaques passives et attaques actives.....	14
4. Menaces, risques et vulnérabilités.....	15
5. Composants de la Sécurité	16
5.1. Disponibilité.....	17
5.2. Intégrité.....	18
5.3. Confidentialité	19
5.4. Concepts additionnels	19
6. Typologies d'attaques Informatiques.....	20
6.1. Bombe logique	21
6.2. Cheval de Troie	22
6.3. Porte dérobée.....	24
6.4. Virus	24
6.5. Ver	25
6.6. Déni de Service (DDoS).....	26
6.7. Social Engineering.....	27
7. Exercices No1	28
Chapitre 2 : Cryptographie	30
1. Contexte	30
2. Concepts basiques	30

3. Cryptage et décryptage.....	32
3.1. Cryptage symétrique.....	33
3.1.1. Description	33
3.1.2. Caractéristiques.....	34
3.2. Cryptage asymétrique	34
3.2.1. Description	34
3.2.2. Caractéristiques.....	35
4. Quelques Chiffrements Classiques.....	36
4.1. Le chiffrement de César.....	37
4.2. Vigenère	38
4.3. Chiffrement autoclave (ou Autokey).....	41
4.4. Cryptanalyse.....	42
5. Exercices No3	45
Chapitre 3 : Cryptage Asymétrique (Algorithme RSA)	47
1. Concept.....	47
2. Principes basiques du RSA.....	48
2.1. La notion de divisibilité et congruence	48
2.2. Arithmétique modulaire	50
2.3. Notion d'inverse.....	52
2.4. Plus grand commun diviseur (PGCD)	53
2.5. Primalité (Primarité).....	55
2.6. Co-primalité (Co-primarité)	58
2.7. Fonction totient d'Euler	58
2.8. Théorème d'Euler.....	60
2.9. Cycles de taille $\phi(n)$	61
2.10. L'inverse multiplicatif modulaire.....	61
2.10.1. Théorème	61
2.10.2. Lemme de Bézout	61
2.10.3. Lemme de Bézout et l'inverse multiplicatif.....	63
2.10.4. Extension de l'Algorithme d'Euclide	63
2.11. L'Exponentiation Modulaire	65
2.11.1. Exponentiation rapide (exponentiation par carré)	65
2.11.2. Exponentiation par théorème d'Euler	66

3. L'algorithme RSA	67
3.1. Etapes	67
3.3. Aspect Sécuritaire	69
3.4. Conclusion.....	70
3.4.1. Utilité pratique.....	70
3.4.2. Restrictions (limites).....	70
6. Exercices no4.....	70
Chapitre 4 : Applications et Considérations Pratiques du RSA.....	73
1. Fonctions de hachage.....	73
1.1. Propriétés	74
1.2. Principaux algorithmes.....	75
1.3. Applications en Cryptographie	76
1.4. Cryptage vs Hachage	77
2. Signature Numérique (Digitale)	77
2.1. Scenario.....	77
2.2. Concept.....	77
2.3. Exemple d'implémentation	79
3. Certificats numériques.....	80
3.2. Notion de certificat et signature numériques	81
3.3. Notion d'infrastructures de clé publique (PKI)	84
3.3.1. Composants	84
3.3.2. Processus de délivrance de certificats	86
3.3.3. Vérification de la validité	86
3.3.4. Etablissement de la fiabilité	86
3.4. Révocation des Certificats.....	90
Chapitre 5 : Cryptage Symétrique (AES)	91
1. Vue d'ensemble de l'algorithme AES.....	91
2. Operations (Fonctions) de l'algorithme AES	94
2.1. SubBytes.....	95
2.2. ShiftRows	95
2.3. MixColumns	96
2.4. AddRoundKey	98
2.5. KeyExpansion (ExpandKey)	99

3.	Le déchiffrement avec AES.....	103
3.1.	invSubBytes	104
3.2.	invShiftRows.....	104
3.3.	invMixColumns	105
4.	Les bloc modes de l'algorithme AES.....	106
4.1.	Electronic Code Book (ECB).....	107
4.2.	CBC : Cipher Block Chaining	108
4.3.	CFB : Cipher Feedback	109
5.	Chiffrement Hybride	110
6.	Exercices No5	111
Chapitre 6 : Le management de la sécurité.....		113
1.	Contexte	113
2.	Le système de management de la sécurité de l'information.....	113
2.1.	Élaboration et mise en place du SMSI	114
2.2.	Suivi et application du SMSI	114
2.3.	Tâches de direction et d'encadrement	115
3.	Conclusion	115
4.	Exercices No6	115

Chapitre 1 : Sécurité Informatique - Généralités

1. Contexte

Les exigences de la sécurité de l'information au sein des organisations ont conduit à deux changements majeurs au cours des dernières décennies. Avant l'usage généralisé d'équipements informatiques, la sécurité de l'information était assurée par des moyens physiques (classeurs fermés par un cadenas) ou administratifs (examen systématique des candidats au cours de leur recrutement).

Avec l'introduction de l'ordinateur, le besoin d'outils automatisés pour protéger fichiers et autres informations stockées est devenu évident. Ce besoin est accentué pour un système accessible via un téléphone public ou un réseau de données. On donne à cette collection d'outils conçus pour protéger des données et contrecarrer les pirates le nom de sécurité informatique.

Le second changement majeur qui affecte la sécurité est l'introduction de systèmes distribués et l'utilisation de réseaux et dispositifs de communication pour transporter des données entre un terminal utilisateur et un ordinateur, et entre ordinateurs. Les mesures de sécurité des réseaux sont nécessaires pour protéger les données durant leur transmission. On parle alors de sécurité des réseaux.

Il n'existe pas de frontières claires entre ces deux formes de sécurité. Par exemple, un des types d'attaque de systèmes d'information les plus médiatisés est le virus (informatique). Un virus peut être physiquement introduit dans un système via une disquette ou via Internet. Dans les deux cas, une fois le virus présent dans le système, des outils informatiques de sécurité sont nécessaires pour le détecter et le détruire.

Pour donner une idée des domaines couverts par la sécurité de l'information, considérons les exemples de violation de sécurité suivants :

- L'utilisateur A transmet un fichier à l'utilisateur B. Ce fichier contient une information sensible (par exemple, un projet de brevet) qui doit être protégée de toute divulgation. L'utilisateur C, qui n'est pas autorisé à consulter ce fichier, est en mesure de contrôler la transmission et de capturer une copie du fichier durant sa transmission ;
- Une application de gestion de réseau D transmet un message à un ordinateur E sous son contrôle. Le message ordonne à l'ordinateur E de mettre à jour un fichier d'autorisation pour inclure l'identité de nouveaux utilisateurs devant avoir accès à cet ordinateur. L'utilisateur F intercepte le message, altère son

contenu en ajoutant ou détruisant des entrées, et fait suivre le message à E, qui l'accepte comme issu du gestionnaire D et met à jour ses fichiers d'autorisation en conséquence ;

- Plutôt que d'intercepter un message, l'utilisateur F construit son propre message avec l'entrée désirée et le transmet à E comme s'il venait du gestionnaire D. L'ordinateur E accepte le message et met à jour ses fichiers d'autorisation ;
- Un employé est victime d'un licenciement qu'il juge abusif. Le responsable du personnel envoie un message à un serveur afin de supprimer son compte. Lorsque la suppression est accomplie, le serveur poste une note à destination du dossier de l'employé pour confirmation de l'action. L'employé est en mesure d'intercepter le message, de le retarder et d'accéder au serveur afin d'y extraire une information sensible. Le message est ensuite réémis, l'action a lieu et la confirmation postée. L'acte de l'employé peut demeurer indétecté pendant un temps considérable ;
- Un client envoie un message à un agent de change avec l'instruction de diverses transactions. Par la suite, l'investissement perd de la valeur et le client nie avoir envoyé le message.

Cette liste n'étant pas exhaustive, elle illustre l'étendue des préoccupations en matière de sécurité des réseaux, que nous abordons dans ce cours en mettant en exergue les principes fondamentaux de la cryptographie.

La sécurité inter-réseau est tout à la fois fascinante et complexe, notamment pour les raisons suivantes :

- La sécurité impliquant communications et réseaux n'est pas aussi simple que pourrait le croire un novice. Les exigences semblent simples. En effet, celles concernant les services de sécurité peuvent se passer d'explication : confidentialité, authentification, non-répudiation, intégrité. Mais les mécanismes utilisés pour satisfaire ces exigences peuvent être très complexes (par exemple, les cryptosystèmes qui sont basés sur des propriétés mathématiques), et leur compréhension nécessiter des raisonnements subtils ;
- En développant un mécanisme ou un algorithme de sécurité particulier, on doit toujours considérer les contre-mesures potentielles. Dans bien des cas, les

contre-mesures sont conçues en considérant le problème différemment, par conséquent en exploitant une faiblesse inattendue du mécanisme ;

- Du fait du point précédent, les procédures utilisées pour fournir un service particulier ne sont pas toujours intuitives. Il n'est pas évident, à partir d'une exigence donnée, de faire le lien avec les mesures compliquées nécessaires à sa réalisation. C'est seulement lorsque les contre-mesures sont considérées que la mesure utilisée prend tout son sens ;
- Une fois conçus divers mécanismes de sécurité, il est nécessaire de décider de leur utilisation. Cela est vrai tant en termes d'emplacement physique (par exemple, à quel point, dans un réseau, certains mécanismes de sécurité sont requis) que de sens logique (à quelle(s) couche(s) d'une architecture telle que TCP/IP placer certains mécanismes) ;
- Les mécanismes de sécurité impliquent habituellement plus d'un algorithme ou protocole. Ils requièrent également que les participants soient en possession d'une information secrète (par exemple, une clef de déchiffrement), ce qui soulève des questions concernant la création, la distribution et la protection de cette information secrète. Le degré de confiance dans les protocoles de communication, dont les comportements peuvent compliquer le développement de mécanismes de sécurité, est un autre souci. Par exemple, si le fonctionnement correct du mécanisme de sécurité requiert de préciser des limites temporelles sur le temps de transit d'un message, alors un protocole ou un réseau qui introduirait des délais variables ou imprévisibles peut rendre ces critères temporels caducs ;

Ainsi, il y a beaucoup de concepts et d'idées à considérer. Néanmoins, ces exemples plantent le décor de l'aboutissement de tous les concepts que nous exploitons dans ce cours, et qui constituent ainsi les noeuds de ce document.

Nous commençons dans ce chapitre, par des notions générales et considérations fondamentales de la sécurité informatique, avant d'approfondir les cryptosystèmes en cryptographie.

2. Sécurité

Pour mieux comprendre la sécurité informatique, il sied de partir de ses définitions basiques ou classiques. Il existe bien évidemment plusieurs connotations au terme « sécurité ». Nous en choisissons quelques-unes pour situer le périmètre de ce cours.

- Le terme « sécurité » peut être compris comme un état d'esprit confiant et tranquille qui résulte du sentiment, bien ou mal fondé, que l'on est à l'abri de tout danger et risqué.
- C'est aussi l'absence de menaces, de difficultés pour une personne ou un groupe social dans un domaine particulier.
- C'est le caractère qui procure un état d'esprit confiant et tranquille. Un sentiment d'assurance et de sûreté.
- C'est aussi le caractère de ce qui est dépourvu de risques, exempt de danger, de ce qui s'effectue sans problèmes.

En informatique ou dans notre contexte numérique, on comprendra la sécurité informatique comme l'ensemble des moyens mis en œuvre pour réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles.

En termes propres, le terme sécurité en informatique, surtout en ce qui concerne les systèmes d'information, est l'ensemble de politiques et les procédures qui permettent d'éviter les intrusions (confidentialité), les incohérences (intégrité) et les pannes (disponibilité) des systèmes d'information, et qui définissent les règles d'authentification. [Nous exploitons en détails les termes soulignés un peu plus tard dans ce chapitre pour parler des composants ou objectifs de la sécurité]

La sécurité des systèmes d'information est un domaine particulièrement stratégique de la sécurité, car, à travers les systèmes de contrôle, les systèmes de gestion, et d'une façon générale à travers l'ingénierie des systèmes, elle doit s'intéresser à l'interopérabilité des systèmes, et faire en sorte que la sécurité soit obtenue au travers de standards et de normes de description des structures de données.

La réalisation de services de sécurité, tels que ceux de gestion des identités, de contrôle d'accès, de détection d'intrusion par exemple, contribue à satisfaire des exigences de sécurité pour protéger des infrastructures numériques. Ce sont des approches complémentaires d'ingénierie et de gestion de la sécurité informatique qui permettent d'offrir un niveau de sécurité cohérent au regard de besoins de sécurité.

3. Sécurisation des Organisations

Pour considérer efficacement les besoins de sécurité d'une organisation et évaluer et choisir les nombreux produits et politiques de sécurité, le responsable de la sécurité a besoin de moyens systématiques de définition des exigences de sécurité et de caractérisation des approches qui satisfont ces exigences. Une approche possible est de considérer trois aspects de la sécurité de l'information :

- Services de sécurité : un service qui améliore la sécurité des systèmes informatiques et des transferts d'information d'une organisation. Les services sont conçus pour contrer les attaques de sécurité, et ils utilisent un ou plusieurs mécanismes de sécurité ;
- Mécanismes de sécurité : un mécanisme est conçu pour détecter, prévenir ou rattraper une attaque de sécurité ;
- Attaque de sécurité : une action qui compromet la sécurité de l'information possédée par une organisation.

3.1. Services

On peut penser aux services de sécurité de l'information par analogie avec les types de fonctions associées aux documents physiques. La plupart des activités humaines, dans des domaines aussi divers que le commerce, la politique étrangère, les actions militaires, dépendent de l'utilisation de documents et de la confiance des deux parties en l'intégrité de ces documents.

Les documents portent signatures et dates ; ils peuvent nécessiter une protection contre la divulgation, la falsification ou la destruction ; être attestés, enregistrés, etc. À mesure que les systèmes d'information deviennent plus diffus et essentiels à la conduite des affaires humaines, l'information électronique prend en charge bien des rôles traditionnellement dévolus aux documents papier. En conséquence, les fonctions associées aux documents papier doivent être accomplies sur des documents au format dématérialisé.

Plusieurs aspects propres aux documents électroniques font qu'assurer ces fonctions ou services est un défi :

- Il est habituellement possible de distinguer entre un document papier original et sa photocopie. Cependant, un document électronique est purement une séquence de bits ; il n'y a pas de différence entre "l'original" et toutes ses copies ;
- Une altération d'un document papier peut laisser des preuves physiques. Par exemple, un effacement peut laisser une tache ou une surface rugueuse. L'altération de bits dans une mémoire d'ordinateur ou un signal ne laisse a priori aucune trace ;
- Tout processus de "preuve" associé à un document physique dépend des caractéristiques physiques du document (par exemple, la forme d'une signature manuelle ou un tampon de notaire). De telles preuves d'authenticité d'un document électronique doivent être basées sur des signes présents dans l'information elle-même.

3.2. Mécanismes

Un seul mécanisme ne peut fournir tous les services de sécurité. On peut noter qu'un élément particulier sous-tend la plupart des mécanismes de sécurité en usage : les techniques cryptographiques.

Le chiffrement - ou des transformations similaires - de l'information est le moyen le plus courant pour fournir une sécurité. Ainsi, dans ce cours on insistera sur le développement, l'utilisation et la gestion de ces techniques.

3.3. Attaques

La sécurité de l'information traite des mesures de prévention d'intrusions, de protection contre le vol, la falsification et la destruction d'information. Des mesures de sécurité dans une organisation concernent aussi la prévention de la fraude, ou, à défaut, de sa détection dans des systèmes d'information et la protection contre la mise hors service de ressources. Ces cas représentent, en effet, des formes d'attaques contre lesquelles une organisation devra se protéger.

Une attaque est réalisée par un ou des agresseurs. Elle se définit par : son origine (qui ou quoi), son commanditaire, sa cible (qui ou quoi), son objectif (pourquoi) et ses moyens (techniques, humains, organisationnels, financiers, temporels). La nature de ces attaques varie considérablement selon les circonstances. Il est possible d'approcher le problème en examinant les types génériques d'attaques pouvant être rencontrées dans une organisation numérisée.

3.3.1. Quelques exemples d'attaques (ou scenarios d'attaques) génériques

- Obtenir un accès non autorisé à l'information (c'est-à-dire, violer secret ou confidentialité) ;
- Usurper l'identité d'un autre utilisateur pour modifier ses attributs de responsabilité ou pour utiliser les droits de ce dernier dans le but de :
 - diffuser une information frauduleuse ;
 - modifier une information légitime ;
 - utiliser une identité frauduleuse pour obtenir un accès non autorisé ;
 - faciliter des transactions frauduleuses ou en tirer parti.
- Refuser la responsabilité d'une information que le fraudeur a diffusée ;
- Prétendre avoir reçu de la part d'un autre utilisateur une information en fait créée par le fraudeur (par exemple, de fausses attributions de responsabilité ou de confiance).

- Prétendre avoir envoyé (à un moment donné) une information qui soit n'a pas été envoyée, soit l'a été à un autre moment ;
- Nier avoir reçu une information ou prétendre qu'elle a été reçue à un autre moment ;
- Étendre des droits d'un fraudeur (pour un accès à des informations) ;
- Modifier (sans autorisation) les droits d'autrui (les inscrire, restreindre ou élargir leurs droits, etc.) ;
- Dissimuler la présence d'information (la communication cachée) dans une autre information (la communication déclarée) ;
- S'insérer dans un lien de communication entre d'autres utilisateurs en tant que point de relai actif (et indétecté) ;
- Apprendre qui a accès à une information donnée (fichiers, etc.) et quand les accès sont réalisés, même si l'information elle-même reste cachée (par exemple, la généralisation de l'analyse de trafic de canaux de communication à des bases de données, des logiciels, etc.)
- Mettre en cause l'intégrité d'un protocole en révélant une information que le fraudeur est censé (selon les termes du protocole) garder secrète ;
- Pervertir la fonction d'un logiciel par l'ajout d'une fonction cachée ;
- Faire qu'autrui viole un protocole en introduisant une information incorrecte ;
- Saper la confiance en un protocole en causant des défaillances visibles dans le système ;
- Empêcher la communication entre d'autres utilisateurs, en particulier par des interférences afin que la communication authentique soit rejetée comme non authentique.

3.3.2. Attaques de systèmes d'information

Les attaques portées à la sécurité d'un ordinateur ou d'un réseau sont mieux caractérisées en considérant le système en tant que fournisseur d'information. En général, il existe un flot d'information issu d'une source - un fichier ou une zone de la mémoire centrale -, vers une destination - un autre fichier ou utilisateur.

Il existe quatre catégories d'attaques : interruption, interception, modification, fabrication.

Interruption

Un atout du système est détruit ou devient indisponible ou inutilisable. C'est une attaque portée à la disponibilité. La destruction d'une pièce matérielle (tel un disque

dur), la coupure d'une ligne de communication, ou la mise hors service d'un système de gestion de fichiers en sont des exemples.

Interception

Une tierce partie non autorisée obtient un accès à un atout. C'est une attaque portée à la confidentialité. Il peut s'agir d'une personne, d'un programme ou d'un ordinateur. Une écoute téléphonique dans le but de capturer des données sur un réseau, ou la copie non autorisée de fichiers ou de programmes en sont des exemples.

Modification

Une tierce partie non autorisée obtient accès à un atout et le modifie de façon (presque) indéetectable. Il s'agit d'une attaque portée à l'intégrité. Changer des valeurs dans un fichier de données, altérer un programme de façon à bouleverser son comportement ou modifier le contenu de messages transmis sur un réseau sont des exemples de telles attaques.

Fabrication

Une tierce partie non autorisée insère des contrefaçons dans le système. C'est une attaque portée à l'authenticité. Il peut s'agir de l'insertion de faux messages dans un réseau ou l'ajout d'enregistrements à un fichier.

3.3.3. Attaques passives et attaques actives

Il peut être utile de distinguer deux catégories d'attaques : les attaques passives et les attaques actives.

Passives

Écoutes indiscrètes ou surveillance de transmissions sont des attaques de nature passive. Le but de l'adversaire est d'obtenir une information qui a été transmise. Ces attaques passives sont la capture du contenu d'un message et l'analyse de trafic. La capture du contenu de messages est facilement compréhensible. Une conversation téléphonique, un courrier électronique ou un fichier transféré peuvent contenir une information sensible ou confidentielle.

La seconde attaque passive, l'analyse de trafic, est plus subtile. Supposons qu'un moyen de masquer le contenu des messages ou des informations soit à disposition (par exemple, un système de chiffrement), de sorte que les adversaires, même en cas de capture, ne pourront en extraire l'information contenue. Cependant l'adversaire pourra être en mesure d'observer le motif de ces messages, déterminer l'origine et l'identité des systèmes en cours de communication, et observer la fréquence et la

longueur des messages échangés. Cette information peut être utile pour deviner la nature de la communication. Les attaques passives sont très difficiles à détecter car elles ne causent aucune altération des données

Actives

La seconde catégorie d'attaques est l'attaque active. Ces attaques impliquent certaines modifications du flot de données ou la création d'un flot frauduleux ; elles peuvent être subdivisées en quatre catégories : mascarade, rejeu, modification de messages et déni de service. Une mascarade a lieu lorsqu'une entité prétend être une autre entité. Une attaque de ce type inclut habituellement une des autres formes d'attaque active. Par exemple, des séquences d'authentification peuvent être capturées et rejouées, permettant ainsi à une entité autorisée munie de peu de priviléges d'en obtenir d'autres en usurpant une identité possédant ces priviléges. Le rejeu implique la capture passive de données et leur retransmission ultérieure en vue de produire un effet non autorisé.

La modification de messages signifie que certaines portions d'un message légitime sont altérées ou que les messages sont retardés ou réorganisés. Par exemple, le message " autoriser X à lire le fichier confidentiel comptes " est modifié en " autoriser Y à lire le fichier confidentiel comptes ". Le déni de service empêche l'utilisation normale ou la gestion de fonctionnalités de communication. Cette attaque peut avoir une cible spécifique ; par exemple, une entité peut supprimer tous les messages dirigés vers une destination particulière. Une autre forme de refus de service est la perturbation d'un réseau dans son intégralité, soit en mettant hors service le réseau, soit en le surchargeant de messages afin de dégrader ses performances.

4. Menaces, risques et vulnérabilités

La sécurité informatique est une discipline de première importance. Particulièrement, la sécurité des systèmes d'information (SI) car le système d'information (SI) est pour toute entreprise un élément absolument vital. En effet, l'information sous plusieurs formes étant au cœur des infrastructures informatiques, sa protection et sécurisation constituent un investissement de grande envergure pour même l'existence et la survie des organisations numérisées.

Conjurer les menaces contre le SI est devenu impératif. Les menaces contre le système d'information entrent dans l'une des catégories suivantes : atteinte à la disponibilité des systèmes et des données, destruction de données, corruption ou falsification de données, vol ou espionnage de données, usage illicite ou sabotage d'un système ou d'un réseau, usage d'un système compromis pour attaquer d'autres cibles. La

falsification de sites web aux fins de détournement de fonds est aujourd’hui la forme d’attaque qui connaît l’expansion la plus rapide.

On peut donc comprendre que ces menaces guettent tout ce qui est connecté sur un réseau ; ainsi donc, tout ce qui est connecté sur un réseau est vulnérable.

Les menaces engendrent des risques et des coûts humains et financiers : perte de confidentialité de données sensibles, indisponibilité des infrastructures et des données, dommages pour le patrimoine intellectuel et la notoriété. Les risques peuvent se réaliser si les systèmes menacés présentent des vulnérabilités.

Il est possible de préciser la notion de risque en la décrivant comme le produit d’un préjudice par une probabilité d’occurrence :

$$\text{risque} = \text{préjudice} \times \text{probabilité d'occurrence}$$

Cette formule exprime qu’un événement dont la probabilité à survenir est assez élevée, par exemple la défaillance d’un disque dur, mais dont il est possible de prévenir le préjudice qu’il peut causer par des sauvegardes régulières, représente un risque acceptable ; il en va de même pour un événement à la gravité imparable, comme l’impact d’une météorite de grande taille, mais à la probabilité d’occurrence faible. Il va de soi que, dans le premier cas, le risque ne devient acceptable que si les mesures de prévention contre le préjudice sont effectives et efficaces : cela irait sans dire, si l’oubli de cette condition n’était très fréquent.

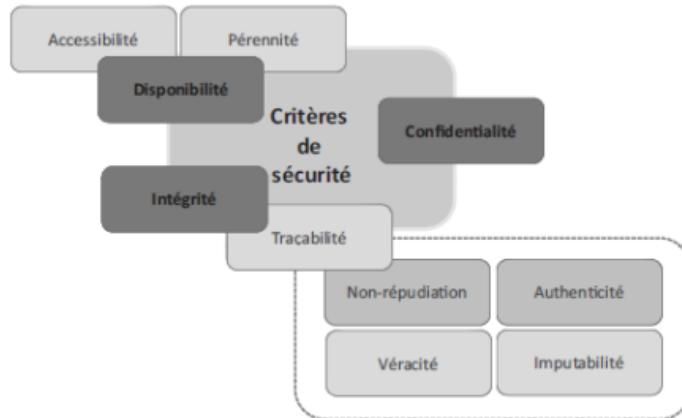
Si la question de la sécurité des systèmes d’information a été radicalement bouleversée par l’évolution rapide de l’Internet, elle ne saurait s’y réduire ; il s’agit d’un vaste problème dont les aspects techniques ne sont qu’une partie. Les aspects juridiques, sociaux, ergonomiques, psychologiques et organisationnels sont aussi importants, sans oublier les aspects immobiliers.

5. Composants de la Sécurité

Quand on parle des composants de la sécurité informatique, on se réfère aux ingrédients qui nous permettent de définir ses objectifs. Nous avions déjà introduit ces termes-clés à la deuxième section de ce chapitre.

En effet, La notion de sécurité fait référence à la propriété d’un système, qui s’exprime généralement en termes de **disponibilité** (D), **d’intégrité** (I) et de **confidentialité** (C). Ces critères de base (dits critères DIC) sont des objectifs de sécurité que la mise en œuvre de fonctions de sécurité permet d’atteindre. Des fonctions additionnelles peuvent offrir des services complémentaires pour confirmer la véracité ou l’authenticité d’une action ou d’une ressource (notion d’**authentification**) ou encore

pour prouver l'existence d'une action à des fins de **non-répudiation** ou **d'imputabilité**, ou de **tracabilité**.



Critères de sécurité

5.1. Disponibilité

La disponibilité d'une ressource est relative à la période de temps pendant laquelle le service qu'elle offre est opérationnel. Le volume potentiel de travail susceptible d'être pris en charge durant la période de disponibilité d'un service détermine la capacité d'une ressource à être utilisée (serveur ou réseau par exemple).

Il ne suffit pas qu'une ressource soit disponible, elle doit pouvoir être utilisable avec des temps de réponse acceptables. Sa disponibilité est indissociable de sa capacité à être accessible par l'ensemble des ayants droit (notion d'accessibilité).

La disponibilité des services, systèmes et données est obtenue par un dimensionnement approprié et une certaine redondance des infrastructures ainsi que par une gestion opérationnelle et une maintenance efficaces des infrastructures, ressources et services.

De nombreuses attaques peuvent résulter en une perte ou une réduction de la disponibilité d'un service ou d'un système. Certaines de ces attaques sont susceptibles d'être l'objet de contre-mesures automatiques, telle que l'authentification et le chiffrement, alors que d'autres exigent une action humaine pour prévenir ou se rétablir de la perte de disponibilité des éléments d'un système.

Des pertes de données, donc une indisponibilité de celles-ci, peuvent être possibles si les procédures de sauvegarde et de restitution ainsi que les supports de mémorisation associés ne sont pas gérés correctement.

Une politique de sauvegarde ainsi qu'un arbitrage entre le coût de la sauvegarde et celui du risque d'indisponibilité supportable par l'organisation doivent être préalablement établis pour que la mise en œuvre des mesures techniques soit efficace et pertinente et que les utilisateurs sachent quelles sont les procédures à suivre.

5.2. Intégrité

Le critère d'intégrité des ressources physiques et logiques (équipements, données, traitements, transactions, services) est relatif au fait qu'elles sont demeurées intactes, qu'elles n'ont pas été détruites (altération totale) ou modifiées (altération partielle) à l'insu de leurs propriétaires tant de manière intentionnelle qu'accidentelle. Une fonction de sécurité appliquée à une ressource pour préserver son intégrité permet de la protéger contre une menace de corruption ou de destruction.

Se prémunir contre l'altération des données et avoir la certitude qu'elles n'ont pas été modifiées lors de leur stockage, de leur traitement ou de leur transfert contribue à la qualité des prises de décision basées sur celles-ci.

Les critères de disponibilité et d'intégrité sont à satisfaire par des mesures appropriées afin de pouvoir atteindre un certain niveau de confiance dans les contenus et le fonctionnement des infrastructures informatiques et télécoms.

L'intégrité des données relève essentiellement de problématiques liées au transfert de données, elle dépend également des aspects purement informatiques de traitement de l'information (logiciels d'application, systèmes d'exploitation, environnements d'exécution, procédures de sauvegarde, de reprise et de restauration des données).

Des contrôles d'intégrité peuvent être effectués pour s'assurer que les données n'ont pas été modifiées lors de leur transfert par des attaques informatiques qui les interceptent et les transforment (notion d'écoutes actives). En revanche, ils seront de peu d'utilité pour détecter des écoutes passives, qui portent atteinte non à l'intégrité des données mais à leur confidentialité. En principe, lors de leur transfert, les données ne sont pas altérées par les protocoles de communication qui les véhiculent en les encapsulant. L'intégrité des données peut être prouvée par les mécanismes de signature électronique.

5.3. Confidentialité

La notion de confidentialité est liée au maintien du secret, elle est réalisée par la protection des données contre une divulgation non autorisée (notion de protection en lecture).

Il existe deux types d'actions complémentaires permettant d'assurer la confidentialité des données :

- Limiter et contrôler leur accès afin que seules les personnes habilitées à les lire ou à les modifier puissent le faire ;
- Les rendre inintelligibles en les chiffrant de telle sorte que les personnes qui ne sont pas autorisées à les déchiffrer ne puissent les utiliser.

La confidentialité est la protection contre les attaques passives des données transmises. Plusieurs niveaux de protection de la confidentialité sont envisageables. Le service le plus général protège toutes les données transmises entre deux utilisateurs pendant une période donnée. Des formes restreintes de ce service peuvent également être définies, incluant la protection d'un message élémentaire ou même de champs spécifiques à l'intérieur d'un message. Un autre aspect de la confidentialité est la protection du flot de trafic contre l'analyse. Cela requiert qu'un attaquant ne puisse observer les sources et destinations, les fréquences, longueurs ou autres caractéristiques du trafic existant sur un équipement de communication.

5.4. Concepts additionnels

Identifier l'auteur présumé d'un tableau signé est une chose, s'assurer que le tableau est authentique en est une autre. Il en est de même en informatique, où des procédures d'identification et d'authentification peuvent être mises en œuvre pour contribuer à réaliser des procédures de contrôle d'accès et des mesures de sécurité assurant :

- La **confidentialité** et l'**intégrité** des données : seuls les ayants droit identifiés et authentifiés peuvent accéder aux ressources (contrôle d'accès) et les modifier s'ils sont habilités à le faire ;
- La **non-répudiation** et l'**imputabilité** : seules les entités identifiées et authentifiées ont pu réaliser une certaine action (preuve de l'origine ou de la destination d'un message, par exemple). L'identification et l'authentification des ressources et des utilisateurs permettent d'imputer la responsabilité de la réalisation d'une action à une entité qui pourra en être tenue responsable et devra éventuellement en rendre compte.

Tous les mécanismes de contrôle d'accès logique aux ressources informatiques nécessitent de gérer l'identification, l'authentification des entités et la gestion des droits et permissions associés. C'est également sur la base de l'identification des personnes et des accès aux ressources que s'établissent des fonctions de facturation et de surveillance.



Identification et authentification

La non-répudiation est le fait de ne pouvoir nier ou rejeter qu'un événement (action, transaction) a eu lieu. À ce critère de sécurité peuvent être associées les notions d'imputabilité, de traçabilité ou encore parfois d'audibilité.

Attribuer une action à une entité déterminée (ressource ou personne) relève de l'imputabilité, qui peut être réalisée par un ensemble de mesures garantissant l'enregistrement fiable d'informations pertinentes relatives à un événement.

La traçabilité permet de reconstituer une séquence d'événements à partir des données numériques laissées dans les systèmes lors de leurs réalisations. Cette fonction comprend l'enregistrement des opérations, de la date de leur réalisation et leur imputation.

L'audibilité d'un système se définit par sa capacité à garantir la présence d'informations nécessaires à une analyse, postérieure à la réalisation d'un événement (courant ou exceptionnel), effectuée dans le cadre de procédures de contrôle et d'audit. L'audit peut être mis en œuvre pour diagnostiquer ou vérifier l'état de la sécurité d'un système, pour déterminer s'il y a eu ou non violation de la politique de sécurité.

6. Typologies d'attaques Informatiques

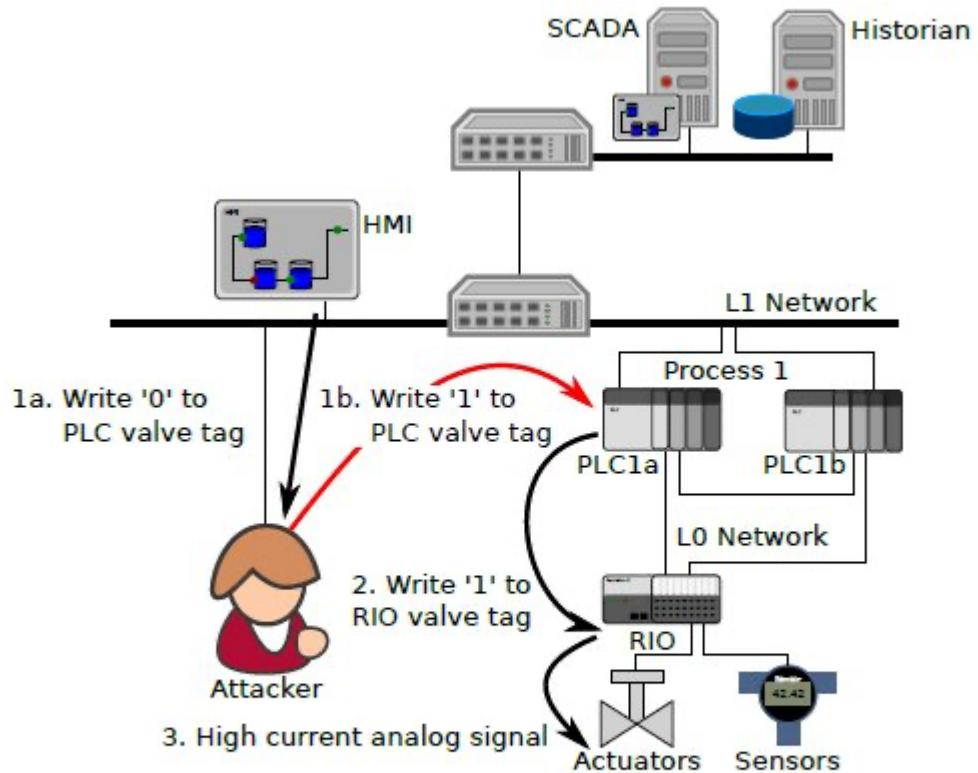
Il existe plusieurs types d'attaques contre lesquelles des mesures sécuritaires devront faire face. Dans cette section, nous présentons quelques-unes d'attaques typiques.

6.1. Bombe logique

Une Bombe logique est une partie d'un programme malveillant (virus, cheval de Troie, etc.) qui reste dormante dans le système hôte jusqu'à ce qu'un instant ou un événement survienne, ou encore que certaines conditions soient réunies, pour déclencher des effets dévastateurs en son sein. Le virus Tchernobyl, qui fut l'un des virus les plus destructeurs, avait une bombe logique qui s'est activée le 26 avril 1999, jour du treizième anniversaire de la catastrophe nucléaire de Tchernobyl.



Illustration d'une Bombe Logique



Exécution d'une attaque de Bombe Logique de manipulation de données de capteurs pour insérer des codes pouvant déclencher un dysfonctionnement.

6.2. Cheval de Troie

Un Cheval de Troie (trojan en anglais) est un programme effectuant une fonction illicite tout en donnant l'apparence d'effectuer une fonction légitime. La fonction illicite peut consister en la divulgation ou l'altération d'informations. Trojan.ByteVerify est un cheval de Troie sous forme d'une applet java. Ce cheval de Troie exploite une vulnérabilité de la machine virtuelle java de Microsoft permettant à un pirate d'exécuter du code arbitraire sur la machine infectée. Par exemple, Trojan.ByteVerify peut modifier la page d'accueil d'Internet Explorer.

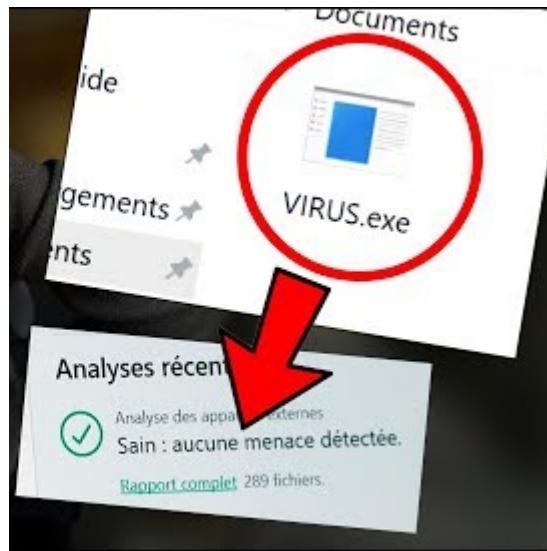
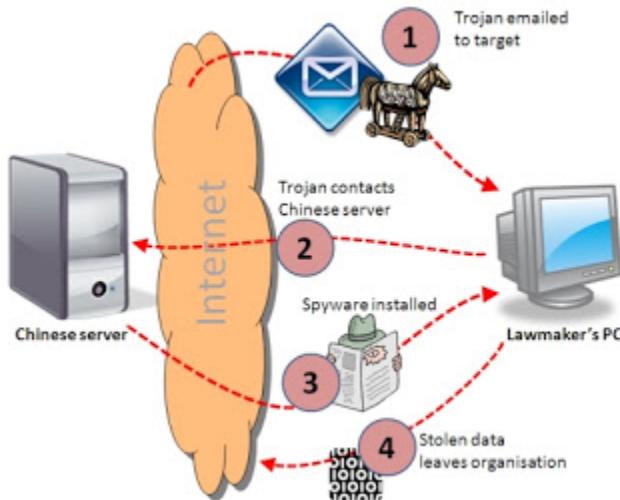


Illustration Cheval de Troie (Virus.exe) apparaissant comme un programme légitime



Scenario de mise en scène d'une attaque de Cheval de Troie

Dans ce scenario, une opération d'attaque se passe sur internet compromettant deux entités (victimes) :

1. Un courriel infecté par des agresseurs est envoyé à un législateur (Lawmaker)
2. De là, un contact est établi avec un serveur chinois
3. Le serveur réagit en envoyant des données au « demandeur » (Lawmaker)
4. De là, les données volées sont récupérées par les agresseurs

6.3. Porte dérobée

Une porte dérobée (ou backdoor en anglais) est un moyen de contourner les mécanismes de contrôle d'accès. Il s'agit d'une faille du système de sécurité due à une faute de conception accidentelle ou intentionnelle (cheval de Troie en particulier). C'est donc une fonctionnalité inconnue de l'utilisateur légitime qui donne un accès secret au logiciel.

Une porte dérobée a été découverte dans le SGBD interbase de Borland au début des années 2000. Il suffisait d'entrer le nom d'utilisateur "politically" et le mot de passe "correct" pour se connecter à la base de données avec les droits d'administrateur.

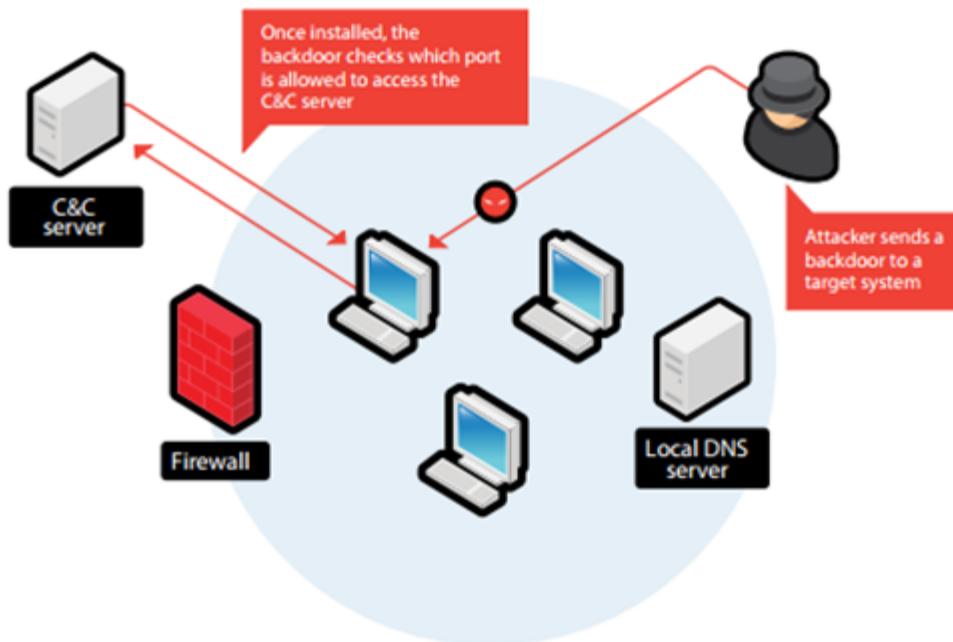


Illustration d'un agresseur violant accès au réseau d'une entreprise avec porte dérobée

6.4. Virus

Un virus est un segment de programme qui, lorsqu'il s'exécute, se reproduit en s'adjoignant à un autre programme (du système ou d'une application), et qui devient

ainsi un cheval de Troie. Puis le virus peut ensuite se propager à d'autres ordinateurs (via un réseau) à l'aide du programme légitime sur lequel il s'est greffé. Il peut également avoir comme effets de nuire en perturbant plus ou moins gravement le fonctionnement de l'ordinateur infecté.

Psyboot, découvert en 2009, est considéré comme étant le seul virus informatique ayant la capacité d'infecter les routeurs et modems haut-débit.

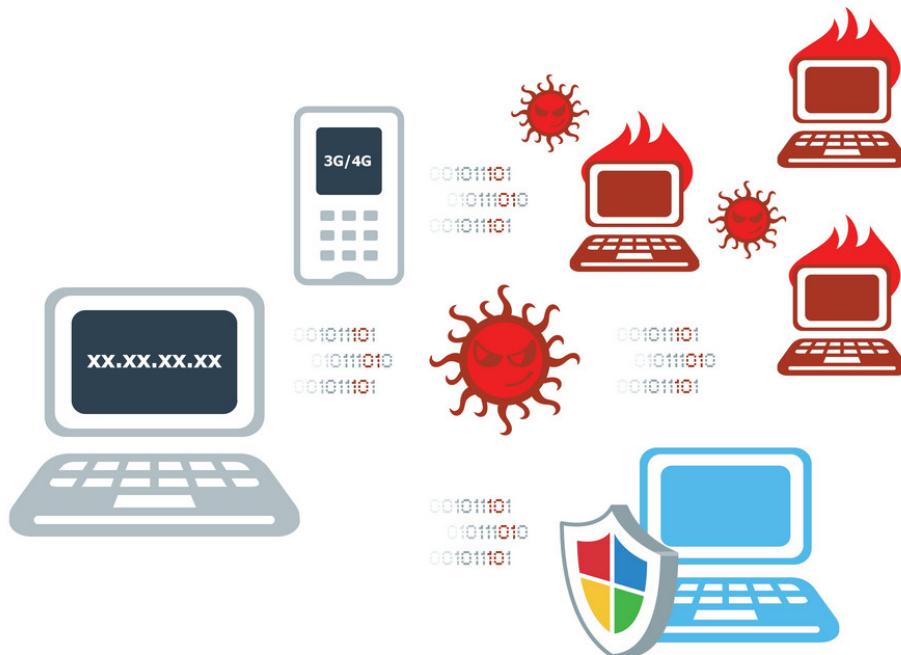


Illustration de propagation d'un virus sur un réseau sans fil avec plusieurs types des dispositifs

6.5. Ver

Un ver est un programme autonome qui se reproduit et se propage à l'insu des utilisateurs. Contrairement aux virus, un ver n'a pas besoin d'un logiciel hôte pour se dupliquer. Le ver a habituellement un objectif malicieux, par exemple :

- Espionner l'ordinateur dans lequel il réside ;
- Offrir une porte dérobée à des pirates informatiques ;
- Détruire des données sur l'ordinateur infecté ;
- Envoyer de multiples requêtes vers un serveur internet dans le but de le saturer.

Le ver Blaster avait pour but de lancer une attaque par déni de service sur le serveur de mises à jour de Microsoft.

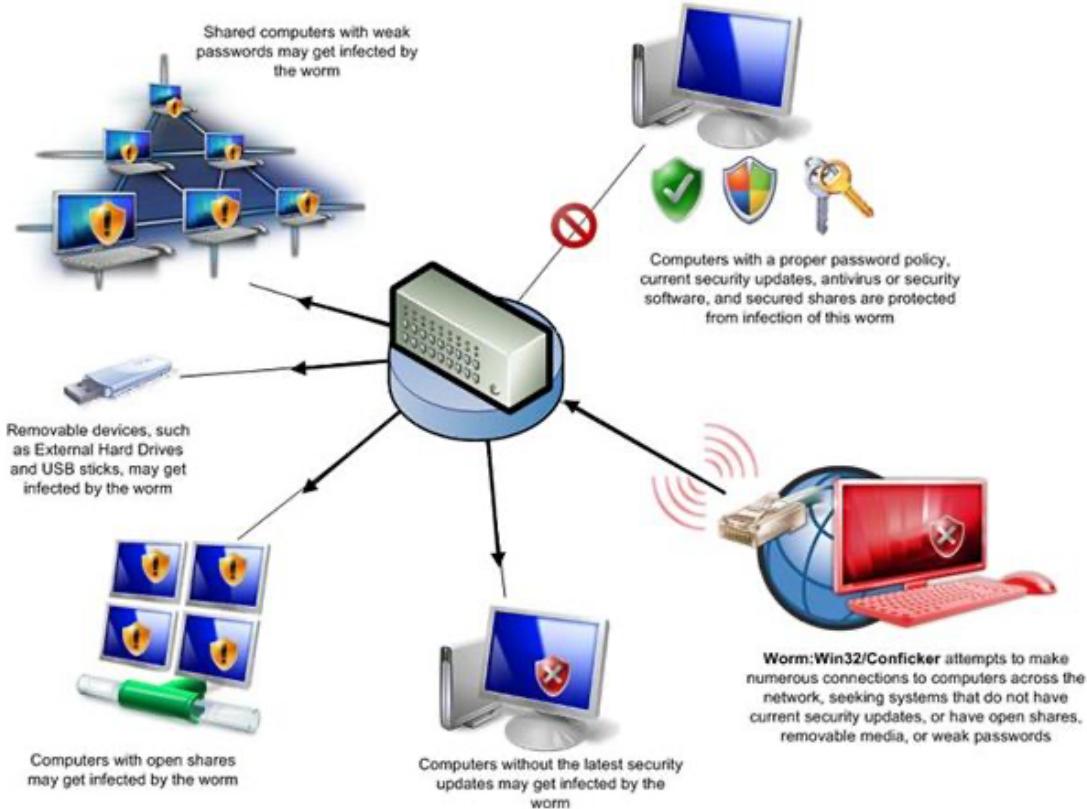
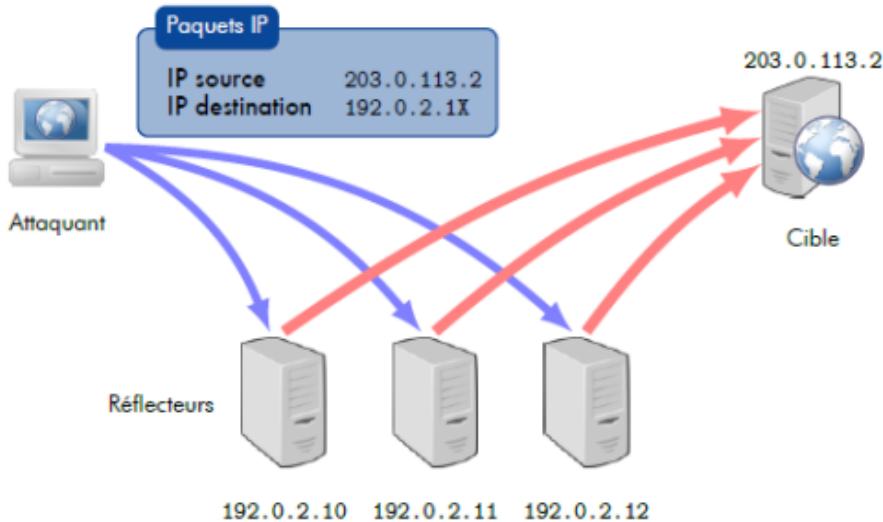


Illustration: Ver Win32/conficker essaie de generer plusieurs connections aux machines sur un reseau n'ayant pas la derniere mises a jour de protection securitaire ou des mots de passe faibles.

6.6. Déni de Service (DDoS)

Une attaque par déni de service vise à rendre indisponible un ou plusieurs services. Un déni de service peut consister à exploiter, par exemple, une vulnérabilité logicielle ou matérielle. L'interruption de service peut également s'effectuer en empêchant l'accès à ce service, par exemple en saturant la bande passante du réseau : on parle alors d'attaques volumétriques. Par ailleurs, une attaque peut solliciter, jusqu'à épuisement, une ou plusieurs ressources d'un service. Il peut s'agir, par exemple, de l'ouverture d'un grand nombre de nouvelles sessions TCP dans un intervalle de temps très court, ou encore d'un nombre trop important de traitements concurrents effectués par une base de données.

On parle de « déni de service distribué » (de l'anglais Distributed Denial of Service ou DDoS) lorsque l'attaque fait intervenir un réseau de machines (souvent compromises) afin d'interrompre le ou les services visés.



Principe d'une attaque de DDoS par réflexion

6.7. Social Engineering

Ce type d'attaque est l'une de plus dangereuse, car elle implique des hommes. En utilisant les moyens usuels (téléphone, courriel...) et en usurpant une identité, un pirate cherche à obtenir des renseignements confidentiels auprès du personnel de l'entreprise en vue d'une intrusion future. Seule une formation du personnel permet de se protéger de cette attaque.

En effet, les gens peuvent être tout aussi dangereux que les systèmes informatiques non protégés. Les gens peuvent être mentis, manipulés, soudoyés, menacés, blessés, torturés, etc. pour donner des informations précieuses et il peut être prouvé que la plupart des humains craquent sous forte pression à moins qu'ils aient été spécialement formés.

Considérez ce scenario :

Quelqu'un vous appelle au milieu de la nuit.

"Avez-vous appelé l'Egypte pour les six dernières heures ?"

"Non"

"Eh bien, nous avons un appel qui est réellement actif en ce moment, c'est sur votre carte d'appel (SIM) et c'est à l'Egypte et en fait, vous avez environ 2000 \$

de frais sur votre carte et ... donnez-moi vos détails (votre numéro de carte, code PIN et PUK) ainsi que votre numéro de carte bancaire puis je vais me débarrasser de la charge pour vous”

Des exemples sont légions, surtout au cours de ces dernières années pendant lesquelles les solutions numériques pullulent nos vies. La solution la plus efficace en ce sens, c'est l'éducation et la formation.

7. Exercices N°1

Pour vous mettre au travail, nous vous demandons de vous familiariser avec deux outils importants en sécurité de réseau : Netkit et Wireshark.

Recommandations :

- Construire un réseau ou avoir accès à un réseau
- Installer VMware ou un environnement virtuel qui vous permettra de simuler un réseau au moyen des machines virtuelles

1. Téléchargez l'outil Netkit

- a. Faites un descriptif de l'outil : ses principales parties et fonctions.
- b. Démontrez avec capture d'écran ses fonctionnalités au moyen de quelques commandes de base suivant un scenario que vous devriez simuler
- c. Démontrez comment vous pouvez commencer, arrêter et pauser un terminal au moyen de commandes Netkit

2. Utilisez et téléchargez si nécessaire l'outil Wireshark

- a. Faites un descriptif de l'outil : ses principales parties et fonctions.
- b. Commencez à capturer le trafic réseau. Pour générer du trafic HTTP, accédez à une page Web avec votre navigateur Web (p. ex. www.radiookapi.net). N'oubliez pas d'arrêter de capturer que vous pouvez obtenir beaucoup de trafic dans votre capture.
- c. Regardez vos paquets capturés et trouvez un paquet HTTP GET. Répondez aux questions suivantes et fournissez les captures d'écran :
 - i. Quelle est la source et la destination de l'adresse MAC de ce paquet HTTP ? Fournir une capture d'écran pour le prouver
 - ii. Quelle est la source et la destination de l'adresse IP de ce paquet HTTP ? Fournir une capture d'écran pour le prouver
 - iii. Quelle est la source et le port de destination de ce paquet HTTP ?

- iv. Quel est le nom d'hôte de ce paquet HTTP GET ? Fournir une capture d'écran pour le prouver
 - v. Recherchez la réponse HTTP appartenant au paquet HTTP GET. Combien de temps s'est écoulé entre la réponse HTTP GET et HTTP ?
3. Qu'avez-vous compris et appris de la sécurité informatique dans ce chapitre ?
4. Comment pourriez-vous relativiser le niveau de numérisation de nos infrastructures face aux défis de sécurité ?

Chapitre 2 : Cryptographie

1. Contexte

En informatique, la quasi-totalité des solutions et mécanismes de protection contre les attaques sont orientés vers la sécurisation des données. Pour des raisons évidentes que vous avez apprises tout au long de votre cursus, l'information est de l'or dans ce domaine.

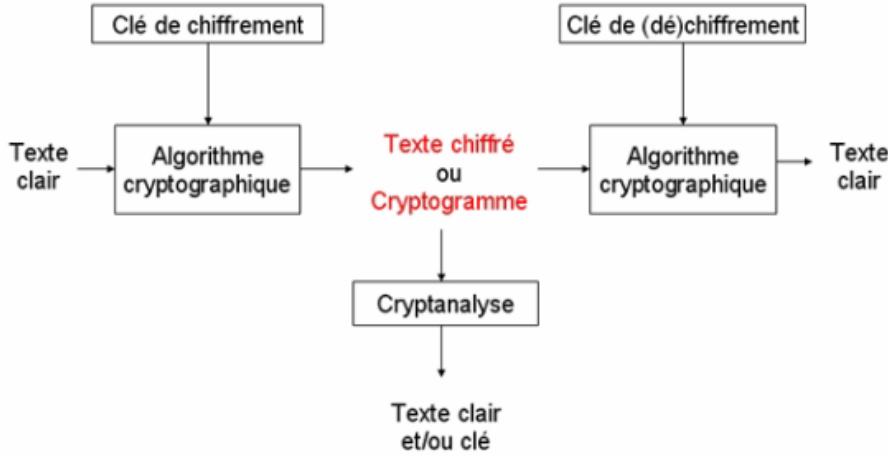
La cryptographie, comme un moyen de protection des données en informatique, constitue le sous-basement sur lequel se fondent toutes ces solutions de sécurité. Pour ce faire, ce chapitre se consacre à poser les jalons par rapport aux concepts basiques de cryptographie que tout académique devra maîtriser.

La cryptographie utilise des concepts issus de nombreux domaines (Informatique, Mathématiques, Electronique). Toutefois, les techniques évoluent et trouvent aujourd'hui régulièrement racine dans d'autres branches (Biologie, Physique, etc.).

2. Concepts basiques

La cryptographie tire aussi ses origines du temps de Jules César. Ce dernier envoyait, ne faisant pas confiance à ses messagers, envoyait des messages à ses généraux en les chiffrant. Il pouvait remplacer, par exemple, tous les A contenus dans ses messages par des D, les B par des E, et ainsi de suite pour tout l'alphabet. Seule la personne connaissant la règle du « décalage par trois » pouvait déchiffrer ses messages. Cette règle devrait donc être considérée comme la clef.

Il s'agit donc de transformer ou encoder des messages (chiffrement) suivant une procédure spécifique (algorithme cryptographique) au moyen d'une clef. Le processus est réversible suivant la même démarche déchiffrement). Il existe un vocabulaire approprié pour capturer tous les concepts importants de la cryptographie :



Protocole de chiffrement

Cryptologie : Il s'agit d'une science mathématique comportant deux branches : la cryptographie et la cryptanalyse

Cryptographie : La cryptographie est l'étude des méthodes donnant la possibilité d'envoyer des données de manière confidentielle sur un support donné.

Chiffrement (ou cryptage) : Le chiffrement consiste à transformer une donnée (texte, message, ...) afin de la rendre incompréhensible par une personne autre que celui qui a créé le message et celui qui en est le destinataire. La fonction permettant de retrouver le texte clair à partir du texte chiffré porte le nom de déchiffrement (ou décryptage).

Texte chiffré : Appelé également cryptogramme, le texte chiffré est le résultat de l'application d'un chiffrement à un texte clair.

Clef : Il s'agit du paramètre impliqué et autorisant des opérations de chiffrement et/ou déchiffrement. Dans le cas d'un algorithme symétrique, la clef est identique lors des deux opérations. Dans le cas d'algorithmes asymétriques, elle diffère pour les deux opérations.

Cryptanalyse : Opposée à la cryptographie, elle a pour but de retrouver le texte clair à partir de textes chiffrés en déterminant les failles des algorithmes utilisés.

Cryptosystème : Il est défini comme l'ensemble des clés possibles (espace de clés), des textes clairs et chiffrés possibles associés à un algorithme donné.

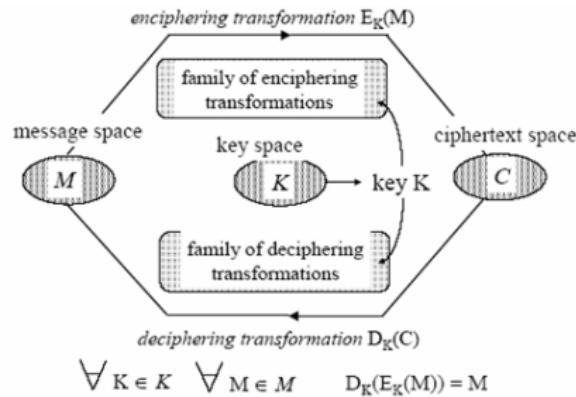


Illustration Schéma d'un Cryptosystème.

On emploie également parfois les termes "cryptage" et "crypter" pour qualifier l'action de chiffrer un message. Les mots "encryptage" et "(en)cryptement" sont des anglicismes dérivés du verbe "to encrypt".

3. Cryptage et décryptage

Nous l'avons dit dans la section précédente, mis ensemble, il est à noté que les données lisibles et compréhensibles sans intervention spécifique sont considérées comme du texte en clair. La méthode permettant de dissimuler du texte en clair en masquant son contenu est appelée le cryptage. Le cryptage consiste à transformer un texte normal en charabia inintelligible appelé texte chiffré.

Cette opération permet de s'assurer que seules les personnes auxquelles les informations sont destinées pourront y accéder. Le processus inverse de transformation du texte chiffré vers le texte d'origine est appelé le décryptage.



Illustration de cryptage et décryptage

Il existe deux principales catégories de cryptage : le cryptage symétrique ou cryptage de clé secrète et le cryptage asymétrique ou cryptage de clé publique.

3.1. Cryptage symétrique

3.1.1. Description

Avec le cryptage symétrique ou cryptage de clé secrète, une seule clé suffit pour le cryptage et le décryptage comme l'illustre l'image ci-dessous. La norme de cryptage de données (DES) ou l'algorithme DES est un exemple de système de cryptage symétrique.

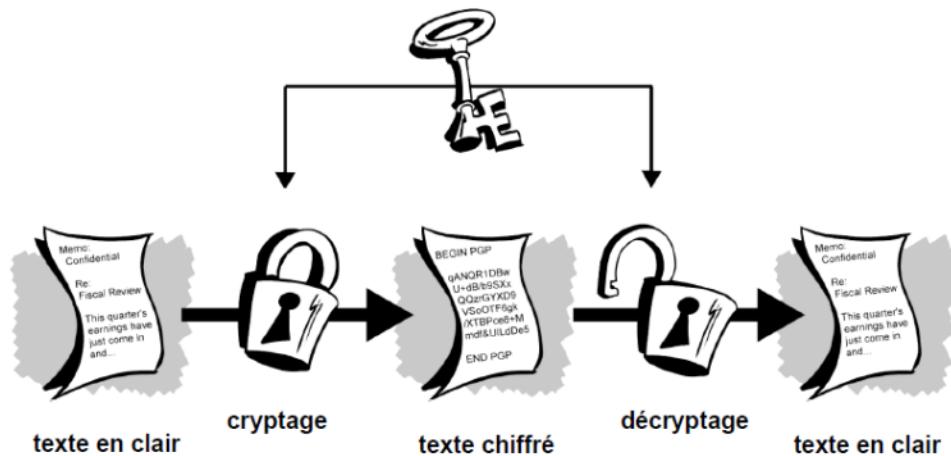


Illustration Cryptage symétrique

Ce cryptage comporte des avantages. Il est très rapide. Mais, il s'avère particulièrement utile pour les données véhiculées par des moyens de transmission sécurisés. Toutefois, il peut entraîner des coûts importants en raison de la difficulté à garantir la confidentialité d'une clé de cryptage lors de la distribution.

Par exemple, un expéditeur et un destinataire souhaitant communiquer de manière sécurisée à l'aide du cryptage symétrique doivent convenir d'une clé et ne pas la divulguer. S'ils se trouvent à des emplacements géographiques différents, ils doivent faire confiance à un coursier, au téléphone de Batman ou à tout autre moyen de communication sécurisé pour éviter la divulgation de la clé secrète lors de la transmission. Toute personne interceptant la clé lors d'un transfert peut ensuite lire, modifier et falsifier toutes les informations cryptées ou authentifiées avec cette clé. De la norme de cryptage de données DES au code secret de Jules César, la distribution des clés reste le problème majeur du cryptage symétrique.

3.1.2. Caractéristiques

- La clé doit rester secrète,
- Les algorithmes les plus répandus sont le DES, AES, 3DES, ...
- Au niveau de la génération des clés, elle est choisie aléatoirement dans l'espace des clés,
- Ces algorithmes sont basés sur des opérations de transposition et de substitution des bits du texte clair en fonction de la clé,
- La taille des clés est souvent de l'ordre de 128 bits. Le DES en utilise 56, mais l'AES peut aller jusqu'à 256,
- L'avantage principal de ce mode de chiffrement est sa rapidité,
- Le principal désavantage réside dans la distribution des clés : pour une meilleure sécurité, on préfèrera l'échange manuel. Malheureusement, pour de grands systèmes, le nombre de clés peut devenir conséquent. C'est pourquoi on utilisera souvent des échanges sécurisés pour transmettre les clés. En effet, pour un système à N utilisateurs, il y aura $N.(N - 1)/2$ paires de clés.

3.2. Cryptage asymétrique

3.2.1. Description

Les problèmes de distribution des clés sont résolus par la cryptographie de clé publique. Ce concept a été introduit par Whitfield Diffie et Martin Hellman en 1975.

Le cryptage de clé publique est un procédé asymétrique utilisant une paire de clés pour le cryptage : une clé publique qui crypte des données et une clé privée ou secrète correspondante pour le décryptage. Vous pouvez ainsi publier votre clé publique tout en conservant votre clé privée secrète. Tout utilisateur possédant une copie de votre clé publique peut ensuite crypter des informations que vous êtes le seul à pouvoir lire. Même les personnes que vous ne connaissez pas personnellement peuvent utiliser votre clé publique.

D'un point de vue informatique, il est impossible de deviner la clé privée à partir de la clé publique. Tout utilisateur possédant une clé publique peut crypter des informations, mais est dans l'impossibilité de les décrypter. Seule la personne disposant de la clé privée correspondante peut les décrypter.

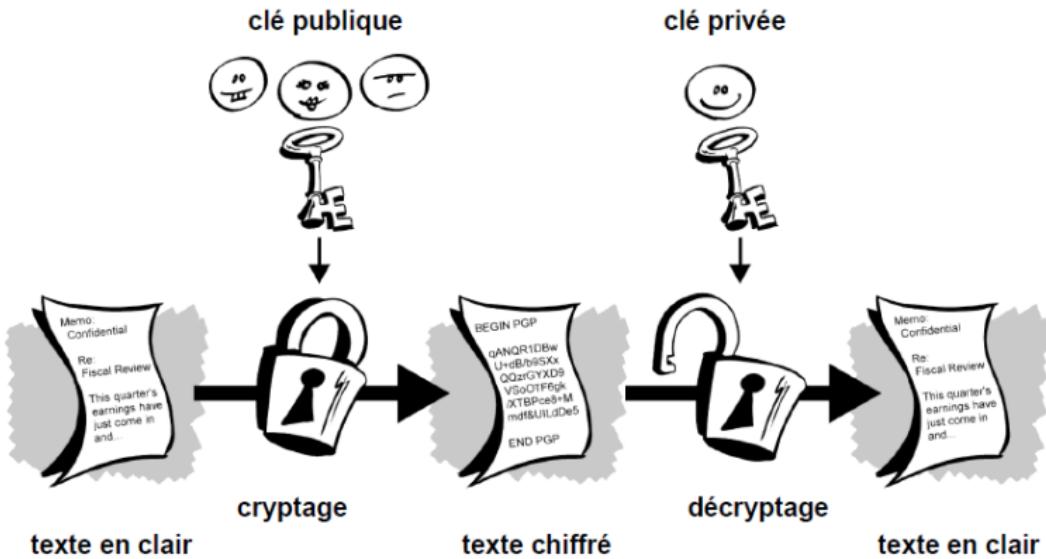


Illustration chiffrement asymétrique

Le cryptage de clé publique présente un avantage majeur : en effet, elle permet d'échanger des messages de manière sécurisée sans aucun dispositif de sécurité. L'expéditeur et le destinataire n'ont plus besoin de partager des clés secrètes via une voie de transmission sécurisée. Les communications impliquent uniquement l'utilisation de clés publiques et plus aucune clé privée n'est transmise ou partagée. Elgamal (d'après le nom de son inventeur, Taher Elgamal), RSA (d'après le nom de ses inventeurs, Ron Rivest, Adi Shamir et Leonard Adleman), Diffie-Hellman (également d'après le nom de ses inventeurs) et DSA, l'algorithme de signature numérique (élaboré par David Kravitz), sont des exemples de systèmes de cryptographie de clé publique.

3.2.2. Caractéristiques

- Une clé publique PK (symbolisée par la clé verticale),
- Une clé privée secrète SK (symbolisée par la clé horizontale),
- Propriété : La connaissance de PK ne permet pas de déduire SK,
- $DSK(EPK(M)) = M$,
- L'algorithme de cryptographie asymétrique le plus connu est le RSA,
- Le principe de ce genre d'algorithme est qu'il s'agit d'une fonction unidirectionnelle à trappe. Une telle fonction à la particularité d'être facile à calculer dans un sens, mais difficile voire impossible dans le sens inverse. La seule manière de pouvoir réaliser le

calcul inverse est de connaître une trappe. Une trappe pourrait par exemple être une faille dans le générateur de clés. Cette faille peut être soit intentionnelle de la part du concepteur (définition stricte d'une trappe) ou accidentelle

- Les algorithmes se basent sur des concepts mathématiques tels que l'exponentiation de grands nombres premiers (RSA), le problème des logarithmes discrets (ElGamal), ou encore le problème du sac à dos (Merkle-Hellman).
- La taille des clés s'étend de 512 bits à 2048 bits en standard. Dans le cas du RSA, une clé de 512 bits n'est plus sûre au sens "militaire" du terme, mais est toujours utilisable de particulier à particulier.
- Au niveau des performances, le chiffrement par voie asymétrique est environ 1000 fois plus lent que le chiffrement symétrique.
- Cependant, à l'inverse du chiffrement symétrique où le nombre de clés est le problème majeur, ici, seules n paires sont nécessaires. En effet, chaque utilisateur possède une paire (SK, PK) et tous les transferts de message ont lieu avec ces clés.
- La distribution des clés est grandement facilitée car l'échange de clés secrètes n'est plus nécessaire. Chaque utilisateur conserve sa clé secrète sans jamais la divulguer. Seule la clé publique devra être distribuée.

4. Quelques Chiffrements Classiques

Pour comprendre le contexte de la cryptographie classique, nous considérons le schéma ci-dessous.

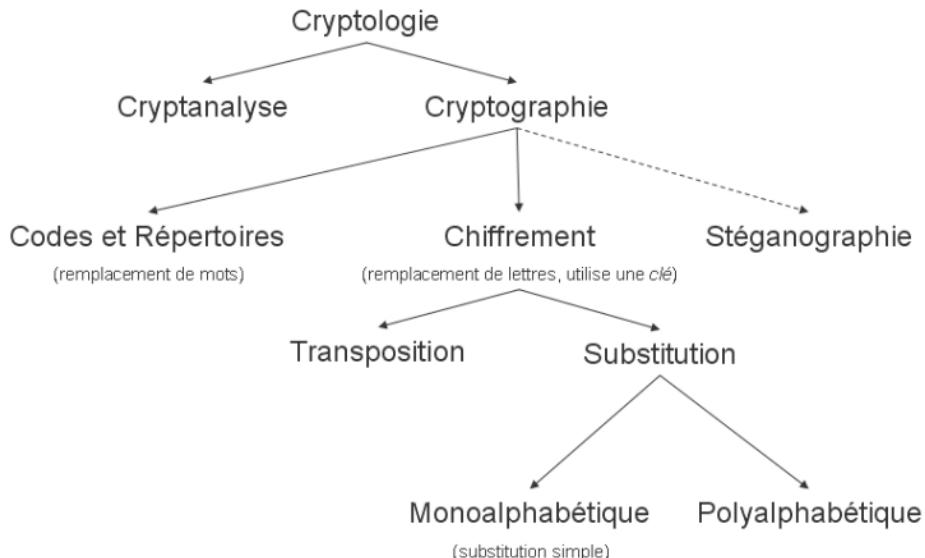


Illustration des branches de la cryptologie

Les premiers algorithmes de chiffrement se fondaient sur la substitution. Il en existe certainement qui se fondent sur la transposition, mais nous prendrons quelques exemples de la substitution mono et polyalphabétique pour illustration.

4.1. Le chiffrement de César

C'est l'un des exemples de chiffrement par substitution monoalphabétique. Tous les chiffrements par substitution peuvent être sensibles à l'analyse de fréquence d'apparition des lettres.

Le chiffrement de César substitue une information par une autre. Cette opération s'effectue généralement en décalant les lettres de l'alphabet. L'algorithme constitue à décaler les lettres de l'alphabet et la clé correspond au nombre de caractères de décalage.

Exemple 1 :

Si vous cryptez le mot « SECRET » à l'aide de la valeur 3 de la clé de César, l'alphabet est décalé de manière à commencer à la lettre D. On peut aussi spécifiquement utiliser la notation « ROT-3 » qui veut dire Rotation-3 pour se référer au code secret 3. Et par défaut, le comptage se fait de gauche à droite. Mais si dans le code, on spécifie « ROT-3 LEFT », le contraire est applicable. « LEFT » qui est un mot anglais signifiant gauche.

Dans cet exemple, en décalant le début de 3 lettres, vous obtenez

DEFGHIJKLMNOPQRSTUVWXYZABC

où D = A, E = B, F = C, etc.

Avec ce procédé, le texte en clair « SECRET » est crypté en « VHFUHW ».

Pour autoriser un autre utilisateur à lire le texte chiffré, indiquez-lui que la valeur de la clé est égale à 3.

Exemple 2 :

Avec le même code secret ou position de décalage 3, on peut chiffrer le mot « CRYPTOGRAPHIE ». Pour cela on écrit les alphabets clair et chiffré comme suit :

ABCDEFGHIJKLMNOPQRSTUVWXYZ

DEFGHIJKLMNOPQRSTUVWXYZABC

Il suffit alors de remplacer les lettres du message clair à l'aide de l'alphabet chiffré : « CRYPTOGRAPHIE » devient alors « FUBSWRJUDSKLH »

4.2. Vigenère

Avec le chiffrement de César, le fait qu'une lettre soit toujours cryptée de la même façon représente une trop grande faiblesse. Le chiffrement de Vigenère remédié à ce problème.

Blaise de Vigenère, né en 1523, fut l'initiateur d'une nouvelle façon de chiffrer les messages qui a pu dominer pendant presque 3 siècles. Le chiffrement de Vigenère est un chiffrement par translation polyalphabetique, c'est-à-dire que chacune des clés m indique une translation affine.

Supposons que nous prenions l'alphabet ordinaire {A, B, ..., Z} en bijection avec $Z/26Z = 0, 1, \dots, 25$. L'addition doit être effectuée dans $Z/26Z$, avec la bijection définie par la table suivante :

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Le chiffrement se fait à l'aide cette table, appelée Tableau de Vigenère :

Considérant chaque lettre et sa correspondante (à la même position), on avance progressivement comme suit :

- Colonne J (du texte clair), ligne M (de la clé) : on obtient la lettre V (l'intersection dans le tableau de Vigenère).
- Colonne A, ligne U : on obtient la lettre U.
- Colonne D, ligne S : on obtient la lettre V.
- Colonne O, ligne I: on obtient la lettre W.
-

Le texte chiffré sera donc : « V'UVWHY IOIMBUL PM LSLYI XAOLM BU NAOJVUY ».

Le principe pour le déchiffrement suit ces étapes :

- Pour chaque lettre de la clé répétée, on regarde la ligne correspondante ;
- Dans la ligne trouvée, on récupère (cherche) la lettre codée ;
- La première lettre de la colonne que l'on trouve ainsi est la lettre décodée.

Ainsi dans notre exemple, on considère et pose le texte chiffré et la clé répétée comme suit :

Texte chiffré : V' UVWHY IOIMBUL PM LSLYI XAOLM BU NAOJVUY

Clé répétée : M USIQU EMUSIQU EM USIQU EMUSI QU EMUSIQU

En décryptant :

- On cherche V sur la ligne M : on trouve la colonne J.
- Sur la ligne U, on cherche U : on trouve la colonne A.
- Sur la ligne S, on cherche V : on trouve la colonne D.
- Ligne I, on cherche W : on trouve la colonne O.

Exemple 2 :

- Texte Clair : fontys...
- Clé répétée : ictict...
- Texte Chiffré : nqgbal...

Le déchiffrement se fait de la même manière que le déchiffrement de Vigenère, en retrouvant les lettres du message clair sur les intersections de la ligne de la clé et la colonne du message chiffré.

Example 2 :

Texte clair : L E T S G O T O T H E C I N E M A .

Clé Allongée : H E L L O L E T S G O T O T H E C .

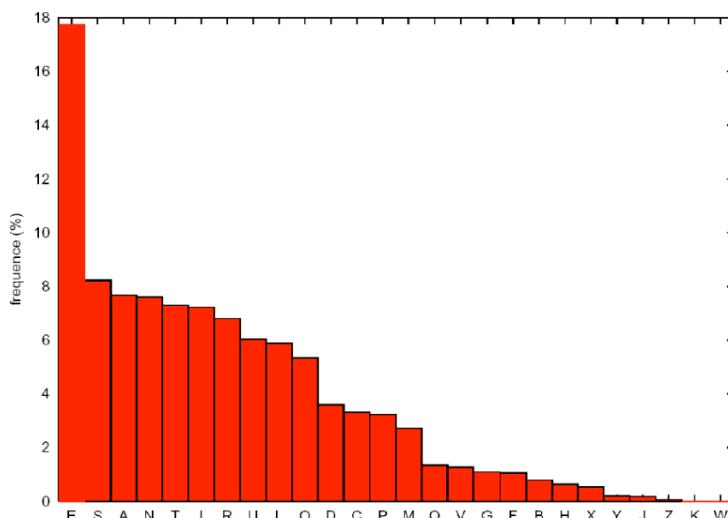
Texte chiffré : S I E D U Z X H L N S V W G L Q C .

4.4. Cryptanalyse

Il existe des méthodes qui peuvent être potentiellement utilisées pour déchiffrer certains messages codés avec ces différents types de chiffrement, même sans la clé initiale. Dans ce cours, nous considérons une seule méthode, l'analyse fréquentielle, pour illustrer ce procédé.

L'analyse fréquentielle, ou analyse de fréquences, est une méthode de cryptanalyse qui consiste à examiner la fréquence des lettres employées dans un message chiffré. Cette méthode est fréquemment utilisée pour décoder des messages chiffrés par substitution (comme par exemple le Chiffre de Vigenère ou le Chiffre de César).

L'analyse fréquentielle est basée sur le fait que, dans chaque langue, certaines lettres ou combinaisons de lettres apparaissent avec une certaine fréquence. Par exemple, en français, le e est la lettre la plus utilisée, suivie du s et du a. Inversement, le w est peu usité.



Fréquence des lettres en Français

Ainsi, à partir de pareils détails on peut faire des hypothèses sur le texte clair, à condition que l'algorithme de chiffrement conserve la répartition des fréquences, ce qui est le cas pour des substitutions mono-alphabétiques et polyalphabetiques.

Néanmoins, il existe une deuxième condition pour appliquer cette technique : c'est la longueur du message à décrypter. En effet, un texte trop court ne reflète pas obligatoirement la répartition générale des fréquences des lettres. De plus, si la clé est de la même longueur que le message, il ne pourra y avoir des répétitions de lettres et l'analyse fréquentielle sera impossible.

Exemple :

Voici un texte chiffre :

« nvxlbg i avxw n ctxnbw ubn dvttbn r bhxqacyb awbgbg i rbn cueciwvn lcni bn vqnbcxz rbn tbwn hxq nxqlbg i qgrvubgin mvtacygvgn rb lvscyb ub gclqwb yuqnncgi nxw ubn yvxoowbn ctbwn c abqgb ubn vgi qun rbavnbn nxw ubn aucgmdbn hxb mbn wvqn rb u ckxw tcucrwvqin bi dvgibxz ucqnnbgi aqibxnbtbg i ubxwn ywcgrbn cqubn eucgmdbn mvttb rbn clqwvgn iwcqgbw c mvib r bxz mb lvscybxw cqub mvttb qu bni ycxmdb bi lbxub uxq gcyxbwb nq ebcx hx qu bni mvtqhxb bi ucqr u xg cycmb nvg ebm clbm xg ewxubyxbxub u cxiwb tqtbg evqicgi u qgoqwtb hxq lvucqi ub avbib bni nbteuceub cx awqgmb rbn gxbbn hxq dcgib uc ibtabib bi nb wqi rb u cwmdbw bzqub nxw ub nvu cx tquqbx rbn dxbbn nbn cqubn rb ybcgi u btabmdbg i rb tcwmdbw»

En analysant la fréquence d'apparition des lettres dans ce texte, on peut avoir cette distribution :

B	N	C	U	X	Q	G	I	W	V
18,7	9,91	7,78	6,90	6,72	6,37	5,84	5,84	5,30	4,60

Or en Français, la distribution des fréquences des lettres se présente comme suit :

E	S	A	N	T	I	R	U	L	O
17,8	8,23	7,68	7,61	7,30	7,23	6,81	6,05	5,89	5,34

On pourrait donc déduire cette corrélation :

$$B \longrightarrow E$$

N ----> S
 C ----> A

En remplaçant chaque lettre correspondante, on obtient le texte chiffre devient :

“svxlegi avxw s atxsew ues dvttes r ehxqaaye aweggegi res aueaiwvs lasies vqseaxz res tews hxq sxqllegi qgrvuegis mvtaaygvgs re lvsaye ue galqwe yuqssagi sxw ues yvxoowes atews a aege ues vgi quis reavses sxw ues auagmdes hxe mes wvqs re u akxw tauarwvqis ei dvgiexz uaqssegi aqiexsetegi uexws ywagres aques euagmdes mvtte res alqwvgs iwaqgew a mvie r exz me lvsayexw aque mvtte qu esi yaxmde ei lexue uxq gayxewe sq eeax hx qu esi mvtqhxe ei uaqr u xg ayame svg eem alem xg ewxueyxexue u axiwe tqte eg evqiagi u qgoqwte hxq lvuaqi ue aveie esi seteuaeue ax awqgme res gxees hxq dagie ua ietaeie ei se wqi re u awmdew ezque sxw ue svu ax tquqex res dxees ses aques re yeagi u etaemdegi re tawmdew.”

Nous considérons aussi la fréquence des bigrammes dans le texte original que nous déchiffrons :

ES	UE	GI	RE	EG	EX	IE	SE	QU	TE	UA	EW	AG ...
25	17	13	12	9	8	8	8	8	8	8	7	7 ...

En Français en revanche, voici les bigrammes les plus fréquents :

ES LE EN DE RE NT ON ER TE SE ET EL QU

Nous pouvons déduire de ces détails, les lettres potentiellement correspondantes suivant leur fréquence dans les bigrammes :

U → L

R → D

G → N

Q → I

I → T

Le texte chiffré devient donc après substitution :

« soulent pour s amuser les dommes d ehuipage prennent des aleatros lastes oiseaux des mers hui suilent indolents mompagnons de losage le nalire glissant sur les

gouoores amers a peine les ont ils déposes sur les planmdes hue mes rois de l akur maladroits et donteux laissent piteusement leurs grandes ailes elanmdes momme des alirons trainer a mote d eux me losageur aile momme il est gaumde et leule lui naguere si eea hu il est momihue et laid l un agame son eem alem un erulegueule l autre mime en eoitant l inoirme hui lolait le poete est semelaele au prinme des nues hui dante la tempete et se rit de l armder exile sur le sol au milieu des duees ses ailes de geant l empemdent de marmder»

5. Exercices No3

1. Comment pouvez-vous différencier entre le chiffrement et la sténographie ? Soutenez votre thèse
2. Identifiez 4 autres chiffrements classiques de substitution et démontrez leur fonctionnement
3. Identifiez 5 chiffrements classiques de transposition et démontrez leur fonctionnement
4. Résolvez ces problèmes en démontrant tout le procédé :
 - a. Décodez le texte suivant obtenu grâce au chiffrement de Vigenère avec comme clé « FONCTION » :

« QSF OTBVRROGKJCSF H SFV MZCC GWRP »

- b. Voici un texte codé par la méthode de chiffrement de Vigenère :
“TMFASAYKBWVJELRIYLNOMGROTRBOYHOZIEBSMAJBINJOK
BJOZPKW
MFYKORYKVFSACUCMEJOTNIVMRYDCAHYAFHBIUXMNTWW
VTCAVRZIEBS
MAZKARYPQAYMMFZACRPOTHOKCEGSAOOVRDZTVWEMERK
URZRWQK”

On ne dispose pas de clé de codage. En combinant deux techniques, il est possible de trouver la longueur de la clé et des clés possibles :

- Repérer des groupes de lettres identiques dans le message, en pratique on cherche des groupes de 3 lettres. On évalue l'écart de position entre le groupe de lettres. Il est probablement un multiple de la longueur de la clé :
- Par le codage de Vigenère, la fréquence des lettres dans la langue française n'est pas modifiée ; comme l'on dispose d'une valeur possible de la longueur de la clé, on découpe le message codé en blocs de cette longueur. On fabrique

ainsi en prenant les lettres en même position dans ces blocs des nouveaux textes dont le décalage avec le texte en clair est constant (chiffrement de César).

Essayez d'appliquer cette méthode au texte ci-dessus.

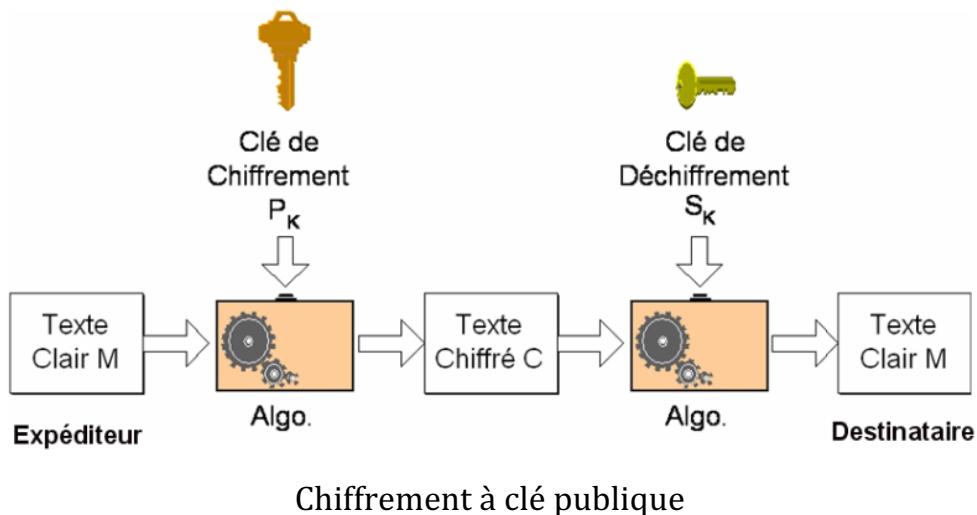
Chapitre 3 : Cryptage Asymétrique (Algorithme RSA)

Nous avons dans ce cours distingué deux catégories principales de chiffrement sur base de l'utilisation des clés : chiffrement (cryptage) symétrique et asymétrique. Dans ce chapitre, nous posons les bases du chiffrement asymétrique alors que le chiffrement symétrique est discuté ultérieurement dans ce document.

1. Concept

Avec le système de cryptage asymétrique, il existe deux (2) clés distinctes (une privée, une publique) utilisées dans la transmission des données entre deux entités qui correspondent. Ces clés s'utilisent selon le principe qu'il est impossible de déduire la clé privée à partir de la clé publique. De ce fait, il est possible de distribuer librement cette dernière.

La distribution de la clé et le transfert de messages cryptés peuvent être illustrés à travers ce scenario :



Formellement, on note que le concept de chiffrement asymétrique date des années 1976 de Diffie et Hellman bien que ses bases soient posées en 1969 par Ellis. La première implémentation a lieu en 1978 par Rivest, Shamir et Adleman (RSA) sous la forme de l'algorithme RSA bien que, là aussi, les fondements de ce système datent de 1973, par Cocks.

Le RSA étant l'algorithme d'implémentation le plus populaire et le plus utilisé, nous nous focalisons dans ce chapitre sur ses fondamentaux pour démontrer empiriquement le fonctionnement des principes du cryptage ou chiffrement asymétrique.

2. Principes basiques du RSA

L'algorithme RSA est basé sur le calcul exponentiel. Sa sécurité repose sur la fonction unidirectionnelle suivante : le calcul du produit de 2 nombres premiers est aisé. La factorisation d'un nombre en ses deux facteurs premiers est beaucoup plus complexe.

Le RSA est donc basé entièrement sur ces principes mathématiques :

- La notion de divisibilité des entiers et congruence
- L'arithmétique modulaire
- La notion d'inverse
- La notion de pgcd
- La notion des nombres premiers
- La fonction d'Euler et les théorèmes d'Euler et de Fermat
- L'exponentiation Modulaire

2.1. La notion de divisibilité et congruence

Rappel : L'ensemble \mathbb{Z}

C'est l'ensemble des nombres entiers relatifs. Un entier relatif est, non seulement, un entier naturel, mais se présente aussi comme un entier naturel muni d'un signe positif ou négatif.

- $\mathbb{Z} = \{..., -3, -2, -1, 0, 1, 2, 3, ...\}$: désigne l'ensemble des entiers (relatifs)
- \mathbb{Z}^+ désigne l'ensemble des entiers positifs : $\mathbb{Z}^+ = \{1, 2, 3, ...\}$
- $\mathbb{Z}^+ = \{x \mid x \in \mathbb{Z}, x > 0\}$: l'ensemble des entiers relatifs x tels que $x > 0$
- $\mathbb{Z}_p = \{0, 1, 2, 3, ..., p-1\}$: l'ensemble des entiers positifs de taille p , allant de 0 à $p-1$.

Divisibilité :

Soient a et b deux entiers. S'il existe $q \in \mathbb{Z}$ tel que $a = qb$ on dit que :

- a est un multiple de b ;
- b est un diviseur de a ;
- b divise a , ce qu'on note « $b \mid a$ ».

Exemple : $7 \mid 21$ (car $7 \cdot 3 = 21$)

Remarque. Si b divise a , soit $a = qb$, on a aussi $a = (-q)(-b)$ donc $-b$ divise a . Les problèmes de divisibilité restent donc inchangés par multiplication par ± 1 .

Chaque entier a est divisible par lui-même (prendre $q = 1$), est divisible par 1 (prendre $q = a$) et divise zéro (prendre $q = 0$). En outre :

Quels que soient $a, b, c, d, e \in \mathbb{Z}$ on a :

- Si $a \mid b$ et $b \mid c$, alors $a \mid c$.
- Si $a \mid b$ et $c \mid d$, alors $ac \mid bd$.
- Si $a \mid b$ et $a \mid c$, alors $a \mid bd + ce$.
- Si $a \mid b$ et $b > 0$, alors $a \leq b$.

Exemple. L'entier 6 divise 42 mais pas 45. En effet, on a $42 = 6 \cdot 7$. En revanche, si $q \leq 7$, on a $6q \leq 42$ alors que, si $q \geq 8$, on a $6q \geq 48$; aucun $q \in \mathbb{Z}$ ne satisfait donc $6q = 45$.

Congruence

$\forall a, b \in \mathbb{Z}, m \in \mathbb{Z}^+$

On dit que a est congru à b modulo m , lorsque $a - b$ est divisible par m ; on note cela $a \equiv b \pmod{m}$

$$\begin{aligned} a &\equiv b \pmod{m} \\ &\Leftrightarrow \\ m &\mid a - b \end{aligned}$$

$$\begin{aligned} &"m \text{ est diviseur de } a - b" \\ &\Leftrightarrow \\ a - b &= x \cdot m \quad (\text{avec } x \in \mathbb{Z}) \\ &\Leftrightarrow \\ a &= b + x \cdot m \quad (\text{avec } x \in \mathbb{Z}) \end{aligned}$$

Remarque : Le reste de la division euclidienne de a par b est le plus petit entier positif congru à a modulo b . Deux entiers sont congrus modulo b si et seulement si leurs divisions euclidiennes par b donnent le même reste.

Exemples :

- a) 42 et 127 sont congrus modulo 17 car $42 - 127 = -85 = -5 \cdot 17$.
On a par ailleurs $42 = 2 \cdot 17 + 8$ et $127 = 7 \cdot 17 + 8$.
- b) $57 \equiv 15 \pmod{7}$ car : $57 = 7 \cdot 8 + 1$ et $15 = 7 \cdot 2 + 1$
- c) Un nombre est toujours congru à son reste modulo n par la division euclidienne par n :
 $2008 \equiv 8 \pmod{10}$ car: $2008 = 10 \cdot 200 + 8$ et $8 = 10 \cdot 0 + 8$

Propriétés de congruence

Soit n un entier non-nul fixé. La relation de congruence modulo n possède de nombreux points communs avec celle d'égalité :

— Transitivité :

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}, \forall c \in \mathbb{Z}, a \equiv b \wedge b \equiv c \Rightarrow a \equiv c$$

— Symétrie :

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}, a \equiv b \Rightarrow b \equiv a$$

— Réflexivité :

$$\forall a \in \mathbb{Z}, a \equiv a$$

On dit ainsi que la congruence est relation d'équivalence. Elle est de surcroit compatible avec les opérations d'addition et de multiplication :

— Addition

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}, \forall c \in \mathbb{Z}, \forall d \in \mathbb{Z}, a \equiv b \wedge c \equiv d \Rightarrow a + c \equiv b + d$$

— Multiplication

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}, \forall c \in \mathbb{Z}, \forall d \in \mathbb{Z}, a \equiv b \wedge c \equiv d \Rightarrow a \cdot c \equiv b \cdot d$$

2.2. Arithmétique modulaire

Dans le système modulo n ou système restreint aux nombres entiers inférieurs à n, on utilise les nombres $0, 1, 2, 3, 4, \dots, (n - 1)$.

Considérons n comme ayant la valeur 12, on peut donc écrire :

$$\mathbb{Z}_{12} = \{0, 1, 2, 3, \dots, 11\}$$

Principe :

Les opérations arithmétiques définies dans ce système sont les mêmes que celles de l'arithmétique élémentaire, mais les nombres utilisés ne peuvent être supérieurs à $(n - 1)$. Lorsqu'un résultat devrait être supérieur à $(n - 1)$, on divise ce résultat par n et on utilise le reste de cette division comme résultat de l'opération. Autrement, si nous tombons sur un nombre en dehors de cet intervalle, nous lui ajoutons n (ou lui soustrayons n), jusqu'à ce qu'il soit compris dans cet intervalle (de 0 à $n - 1$).

Exemples :

- $15 \equiv 27 \pmod{12}$
- $15 \pmod{12} = 3,$
 $27 \pmod{12} = 3,$
 $39 \pmod{12} = 3$

On définit les opérations algébriques d'addition, soustraction, multiplication par :

$$(a \pmod{n}) + (b \pmod{n}) = (a + b \pmod{n})$$

$$(a \pmod{n}) - (b \pmod{n}) = (a - b \pmod{n})$$

$$(a \pmod{n}) \times (b \pmod{n}) = (a \times b \pmod{n})$$

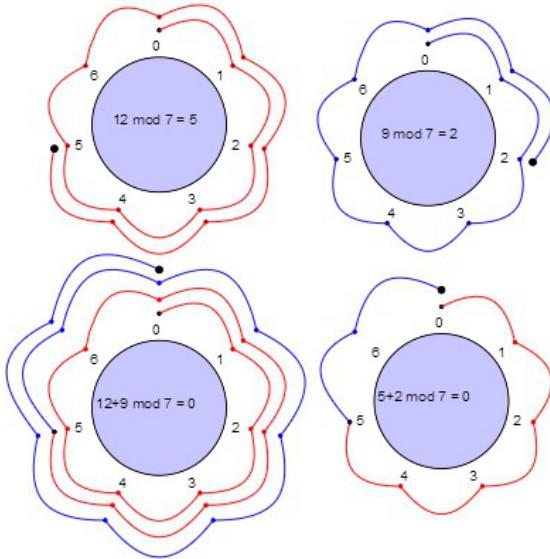
Exemples :

$$3 + 4 \pmod{24} = 7 \pmod{24}, 3 \times 4 \pmod{24} = 12 \pmod{24}$$

Nous pouvons réécrire cet exemple comme suit :

$$3 + 4 \equiv 7 \pmod{24}, 3 \times 4 \equiv 12 \pmod{24}$$

En mod 7, on peut illustrer quelques opérations par un comptage dans le sens de l'aiguille d'une horloge pour passer en revue tous les éléments de la classe d'équivalence (intervalle).



$$3 + 2 \equiv 5 \pmod{7}$$

$5 + 6 \equiv 11 \pmod{7}$, mais 11 n'est pas dans l'intervalle, on va donc enlever 7 de cette somme pour trouver un nombre inférieur à 7.

$$5 + 6 \equiv 11 (11-7) \equiv 5 \pmod{7}$$

Remarque : La représentation de -3 en mod 7 est 4, puisque $-3 + 7 = 4$, qui est un nombre compris dans notre intervalle.

$$5 - 6 \equiv -1 (-1 + 6) \equiv 6 \pmod{7}$$

$$3 \times 5 \equiv 15 (15-7-7) \equiv 1 \pmod{7}$$

2.3. Notion d'inverse

Soit $a, b \in \mathbb{Z}$:

- 0 est l'élément neutre pour l'addition, si $a + 0 = a$
- 1 l'élément neutre pour la multiplication, si $a \cdot 1 = a$
- b est l'opposé de a ou l'inverse de a pour l'addition, si $a + b = 0$
- c est l'inverse multiplicatif de a , si $a \cdot c = 1$

Exemples

Partant de ces principes, nous pouvons déterminer l'inverse de ces nombres en mod 7 :

- $5 + 2 = 0$, donc 2 est l'opposé ou l'inverse pour l'addition de 5 en mod 7
- $3 \cdot 5 = 1$, donc 5 est l'inverse multiplicatif de 3 en mod 7

Exercice : Quelle est la valeur de x dans cette équation ?

$$25 \cdot x = 1 \pmod{42}$$

2.4. Plus grand commun diviseur (PGCD)

Soient a, b et d trois entiers.

Le plus grand commun diviseur (PGCD) d, est le plus grand entier positif divisant a et b.

$$\text{pgcd}(a, b) = d$$

\Leftrightarrow

$$d = (\text{MAX } c : c \in \mathbb{Z} \wedge c \mid a \wedge c \mid b : c)$$

On peut naïvement calculer un plus grand commun diviseur, par exemple $\text{pgcd}(42, 60)$, en énumérant les diviseurs des deux nombres :

on a $\text{div}(42) = \{1, 2, 3, 6, 7, 14, 21, 42\}$ et

$\text{div}(60) = \{1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60\}$,

d'où $\text{div}(42) \cap \text{div}(60) = \{1, 2, 3, 6\}$

ce qui donne $\text{pgcd}(42, 60) = 6$ parce que 6 est le plus grand de ces diviseurs communs.

Si vous prenez des nombres plus grands (tels que 7895, 23763...), cette méthode s'avèrera pratiquement inefficace. Ainsi, l'algorithme Euclidien ou l'algorithme d'Euclide.



*Euclide
d'Alexandrie
 $\pm 265 - 200 \text{ BC}$*
Εὐκλείδης

Algorithme (Euclide).

ENTRÉE : Deux entiers positifs a et b .

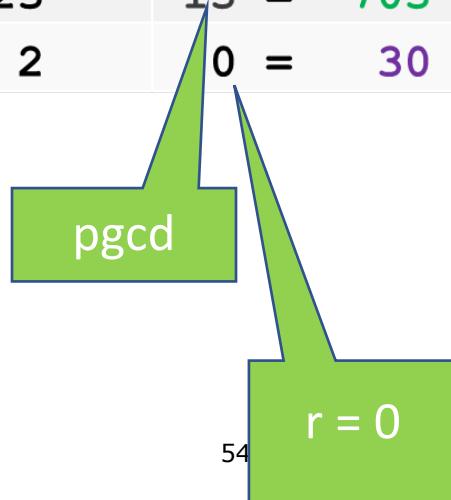
SORTIE : Leur pgcd.

1. *Calculer le reste r de la division euclidienne de a par b .*
2. *Si $r = 0$, renvoyer b .*
3. *Sinon, assigner $a \leftarrow b$ et $b \leftarrow r$ puis retourner en 1.*

Pour expliquer les étapes de cet algorithme, nous posons les termes de la division euclidienne et suivons ces étapes dans cet exemple.

Exemple : Calculer pgcd (735, 2175)

b	a	$q = b \text{ div } a$	$r = b \bmod a = b - q*a$
2175	735	2	705 = 2175 - 2*735
735	705	1	30 = 735 - 1*705
705	30	23	15 = 705 - 23*30
30	15	2	0 = 30 - 2*15



Dans ce tableau, nous considérons le plus grand nombre (2175) comme dividende et le plus petit comme diviseur. Nous calculons le quotient (q) et le reste (r) progressivement jusqu'à trouver un reste égale à 0. Si le reste est différent de 0, il devient diviseur et le diviseur de l'opération précédente devient le dividende de notre opération.

Quelques propriétés du pgcd

Soient $a, b \in \mathbb{Z}$, on peut remarquer ce qui suit :

$$\text{pgcd}(a, b)$$

=

$$\text{pgcd}(b, a)$$

=

$$\text{pgcd}(a-b, b)$$

=

$$\text{pgcd}(a, b-a)$$

2.5. Primalité (Primarité)

On dit qu'un entier $a > 1$ est premier s'il n'admet comme diviseurs positifs que 1 et a .

$$a \in \mathbb{Z}$$

a est premier

\Leftrightarrow

$a > 1 \wedge$ seuls diviseurs sont 1, a

Exemples : 2, 3, 5, 7, 11, 13, 17, 19, 23

Note : Tout entier $n > 0$ peut se décomposer comme produit de nombres premiers. Il faut aussi noter que cette décomposition est unique à l'ordre des facteurs près. C'est pour cela qu'on adopte la convention que le nombre 1 n'est pas premier.

Exemples :

- $42 = 2 \cdot 3 \cdot 7$

$$- \quad 60 = 2 \cdot 2 \cdot 3 \cdot 5$$

On peut formaliser l'énumération des nombres premiers en suivant la méthode suivante :

Algorithme (crible d'Ératosthène).

ENTRÉE : Un entier n .

SORTIE : Les nombres premiers inférieurs à n .

1. *Construire l'ensemble $E = \{2, \dots, n\}$.*
2. *Tant que E est non vide :*
3. *Soit p le plus élément de E .*
4. *Enlever de E tous les multiples de p .*
5. *Afficher p .*

Rappel important : Décomposition en produit de facteurs premiers

Dans le contexte du chiffrement RSA, il est aussi important de recourir aux notions de factorisation ou décomposition en facteurs premiers. Nous en verrons la portée dans les sections un peu plus tard dans ce document.

En effet, la décomposition en produit de facteurs premiers, aussi connue comme la factorisation entière en nombres premiers, consiste à chercher à écrire un entier naturel non nul sous forme d'un produit de nombres premiers.

Par exemple, si le nombre donné est 45, la factorisation en nombres premiers est $3^2 \times 5$, soit $3 \times 3 \times 5$.

L'écriture des nombres entiers en produits de facteurs premiers en facilite la manipulation dans des problèmes de divisibilité, de fraction ou de racine carrée.

La méthode la plus naïve qui nous permet de décomposer un nombre en facteurs premiers est celle que tout le monde apprend à l'école primaire :

- On établit si le nombre est divisible par 2 et en cas affirmatif, on calcule le quotient
- On continue en divisant par deux jusqu'à ce que l'on trouve un quotient qui n'est plus divisible par deux
- Si le premier nombre ou le dernier quotient n'est pas divisible par deux, on continue de la même façon avec les nombres premiers suivants (3, 5, 7...) jusqu'à ce que l'on obtienne un quotient qui est un nombre premier

Exemples :

$$\begin{array}{rcl} 4\,220 & | & 2 \\ 2\,110 & | & 2 \\ 1\,055 & | & 5 \\ 211 & | & 211 \\ 1 & | & \end{array} \qquad \begin{array}{rcl} 728 & | & 2 \\ 364 & | & 2 \\ 182 & | & 2 \\ 91 & | & 7 \\ 13 & | & 13 \\ 1 & | & \end{array}$$

$$4\,220 = 2 \times 2 \times 5 \times 211 \quad 728 = 2 \times 2 \times 2 \times 7 \times 13$$

Il existe plusieurs théories proposant cette décomposition, mais l'explication de Jean-Paul Delahaye paraît un peu plus compréhensible pour les lecteurs de ce document.

Selon Jean-Paul Delahaye, on peut suivre ces étapes :

Initialisation : 2 s'écrit comme produit de nombres premiers, car $2 = 2$ (par convention, un nombre seul est considéré comme un produit d'un facteur). Soit n un entier supérieur à 2. Supposons que tous les entiers entre 2 et $n - 1$ s'écrivent comme produit de nombres premiers (hypothèse de récurrence), et montrons que cela est aussi vrai pour n .

Nous savons (d'après la proposition selon laquelle tout nombre supérieur à 1 est divisible par un nombre premier) que n est divisible par un nombre premier p . Donc $n = q \times p$ avec $1 < p \leq n$.

- Si $p = n$ (et $q = 1$), c'est terminé, car le nombre premier p est un produit de nombres premiers.
- Si p est inférieur à n , alors q est compris entre 2 et $n - 1$ et, d'après l'hypothèse de récurrence, q est un produit de nombres premiers. Par conséquent, n l'est aussi, puisqu'il est le produit de q par le nombre premier p .

Ce théorème de décomposition en facteurs premiers se traduit immédiatement en algorithme pour la décomposition des nombres et la recherche de leurs diviseurs :

- tenter toutes les divisions de a par les nombres premiers entre 2 et \sqrt{a} ;
- dès qu'un facteur premier p est trouvé, mémoriser p , diviser a par p , ce qui donne un nouveau nombre a , puis reprendre l'algorithme avec ce nouveau nombre ;

- si aucun diviseur n'est trouvé, c'est que a est premier ; l'ajouter à la liste ;
- la liste des nombres premiers ainsi constituée est la décomposition du nombre a initial en facteurs premiers ; en combinant ces facteurs de toutes les façons possibles, on obtient les diviseurs de a .

Exemple, décomposons $a = 220$ à l'aide de l'algorithme :

- dès la première division, on trouve $p = 2$; a devient $220/2 = 110$;
- dès la deuxième division, on trouve $p = 2$; a devient $110/2 = 55$;
- à la troisième division, on trouve $p = 5$; a devient $55/5 = 11$;
- comme 11 est premier, on s'arrête ;
- la liste des facteurs premiers de 220 , obtenue au bout de seulement cinq divisions, est $2, 2, 5, 11$;
- la liste des diviseurs de 220 est donc : $1 ; 2 ; 5 ; 11 ; 2 \times 2 ; 2 \times 5 ; 2 \times 11 ; 5 \times 11 ; 2 \times 2 \times 5 ; 2 \times 2 \times 11 ; 2 \times 5 \times 11 ; 2 \times 2 \times 5 \times 11$.

2.6. Co-primalité (Co-primarité)

Soient a et $b \in \mathbb{Z}$

a et b sont copremiers ou premiers entre eux

\Leftrightarrow

$$\text{pgcd}(a,b)=1$$

Exemples:

- 14 et 15 sont premiers entre eux :

$$14=2 \cdot 7$$

$$15=3 \cdot 5$$

$$\text{pgcd}(a,b) = 1$$

- 11 and 13 sont aussi premiers entre eux car le pgcd (11,13) = 1.
- 15 and 12 ne sont pas premiers entre eux, le pgcd (15,12) = 3

2.7. Fonction totient d'Euler

Soit $n \in \mathbb{Z}^+$, la fonction d'Euler définie par :

$$\varphi(n) =$$

$$(\underline{\text{Ni}} : i \in \{1, \dots, n\} : \text{gcd}(n,i) = 1) =$$

$$(\exists i : i \in \{1, \dots, n\} : n \text{ and } i \text{ coprime})$$

$\varphi(n)$ est l'indicateur d'Euler, c'est-à-dire le cardinal des entiers inversibles de n .

En d'autres termes, on pourra lire la fonction φ du nombre n a pour résultat le nombre des i inférieur ou égal à n et tel que le plus grand commun diviseur de n et i soit 1. Dans ce cas, n et i sont co-premiers ou premiers entre eux.

Exemples :

- $\varphi(10) = 4$ car de cet ensemble $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$, seuls les quatre nombres 1, 3, 7 et 9 sont premiers avec 10;
- $\varphi(8) = 4$ car parmi les nombres de 1 à 8, seuls les quatre nombres 1, 3, 5 et 7 sont premiers avec 8 ;
- $\varphi(12) = 4$ car parmi les nombres de 1 à 12, seuls les quatre nombres 1, 5, 7 et 11 sont premiers avec 12 ;
- $\varphi(1) = 1$ car 1 est premier avec lui-même
- $\varphi(11) = 10$

Propriétés de φ

Si p est premier:

$$- \quad \varphi(p) = p-1$$

De plus, soient p et q deux nombres premiers et $n = pq$. Il vient

$$\varphi(n) = \varphi(pq)$$

$$= \varphi(p) * \varphi(q)$$

$$= (p - 1) (q - 1)$$

Si a et b sont premiers entre eux :

$$- \quad \varphi(a \cdot b) = \varphi(a) \cdot \varphi(b)$$

Si p est premier, et $k \in \mathbb{Z}^+$:

$$- \quad \varphi(p^k) = (p-1) \cdot p^{k-1}$$

Il est facile de calculer la valeur d'indicateur d'Euler quand n est premier. Au cas contraire, on peut appliquer cette formule générale :

- réaliser la décomposition en facteurs premiers de n
- Soient p_i les m facteurs premiers distincts diviseurs de n. La formule devient :

$$\varphi(n) = n \prod_{i=1}^m \left(1 - \frac{1}{p_i}\right)$$

Exemple :

- Pour $n = 6$, seuls les nombres 1 et 5 sont premiers avec 6 donc $\varphi(6) = 2$. Ce que confirme la formule pour $n = 6 = 2^1 \times 3^1$:

$$\varphi(6) = 6(1 - 1/2)(1 - 1/3) = 2$$

2.8. Théorème d'Euler

Soient $n \in \mathbb{Z}_+$, $k \in \mathbb{Z}$, $a \in \mathbb{Z}_n$, a et n premiers entre eux, alors :

$$a^{\varphi(n)} \equiv 1 \pmod{n}$$

et particulièrement:

$$a^{(1+k \cdot \varphi(n))} \equiv a \pmod{n}$$

2.9. Cycles de taille $\varphi(n)$

Soit $n \in \mathbb{Z}^+$, les exposants se répètent après toutes les $\varphi(n)$ étapes ou pas.

Exemple :

- Supposons que $a=3$ et $n=5$, alors:

$a^0 \equiv 1$		Toutes les fois, après 4 étapes (pas), nous avons un cycle qui se répète parce que $\varphi(5)=4$
$a^1 \equiv 3$		
$a^2 \equiv 4$		
$a^3 \equiv 2$		
$a^4 \equiv 1$		
$a^5 \equiv 3$		
$a^6 \equiv 4$		
$a^7 \equiv 2$		
$a^8 \equiv 1$		
....		

2.10. L'inverse multiplicatif modulaire

2.10.1. Théorème

Soit $a \in \mathbb{Z}$, $m \in \mathbb{Z}^+$, alors

$$\text{Si } \text{pgcd}(a,m) = 1$$

\Leftrightarrow

l'inverse multiplicatif modulaire
de $a \bmod m$ existe

\Leftrightarrow

Il existe un $x \in \mathbb{Z}$, tel que
 $a \cdot x \equiv 1 \bmod m$

2.10.2. Lemme de Bézout

Soit $a, m \in \mathbb{Z}$,

Il existe $u \in \mathbb{Z}$ et $v \in \mathbb{Z}$, tel que

$$\text{pgcd}(a,m) = u \cdot a + v \cdot m$$

2.10.3. Lemme de Bézout et l'inverse multiplicatif

Supposons $\text{pgcd}(a,m)=1$, en utilisant le lemme de Bézout, nous pouvons calculer l'inverse multiplicatif de a , modulo m :

- $\text{pgcd}(a,m) = 1$, alors u et v existent tel que
 $1 = u \cdot a + v \cdot m$
- Parce que $v \cdot m$ est un multiple de m , il en découle que :
 $1 \equiv u \cdot a \pmod{m}$
- Donc, u est l'inverse multiplicatif de a , modulo m

Comment pouvons-nous retrouver les valeurs de u et v connaissant a et m pour appliquer le lemme de Bézout ?

La réponse à cette question repose dans l'adoption de l'extension de l'algorithme d'Euclide.

2.10.4. Extension de l'Algorithme d'Euclide

En effet, il existe plusieurs variantes de cet algorithme. Dans ce document, nous optons pour la variante de W.A Blankinship. Cette variante propose ces étapes :

- Ajouter une colonne à droite
- Ajouter deux lignes au-dessus (début)
- Chaque ligne est une combinaison des équations dans les deux lignes précédentes

Exemple : Calculer l'inverse multiplicatif de 295 modulo 3876

a	B	$r = b \bmod a$	$r = u^*a + v^*b$
			$3876 = 0*295 + 1*3876$
			$295 = 1*295 + 0*3876$
3876	295	$41 = 3876 - 13*295$	$41 = -13*295 + 1*3876$
295	41	$8 = 295 - 7*41$	$8 = 92*295 - 7*3876$
41	8	$1 = 41 - 5*8$	$1 = -473*295 + 36*3876$
8	1	$0 = 8 - 8*1$	$1 - 7*(-13) = 92$ $-13 - 5*92 = -473$ $0 - 7*1 = -7$ $1 - 5*(-7) = 36$
Chaque ligne est une combinaison de 2 lignes précédentes			

- Nous obtenons :

$$1 = -473*295 + 36*3876$$

- Etant donné que nous travaillons en mod 3876, nous pouvons écrire :

$$-473*295 + 36*3876 \equiv 1 \pmod{3876}$$

- $36*3876$ étant un multiple de 3876, nous pouvons écrire :

$$-473*295 \equiv 1 \pmod{3876}$$

- En d'autres termes :

-473 est l'inverse de 295 , mod 3876

- Appliquant la règle de modulo pour les nombres négatifs, nous devons trouver une valeur entre 0 et 3876 :

$$-473 \bmod 3876 = 3876 - 473 = 3403$$

Remarque:

- On utilise l'algorithme d'Euclide pour calculer le pgcd de 2 nombres
- On utilise l'algorithme d'Euclide étendu (la variante de Blankinship dans ce cas) pour trouver l'inverse multiplicatif modulaire d'un nombre

2.11. L'Exponentiation Modulaire

Quelques règles d'exponentiation :

$$- a^{n+m} = a^n \cdot a^m$$

$$- a^{n \cdot m} = (a^n)^m$$

La règle suivante s'applique aux exposants en modulo :

$$- a^k \equiv (a \pmod{n})^k \pmod{n}$$

Il existe deux possibilités qui nous permettent de calculer une expression exponentielle en modulo :

2.11.1. Exponentiation rapide (exponentiation par carré)

L'exponentiation rapide repose sur trois étapes. Considérons que nous voulons calculer a^{1063} modulo 2159

Etape 1 : convertir l'exposant de a (1063) en binaire, c'est-à-dire qu'on écrit :

$$1063 = 10000100111$$

d'où l'on déduit la décomposition de 1063 en somme de puissances de 2 en ne considérant que les positions avec 1:

$$1063 = 2^{10} + 2^5 + 2^2 + 2^1 + 2^0 = 1024 + 32 + 4 + 2 + 1.$$

Etape 2 : Appliquant les propriétés des puissances, on peut écrire :

$$a^{1063} = a^{1+2+4+32+1024} = a \times a^2 \times a^4 \times a^{32} \times a^{1024}$$

Etape 3 : on procède ensuite à des élévarions au carré successives à partir de a :

$$[a, a^2, a^4, a^8, a^{16}, a^{32}, a^{64}, a^{128}, a^{256}, a^{512}, a^{1024}]$$

Les valeurs entourées nous permettent de calculer a^{1063} comme on peut le rappeler.

Exemple : calculer $14^{114} \bmod 145$

$$14^{114} = 14^{(64+32+16+2)} = 14^{64} \cdot 14^{32} \cdot 14^{16} \cdot 14^2$$

$$\begin{array}{rclcl} 14^1 & & & = & 14 \\ 14^2 & = & 14^2 & = & 196 \\ 14^4 & = & 51^2 & = & 2601 \\ 14^8 & = & 136^2 & = & 18496 \\ 14^{16} & = & 81^2 & = & 6561 \\ 14^{32} & = & 36^2 & = & 1296 \\ 14^{64} & = & 136^2 & = & 18496 \end{array} \quad \begin{array}{rclcl} & & & = & 14 \\ & & & = & 51 \\ & & & = & 136 \\ & & & = & 81 \\ & & & = & 36 \\ & & & = & 136 \\ & & & = & 81 \end{array}$$

$$\text{Donc : } 14^{114} \bmod 145 = 81 \cdot 136 \cdot 36 \cdot 51 \bmod 145 = 51$$

2.11.2. Exponentiation par théorème d'Euler

Utilisant le même exemple : calculer $14^{114} \bmod 145$

- On note que $145 = 5 \cdot 29$, ainsi $\varphi(145) = 4 \cdot 28 = 112$
- On remarque que 14 et 145 sont premiers entre eux
- Nous déduisons (en appliquant le théorème d'Euler): $14^{112} \equiv 1 \pmod{145}$

$$14^{114} = 14^{112+2}$$

$$= 14^{112} \cdot 14^2$$

$$\equiv 1 \cdot 14^2 \pmod{145}$$

$$\equiv 196 \pmod{145}$$

$$\equiv 51 \pmod{145}$$

La stratégie globale pour effectuer l'exponentiation modulaire, repose sur ces 2 considérations :

- En calculant par exemple $a^b \text{ mod } n$ on applique le théorème d'Euler, on vérifie ces 2 conditions :
 - a et n sont-ils premiers entre eux?
 - $\varphi(n) < b$?
- Si ces conditions ne sont pas vérifiables, on utilise la méthode d'exponentiation rapide.

3. L'algorithme RSA

3.1. Etapes

Les étapes principales de l'algorithme RSA se présentent comme suit :

- a. Choisir deux nombres premiers différents p, q
- b. Calculer $n = p \cdot q$
- c. Calculer $\varphi(n)$
- d. Choisir e , sachant $\text{pgcd}(e, \varphi(n))=1$
(conseil: choisir e premier et $e>p, e>q$)
- e. Calculer d , sachant que $e \cdot d \equiv 1 \pmod{\varphi(n)}$
- f. Publier e et n ; garder d prive
détruire (jeter) $p, q, \varphi(n)$

En d'autres termes, pour crypter un message avec RSA on commence par le transformer en un ou plusieurs nombres. Les processus de chiffrement et déchiffrement font appel à plusieurs notions que nous avons apprises dans ce chapitre :

- On choisit deux nombres premiers p et q que l'on garde secrets et on pose $n = p \times q$. Le principe étant que même connaissant n il est très difficile de retrouver p et q (qui sont des nombres ayant des centaines de chiffres).
- La clé secrète et la clé publique se calculent à l'aide de l'algorithme d'Euclide et des coefficients de Bézout.
- Les calculs de cryptage se feront modulo n .

3.2. Implémentation

En informatique, les messages sont des nombres (il est toujours possible de convertir un texte plain en nombres en utilisant le code ASCII).

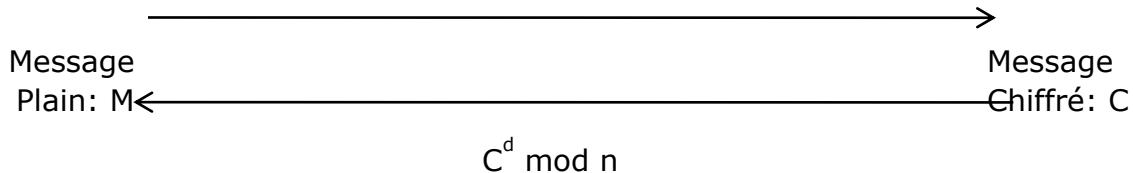
- Une fois converti, le message M est prêt pour le chiffrement :

$$\text{cryptage}(M) = M^e \bmod n$$

- Le déchiffrement du message chiffré C suit cette formule :

$$\text{decryptage}(C) = C^d \bmod n$$

$$C = M^e \bmod (n=pq)$$



Il faut noter qu'en implémentant le RSA :

- Il est difficile de déduire p, q à partir de n ,
- Il est difficile de déduire d à partir de (n, e) ,
- Il sera difficile de déduire M à partir de (n, e) et C tel que $C = M^e$.

Exemple :

Supposons, pour déterminer les clés nous procédonss de cette manière :

- $p = 11, q = 7$,
Après calcul:
- $n = 77$,
- $\Phi(n) = 60$
- $d = 13, e = 37$ ($e.d = 481; e.d \bmod 60 = 1$)

Considérons que notre $M = 15$. En appliquant ces clés nous aurons :

- Pour chiffrer M en C , nous appliquons la formule : $C \equiv M^e \bmod n$
$$C \equiv 15^{37} \pmod{77} = 71$$
- Pour déchiffrer et retrouver le message plein M , nous nous servons de la formule : $M \equiv C^d \bmod n$

$$M \equiv 71^{13} \pmod{77} = 15$$

En réalité, les messages sont souvent longs et leur code ASCII correspondant aura naturellement une taille proportionnée.

Exemple :

```
p =
1093376618363257581761151703473066828715579998463222345413874567112
1273456287670082908433028755212749702453145932229461290645383585810
18615539828479146517
```

```
q =
1091061696734911023172373407861492264533706088214174896820983422513
8976011179993942998101597369044685540217082898243965534121805148279
96444845438176099863
```

```
n = p x q =
1192941348401695090555272113312556496446065696615276380120674819549
4305685115033806315957037715620297305000118628770846689969112892212
2454571180605749959895170004210526342737632227426639311619351783957
0773505632231596681121927337473973220125125990612313222509455062600
```

3.3. Aspect Sécuritaire

L'aspect sécuritaire de l'algorithme RSA résulte de ces considérations :

- La sécurité dépend de la difficulté à factoriser n

Factoriser n => $\Phi(n)$ => calculer d à partir de (e, $\Phi(n)$)

- La taille de n=p*q reflète la force

n de taille 700-bit déjà factorisé en 2007

n de 768 bit déjà factorisé en 2009

- Aujourd'hui, la taille 1024 bit est requise pour un niveau minimum de sécurité
- AU contraire, on recommande une taille minimale de 2048 bits pour l'usage moderne
- La vitesse (performance) du RSA est quadratique à la taille de la clé

3.4. Conclusion

3.4.1. Utilité pratique

On peut adopter et utiliser l'algorithme RSA dans ces deux cas les plus populaires :

- Lorsqu'on veut s'assurer de la confidentialité des données(message)
 - Chiffrer avec (clé publique) e, déchiffrer avec la (clé privée) d
 - Tout le monde peut chiffrer par conséquent, mais seulement celui qui est en possession de d pourra décrypter
- Lorsqu'on veut s'assurer d'une bonne authentification de la source des données (message)
 - L'auteur peut chiffrer avec (la clé privée) d, tout le monde peut décrypter avec la clé publique e

3.4.2. Restrictions (limites)

L'algorithme RSA, bien que populaire et même très efficace, présente quelques restrictions :

- RSA n'est pas approprié pour l'usage avec des messages plus larges (de grande taille) :
 - On déchiffre avec la formule :
 $C^d \text{ mod } n$
sous-entendu que le message devra être plus petit que n
 - RSA nécessite beaucoup de temps du CPU, même quand on veut découper le message en plusieurs blocks, cela est toujours inefficace.
- Pour palier à cette restriction, on peut adopter l'algorithme AES (cryptage symétrique) que nous détaillons dans le prochain chapitre.

6. Exercices no4

1. Que vaut :
 - a. $42^{777} \text{ mod } 6$
 - b. $43^{888} \text{ mod } 6$
 - c. $41^{999} \text{ mod } 6$

- d. $12+9 \equiv ? \pmod{5}$
- e. $12-9 \equiv ? \pmod{5}$
- f. $12+3 \equiv ? \pmod{5}$
- g. $15-23 \equiv ? \pmod{5}$
- h. $35*7 \equiv ? \pmod{5}$
- i. $-47*(5+1) \equiv ? \pmod{5}$
- j. $373 \equiv ? \pmod{5}$

2. Soit x un entier et n un entier positif.

On désigne par $x \bmod n$ le reste de la division de x par n .

Évaluer les quantités suivantes :

- a) $13 \bmod 3$
- b) $155 \bmod 19$
- c) $-97 \bmod 11$
- d) $-221 \bmod 23$

3. Que vaut x dans ces expressions :

- a) $x = \text{pgcd}(168, 540)$
- b) $x = \text{pgcd}(1365, 19320)$
- c) $x = \varphi(2701)$
- d) $x = \varphi(67)$
- e) $x = \varphi(1024)$
- f) $x = \varphi(25)$
- g) $x = \varphi(143)$
- h) $x = \varphi(24)$
- i) $x = \varphi(20)$

4. Que vaut x dans ces expressions :

- a) $168 \cdot x \equiv 1 \pmod{557}$
- b) $1100 \cdot x \equiv 1 \pmod{5307}$
- c) $1365 \cdot x \equiv 1 \pmod{19320}$
- d) $853 \cdot x \equiv 1 \pmod{7816}$

5. Que vaut x dans ces expressions :

- a) $10^{123} \equiv x \pmod{143}$
- b) $10^{113} \equiv x \pmod{143}$
- c) $3^{297} \equiv x \pmod{1000}$
- d) $3^{29203} \equiv x \pmod{1000}$
- e) $154^{531} = x \pmod{180}$

f) $269^{876} = x \pmod{1156}$

6. Démontrez que le nombre 6 n'a pas d'inverse multiplicatif modulo 15.

7. Donnez tous les nombres de l'ensemble :

$\{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14\}$ qui sont premiers avec 15.

8. Donnez tous les nombres de l'ensemble :

$\{0,1,2,3,4,5, \dots, 32742\}$ qui sont premiers avec 32743.

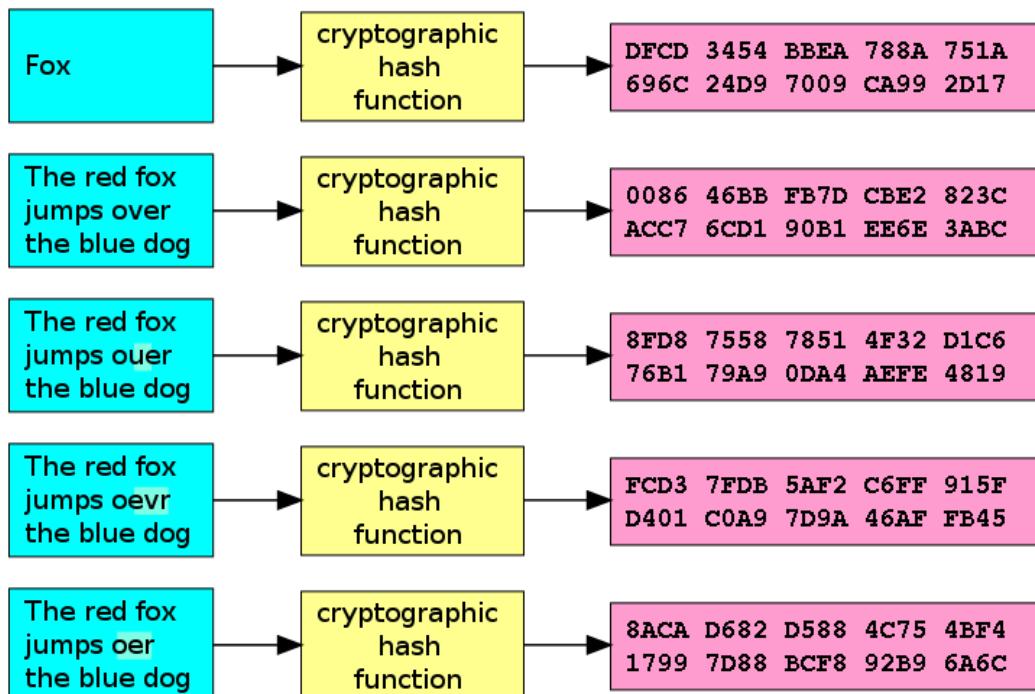
Chapitre 4 : Applications et Considérations Pratiques du RSA

L'algorithme RSA, bien que limité dans quelques cas dans son inefficacité quant aux messages très longs, demeure cependant l'un des algorithmes les plus utilisés dans beaucoup d'applications modernes. En plus, il existe des dispositions pratiques nécessaires dans son application qui méritent d'être explorées entre autres la gestion pratique des clés publiques et des infrastructures y afférentes.

C'est à ces fins que nous incluons ce chapitre dans ce document.

1. Fonctions de hachage

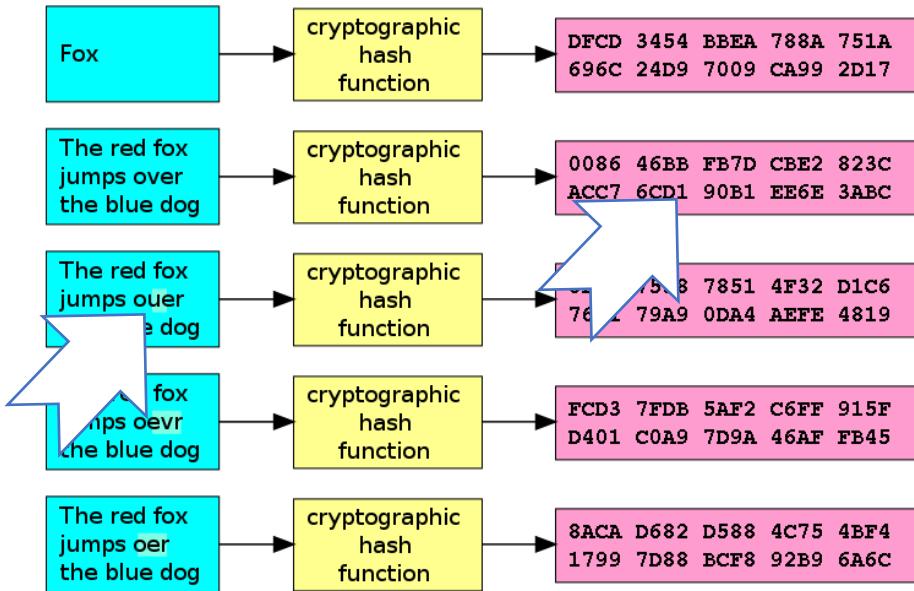
Une fonction de hachage est une fonction permettant d'obtenir un résumé d'un texte, c-à-d. une suite de caractères assez courte représentant le texte qu'il résume. Elle traite une entrée de longueur variable (dans ce cas, un message pouvant contenir indifféremment des milliers ou des millions de bits), afin d'obtenir en sortie un élément de longueur fixe. En cas de modification des données (même d'un seul bit), la fonction de hachage garantit la production d'une valeur de sortie complètement différente.



Exemple d'utilisation d'une fonction de hachage sur un message pour produire une valeur hachée (**résumé de message**)

La fonction de hachage doit :

- être telle qu'elle **associe un et un seul** résumé à un texte en clair (cela signifie que la moindre modification du document entraîne la modification de son résumé).



Le changement d'une lettre dans le message original produit une toute autre valeur hachée complètement différente.

- être une **fonction à sens unique** (one-way function) afin qu'il soit impossible de retrouver le message original à partir du résumé.

$y = F(x)$, mais il est impossible de retrouver x à partir de y !

1.1. Propriétés

Une fonction de hachage "H" transforme une entrée de données d'une dimension variable "m" et donne comme résultat une sortie de données inférieure et fixe "h" ($h = H(m)$).

- l'entrée peut être de dimension variable ;
- la sortie doit être de dimension fixe ;
- $H(m)$ doit être relativement facile à calculer ;
- $H(m)$ doit être une fonction à sens unique ;
- $H(m)$ doit être sans collision. En d'autres termes, elle associe un et un seul résumé à un texte en clair.

Principaux algorithmes

Il existe différents algorithmes réalisant de traitement :

— MD2, MD4 et MD5 (MD signifiant Message Digest), développé par Ron Rivest (société RSA Security), créant une empreinte digitale de 128 bits pour MD5.

Il est courant de voir des documents en téléchargement sur Internet accompagnés d'un fichier MD5, il s'agit du résumé du document permettant de vérifier l'intégrité de ce dernier

— SHA (pour Secure Hash Algorithm, pouvant être traduit par Algorithme de hachage sécurisé), développé par le NIST en 1995. il crée des empreintes d'une longueur de 160 bits.

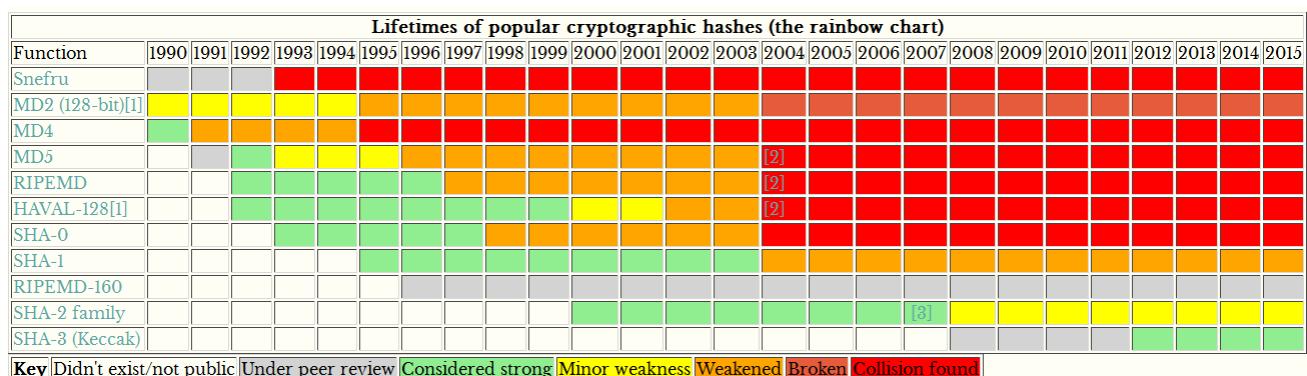
C'est un standard SHA0 et SHA1 (devenu le standard SHS)

— RACE Integrity Primitives Evaluation Message Digest, développé par Hans Dobbertin, Antoon Bosselaers et Bart Preneel ;

— RIPEMD-128 et RIPEMD-160, créé entre 88 et 92 ;

— Tiger, développé par Ross Anderson et Eli Biham, plus rapide que MD5 (132Mb/s contre 37Mb/s sur une même machine, optimisé pour processeur 64bit).

Le tableau suivant donne un aperçu des algorithmes ainsi que leur évolution jusqu'en 2015.



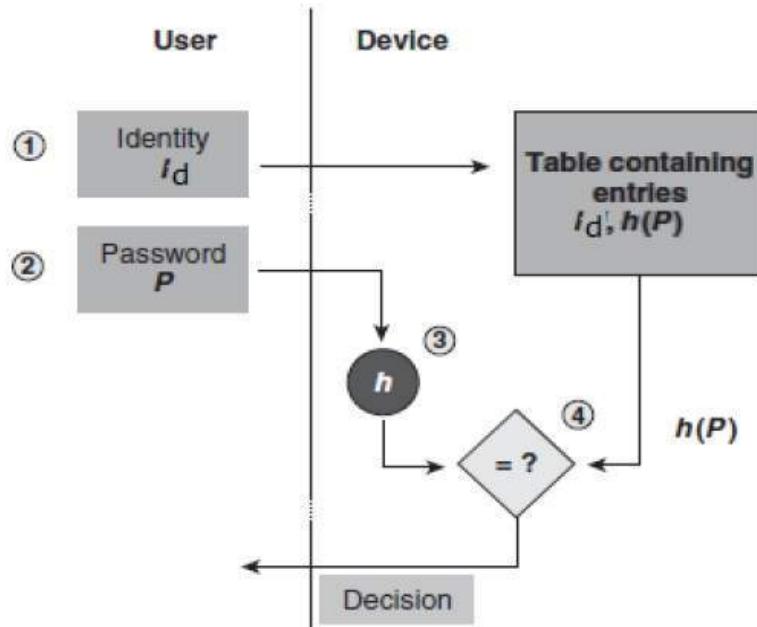
Ces algorithmes produisent des valeurs hachées correspondant aux tailles suivantes :

- MD4 → 128 bit
- MD5 → 128 bit
- SHA-1 → 160 bit
- SHA-2 → 256/512 bit
- SHA-3 → 224/256/384/512 bit

1.2. Applications en Cryptographie

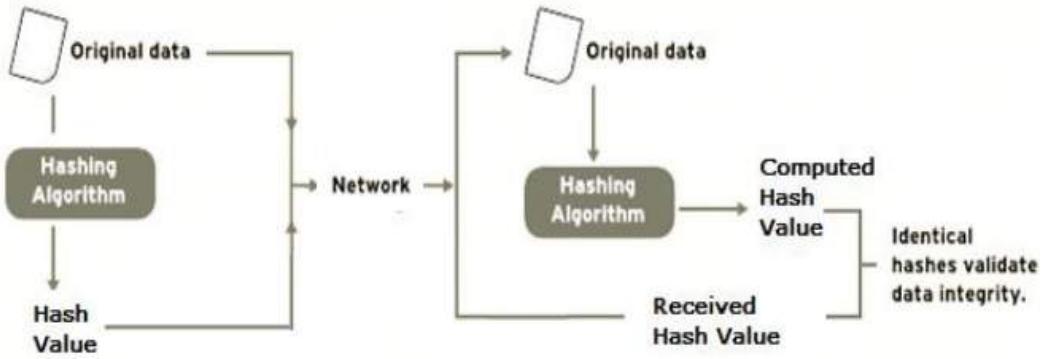
- Stockage de mots de passe :

Au lieu de stocker le mot de passe en clair, la plupart des processus d'ouverture de session stockent les valeurs de hachage des mots de passe dans le fichier. Le fichier Mot de passe se compose d'une table de paires qui sont dans le formulaire (id d'utilisateur, $h(P)$).



Un intrus ne peut voir les hachages des mots de passe, même s'il a accédé au mot de passe. Il ne peut ni logon en utilisant le hachage et ne peut pas dériver le mot de passe de la valeur de hachage puisque la fonction de hachage possède la propriété de la résistance de pré-image.

- Le plus souvent, le hachage est utilisé pour générer les checksums sur les fichiers de données. Cette application fournit l'assurance à l'utilisateur de la justesse des données.



1.3. Cryptage vs Hachage

La principale différence entre le chiffrement et le hachage est que les chaînes cryptées peuvent être inversées dans leur forme d'origine décryptée si vous avez la bonne clé alors que le hachage est un processus à sens unique (du message original on ne peut produire qu'une valeur hachée).

2. Signature Numérique (Digitale)

2.1. Scenario

- Bob envoie un message à Alice

- Alice reçoit un message :

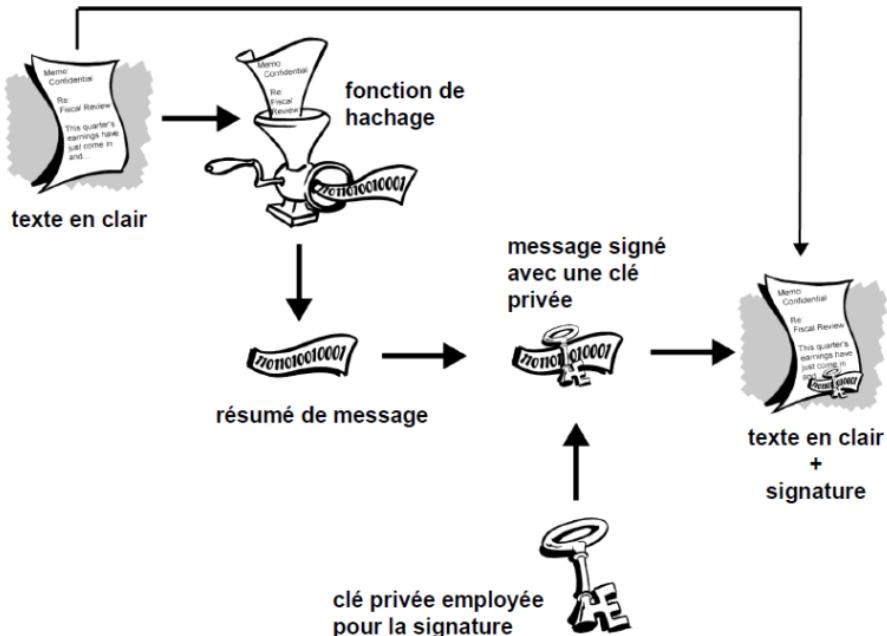
- Le message a été crypté à l'aide de la clé publique d'Alice
- Tout le monde a accès à la clé publique
- Comment peut-elle vérifier qu'il a été envoyé par Bob ?
- Elle ne peut utiliser l'information à laquelle seuls (elle-même et Bob) ont accès : authentification.

2.2. Concept

Avec tous les concepts que nous avons vus jusque-là, il est possible de joindre à un document sa signature obtenue à l'aide d'une fonction de hachage en la chiffrant à l'aide de sa clé privée.

Le document peut être identifié comme provenant de la personne (Authentification) et cela assure également la non-répudiation (utilisation de la clé privée). Il est possible

de déchiffrer cette signature à l'aide de la clé publique de la personne. Cette signature permet de contrôler l'intégrité du document.



Signatures numériques sécurisées

Fonctionnement :

1. L'expéditeur calcule l'empreinte de son texte en clair à l'aide d'une fonction de hachage ;
2. L'expéditeur chiffre l'empreinte avec sa clé privée (Le chiffrement du document est optionnel si la confidentialité n'est pas nécessaire) ;
3. L'expéditeur chiffre le texte en clair et l'empreinte chiffrée à l'aide de la clé publique du destinataire.
4. L'expéditeur envoie le document chiffré au destinataire ;
5. Le destinataire déchiffre le document avec sa clé privée ;
6. Le destinataire déchiffre l'empreinte avec la clé publique de l'expéditeur (authentification) ;
7. Le destinataire calcule l'empreinte du texte clair à l'aide de la même fonction de hachage que l'expéditeur ;

8. Le destinataire compare les deux empreintes (Deux empreintes identiques impliquent que le texte en clair n'a pas été modifié (intégrité)).

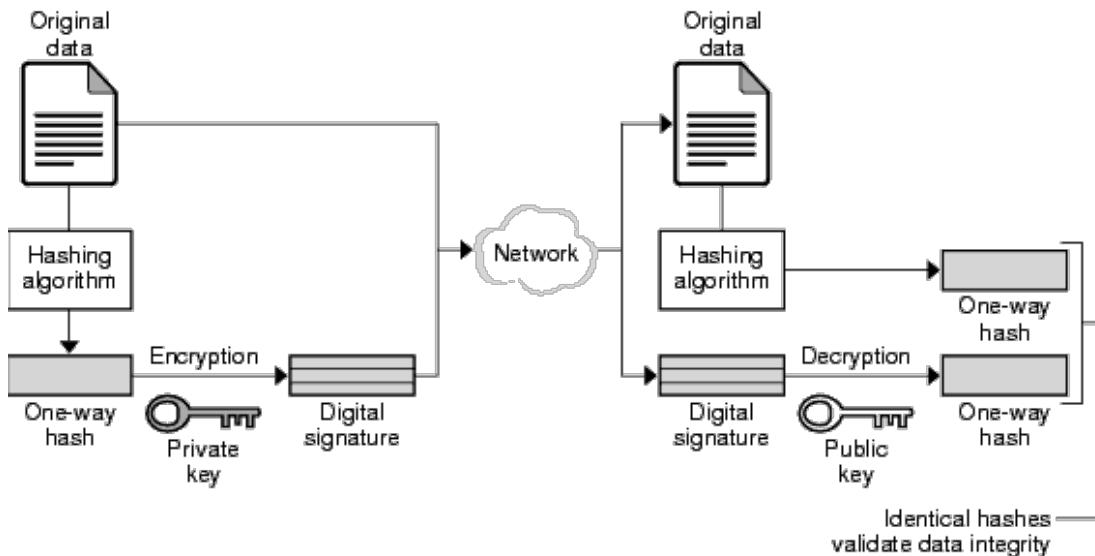


Illustration d'authentification avec hachage et RSA

2.3. Exemple d'implémentation

- Les clés d'Alice : $n_a = 689837$, $e_a = 401$, $d_a = 452945$
- Les clés de Bob : $n_b = 165197$, $e_b = 2333$, $d_b = 23093$
- Message : $M = 107734$
 - M en binaire : 11010010011010110
 - Fonction de hachage : bloc XOR de 6 bits
 - $H = \text{hash}(M) = 111110 = 62$
- Alice envoie un message à Bob
- Elle crypte M avec $(n_b, e_b) = C$ (avec les clés de Bob telles que présentées)
- Sa signature numérique : elle crypte H avec (n_a, d_a) (avec ses propres clés telles que présentées)
 - $DS = 62^{452945} \bmod 689837$
 - $DS = 138031$

- Elle envoie le texte chiffre avec sa signature numérique
- Bob reçoit le message C, DS
- Il décrypte C avec $(n_b, d_b) = M$ (ses propres clés)
- Il calcule la valeur hachée de M = H'
- Il décrypte DS avec (n_a, e_a)
 - $DS = 138031^{401} \text{ mod } 689837$
 - $H = 62$
- Si $H = H'$, alors le message vient d'Alice

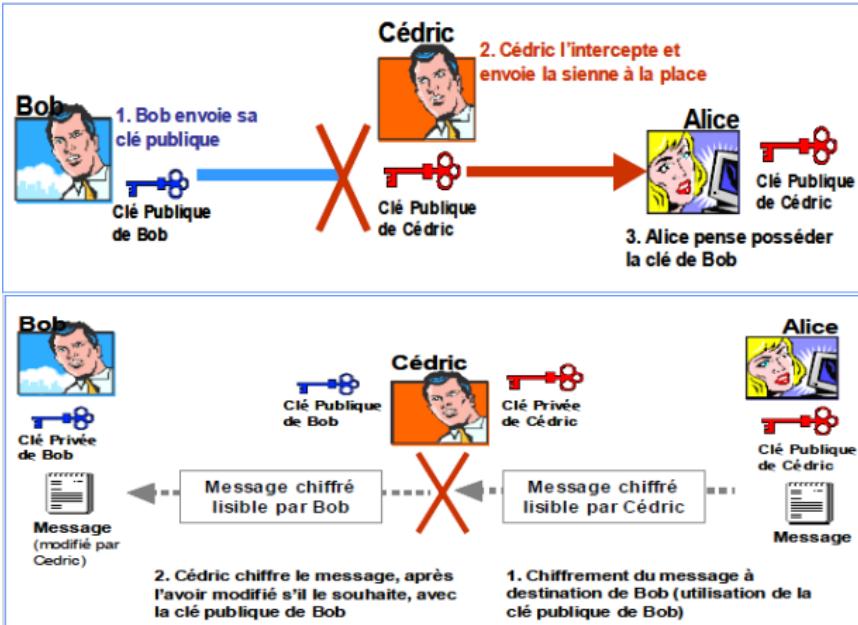
3. Certificats numériques

Lors de l'utilisation des systèmes de cryptographie de clé publique, les utilisateurs doivent constamment vérifier qu'ils cryptent vers la clé du bon utilisateur, ce qui constitue un problème. Dans un environnement où le libre-échange de clés via des serveurs publics est sécurisé, toute attaque menée par une personne intermédiaire, encore appelée un intercepteur, représente une menace éventuelle. Dans ce type d'attaque, une personne place une fausse clé comportant le nom et l'ID utilisateur du destinataire. Les données cryptées (et interceptées) vers le détenteur réel de cette clé erronée sont dorénavant entre de mauvaises mains.

Dans un environnement de clé publique, il est essentiel de s'assurer que la clé publique vers laquelle vous cryptez les données est celle du destinataire concerné et non une contrefaçon. Vous pouvez crypter uniquement vers les clés qui vous ont été distribuées physiquement.

Supposons maintenant que vous devez échanger des informations avec des personnes que vous ne connaissez pas, comment savoir que vous êtes en possession de la bonne clé ?

3.1. Scenario



Ces scénarios mettent en exergue le danger lié à la distribution inadéquate des clés publiques. Ils exemplifient l'attaque la plus fréquente pour ces scénarios. Dans cette attaque, un pirate peut **remplacer la clé publique** présente dans un annuaire par **sa** clé publique. Ainsi, il peut déchiffrer tous les messages ayant été chiffrés avec cette clé. Il peut même ensuite renvoyer à son véritable destinataire le message (modifié ou non) en chiffrant avec la clé originale pour ne pas être démasqué !

Pour résoudre ce problème, en sécurité informatique, on recourt à la notion des certificats numériques. Les certificats numériques ou certificats simplifient la tâche qui consiste à déterminer si une clé publique appartient réellement à son détenteur supposé.

3.2. Notion de certificat et signature numériques

Un certificat correspond à une référence. Il peut s'agir par exemple de votre permis de conduire, de votre carte de sécurité sociale ou de votre certificat de naissance. Chacun de ces éléments contient des informations vous identifiant et déclarant qu'une autre personne a confirmé votre identité. Certains certificats, tels que votre passeport, représentent une confirmation de votre identité suffisamment importante pour ne pas les perdre, de crainte qu'une autre personne ne les utilise pour usurper votre identité.

Un certificat numérique contient des données similaires à celles d'un certificat physique. Il contient des informations associées à la clé publique d'une personne, aidant d'autres personnes à vérifier qu'une clé est authentique ou valide.

Les certificats numériques permettent de contrecarrer les tentatives de substitution de la clé d'une personne par une autre.

Un certificat numérique se compose de trois éléments :

- Une clé publique.
- Des informations sur le certificat. (Informations sur l'« identité » de l'utilisateur, telles que son nom, son ID utilisateur, etc.)
- Une ou plusieurs signatures numériques.

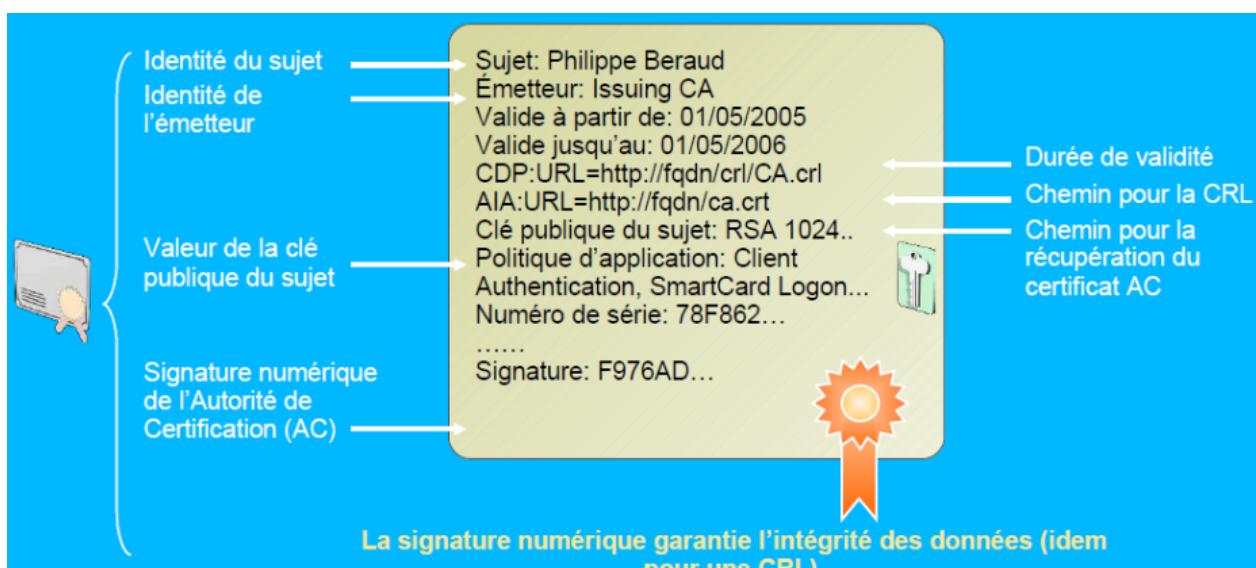
Un certificat permet d'associer une clé publique à une entité (une personne, une machine, ...) afin d'en assurer la validité.

Le certificat est la carte d'identité de la clé publique, délivré par un organisme appelé autorité de certification.

Ces certificats sont émis et signé par une tierce partie, l'autorité de certification ou CA (Certificate Authority).

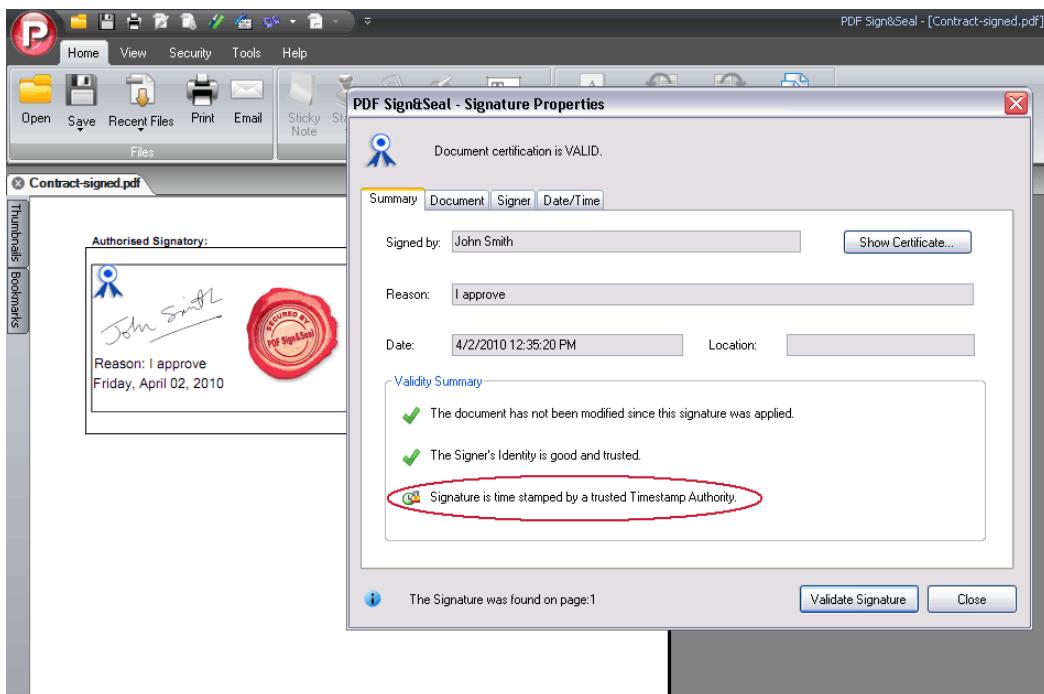
L'autorité de certification est chargée de

- délivrer les certificats ;
- d'assigner une date de validité aux certificats (équivalent à la date limite de péremption des produits alimentaires) ;
- révoquer éventuellement des certificats avant cette date en cas de compromission de la clé (ou du propriétaire).



Exemple de champs d'un certificat numérique

La signature numérique d'un certificat permet de déclarer que ses informations ont été attestées par une autre personne ou entité.



Exemple d'un certificat numérique



Exemple d'un certificat numérique utilisé sur le web

Les certificats sont utilisés lors de l'échange de clés publiques avec un autre utilisateur. Pour un petit groupe de personnes souhaitant communiquer de manière sécurisée, il est facile d'échanger manuellement des disquettes ou des courriels contenant la clé publique de chaque détenteur. Cette distribution manuelle de clés publiques s'avère limitée. Au-delà d'un certain point, il est nécessaire de mettre en place des systèmes pouvant fournir des mécanismes de sécurité, de stockage et d'échanges nécessaires pour que vos collègues ou d'autres personnes puissent communiquer. Ces systèmes peuvent se présenter sous la forme de référentiels de stockage uniquement, appelés serveurs de certificats ou sous la forme de systèmes structurés offrant des fonctions de gestion de clés, appelés infrastructures de clé publique (PKI).

3.3. Notion d'infrastructures de clé publique (PKI)

Une PKI contient les fonctions de stockage de certificats d'un serveur de certificats, qui est un serveur de certificats, également appelé serveur de clés, est une base de données permettant aux utilisateurs de soumettre et de récupérer des certificats numériques

Une PKI offre également des fonctions de gestion de certificats (émission, révocation, stockage, récupération et fiabilité des certificats).

La principale fonction d'une PKI est de présenter l'autorité de certification ou la CA, à savoir une entité humaine (une personne, un groupe, un service, une entreprise ou une autre association) autorisée par une société à émettre des certificats à l'attention de ses utilisateurs informatiques. Une CA fonctionne comme un service de contrôle des passeports du gouvernement d'un pays. Elle crée des certificats et les signe de façon numérique à l'aide d'une clé privée de CA.

Ainsi, la CA est l'élément central d'une PKI. A l'aide de la clé publique de la CA, quiconque souhaite vérifier l'authenticité d'un certificat doit vérifier la signature numérique de la CA émettrice et, par conséquent, l'intégrité du contenu du certificat (essentiellement, la clé publique et l'identité du détenteur du certificat).

3.3.1. Composants

Une PKI est composée :

- d'au moins une autorité de certification (CA),
- d'au moins une autorité d'enregistrement (RA) chargée :

- de vérifier les données d'identification des utilisateurs de certificat électronique, et
- de contrôler les droits liés à l'utilisation des certificats électroniques conformément à la politique de certification.

Une PKI fournit :

- les fonctions de stockage de certificats d'un serveur de certificats,
- des fonctions de gestion de certificats (émission, révocation, stockage, récupération et fiabilité des certificats).

Vérification d'un certificat

- vérifier que le certificat n'a pas expiré, que sa date de validité est correcte ;
- authentifier l'empreinte (provenance de l'AC) et l'intégrité (pas de modification du certificat) ;
- consulter la liste de révocation de l'AC pour savoir s'il n'a pas été révoqué.

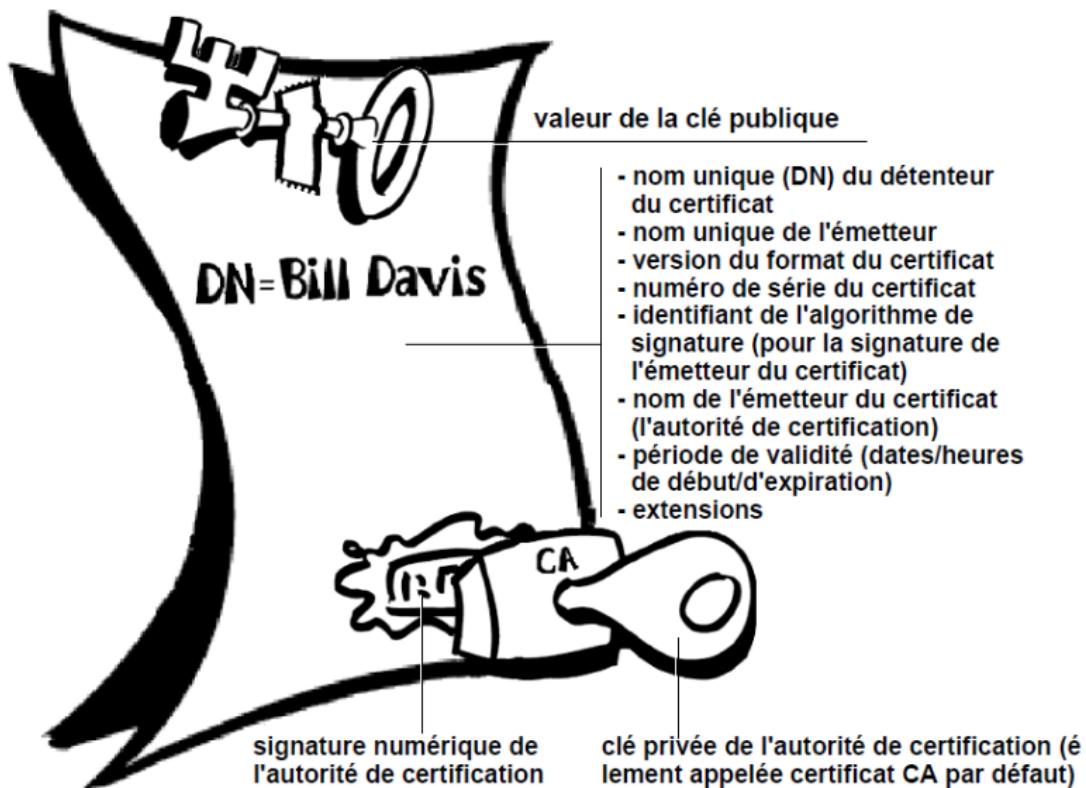


Illustration d'un certificat X.509

3.3.2. Processus de délivrance de certificats

- Bob présente une RA en personne, avec :
 - sa clé publique
 - son passeport/papiers d'identité
- RA vérifie les informations d'identité
 - en faisant usage des systèmes externes
- Bob introduit une requête pour un certificat auprès de la CA
 - CA reçoit les informations de la RA
 - CA crée le certificat (avec l'id de Bob avec sa clé publique)
 - CA introduit aussi son id
 - CA appose sa signature numérique sur le certificat (avec sa clé privée)
 - CA envoie le certificat à Bob
 - CA enregistre le certificat dans le registre public

3.3.3. Vérification de la validité

- Vous pouvez établir la validité manuellement et ce, de plusieurs manières. Vous pouvez demander à votre destinataire de vous remettre physiquement une copie de sa clé publique. Cependant, cette méthode peut s'avérer peu pratique et inefficace.
- Vous pouvez également procéder à une vérification manuelle de l'empreinte digitale du certificat.
- Vous pouvez vérifier la validité d'un certificat en appelant le détenteur de la clé (vous débutez ainsi la transaction) et en l'invitant à lire l'empreinte digitale de sa clé pour vérifier son authenticité. Cette méthode fonctionne si vous connaissez la voix du détenteur.
- Une autre manière d'établir la validité du certificat d'un utilisateur est de faire confiance au tiers qui a effectué le processus de validation.
Par exemple, une CA se doit de vérifier que la partie de clé publique appartient bien au détenteur supposé avant d'émettre un certificat. Toute personne faisant confiance à la CA considère alors que tous les certificats signés par cette CA sont valides.
- Un autre aspect de la vérification de la validité consiste à garantir la non-révocation du certificat.

3.3.4. Etablissement de la fiabilité

Globalement, la CA inspire une confiance totale pour établir la validité des certificats et effectuer tout le processus de validation manuelle. Cette procédure est appropriée pour un nombre défini d'utilisateurs ou de postes de travail.

Au-delà de cette limite, la CA ne peut pas conserver le même niveau de qualité de validation. Dans ce cas, l'intervention d'autres validateurs s'avère nécessaire.

Une CA peut également désigner un gestionnaire en chef de la sécurité. Outre la validité des clés, un gestionnaire en chef de la sécurité définit également leur fiabilité. Le gestionnaire en chef de la sécurité permet à d'autres utilisateurs d'agir en tant que correspondants fiables. Ces correspondants fiables peuvent procéder à la validation des clés de la même manière que le gestionnaire en chef de la sécurité. Ils ne peuvent toutefois pas créer de nouveaux correspondants fiables.

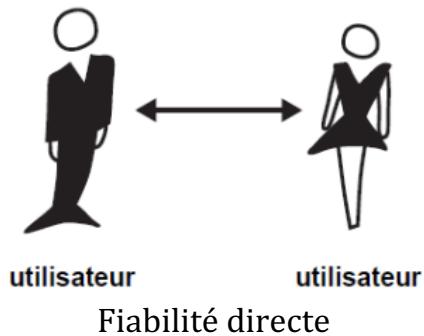
Il existe trois modèles de fiabilité différents :

- Fiabilité directe
- Fiabilité hiérarchique
- Fiabilité du Web

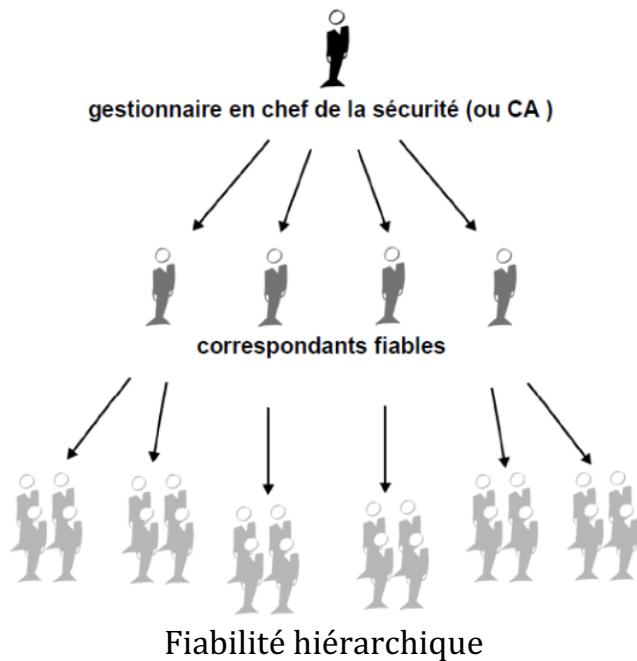
Fiabilité directe

Dans ce modèle, un utilisateur est sûr qu'une clé est valide parce qu'il en connaît la provenance.

Tous les systèmes de cryptographie utilisent cette forme de fiabilité d'une façon ou d'une autre. Par exemple, dans les navigateurs Web, les clés de l'autorité de certification par défaut disposent d'une fiabilité directe, car elles ont été envoyées par le fabricant. S'il existe une forme quelconque de hiérarchie, elle se décline à partir de ces certificats à fiabilité directe.

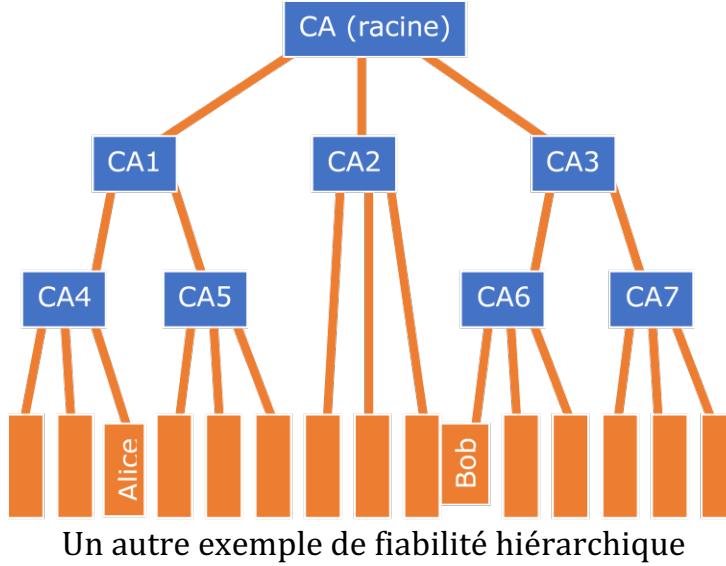


Fiabilité hiérarchique



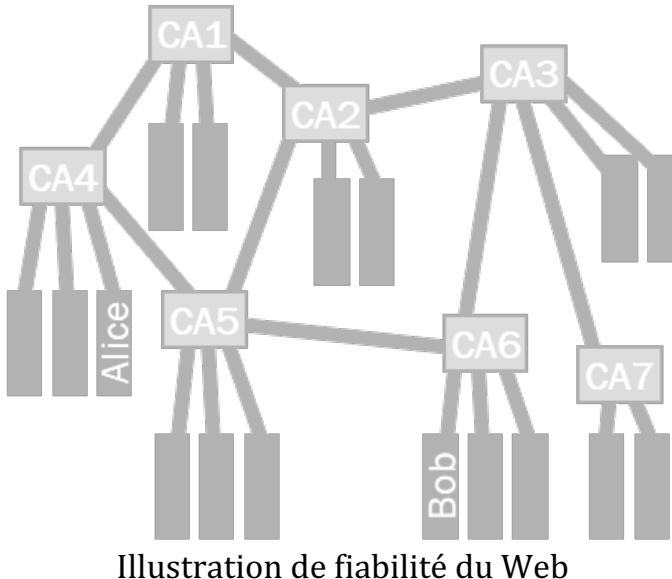
Il existe un certain nombre de certificats « par défaut » sur lesquels se fonde la fiabilité. Ces certificats peuvent directement certifier d'autres certificats ou certifier des certificats qui en certifient d'autres, et ainsi de suite. On peut comparer cette structure à un grand « arbre » de fiabilité.

La validité du certificat « feuille » est vérifiée en remontant vers le certificat qui l'a rendu fiable, puis vers d'autres certificats ayant rendu fiable ce dernier et enfin d'autres le précédent, jusqu'à atteindre le certificat par défaut à fiabilité directe.



Fiabilité du Web

La fiabilité du Web comprend les deux modèles précédents, mais ajoute également la notion selon laquelle la fiabilité dépend de l'opinion de l'utilisateur (qui a une vue réaliste) et l'idée que plus on dispose d'informations, mieux c'est. Il s'agit donc d'un modèle de fiabilité cumulatif. Un certificat peut être rendu fiable directement ou par une chaîne remontant vers un certificat par défaut à fiabilité directe (le gestionnaire en chef de la sécurité) ou par un groupe de correspondants.



3.4. Révocation des Certificats

Les certificats sont utiles tant qu'ils sont valides. Si vous considérez que la validité d'un certificat est permanente, la sécurité n'est plus garantie. Les certificats sont donc créés avec une période de validité par défaut : une date/heure de début et une date/heure d'expiration. Ce certificat peut être utilisé pendant la totalité de sa période de validité (sa durée de vie). Lorsque ce certificat arrive à expiration, il n'est plus valide, car l'authenticité de sa paire de clés/d'identification n'est plus assurée.

L'annulation d'un certificat préalablement à sa date d'expiration peut parfois s'avérer nécessaire, en particulier lorsque son détenteur quitte l'entreprise ou pense que la clé privée correspondante est compromise. Dans ce cas, on parle de révocation.

- On gère la révocation des certificats au moyen de la CRL (Certificate Revocation List)
 - "liste noire" des certificats révoqués
 - La liste est régulièrement actualisée
 - Une autre alternative est le OCSP (Online Certificate Status Protocol)
- Les quelques raisons de révocation:
 - fausse identification du propriétaire du certificat
 - une requête de révocation du propriétaire du certificat
 - quand la cle privee est compromise
 - quand on quitte une organization
 - les détails du propriétaire ont change
 - CA est compromise
 - Le nom ou autres détails de la CA ont change

Chapitre 5 : Cryptage Symétrique (AES)

L'algorithme AES (Advanced Encryption Standard) est le chiffre symétrique le plus largement utilisé aujourd'hui. Parmi les normes commerciales qui incluent AES figurent la norme de sécurité Internet IPsec, TLS, la norme de cryptage Wi-Fi IEEE 802.11i, le protocole réseau shell sécurisé SSH (Secure Shell), le téléphone Internet Skype et de nombreux produits de sécurité à travers le monde. À ce jour, il n'y a pas d'attaques meilleures que la force brute connue contre AES.

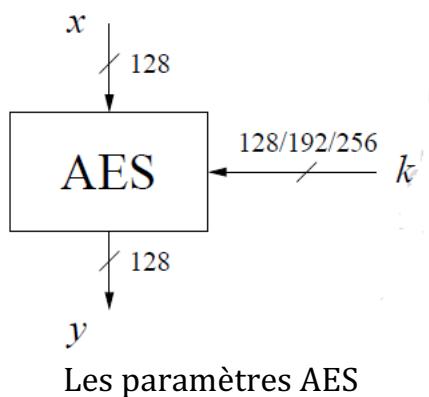
1. Vue d'ensemble de l'algorithme AES

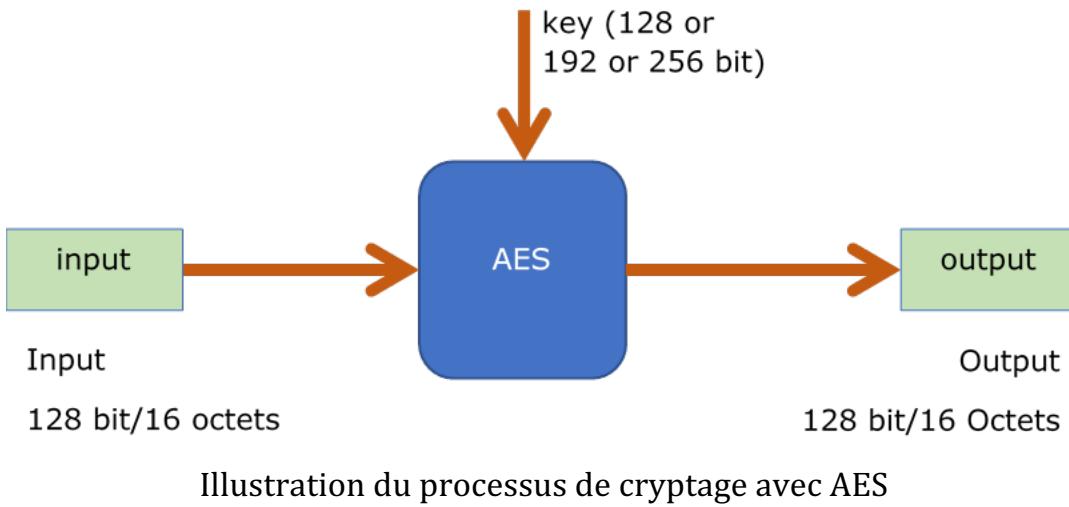
Le système de chiffrement à clé secrète AES est un système basé sur le système Rijndael construit par deux cryptographes belges, Joan Daemen et Vincent Rijmen, en 2000.



C'est un système cryptographique constitué d'une suite d'opérations de permutation et de substitution. Une même clef secrète est utilisée pour les opérations de chiffrement et de déchiffrement.

AES travaille sur des blocs de 128 bits avec des clés de longueur 128, 192 ou 256 bits. Les blocs de données en entrée et en sortie sont des blocs de 128 bits, c'est à dire de 16 octets.





Le process comporte un nombre des rounds(cycles) qui dépend de la taille de la clé selon ce tableau :

Taille de la clé	# rounds (cycles)
128 bit	10
192 bit	12
256 bit	14

On découpe les données et les clés en octets et on les place dans des tableaux pour constituer un état (state) suivant ce schéma :

- On considère un texte input de 16 octets (128 bit)
- Le texte est converti en codes ASCII
- Après la conversion, on constitue la matrice d'état 4x4, colonne par colonne

texte, 16 caract:

A	E	S		U	S	E	S		M	A	T	R	I	X	!
---	---	---	--	---	---	---	---	--	---	---	---	---	---	---	---

ASCII code, 16 bytes:

41	45	53	20	55	53	45	55	20	4D	41	54	52	49	58	21
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Matrice 4x4 (ETAT)

41	55	20	52
45	53	4D	49
53	45	41	58
20	55	54	21

Considérant toutes ces informations, nous pouvons maintenant techniquement dire que l'AES est un système de chiffrement itéré constitué d'un algorithme de chiffrement, la fonction d'étage (cycle ou round), répété de N_r fois, avec N_r allant de 10 à 14, et d'une clé d'une longueur de 128, 192 ou 256 bits pour chaque étage.

Un algorithme de diversification de clé, $\text{ExpandKey}[i]$, permet de créer une clé pour chacun des étages, i , à partir de la clé secrète K .

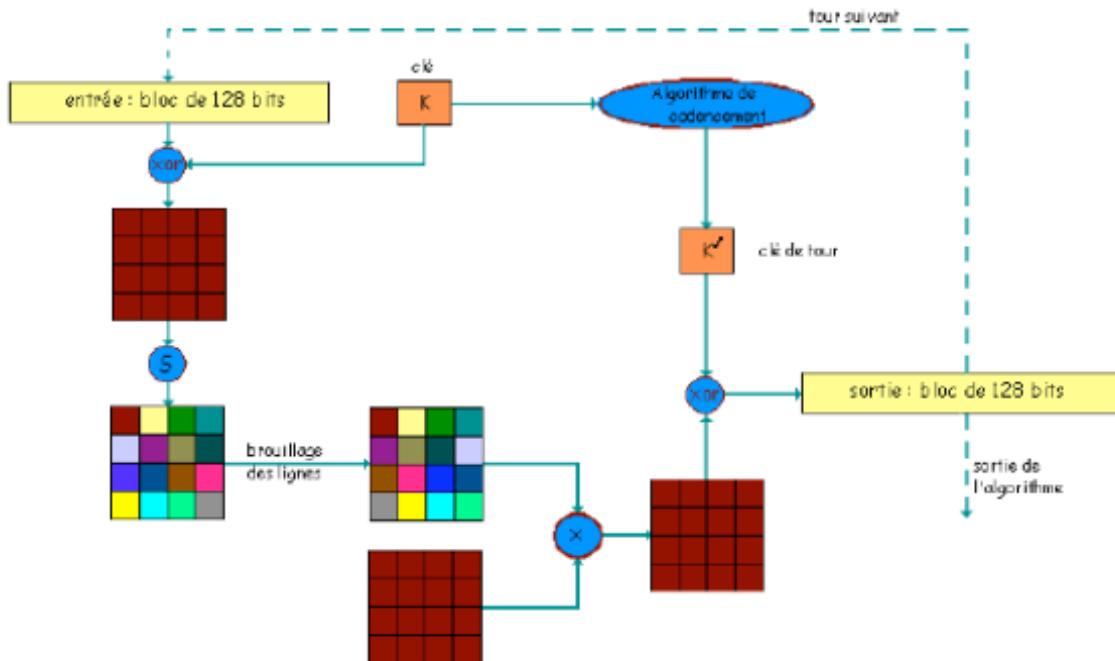
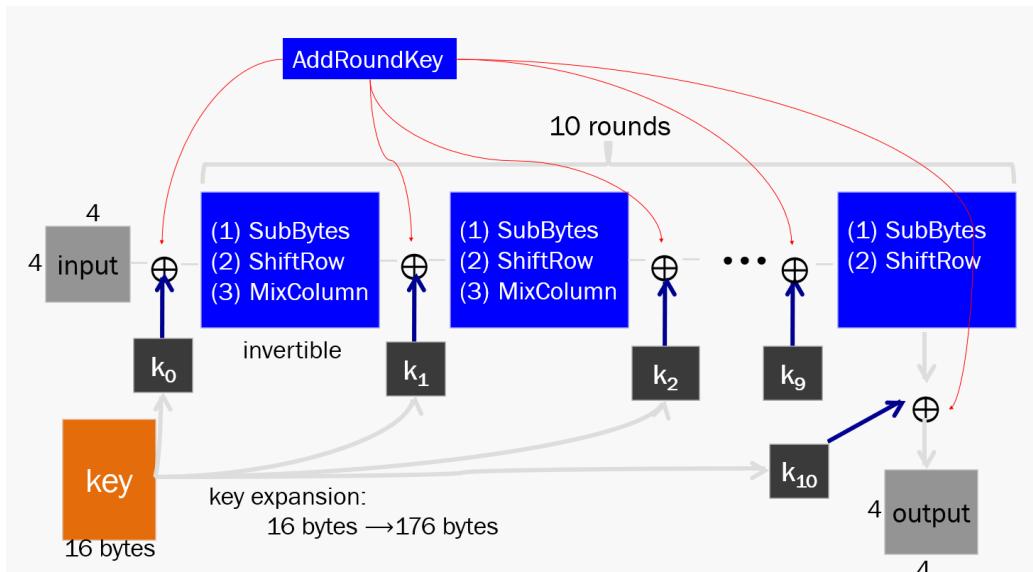


Illustration du déroulement de l'AES

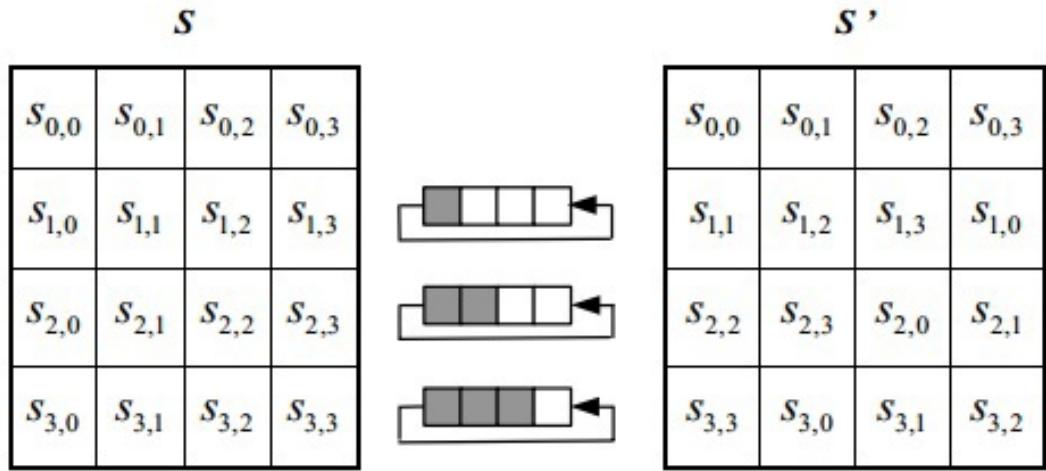
2. Opérations (Fonctions) de l'algorithme AES

Pour chaque étage ou round du chiffrement AES, quatre opérations distinctes sont exécutées successivement : SubBytes, ShiftRows, MixColumns et AddRoundKey.



Schématique du déroulement du chiffrement AES avec toutes les fonctions

Dans chaque cycle ou round, on effectue :



a_0	a_4	a_8	a_{12}
a_1	a_5	a_9	a_{13}
a_2	a_6	a_{10}	a_{14}
a_3	a_7	a_{11}	a_{15}

a_0	a_4	a_8	a_{12}
a_5	a_9	a_{13}	a_1
a_{10}	a_{14}	a_2	a_6
a_{15}	a_3	a_7	a_{11}

Transformation ShiftRows

2.3. MixColumns

C'est est une fonction qui transforme chaque octet d'entrée en une combinaison linéaire d'octets d'entrée. En pratique, il y a deux opérations ici : on effectue la multiplication de chaque colonne et la matrice donnée et on effectue une substitution selon les tables de multiplication suivantes (mul2, mul3..). Nous appliquons exactement le même principe que dans la première fonction SubBytes) avant d'exécuter l'opération XOR.

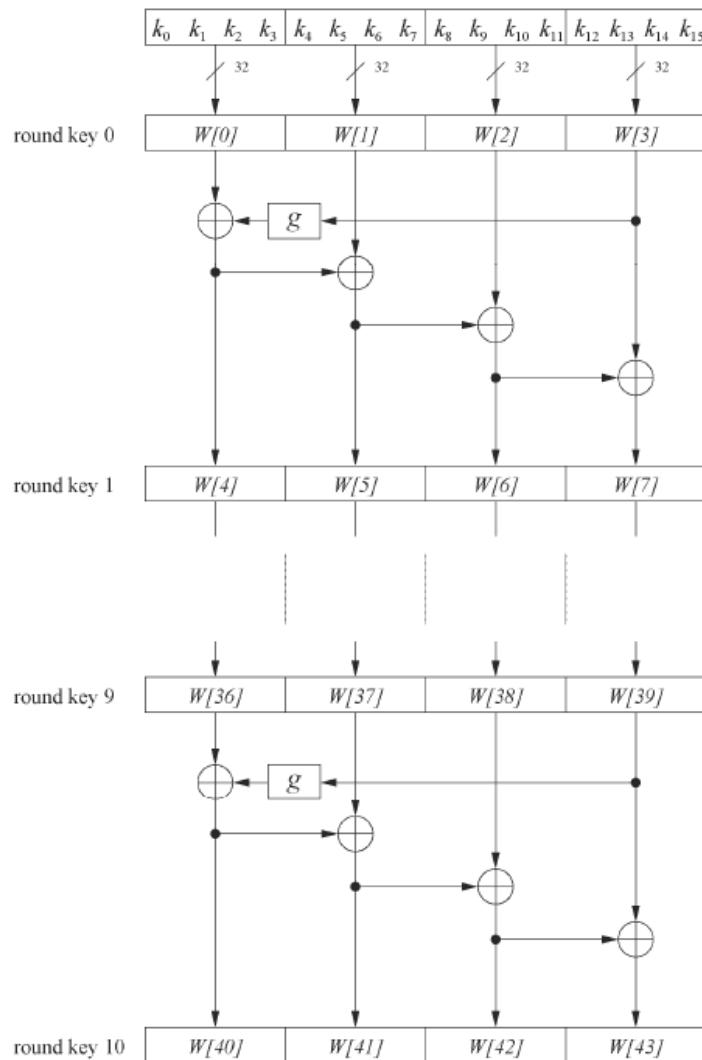
$$\begin{array}{|c|c|c|c|} \hline a_{0,0} & a_{0,1} & a_{0,2} & a_{0,3} \\ \hline a_{1,0} & a_{1,1} & a_{1,2} & a_{1,3} \\ \hline a_{2,0} & a_{2,1} & a_{2,2} & a_{2,3} \\ \hline a_{3,0} & a_{3,1} & a_{3,2} & a_{3,3} \\ \hline \end{array} \oplus \begin{array}{|c|c|c|c|} \hline k_{0,0} & k_{0,1} & k_{0,2} & k_{0,3} \\ \hline k_{1,0} & k_{1,1} & k_{1,2} & k_{1,3} \\ \hline k_{2,0} & k_{2,1} & k_{2,2} & k_{2,3} \\ \hline k_{3,0} & k_{3,1} & k_{3,2} & k_{3,3} \\ \hline \end{array} = \begin{array}{|c|c|c|c|} \hline b_{0,0} & b_{0,1} & b_{0,2} & b_{0,3} \\ \hline b_{1,0} & b_{1,1} & b_{1,2} & b_{1,3} \\ \hline b_{2,0} & b_{2,1} & b_{2,2} & b_{2,3} \\ \hline b_{3,0} & b_{3,1} & b_{3,2} & b_{3,3} \\ \hline \end{array}$$

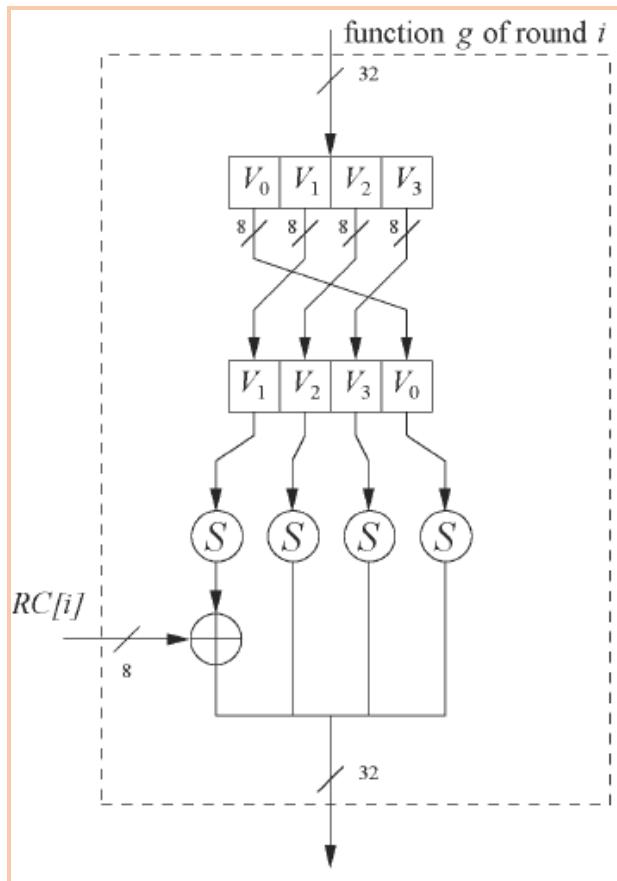
Illustration d'une opération AddRoundKey entre un Etat et une clé.

2.5. Extension de la Clé (ExpandKey)

L'opération ExpandKey dépend de la taille de blocs choisie, $N_b \times 32$, qui peut être égale à 128, 160, 192 ou 224, 256 bits, et de la taille de clé choisie, $N_k \times 32$, qui peut être égale à 128, 160, 192, 224 ou 256 bits.

On supposera dans la suite que la longueur de la clef est de 192 bits avec donc $N_k = 6$.





On écrit la clé K sous forme d'une matrice de N_k colonne de 4 bytes chacune (donc 4 lignes l'élément d'une ligne étant un mot de 8 bits) dénotés k_0, \dots, k_5

k_0	k_1	k_2	k_3	k_4	k_5
-------	-------	-------	-------	-------	-------

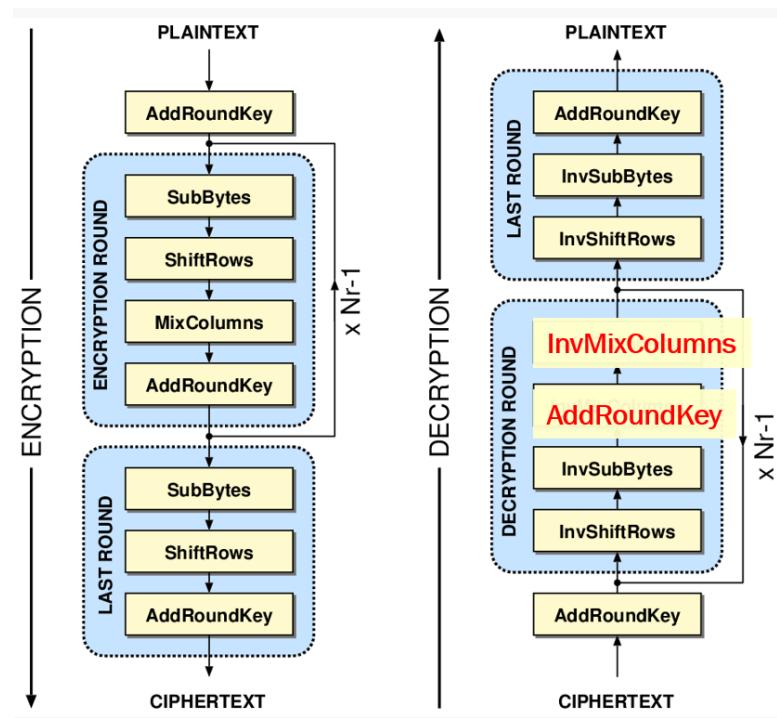
Ensuite cette matrice est étendue en une matrice de taille $N_b (N_r + 1)$ ou N_r est le nombre d'étages par l'algorithme suivant :

ETAT après MixColumns			
89	06	DE	CB
B8	09	BC	32
1A	27	9D	63
12	74	9A	F8

Etes-vous en mesure de démontrer les étapes de ces transformations ?

3. Le déchiffrement avec AES

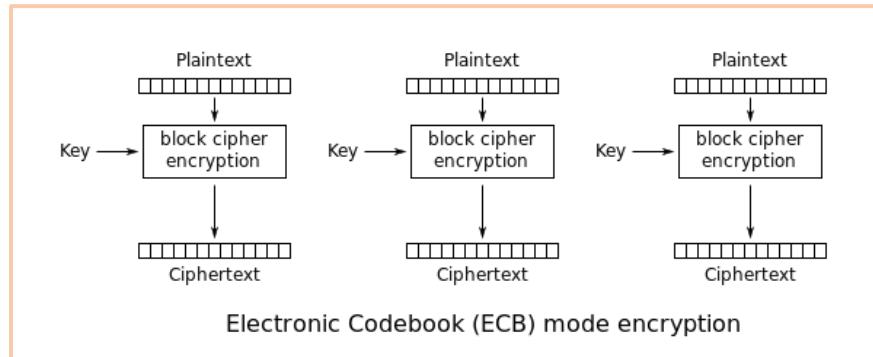
Pendant le déchiffrement, les clés de cycle (rounds ou rondes) sont utilisées dans l'ordre inverse de celui du chiffrement. On notera que l'ordre des transformations diffère de celui du chiffrement. Le déchiffrement consiste à appliquer dans l'ordre inverse du chiffrement les transformations inverses correspondantes ("détricoter le chiffrement") comme le démontre le graphique suivant.



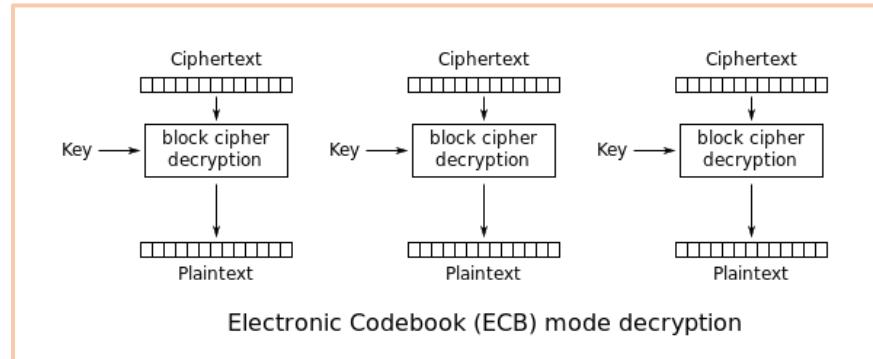
Les opérations du chiffrement et déchiffrement avec AES

4.1. Electronic Code Book (ECB)

Mode d'opération normal : il applique l'algorithme au texte clair en transformant normalement chaque bloc de texte clair. Il peut s'exécuter en parallèle.

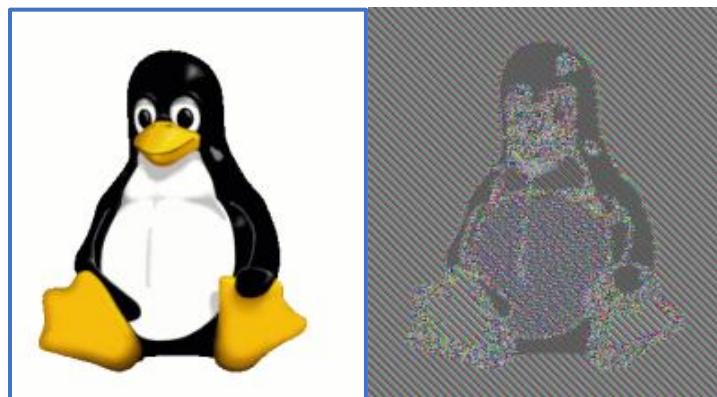


Chiffrement en mode ECB

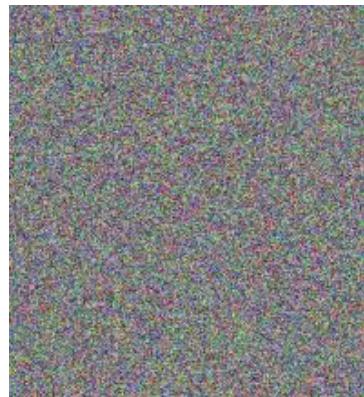


Déchiffrement en mode ECB

Le problème avec ce mode est que si on utilise deux fois le même texte clair et la même clé de chiffrement, le résultat du chiffrement sera identique.



Pas vraiment sécurisant



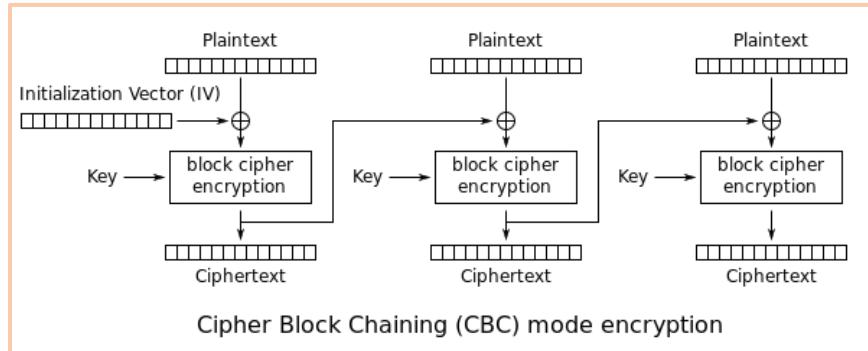
Meilleur chiffrement

4.2. CBC : Cipher Block Chaining

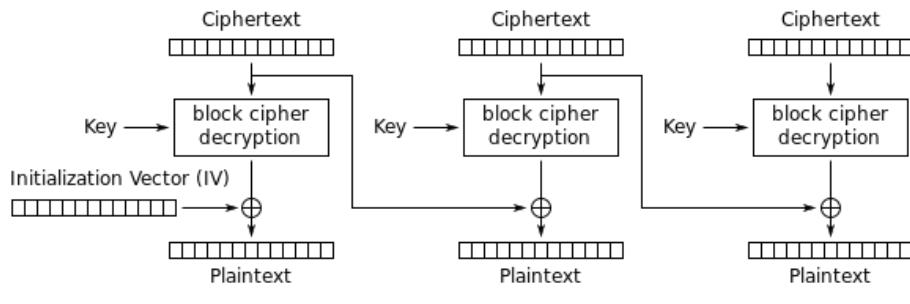
C'est un des modes les plus populaires. Il apporte une solution au premier problème du mode ECB :

- avant d'être chiffré, l'opération binaire « XOR » est appliquée entre le bloc actuel de texte en clair et le bloc précédent de texte chiffré ;
- pour le tout premier bloc, un bloc de contenu aléatoire est généré et utilisé, appelé « vecteur d'initialisation » (initialization vector, ou IV).

Ce premier bloc est envoyé tel quel avec le message chiffré.



Chiffrement en mode CBC



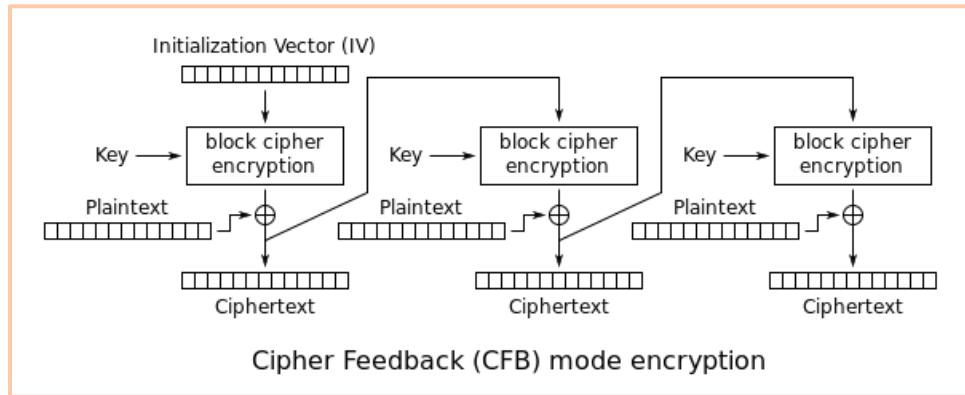
Cipher Block Chaining (CBC) mode decryption

Déchiffrement en mode CBC

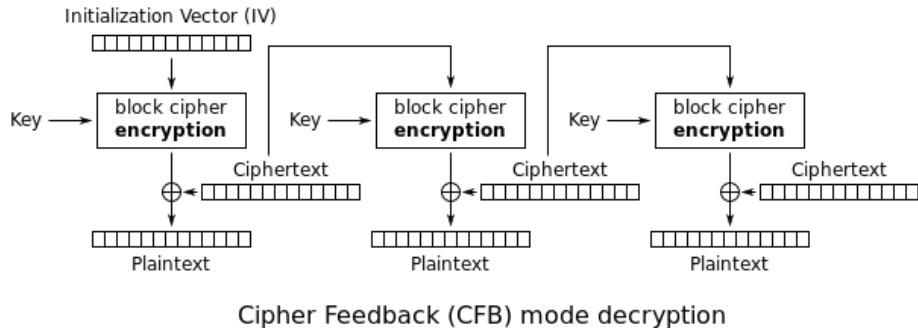
- Vecteur d'initialisation : un bloc aléatoire imprévisible de 128 bits
 - Premier bloc en texte brut randomisé avec IV
 - Bloc de texte brut suivant randomisé avec le texte chiffré précédent
- Randomisé
- Non parallélisable

4.3. CFB : Cipher Feedback

Dans ce mode, l'opération XOR est appliquée entre le bloc de texte clair et le résultat précédent chiffré à nouveau par la fonction de chiffrement. Il offre une grande sécurité. Pour le premier bloc de texte clair, on génère un vecteur d'initialisation.



Chiffrement en mode CFB



Déchiffrement en mode CFB

- Vecteur d'initialisation : un bloc imprévisible, aléatoire et non secret de 128 bits
 - IV est cryptée dans la première étape, puis XOR avec le texte brut
 - Le chiffrement résultant est utilisé comme IV à l'étape suivante

- Randomisés
- Non parallélisable

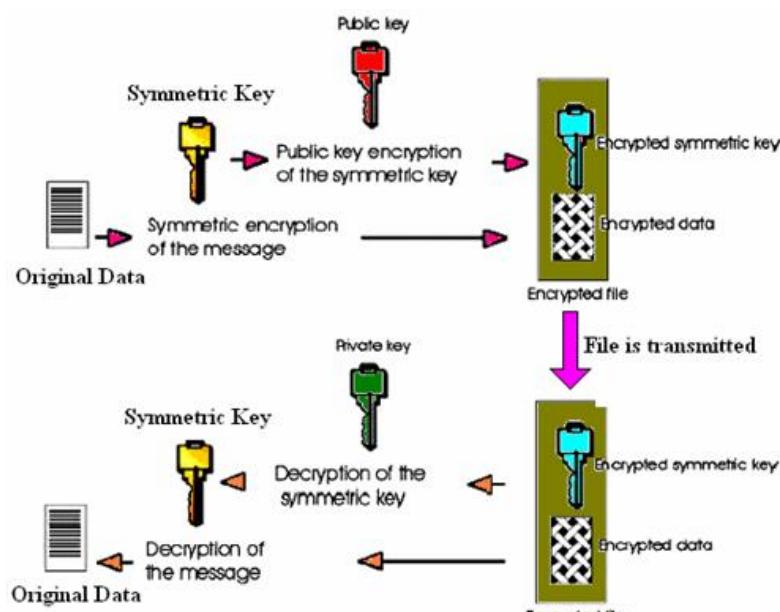
5. Chiffrement Hybride

Nous avons vu que :

- AES est utile aux messages plus larges et de grande taille : mais comment est-il possible de partager la clé ?
- RSA est utile au message plus court : étant un algorithme très efficace, il peut être utilisé ensemble avec AES pour faciliter le partage ou transfert de la clé.

Le processus hybride suivrait ces étapes :

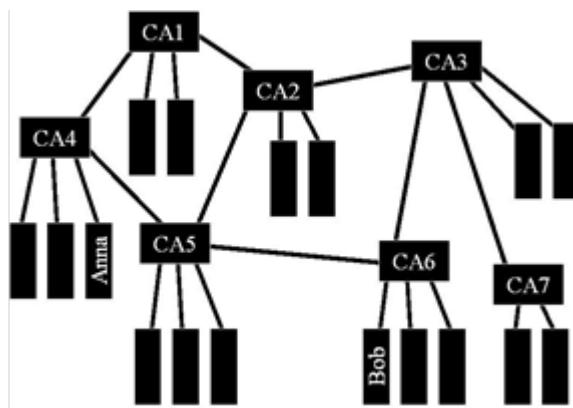
- générer aléatoirement une clé de taille raisonnable utilisée pour un algorithme de cryptage symétrique (AES par exemple) ;
- chiffrer cette clé à l'aide d'un algorithme de cryptage à clé publique (à l'aide de la clé publique du destinataire) ; Cela impose que l'un des interlocuteurs possède la clé publique de l'autre (pas toujours facile de s'assurer que la clé publique appartient bien à la bonne personne) ;
- l'expéditeur chiffre le (long) message avec AES ;
- le récepteur décrypte le message avec la clé appropriée.



Chiffrement et déchiffrement hybride

6. Exercices No5

2. Expliquez pourquoi, ensemble, (e, n) et (d, n) sont, oui ou non, un système RSA complet.
3. Supposons que nous ayons cette structure des autorités de certification (ca):



Comment Anna peut-elle vérifier la validité du certificat de Bob, s'il est connu que Ca5 est corrompu ?

4. Tenant compte de l'état suivant à décrypter, donnez la valeur correspondante pour les octets marqués après l'exécution de la fonction invMixColumns:

04	9F	82	53
66	21	4D	FC
81	59	A1	B7
E5	B6	8A	49

5. Pour les 3 modes de bloc de chiffrement, citez des problèmes concrets qui seront particulièrement et distinctement appropriés pour chacun. Soutenez votre argument de façon succincte

6. Mini Projet :

Cet exercice concerne le processus de génération et d'extension des clés AES. Vous êtes libre de consulter toutes les sources supplémentaires pour vous assurer que vous produisez un travail de qualité. Vous devez

étudier le processus et le mettre en œuvre en utilisant n'importe quel langage de programmation avec lequel vous êtes à l'aise en fonction de votre propre exemple.

En fin de compte, vous devez soumettre :

- Un rapport technique sur le processus du calendrier clé dans AES (pour 3 tailles différentes)
- Le code source
- L'interface démontrant le projet implémentant l'algorithme
- Présentation PowerPoint.

Chapitre 6 : Le management de la sécurité

1. Contexte

Après avoir parcouru les cinq premiers chapitres de ce document, il convient de conclure ce cours par des normes importantes et des notions essentielles sur la gestion et le management de la sécurité.

L'Organisation internationale de normalisation, ou International organization for standardization en anglais (ISO pour la forme abrégée) est l'organisation internationale qui produit des normes internationales dans des domaines industriels et commerciaux. Selon la norme IS (pour International Standard) 9000, l'ISO définit un système de management comme un système qui permet :

- d'établir une politique ;
- de fixer des objectifs ;
- de vérifier que l'on a atteint les objectifs fixés.

Pour déterminer le contexte dans lequel des décisions et stratégies de maintien et de gestion de la sécurité informatique, l'ISO suggère des normes suivant un système de management. Un système de management comporte un ensemble de mesures organisationnelles et techniques destinées à mettre en place un certain contexte organisationnel et à en assurer la pérennité et l'amélioration.

L'objectif est que parce que le système de management repose sur référentiel écrit, le document de management de sécurité informatique sera donc vérifiable, au moyen d'un audit qui consistera à comparer le référentiel à la réalité pour relever les divergences, nommées écarts ou non-conformités.

2. Le système de management de la sécurité de l'information

La norme IS 27001 [79] est destinée à s'appliquer à un système de management de la sécurité de l'information (SMSI) ; elle comporte notamment un schéma de certification susceptible d'être appliqué au SMSI au moyen d'un audit. La norme IS 27001 repose sur une approche par processus, et plus précisément sur le modèle de processus nommé roue de Deming, ou PDCA, comme Plan, Do, Check, Act :

- phase Plan : définir le champ du SMSI, identifier et évaluer les risques, produire le document (Statement of Applicability, SOA) qui énumère les mesures de sécurité à appliquer ;

- phase Do : affecter les ressources nécessaires, rédiger la documentation, former le personnel, appliquer les mesures décidées, identifier les risques résiduels;
- phase Check : audit et revue périodiques du SMSI, qui produisent des constats et permettent d'imaginer des corrections et des améliorations ;
- phase Act : prendre les mesures qui permettent de réaliser les corrections et les améliorations dont l'opportunité a été mise en lumière par la phase Check, préparer une nouvelle itération de la phase Plan.

Le SMSI a pour but de maintenir et d'améliorer la position de l'organisme qui le met en œuvre du point de vue, selon les cas, de la compétitivité, de la profitabilité, de la conformité aux lois et aux règlements, et de l'image de marque. Pour cela il doit contribuer à protéger les actifs (assets) de l'organisme, définis au sens large comme tout ce qui compte pour lui.

2.1. Élaboration et mise en place du SMSI

La norme IS 27001 précise la démarche qui doit être suivie pour élaborer et mettre en place le SMSI :

- définir le champ du SMSI ;
- en formuler la politique de management ;
- préciser la méthode d'analyse de risques utilisée ;
- identifier, analyser et évaluer les risques ;
- déterminer les traitements qui seront appliqués aux différents risques, ainsi que les moyens d'en vérifier les effets ;
- attester l'engagement de la direction de l'organisme dans la démarche du SMSI ;
- rédiger le Statement of Applicability (SOA) qui sera la charte du SMSI et qui permettra de le soumettre à un audit.

2.2. Suivi et application du SMSI

Ici, la norme précise que, une fois que le SMSI a été formulé, il faut faire ce qu'il stipule, vérifier que c'est fait, identifier les erreurs dans son application, les failles qui s'y manifestent et les modifications du contexte nécessitant sa mise à jour ou sa modification.

Pour ces tâches elles-mêmes, l'ISO a produit des documents normatifs : IS 27003 pour l'implémentation, IS 27004 définit des indicateurs de qualité pour le SMSI, IS 27006 encadre le processus de certification du SMSI, IS 27007 le processus d'audit.

2.3. Tâches de direction et d'encadrement

Essentiellement, il incombe à la cellule or équipe dirigeante s'occupant du SMSI, un certain nombre d'obligations : vérifier que tout est bien fait selon les règles, affecter à la démarche du SMSI des ressources suffisantes en personnel et en moyens matériels, déterminer les besoins qui en résultent en termes de compétence et de formation, fournir les efforts qui conviennent en termes de sensibilisation et de formation, effectuer le contrôle des effets de ces efforts. Il faut aussi organiser des revues et des exercices, etc., tout cela afin d'assurer l'amélioration continue du SMSI.

3. Conclusion

Lorsque la gestion des procédures d'établissement et d'implémentation de politique de sécurité dans une entreprise reçoit le même degré de priorité que toutes les démarches stratégiques, cette entreprise démontre des signaux imposants de réussite.

Les menaces sont réelles, les risques inhérents et les vulnérabilités évidentes. Plusieurs solutions, des applications et des systèmes de sécurité incorporant les notions que nous avons apprises existent sur le marché. Cependant, il sied de noter qu'une politique de gestion des questions de sécurité est nécessaire.

Sur ce point, plusieurs paradigmes et points de vue se proposent de montrer la donne démarche mais il sera raisonnable de reconnaître que chaque environnement est différent et que la meilleure approche est celle adaptée aux besoins de cet environnement, aussi longtemps qu'elle soit aussi procédurale que rationnelle.

4. Exercices N°6

Avec cet exercice, vous êtes appelés à démontrer votre maitrise et surtout compréhension du besoin de management de la sécurité dans une entreprise :

1. Considérez une entreprise de télécommunications en RDC. Imaginez les menaces, risques et vulnérabilités.
2. Expliquez et décrivez pareil scenario. Motivez votre choix
3. Pourriez-vous démontrez les conséquences de la non-prise en compte du management de la sécurité dans cet environnement ?
4. Développez un SMSI pour cette entreprise. Exposez-en les grandes parties
5. Le SMSI mis en place, en quoi est-il différent des principes de la norme ISO 270001 ? Pourquoi ?

Bibliographie

1. Carpentier, Jean-François. La sécurité informatique dans la petite entreprise : état de l'art et bonnes pratiques. Editions ENI, 2009.
2. Ghernaouti-Helie, Solange. Sécurité informatique et réseaux - Cours et exercices corrigés. Editions DUNOD, 2006.
3. Ghernaouti-Helie, Solange. Cybersécurité : Sécurité informatique et réseaux - Cours et exercices corrigés. Editions DUNOD, 2011. 5^e Edition
4. Bloch, Laurent et al. Sécurité informatique : Pour les DSI, RSSI et administrateurs. Éditions Eyrolles, 2016.
5. Poinsot, Laurent. Cours : Introduction à la sécurité Informatique.
6. Agence nationale de la sécurité des systèmes d'information, [pas de date]. Comprendre et anticiper les attaques DDoS.
Lien : https://www.ssi.gouv.fr/uploads/2015/03/NP_Guide_DDoS.pdf
7. Dumont, Renaud, 2010. Cryptographie et Sécurité informatique : Notes du cours
8. Network Associates. Introduction à la cryptographie. Network Associates International, 1999.
9. Auteur Inconnu, Les chiffres Autoclaves. Disponible au lien :
http://www.bibmath.net/crypto/index.php?action=affiche&quoi=ancienne/a_utoclave
10. Le Van Kim, Rachel. Les ensembles de nombres en mathématiques N Z D Q R, 2018.
Disponible au lien :
<https://jeretiens.net/ensembles-de-nombres-mathematiques/>
11. La décomposition en facteurs premiers. Disponible en ligne au lien :
https://fr.wikipedia.org/wiki/D%C3%A9composition_en_produit_de_facteurs_premiers
12. Delahaye, Jean-Paul, La décomposition en facteurs premiers. Disponible au lien :
<https://www.futura-sciences.com/sciences/dossiers/mathematiques-merveilleux-nombres-premiers-1791/page/3/>
13. Bonnefoi, Pierre-François. Notes du Cours de Sécurité Informatique