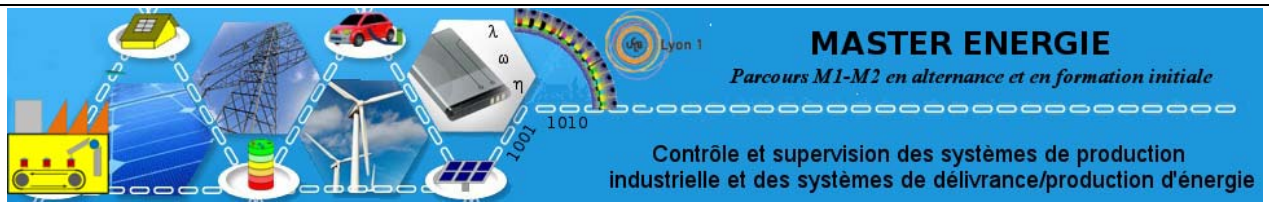


Master Energie, Sécurité des Réseaux, Jean-Marc THIRIET

Jean-marc.thiriet@univ-grenoble-alpes.fr

DOCUMENT DE TRAVAIL A DISTANCE REMPLACANT LE COURS PREVU LE 20 MARS 2020

Année 2019-2020

Préambule

Je vous ai préparé ce document de travail pour remplacer la séance que nous aurions dû avoir ensemble le vendredi 20 mars. J'espère que vous êtes tous dans des conditions correctes de confinement, avec peut-être pour certains du travail à distance avec votre entreprise (je sais que la plupart d'entre vous sont alternants) ou même sur site pour d'autres.

Nous vivons des conditions exceptionnelles, essayons d'en tirer des leçons de vie, en espérant que tout se passera bien pour vous et vos proches. Je vous souhaite le meilleur.

Jean-Marc Thiriet

Introduction

Ce document est un guide de travail, avec un rendu à la fin, qui est estimé l'équivalent de 6 heures d'enseignement présentiel + 3 à 4 heures de travail personnel maximum, soit une petite dizaine d'heures. Ce document comprend 7 pages de cours et 2 pages de devoir. Nous avons convenu de vous donner 3 à 4 semaines, je vous propose donc de me renvoyer votre devoir (en fin de ce document + un document personnalisé annexe) pour

Le **mercredi 22 avril 2020 à 20 heures** (ou le **vendredi 17 avril 2020**, si vous prenez les congés scolaires).

Je rappelle que mes transparents et le poly que je vous ai distribués sont accessibles :

http://www.gipsa-lab.grenoble-inp.fr/~jean-marc.thiriet/lyon/

N'hésitez pas à me contacter si vous avez des questions :

jean-marc.thiriet@univ-grenoble-alpes.fr

Tout d'abord, lors de la première séance de cours, le 6 mars, nous avons traité les points suivants décrits ci-après.

1. Analyse de risques.

Nous avons abordé un certain nombre de concepts, autour de la **sûreté de fonctionnement**, la sécurité, la **cyber-sécurité**. Ce qui nous intéresse globalement est l'étude de la **cyber-sécurité des systèmes cyber-physiques**. Nous avons réfléchi à l'ordre des **priorités en termes de sécurité** selon que l'on s'intéresse à la technologie de l'information « traditionnelle » ou aux systèmes industriels, voire critiques. Nous avons vu que la sécurité couvrait un grand nombre de domaines, de métiers, d'approches, en particulier les concepts de **sécurité physique, sécurité de**

l'exploitation, sécurité logique, sécurité applicative.
Lorsque l'on parle de sécurité, il faut circonscrire l'objet d'études. Il ne s'agit pas de déployer une infrastructure de sécurité complexe, incompréhensible, voire même incohérente, pour protéger quelque chose sans importance, autrement dit il faudra avoir une vision globale, c'est ce que l'on appelle la **sécurité en profondeur** (*in-depth security*). Il nous faudra quantifier le risque et donc effectuer une **analyse de risques** qui nous permettra de mettre en œuvre un **plan d'action** de façon à ce que l'on soit capable de montrer (documentation, certifications) que l'on a une bonne **maîtrise de risques** (en regard de la législation, des contextes applicatifs, des normes en vigueur, de l'état de l'art et du savoir-faire…).
Nous avons fait ensemble un exercice par groupe, permettant de s'initier à l'analyse de risques.

2. Crypto-systèmes à clés symétriques.

Les systèmes à clés symétriques se caractérisent par le fait que la même clé est utilisée pour crypter et décrypter. Ce sont des systèmes qui fonctionnement bien mais dont la principale vulnérabilité est l'échange de cette clé. Nous sommes partis de l'algorithme de translation de César, aisément cassable par analyse fréquentielle. Nous avons vu le code poly-alphabétique, qui est robuste à l'analyse fréquentielle. Ces techniques sont anciennes pour certaines et traitent des lettres.
Dans le monde numérique d'aujourd'hui, les systèmes de cryptage s'intéressent à des suites binaires. Nous avons vu des mécanismes simples dits de « brouillage numérique », tels que l'inversion des bits suivant une suite aléatoire, la permutation de distance. Brouillage numérique (*digital jamming*) signifie que l'on transforme les suites (blocs) binaires en vue du cryptage de telle sorte qu'à partir de la suite cryptée, il soit impossible de remonter à l'original sans la clé, par des techniques de cryptanalyse (analyse fréquentielle par exemple). Ces techniques de brouillage lorsqu'on y regarde de plus près, sont toutes simples, et peuvent en fait être aisément implémentées sur des circuits à base de fonctions logiques (type portes XOR). Ceci est une caractéristique essentielle des systèmes de cryptage symétrique, c'est que les temps de calcul sont très courts, ils peuvent aisément être implémentés sur des cartes réseau pour du cryptage en temps réel.

Nous avons également réfléchi ensemble sur l'influence de la taille de la clé sur la robustesse du système cryptographique, en lien notamment avec le temps de calcul.

A partir d'ici, je vous invite à parcourir les transparents, je commente les points les plus marquants.
Nous nous sommes arrêtés au niveau du transparent 31 du chapitre 2 qui présente le DES qui est le « père » des algorithmes symétriques (vous pouvez noter sa date de naissance, comme un ordre de grandeur par rapport aux technologies numériques).
Les transparents 33 à 37 présentent l'architecture du DES qui travaille sur des blocs de 64 bits. Ce qu'il est intéressant de noter, c'est le fait que pour crypter un bloc, on effectue 16 fois (16 rondes/16 rounds) (transparent 33) une série d'opérations toutes simples de brouillage numérique, comme celles que l'on a vues ensemble. On voit apparaître des XOR à différents endroits.
Quelques remarques :

- Pour chaque ronde, une sous-clé Ki est calculée à partir de la clé (cf étape 3 tr. 34). Il s'agit ici d'utiliser pour chaque ronde une sous-clé différente à partir de la clé principale, de façon à ne pas utiliser directement la clé principale pour la « protéger » (moins la clé principale est utilisée directement dans les calculs, moins elle risquera d'être « volée »). Pour déterminer ces sous-clés, on peut utiliser une fonction irréversible du type des « divisions modulo », souvenez-vous de l'exercice sur l'inversion binaire selon une suite aléatoire, ou nous avions calculé une sous-clé b_n à partir de la clé a_n. C'est le même esprit ;
- La quatrième étape (tr. 33 et 34) consiste à utiliser une substitution par table, cela ressemble dans une certaine mesure à la table du code poly-alphabétique (tr. 20). *A ce stade une remarque très importante : il existe des versions d'algorithme de cryptage qui peuvent être volontairement « vulnérables ». En effet, les algorithmes de cryptage sont disponibles « gratuitement » parfois sur certains sites. Il se peut qu'une telle version inclue par exemple*

une porte dérobée (back door) permettant à un pirate d'avoir accès à votre cryptage. Un élément important en sécurité est la ***confiance (trust)*** *que vous devez avoir dans votre fournisseur avant d'utiliser un programme dédié à la sécurité, surtout pour une utilisation dans des applications critiques (nucléaire, drones, supervision de la santé…).*

- Le transparent 39 nous rappelle que l'on est dans le cas du cryptage symétrique, ce qui veut dire que la même clé est utilisée par l'émetteur et par le récepteur. De la même manière le même algorithme sera utilisé pour le cryptage et le décryptage, la configuration d'un paramètre est suffisante pour préciser si l'on crypte ou décrypte.
- Le transparent 40 précise quelques considérations liées à la sécurité du DES
- Aujourd'hui nous utilisons plutôt des extensions ou nouvelles versions (3DES ou AES) : voir tr. 42 à 44, notamment le tr. 43 sur les considérations concernant la sécurité du DES et ses descendants.

On peut conclure de ce DES que même s'il est ancien, il est efficace est robuste, qu'il a été testé et utilisé, il existe un fort retour d'expérience sur cet algorithme. Il n'est actuellement plus utilisé essentiellement à cause de la taille de la clé qui est aujourd'hui trop faible. 3DES et AES et les autres algorithmes de cryptage symétrique sont basés sur les mêmes approches, avec des clés plus grandes.

Finalement les algorithmes de cryptage symétriques sont basés sur des techniques de calcul très simples, de type fonctions logiques basées sur des OU exclusifs (tr.33), ces fonctions logiques étant elles-mêmes organisées au sein d'un système « logique séquentielle », les 16 rondes par exemple du DES. La sécurité est basée sur cette combinaison d'étapes de brouillage successives organisées séquentiellement. Ceci évite de pouvoir remonter en arrière (on ne peut pas exécuter l'algorithme à l'envers car les OU exclusifs ne sont pas réversibles). Une manière de « casser », outre l'attaque par « force brute » (tel que nous l'avions envisagée dans l'exercice 3 page 7 du poly de cours), pourrait être d'« observer » le processeur au moment du cryptage. Le processeur peut être observé « numériquement » dans le cas par exemple d'une machine corrompue, ou « physiquement » (il existe des techniques d'observation du fonctionnement de processeurs par analyse des ondes électromagnétiques émises par les processeurs et autres composants…).

Globalement donc c'est un système sûr si il est utilisé correctement, rapide, mais avec cette faiblesse de la transmission de la clé commune/clé partagée/clé de session (ces expressions sont synonymes et expriment le fait que la clé doit être au moins à deux endroits : la machine de l'émetteur et celle du récepteur, mais pas plus…).

A ce stade, on pourra préparer la première partie de l'exercice 4 proposé pour le devoir. Cette étude permettra de se poser des questions sur la pertinence des systèmes symétriques, notamment en termes du nombre de clés partagées à gérer en fonction du nombre d'utilisateurs …

Première partie de l'exercice 5 du devoir

Un groupe de n personnes souhaite utiliser un système cryptographique pour s'échanger deux à deux des informations confidentielles. Les informations échangées entre deux membres du groupe ne devront pas pouvoir être lues par un autre membre. Le groupe décide d'utiliser un système de chiffrement **symétrique**.

a. Quel est le nombre minimal de clés symétriques nécessaires ?
b. Donner le nom d'un algorithme de chiffrement symétrique reconnu.

3. Crypto-systèmes à clés asymétriques.

Le cryptage asymétrique a été inventé à la même époque que le cryptage symétrique, à une époque de la guerre froide où les Etats-Unis ont beaucoup investi dans les développements informatiques.

Le cryptage asymétrique se caractérise (tr. 48 et 49) par le fait que les clés sont différentes du côté de l'émetteur et du récepteur. Nous sommes en réalité en présence d'une **paire de clés (clé privée, clé publique**) dont l'une est rendue publique tandis que l'autre est gardée secrètement par son propriétaire, ces clés peuvent être appelées **clés associées**. Les deux clés étant différentes, c'est pour

cela que l'on parle de cryptage **asymétrique**. Les deux clés sont liées entre elles, dans une relation **irréversible** qui fait que :

- A partir de la clé privée, il est possible de générer des clés publiques liées à cette clé privée. Des opérations de type « division modulo » (ou « reste de la division entière ») sont utilisées pour cela.
- A partir de la clé publique, il n'est pas possible de retrouver la clé privée car les opérations modulo ne sont pas réversibles.

L'algorithme principal de cryptage asymétrique s'appelle le RSA (du nom de ses inventeurs, tr. 56). L'algorithme du RSA pour la confidentialité est présenté tr.57. En complément du transparent on peut noter :

- L'opération effectuée est une exponentiation modulo, c'est-à-dire une exponentiation suivie du calcul du reste d'une division entière. Sur le plan calculatoire, on peut noter qu'ici (contrairement au cryptage symétrique qui était une séquence de fonctions logiques de type XOR) des capacités de calcul (type de processeur) beaucoup plus performantes (voir tr.68) sont nécessaires.
- Dans la phase de calcul on utilise un nombre n qui est le produit de deux « grands » nombres premiers (vous pourrez trouver tr. 54 des considérations sur les nombres premiers. Vous pouvez trouver des informations sur des sites internet si vous êtes intéressés).

Pour prendre conscience des besoins en capacité de calcul vous pouvez tester votre calculatrice et comparer à la calculatrice scientifique de Windows, on peut tester le tr. 63.

726^13 mod 21209 = 11600

On peut noter que 726 représente un bloc à **crypter**, 13 représente la clé utilisée pour le cryptage (cette clé est très « petite ») et 21209 est une valeur qui est obtenue à partie de l'algorithme RSA (vous pouvez approfondir à partir des tr. 61 et 62 mais ce n'est pas nécessaire ici) à partir des nombres premiers.

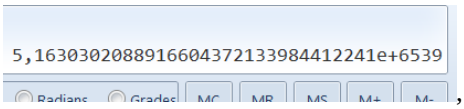
On va maintenant effectuer le calcul

1ère phase de calcul : 726^13 = 1,5565e+37 : tout va bien, mais le nombre est déjà en 10^{37} .

2ème phase de calcul : la fonction « modulo » existe sur la calculatrice scientifique de Windows, on obtient aisément la valeur résultat de 11600.

Pour le **décryptage** on utilisera la clé **associée**, la valeur de la clé associée est 1609

11600^1609 mod 21209 = 726

1ère phase de calcul : 11600^1609 = 

, on voit ici que le nombre commence à être très grand (encore calculable sur la calculatrice scientifique de Word mais généralement pas sur vos propres calculatrices...).

2ème phase de calcul : la fonction « modulo » existe sur la calculatrice scientifique de Windows, on retrouve l'original : 726.

Pourquoi avoir fait cette expérimentation ? Pour montrer que le calcul, pour le cryptage asymétrique nécessite des ressources de calcul et que même avec des petites clés (13, 1609) sur un petit bloc de 3 chiffres, les ressources nécessaires sont importantes, vos calculatrices plantent en général !

Cet aspect du temps de calcul sera un des points faibles du cryptage asymétique, et limitera son utilisabilité par exemple pour des applications de systèmes embarqués et réseaux de capteurs limitées en capacités de calcul.

Une propriété intéressante des systèmes asymétriques est que la clé privée, unique, jamais transmise, permet non seulement d'être utilisé pour assurer la **confidentialité**, mais également pour effectuer une **signature numérique** (tr. 49, tr. 58). La signature numérique permet de protéger **l'authenticité et l'intégrité**.

Il faut reprendre ici un concept essentiel, si on veut comprendre la suite. Lorsqu'on parle de cryptage numérique, deux applications très différentes sont envisagées :

- La première est la **confidentialité**. On protège ici contre la **lecture** par des personnes non autorisées. Dans ce cas l'émetteur n'est pas très important, car tout le monde peut envoyer des fichiers cryptés à qui il veut, mais le **récepteur** est important car seul le récepteur-cible du message crypté et personne d'autres ne doit être en mesure de lire le contenu du message. Dans ce cas de figure, la **clé publique** est sur la machine émettrice et la **clé privée** sur la machine réceptrice. C'est le schéma du tr. 48. La clé privée, unique, est celle du récepteur, elle est utilisée pour décrypter, et la clé publique est utilisée par l'émetteur, cette clé publique est associée de la clé privée du récepteur. Cela veut dire que si vous souhaitez envoyer un message confidentiel crypté à quelqu'un, vous devez d'abord lui demander sa clé publique (la clé publique associée à sa clé privée). Si vous avez l'habitude d'envoyer « paire à paire » (ce type de système s'utilise par paire, pas par groupe) des messages confidentiels à plusieurs personnes, vous devrez les crypter en utilisant pour chaque personne la clé publique associée (ou complémentaire) à sa clé privée. De la même façon si ces personnes souhaitent vous envoyer des messages confidentiels, elles devront utiliser votre clé publique (la clé publique associée à votre clé privée) pour crypter, et vous pourrez décrypter avec votre clé privée, unique, garantissant que personne d'autre n'a pu décrypter.
- La seconde est l'**intégrité**. Ici on considère **que le fichier n'est pas confidentiel**. On protège ici contre la modification autrement dit contre l'**écriture** par des personnes non autorisées. Dans ce cas le récepteur n'est pas très important, car tout le monde peut vérifier l'intégrité d'un fichier (garantie qu'il est conforme à l'original). L'**émetteur** est important car seul l'émetteur doit garantir l'intégrité du fichier, cette garantie s'effectue par une « signature numérique », cela veut dire qu'ici le cryptage est effectué par l'émetteur en utilisant sa clé privée (tr. 49) ; les récepteurs décrypteront (ce qui reviendra ici à « vérifier » la signature) en utilisant la clé publique associée à la clé privée de l'émetteur.

En résumé le cryptage peut être utilisé pour protéger la confidentialité auquel cas l'émetteur utilise la clé publique du récepteur pour crypter et le récepteur sa propre clé privée pour décrypter.
Le cryptage peut être utilisé pour protéger l'authenticité et l'intégrité (ce que l'on appelle signature numérique) auquel cas l'émetteur utilise sa propre clé privée pour crypter et le récepteur la clé publique de l'émetteur pour décrypter. J'espère que vous n'êtes pas trop perdus…
Si un émetteur souhaite faire les deux, c'est-à-dire protéger la confidentialité d'une part, et l'authenticité/intégrité d'autre part, il faudra crypter deux fois, une première fois pour la confidentialité et une seconde fois pour l'intégrité/authenticité, en respectant les règles ci-dessus.

Vous pouvez à ce stade résoudre la seconde partie ci-dessous de l'exercice 4 du devoir.

Suite de l'exercice 5 du Devoir

Le groupe décide ensuite de remplacer ce système par un système **asymétrique**.

c. Quel est le nombre minimal de couples de clés asymétriques nécessaires pour que chaque membre puisse envoyer et recevoir des informations chiffrées et/ou signées ? Si l'on considère que chacun peut communiquer avec tout le monde, combien de clés privées et publiques chaque utilisateur devra-t-il détenir ?
d. Bob souhaite envoyer des informations chiffrées et signées à Alice (Bob et Alice appartiennent tous les deux au groupe). Quelle(s) clé(s) Bob doit-il utiliser ?
e. Donner le nom d'un algorithme de chiffrement asymétrique reconnu.

Pour terminer la partie sur le cryptage asymétrique, vous pouvez étudier les transparents 65 et 66.

4. Comparaison des systèmes de cryptage, cryptographie hybride

Le transparent 68 conclut sur les algorithmes de cryptage symétriques et asymétriques, montrant que finalement, la bonne solution est de combiner les deux approches de cryptage, pour bénéficier des avantages combinés des deux approches. Lorsqu'on combine les deux approches, on parle de **cryptographie hybride**. Il y a deux grandes manières d'envisager la cryptographie hybride, les

deux approches faisant l'hypothèse que les transactions sont sécurisées (par exemple grâce à des échanges de certificats).

- La première technique consiste en ce qu'un utilisateur génère une clé « symétrique » (ou clé de session, ou clé de partage), ensuite cet utilisateur qui devient l'émetteur crypte cette clé symétrique en utilisant la clé publique du récepteur (on suppose ici que l'émetteur est détenteur de la clé publique du récepteur) afin de lui envoyer la clé symétrique cryptée (bien sûr pour qu'un pirate potentiel ne puisse pas récupérer la clé symétrique en clair…). Lorsque le récepteur reçoit la clé symétrique qui a été cryptée par l'émetteur en utilisant la clé publique du récepteur (vous me suivez toujours ?), alors le récepteur peut décrypter la clé symétrique reçue en utilisant sa clé privée (associée à la clé publique qui a été utilisée par l'émetteur, et qui est la clé publique du récepteur). Cette fois-ci l'émetteur et le récepteur partagent la même clé symétrique, qui a été échangée via un échange sécurisé (crypté). En quoi cette approche est-elle hybride ? elle est hybride car elle combine les deux techniques, une approche asymétrique (cryptage par l'émetteur avec la clé publique du récepteur, décryptage par le récepteur avec sa clé privée) pour échanger en sécurité la clé symétrique, et ensuite une fois que la clé symétrique est partagée, les deux utilisateurs peuvent utiliser le système symétrique. L'intérêt est d'utiliser majoritairement le système symétrique (moins consommateur en temps de calcul) tout en garantissant l'échange de la clé symétrique au moment de l'initialisation de la transaction. Il est conseillé de renouveler de temps en temps la clé symétrique.
- La seconde technique est l'algorithme de Diffie-Helman décrit dans les transparents 69 à 71. Dans cette technique, les deux utilisateurs, chacun détenteur de sa clé privée (X_1 et X_2, bien protégées et jamais échangées), vont échanger leurs clés publiques Y, puis à partir de là, en combinant leur propre clé privée et la clé publique de l'autre utilisateur, vont générer une clé de session (ou clé partagée) « c » qui sera la même sur les deux machines (émetteur et récepteur) sans jamais avoir été échangée et donc n'avoir jamais été potentiellement captée par une tierce personne (pirate). Le transparent 71 propose une petite application qui vous permettra de vous familiariser avec la technique. Vous pouvez noter que dans ce petit exercice on utilise des nombres premiers très petits (et donc facilement « craquables »et que déjà les calculs, si vous les faites à la main, ne sont pas difficiles mais nécessitent un peu de ressources…).

Conclusion sur la cryptographie

Vous trouverez dans les transparents suivants :

- Tr. 72 et 73 : des considérations sur les temps de calcul et le cassage de clé
- A partir du tr. 75 : des considérations sur le hash (utilisé notamment pour stocker les mots de passe et pour la signature numérique), les certificats (qui fonctionnent finalement dans une philosophie proche de l'algorithme de Diffie-Helman vu auparavant), les PKI (Public Key Infrastructure) qui sont des infrastructures permettant de gérer et d'échanger les clés publiques de plusieurs utilisateurs.
- Le chapitre 3 propose un cours sur les protocoles de sécurité. Je ne les passe en général pas tous en cours, je mets l'accent généralement sur IPSec qui est très utilisé, il peut fonctionner en mode **transport** (géré de machine à machine par deux utilisateurs) ou en mode **tunnel** (qui permet d'interconnecter de manière sécurisée (**crypté** et **signé**) par exemple deux parties de réseau distantes géographiquement). Vous pouvez regarder en particulier les transparents 19 à 23 qui explique les fonctionnements en mode transport et tunnel. Le transparent 28 montre l'organisation en couche du protocole IPSec inclus dans les couches IP, entre les couches 3 et 4 du modèle OSI. Ce qui est intéressant de noter c'est le fait que le paquet IP construit et montré peut véhiculer, grâce à IPSec, des contenus de données cryptés pour protéger la confidentialité et/ou signés pour protéger l'authenticité/intégrité.

CONCLUSION GENERALE

En tant que spécialiste du Génie Electrique, que faut-il retenir de ces aspects de cryptographie ? Il existe deux types de cryptographie :

- la cryptographie **symétrique** qui est basée sur des formes mathématiques simples (globalement système logique séquentiel à plusieurs étapes constitués d'opérations à base de OU exclusifs), facilement implantable, et nécessitant des ressources calculatoires « légères » permettant le calcul en temps réel au moment de la transmission (ou du stockage, le cryptage pouvant être utilisé aussi pour le stockage de données ou de fichiers sécurisé).
- La cryptographie **asymétrique** qui est basée sur des formes mathématiques plus lourdes (exponentielles modulo), qui est de fait rarement utilisée seule mais est utilisée pour initier en sécurité (« *securely* ») un échange ou une génération de clé partagée pour ensuite utiliser un système de cryptage symétrique.

La cryptographie peut servir à protéger soit la **confidentialité** soit **l'authenticité & intégrité**. Les jeux de clés à utiliser ne sont pas les mêmes.

En termes de sécurité plus globalement, ce qu'il faut retenir sont les aspects suivants :

- La sécurité se gère finalement un peu comme de la qualité, elle doit être organisée, sous la forme d'une « **politique de sécurité** » (vous pouvez parcourir mes transparents chapitre 7), elle doit être documentée, et les personnes en charge de la sécurité doivent pouvoir « **prouver qu'elles sont *à l'état de l'art,* qu'elles ont déployé les bonnes procédures de sécurité** », pour cela la **documentation** sera importante.
- Les choix de politique de sécurité que l'on met en œuvre sont une réponse à une **analyse de risques**. Il ne s'agit pas de déployer des mécanismes de sécurité au hasard, il faut réfléchir et analyser ce que l'on veut protéger et pourquoi, mettre un coût sur les données à protéger, puis proposer des stratégies de solution pour répondre à ce besoin.
- Un concept important également en sécurité est que « ***tout ce qui n'est pas explicitement autorisé est interdit*** ». Cela veut dire que les usagers doivent spécifier explicitement leurs besoins (par exemple transfert de fichiers de telle heure à telle heures sur tel site, accès limité à un certain nombre de sites internet. Ce concept permet d'éviter que des ports soient ouverts ou des services activés, sans que l'on en soit conscient.
- La **dimension humaine** est importante : les utilisateurs d'un système d'information (employés, fournisseurs et clients) doivent être sensibilisés à la sécurité, bien comprendre qu'il ne s'agit pas d'un jeu, et les acteurs humains doivent être des alliés de la sécurité. Pour cela, ils doivent avoir des formations régulières, le niveau de ces formations et leur fréquence doit être une réponse à l'analyse de risques, et les responsables de la sécurité doivent être à l'écoute des besoins des utilisateurs pour les prendre en compte dans la démarche de sécurité, et éviter que les utilisateurs aient tendance à contourner la sécurité.
- Un dernier concept est la **sécurité en profondeur** (*In-depth security*). Ce concept précise que les différents aspects de la politique de sécurité, déploiement technologique, aspects de ressources humaines, aspects contractuels avec les clients et les fournisseurs, procédures, prise en compte de la législation en vigueur et les normes (par exemple normes nucléaires, normes sur la production et la distribution électrique, normes aéronautiques…) doivent être pensées globalement, de façon, là aussi, à apporter la meilleure réponse à l'analyse de risques.

Impact potentiel d'attaques informatiques sur le dysfonctionnement de process physiques pouvant entraîner des conséquences sur la sécurité physique des personnes, des biens et de l'environnement.