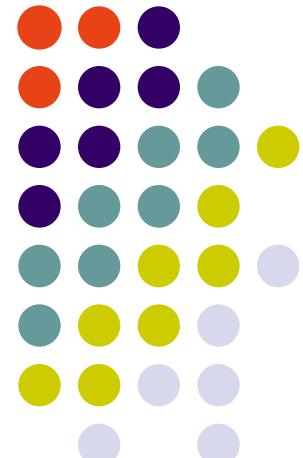
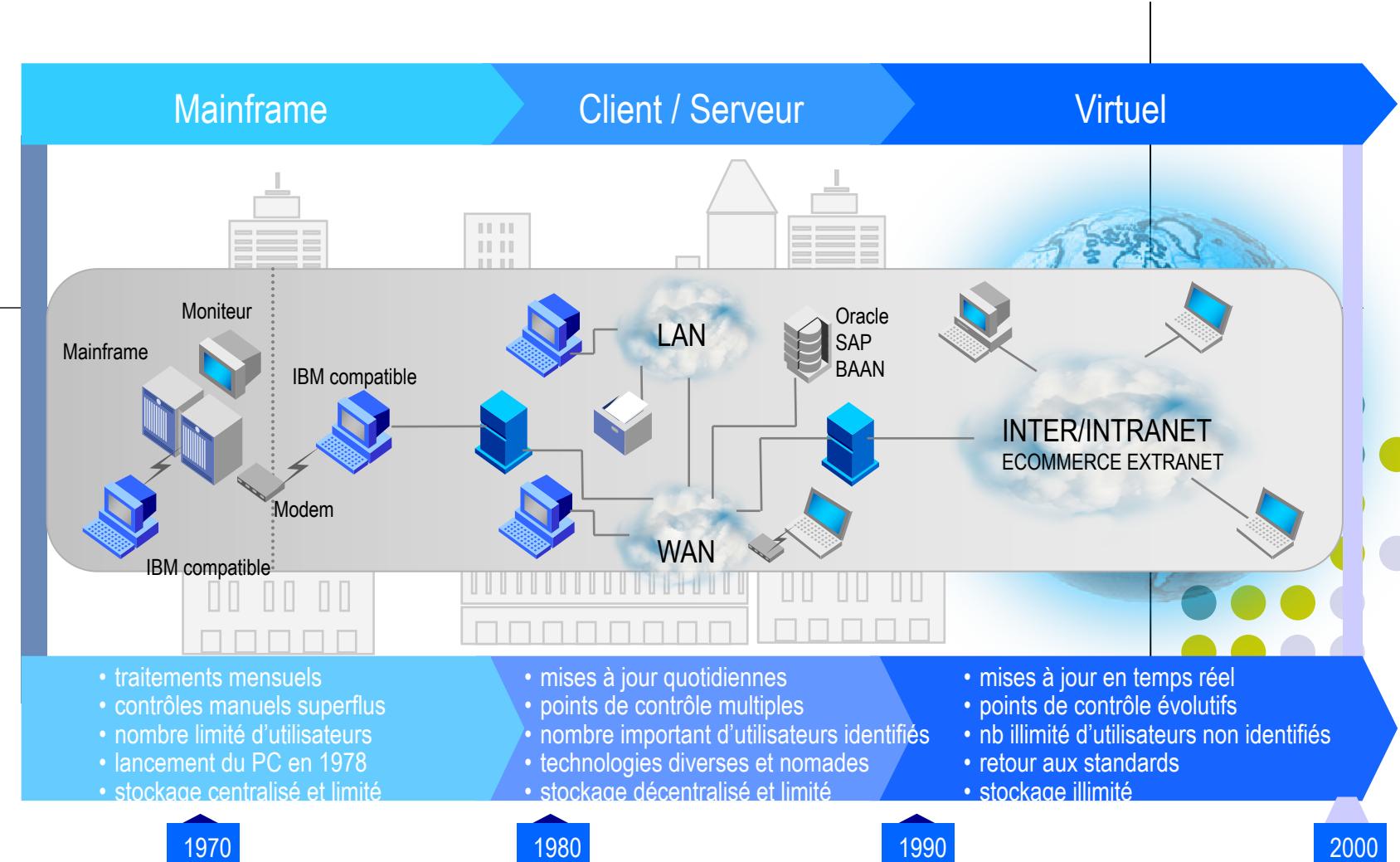


Sécurité Informatique

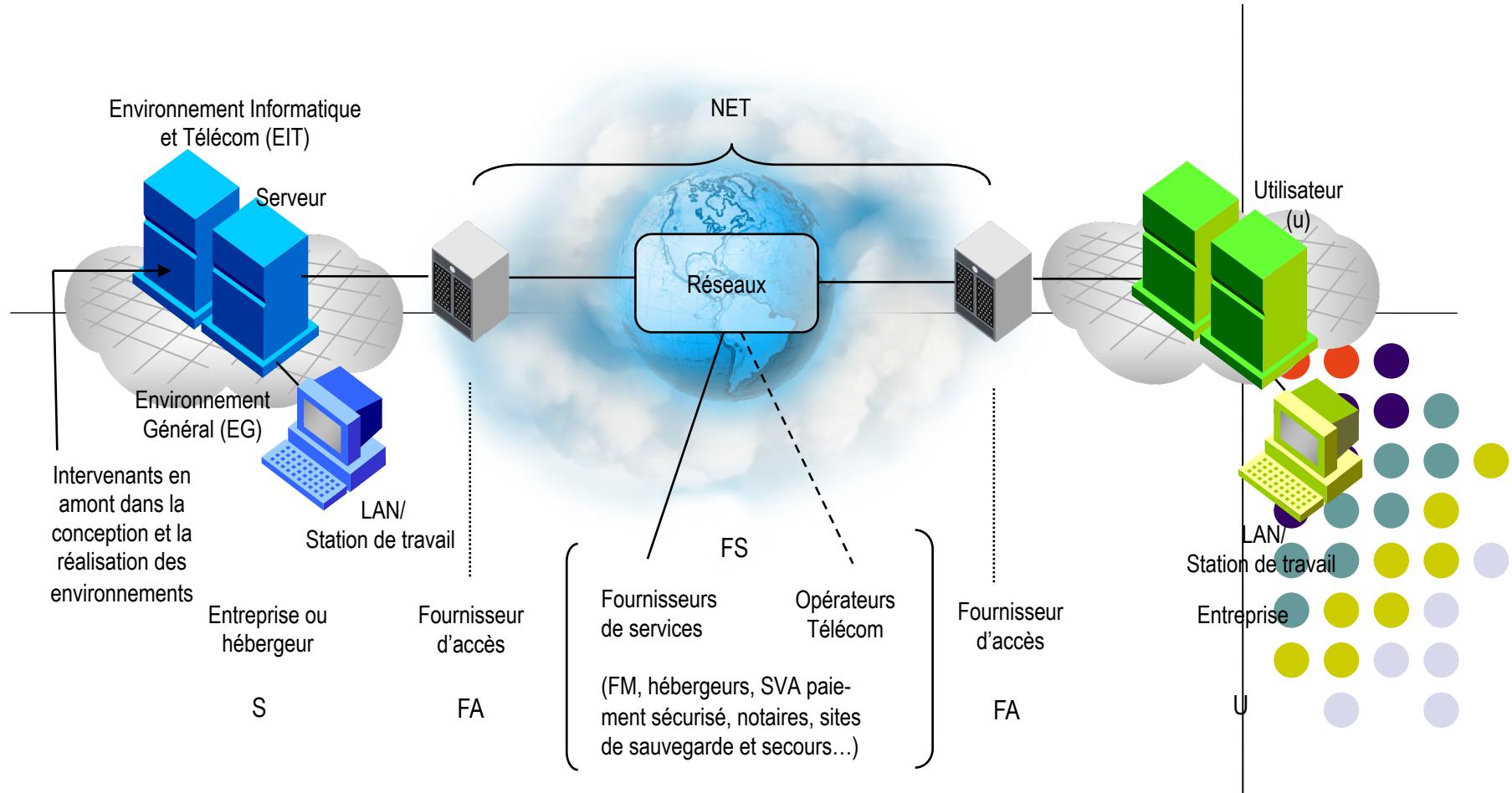
Applications aux Systèmes d'Information



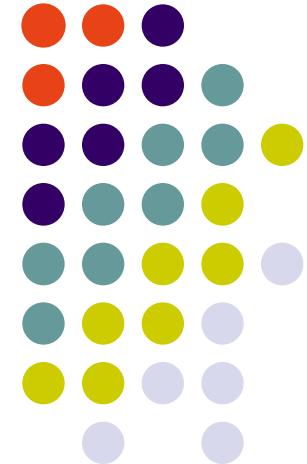
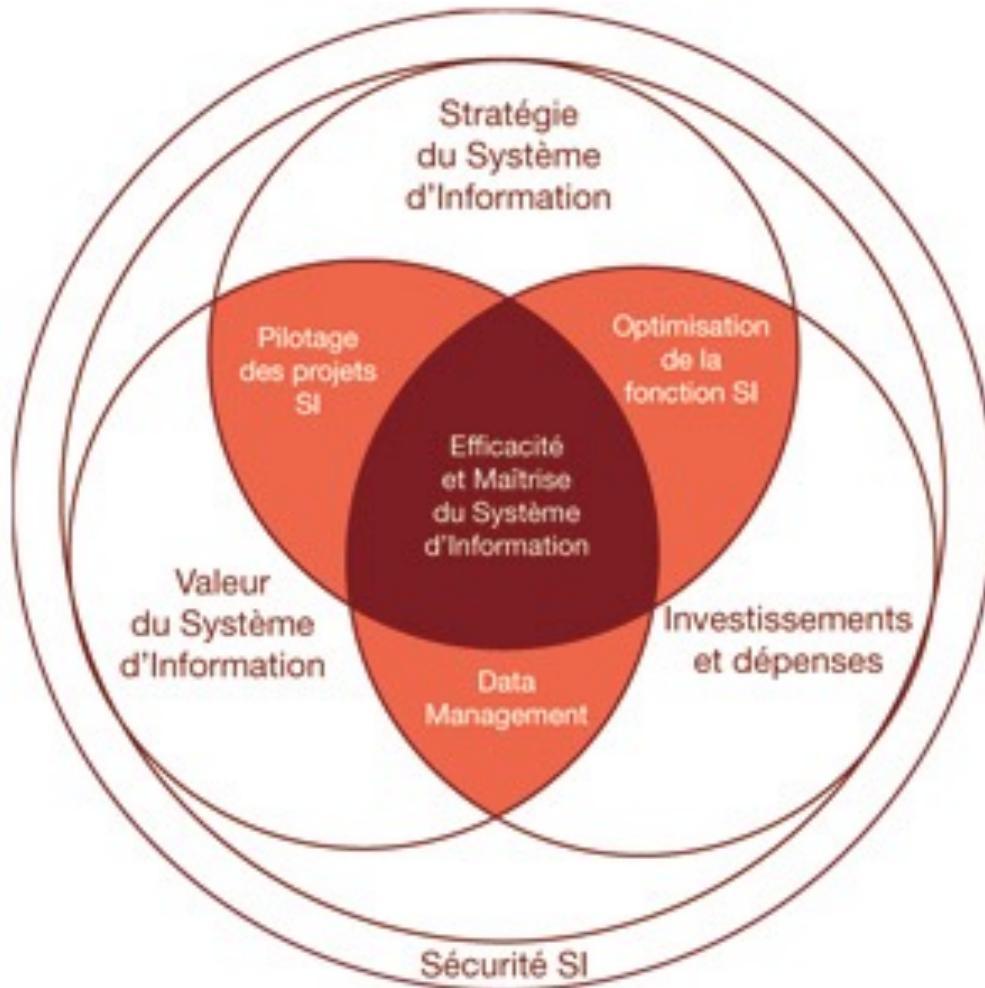
1 - Evolution des architectures SI



2 - Globalisation des SI



Facteurs clés de réussite d'un Système d'information



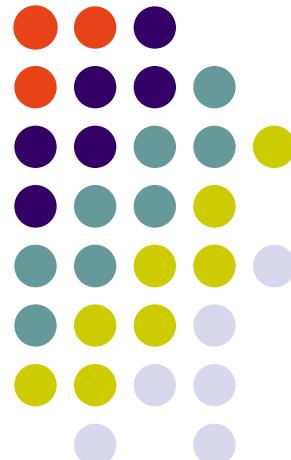
Système d'information

L'information se présente sous trois formes : les données, les connaissances et les messages.

On a l'habitude de désigner par «système d'information» l'ensemble des moyens techniques et humains qui permet de stocker, de traiter ou de transmettre l'information.

On confond souvent, même si ce n'est pas très exact, la notion de «systèmes et réseaux informatiques» et celle de «systèmes d'information (SI) ».

On dira donc qu'un système d'information est « *tout moyen dont le fonctionnement fait appel d'une façon ou d'une autre à l'électricité et qui est destiné à élaborer, traiter, stocker, acheminer, présenter ou détruire l'information* »



Sécurité - définitions

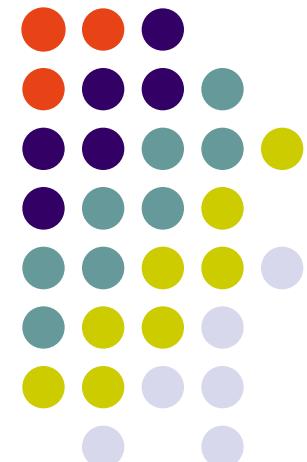
Sécurité

Situation dans laquelle quelqu'un, quelque chose n'est exposé à aucun danger, à aucun risque d'agression physique, d'accident, de vol, de détérioration

Situation de quelqu'un qui se sent à l'abri du danger, qui est rassuré

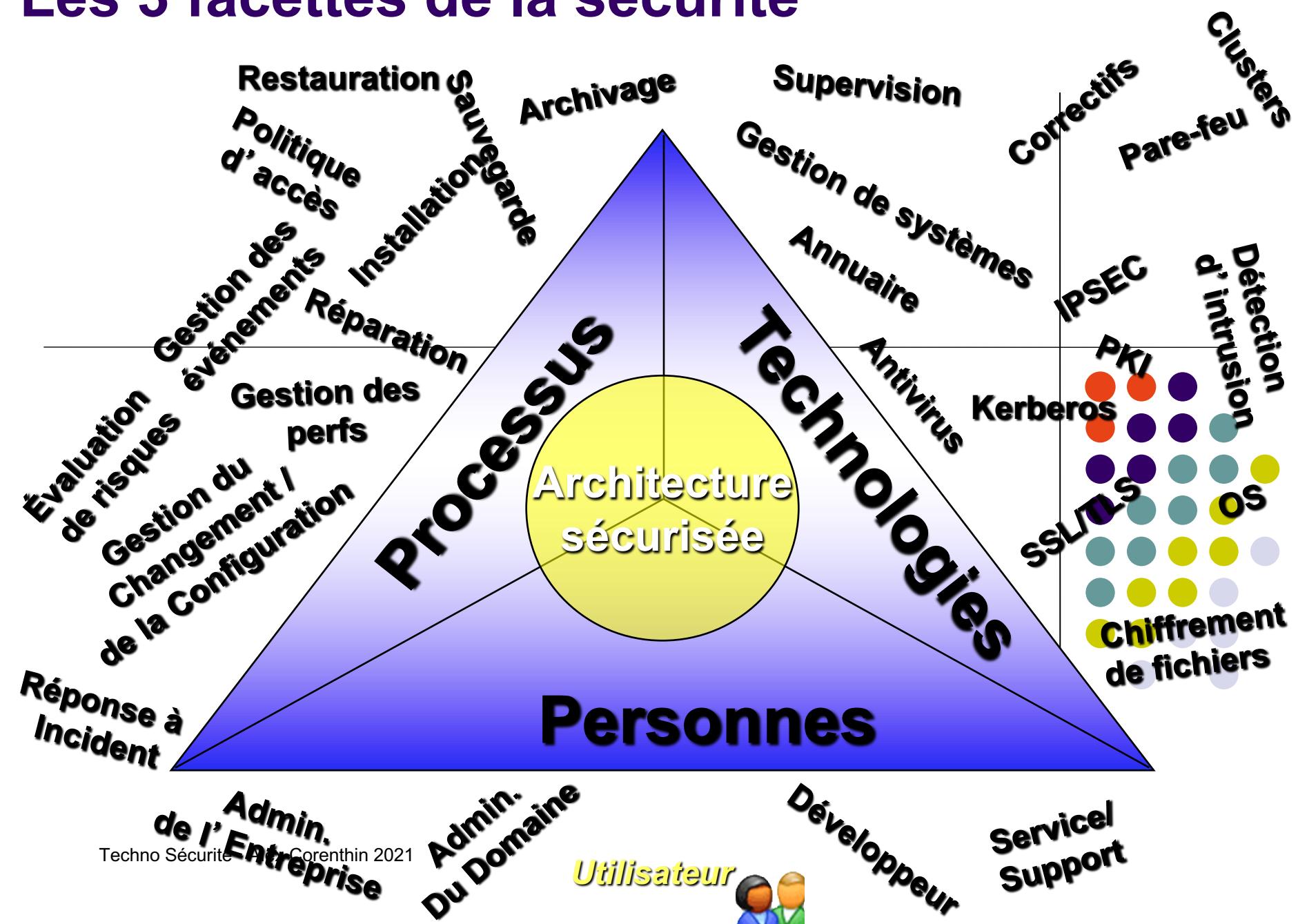
Sécurisé

Qu'est-ce qui a été sécurisé ? Par rapport à quoi ? A qui ? Contre quoi ? Pour combien de temps ? Jusqu'à quel niveau d'attaque ?



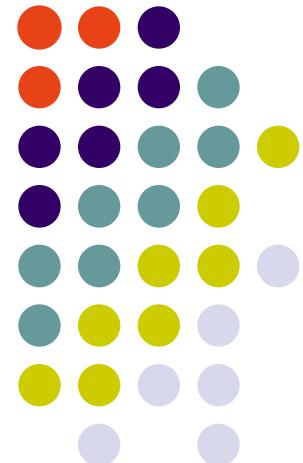
→ La sécurité est une notion relative à un contexte

Les 3 facettes de la sécurité



Définition de la SECURITE DES SI

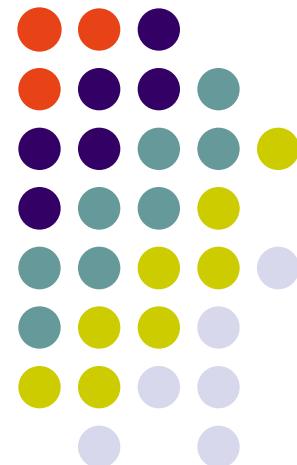
Mise en place d'un ensemble de mesures de sécurité afin d'assurer la protection des biens informatiques matériels (niveau physique) et applicatifs d'une organisation ainsi que les données (niveau logique) de son système d'information



Qu'est ce que la sécurité ?

La sécurité recouvre donc l'ensemble de techniques informatiques permettant de **réduire au maximum** les chances de fuites d'information, de modification de données ou de détérioration des services.

Elle consiste à un très grand nombre de **méthodes, de technologies, d'architectures** permettant d'atteindre un certain niveau de protection.



Qu'est ce que la sécurité (2) ?

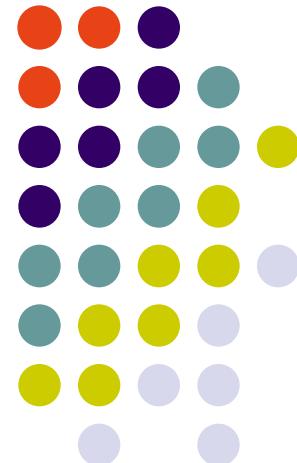
"Sécuriser" consiste à utiliser une ou plusieurs de ces techniques dans le but d'élever le niveau de sécurité d'un système ou d'une architecture.

$$\text{Risque} = \frac{\text{Menace} \times \text{Vulnérabilité}}{\text{Contre-mesure}}$$

La **menace** représente le type d'action susceptible de nuire dans l'absolu

La **vulnérabilité** représente le niveau d'exposition face à la menace dans un contexte particulier.

Enfin la **contre-mesure** est l'ensemble des actions mises en oeuvre en prévention de la menace.

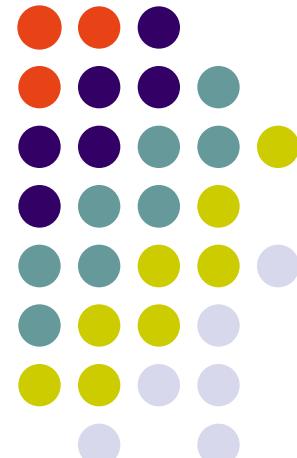


Limites de la définition

La sécurité est un concept relatif. On est plus ou moins en sécurité qu'avant, plus ou moins qu'ailleurs, mais on ne peut jamais être sûrs d'être parfaitement en sécurité

La sécurité absolue n'existe pas !!!

- Parce qu'elle dépend de la conjoncture extérieure (l'apparition de nouveaux risques);
- Parce que les ressources à y consacrer sont limitées.



Exigences fondamentales et objectifs

Origine des attaques

50% interne

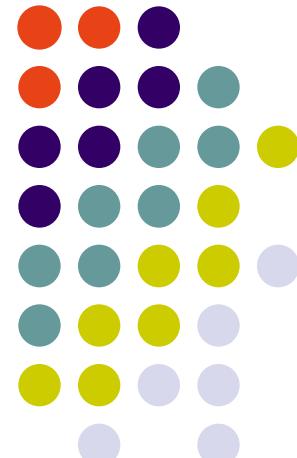
50% externe

Exemple :

- utilisateur malveillant,
erreur involontaire,...

Exemple:

- Piratage, virus,
intrusion....



Objectifs de ces attaques

Récupération d'informations confidentielles

→ *Atteinte à la confidentialité*

Modification du contenu

- Données internes (BDD financière, client...)
- Sites Web publiques (façade commerciale)

→ *Atteinte à l'intégrité*

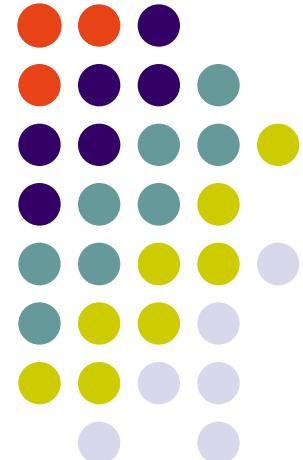
Rendre le service indisponible (Déni de Service – DoS)

- Temporaire
- Permanent

→ *Atteinte à la disponibilité*

Rebond

- Point d'entrée privilégié
- Atteindre des machines internes (Relais pour attaques externes)



Conséquences

Récupération d'informations confidentielles

- Secrets industriels, annonces commerciales, résultats, données clientes, webmails...

Modification du contenu

- Résultats incorrects/incohérents, Comptes faussés, Défiguration, BDD corrompues
- Atteinte à l'image de la société (référencement de défiguration (zataz, zone-h...))

Indisponibilité des services

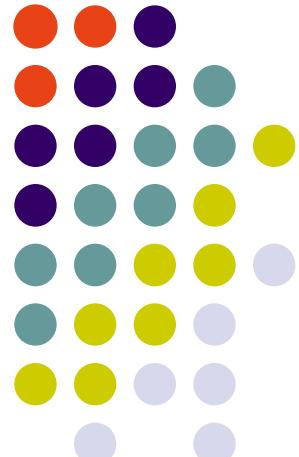
- Production : Perte financière
- Annuaire serveur de fichier... : Force de travail bloquée

Rebond

- Attaque de serveurs internes
- Attaques externes : imputabilité, responsabilité juridique

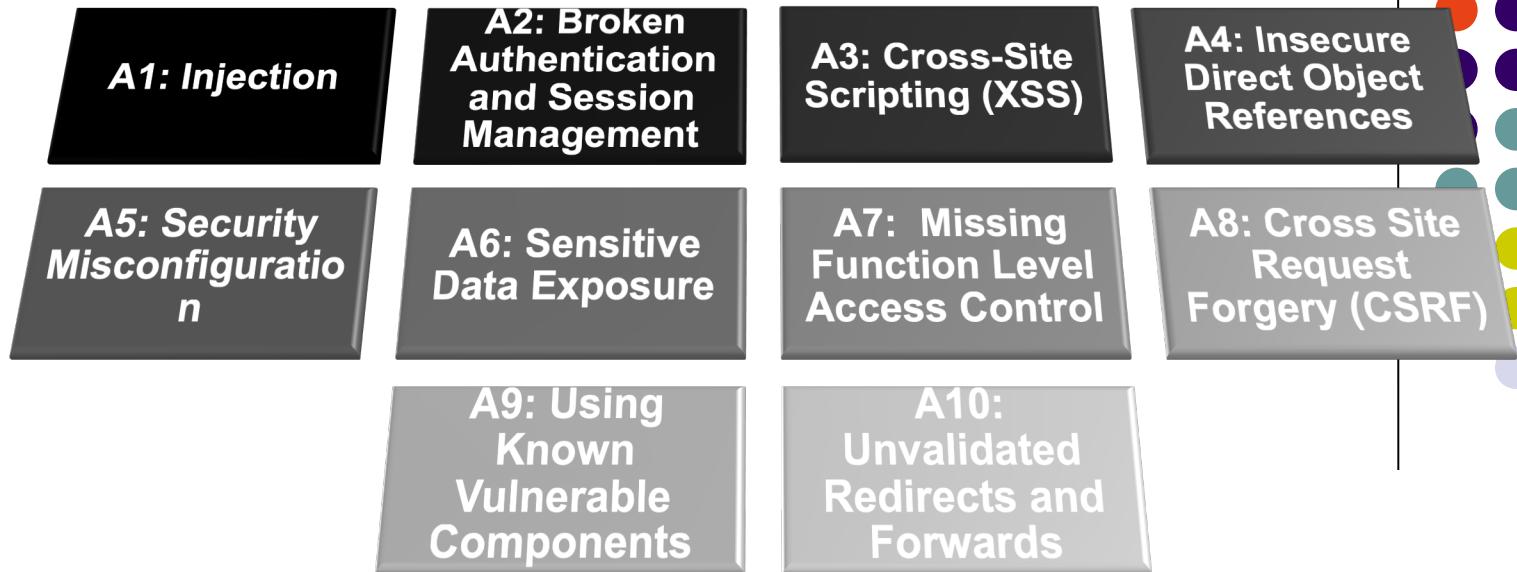


Les risques sont énormes ...



Vulnérabilités applicatives majeures

- Présentation de vulnérabilités applicatives majeures
- Repose sur une classification évolutive de l'OWASP
 - Projet « Top Ten »
 - http://www.owasp.org/index.php/OWASP_Top_Ten_Project
- 10 types de vulnérabilités les plus critiques (2017 RC)

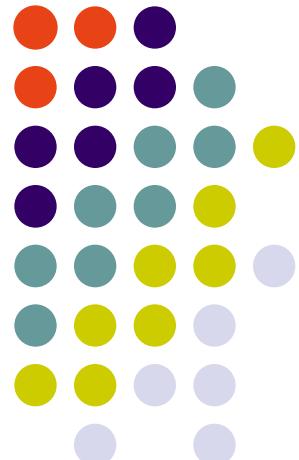


Contre qui ? - Critères

Comment caractériser les agresseurs ?

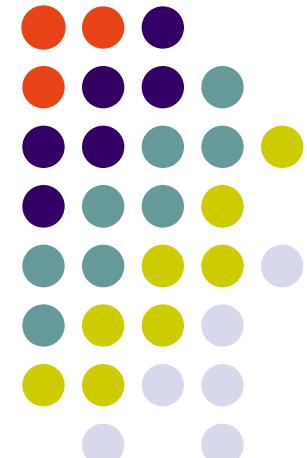
Par :

- ✓ **Leurs compétences techniques**
- ✓ **Le temps qu'ils sont prêts à passer pour réussir**
- ✓ **Leurs motivations**
- ✓ **Leurs moyens**
- ✓ **Leurs connaissances préalables de la cible**

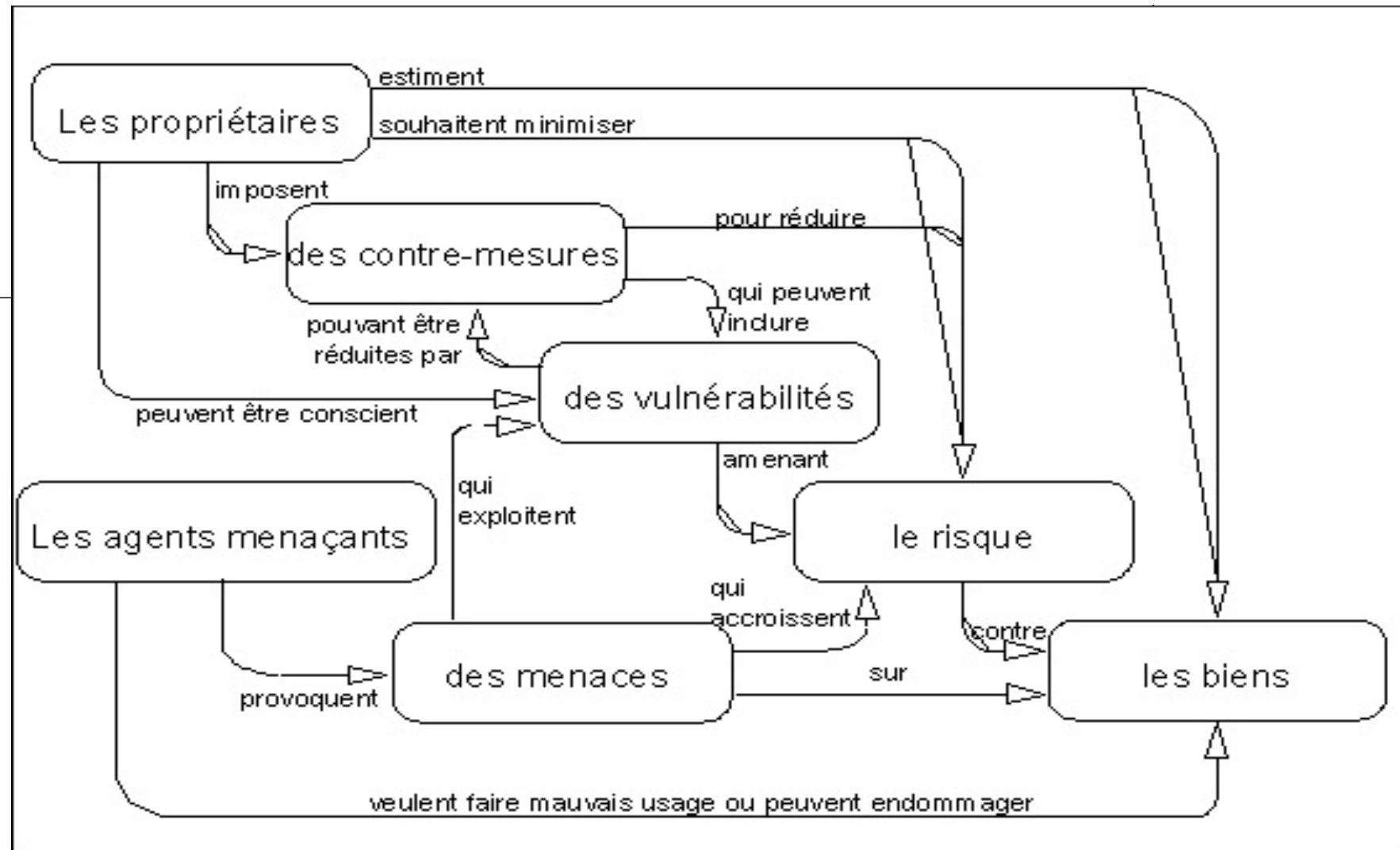


Classement des agresseurs

	Compétence	Temps	Motivation
Un hacker / étudiant externe pour le plaisir	Forte	Fort	Moyenne
Un concurrent	Forte	Faible	Forte
Un escroc (enjeu financier)	Moyenne	Moyen	Moyenne
Un opportuniste	Faible	Faible	Faible
Un membre de société de service	Forte	Faible	Faible
Un ancien membre du personnel	Moyenne	Faible	Moyenne
Un membre du personnel	Moyenne	Faible	Faible
Un stagiaire	Forte	Moyen	Faible



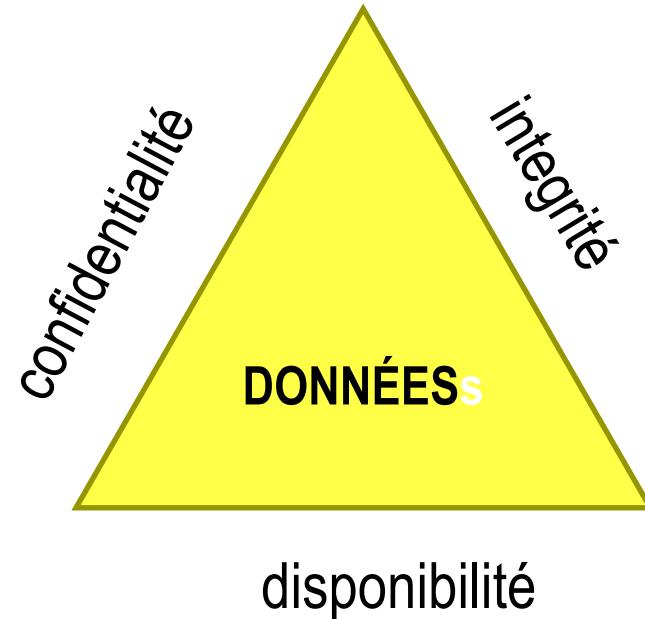
SECURITE : Concepts et relations



Les trois éléments de la sécurité des données

- Le triangle CID ou CIA (en anglais)

- Confidentialité
- Intégrité
- Disponibilité (Availability)



Contexte du cours: Les permissions sur le données

- Les propriétés CID ont beaucoup d'implications et leur assurance est une tâche complexe
- Dans ce cours, nous allons nous concentrer sur l'assurance qui peut être obtenue avec le contrôle des **permissions** sur les données
- Que peut usager X faire avec les données?
- Exemple : modèle UNIX-Linux:
 - Lecture
 - Écriture
 - Exécution

Permissions: autres possibilités

- Effacement – peut-il être considéré un cas d'écriture?
- Concaténation – lectures suivies par écritures?
- Au lieu des données nous pouvons aussi considérer des entités physiques:
 - Permission d'accès à un local
 - Permission d'utiliser un objet (p.ex. un ordinateur)
- Ou logiques
 - bases de données ou fichiers

Concepts de sûreté

- **Défaillance** : Service délivré \neq Service spécifié
- **Erreur** : état du système susceptible d'entrainer une défaillance
- **Faute** : Cause de l'erreur

Relations erreurs/fautes/défaillances:

L'erreur est la manifestation de la faute sur le système

La défaillance est l'effet d'une erreur sur le service

Concept de sûreté : Faute

Une faute devient active lorsqu'elle produit une erreur

Une **faute active** est :

- soit une faute dormante activée par le traitement
- soit une faute externe

Une **faute interne** peut passer cycliquement de l'état
dormant à actif, ...

Concept de sûreté : Erreur

Temporaire par nature

Latente ou détectée

- Latente, tant qu'elle n'a pas été reconnue
- Détectée, soit par les mécanismes de détection et traitement d'erreurs, soit par son effet sur le service (défaillance)

1 erreur → propagation d'autres erreurs dans d'autres parties du système (effet papillon)

Concept de sûreté : Défaillance

Une défaillance survient lorsqu'une erreur traverse l'interface Système/Utilisateur et altère le service délivré par le système

Dans un système constitué d'un ensemble de composants, la conséquence de la défaillance d'un composant est :

- une faute interne pour le composant englobant,
- une faute externe pour les composants avec lesquels il interagit

Type de défaillance

- Fautes franches (*fail stop*) : arrêt pur et simple
- Omissions : perte de messages
- Temporaires : déviations temporelles / spécifications
- Byzantin : comportement aléatoire ou malveillant

Propriétés de sécurité

Avant de pouvoir effectivement sécuriser des applications, vous devez comprendre les concepts fondamentaux de la sécurité.

Les 5 piliers de la sécurité sont

- ❖ **Authentification**
- ❖ **Non répudiation**
- ❖ **Intégrité**
- ❖ **Confidentialité**
- ❖ **Auditabilité**

Propriété de sécurité: l'authentification

C'est la propriété qui assure la reconnaissance sûre de l'identité d'une entité

L'authentification protège de l'usurpation d'identité

Signature = Authentification

- ✓ Authentification: Première idée contenue dans la notion habituelle de signature
- ✓ le signataire est le seul à pouvoir réaliser le graphisme (caractérisation psychomotrice)

Entités à authentifier:

- une personne
- un programme qui s'exécute (processus)
- une machine dans un réseau

Authentification

- Concept permettant de s'assurer que l'identité de l'interlocuteur est bien celle qu'il prétend
- **Techniques traditionnelles :**
 - *Some Thing you Know* : mot de passe
 - *Some Thing you Have* : carte à puce
 - *Some Thing you Are* : empreinte digitale

Propriété de sécurité: la non répudiation

C'est la propriété qui assure que l'auteur d'un acte ne peut ensuite dénier l'avoir effectué

Signature = Authentification + Non répudiation :

- Seconde idée contenue dans la notion habituelle de signature
- le signataire s'engage à honorer sa signature
- engagement contractuel/juridique, on ne peut pas revenir en arrière

Deux aspects spécifiques de la non répudiation dans les transactions électroniques:

- ✓ a) *La preuve d'origine* : Un message (une transaction) ne peut être nié par son émetteur.
- ✓ b) *La preuve de réception* : Un récepteur ne peut ultérieurement nier avoir reçu un ordre s'il ne lui a pas plu de l'exécuter alors qu'il le devait juridiquement.

Exemple: Exécution d'ordre boursier, de commande, ...

Propriété de sécurité: l'intégrité

C'est la propriété qui assure qu'une information n'est modifiée que par des entités habilitées (selon des contraintes précises)

Exemples :

- Une modification intempestive (même très temporaire) est à interdire sur une écriture comptable validée
- Le code binaire des programmes ne doit pas pouvoir être altéré
- Les messages de l'ingénieur système doivent pouvoir être lus et non modifiés

Propriété de sécurité: la confidentialité

C'est la propriété qui assure qu'une information ne peut être lue que par des entités habilitées (selon des contraintes précises)

Exemples :

- Un mot de passe ne doit jamais pouvoir être lu par une autre personne que son possesseur
- Un dossier médical ne doit pouvoir être consulté que par les malades et le personnel médical habilité
- On ne doit pas pouvoir intercepter le contenu d'un courrier

Confidentialité des Données

- Concept permettant de s'assurer que **l'information ne peut être lue que par les personnes autorisées**
- **Solutions dans le monde réel :**
 - Utilisation d'enveloppes scellées
 - Verrouillage avec clés
 - Mesures de Sécurité physique
 - Utilisation de l'encre invisible
 - etc

Propriété de sécurité: l'auditabilité

C'est la propriété qui assure la capacité à détecter et à enregistrer de façon infalsifiable les tentatives de violation de la politique de sécurité.

Audit : Examen méthodique d'une situation relative à un produit, un processus, une organisation, réalisé en coopération avec les intéressés en vue de vérifier la conformité de cette situation aux dispositions préétablies, et l'adéquation de ces dernières à l'objectif recherché [définition ISO, d'après la norme AFNOR Z61-102]

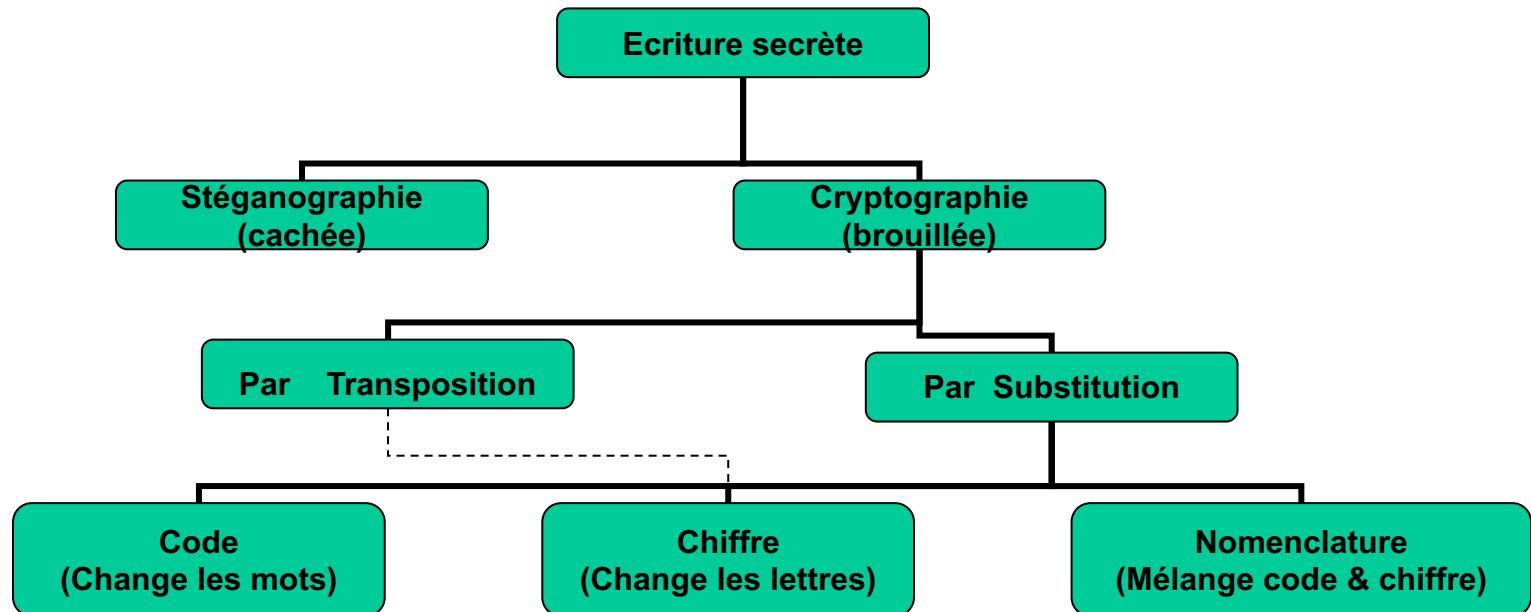
Auditabilité : Garantir une maîtrise complète et permanente sur le système et en particulier pouvoir retracer tous les événements au cours d'une certaine période.

EN CONCLUSION

Ce qu'il faut retenir :

- La Sécurité = 5 Objectifs principaux
 - Confidentialité;
 - Authenticité;
 - Intégrité;
 - Accessibilité;
 - Irrépudiabilité.

La Cryptographie



Modèle de Système de communication

Source : entité qui génère le message. Exemples :

Une personne qui parle : message = mots prononcés

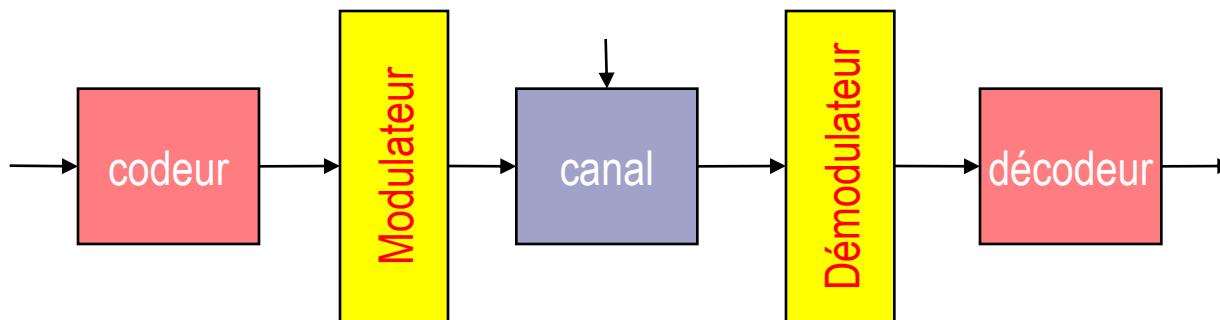
Un ordinateur : message = bits égaux à 0 ou 1

Canal : le support de la communication. Ex. : Une ligne téléphonique, une transmission satellite, ...

Codeur : il met en forme le message de la source pour l'adapter au canal. Ex. : Compression, cryptographie, code correcteur d'erreur...

Décodeur : il restitue l'information émise par la source à partir de la sortie du canal

Modulateur : il met en forme le signal analogique émis sur le canal.



Codage ≠ cryptographie

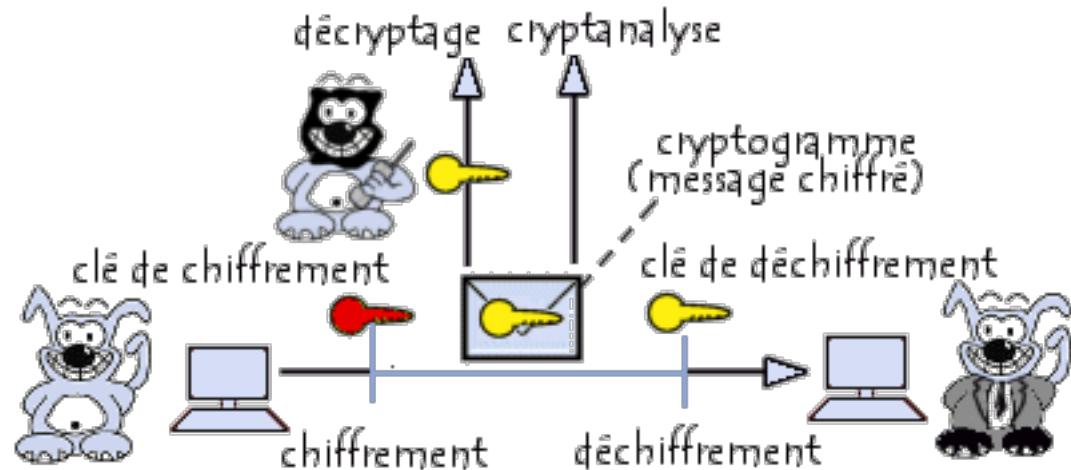
Le codage sert à mettre en forme le signal binaire pour l'adapter au canal : économie de bits / détection – correction d'erreurs.

La cryptographie consiste à chiffrer le message émis dans le but d'assurer :

- **La confidentialité du message : seul son destinataire peut le lire**
- **L'authenticité du message : le destinataire est sûr que le message a été émis par la bonne personne**
- **L'intégrité du message : une tierce personne n'a pas pu modifier le contenu du message en cours de transmission**

Terminologie

- **Cryptographie:** art et science du chiffrement
- **Cryptanalyse:** Art et science du déchiffrement
- **Cryptologie:** Branche de l'informatique théorique qui traite de la cryptographie et de la cryptanalyse



Cryptologie

Cryptologie : science des messages secrets qui englobe :

**La cryptographie, art de rendre inintelligible un message, et
la cryptanalyse, art de trouver le message clair caché.**

- La cryptographie utilise un chiffre pour coder un message. Le déchiffrement est l'opération inverse, par une personne autorisée à retrouver le message clair.
- La cryptanalyse est l'ensemble des techniques permettant à une personne non autorisée de trouver le contenu d'un message.

Système cryptographique ou crypto-système

- mécanisme permettant de camoufler des messages (i.e., de le rendre incompréhensible pour quiconque n'est pas autorisé)

Rappel Terminologie

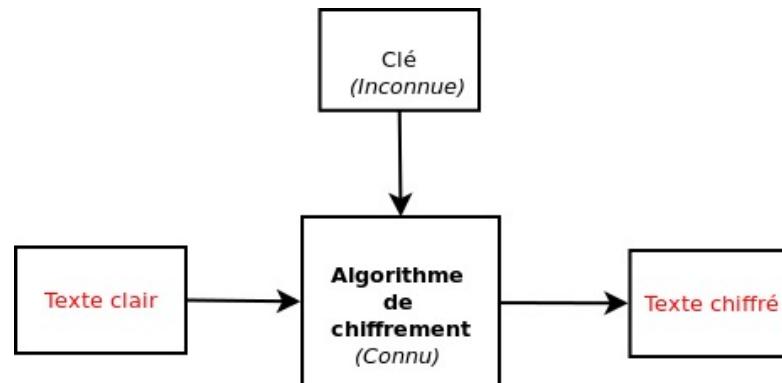
- ✓ **Texte en clair** : c'est le message à protéger.
- ✓ **Texte chiffré** : c'est le résultat du **chiffrement** du **texte en clair**.
- ✓ **Chiffrement** : c'est la méthode ou l'algorithme utilisé pour transformer un **texte en clair** en **texte chiffré**.
- ✓ **Déchiffrement** c'est la méthode ou l'algorithme utilisé pour transformer un **texte chiffré** en **texte en clair**.
- ✓ **Clé** : c'est le secret partagé utilisé pour **chiffrer** le **texte en clair** en **texte chiffré** et pour **déchiffrer** le **texte chiffré** en **texte en clair**. On peut parfaitement concevoir un algorithme qui n'utilise pas de **clé**, dans ce cas c'est l'algorithme lui-même qui constitue **la clé**, et son principe ne doit donc en aucun cas être dévoilé.
- ✓ **Cryptographie** : cette branche regroupe l'ensemble des méthodes qui permettent de **chiffrer** et de **déchiffrer** un **texte en clair** afin de le rendre incompréhensible pour quiconque n'est pas en possession de la **clé** à utiliser pour le **déchiffrer**.

Terminologie (suite)

- **Cryptanalyse** : c'est l'art de révéler les **textes en clair** qui ont fait l'objet d'un **chiffrement** sans connaître la **clé** utilisée pour **chiffrer le texte en clair**.
- **Décrypter** : c'est l'action de retrouver le **texte en clair** correspondant à un **texte chiffré** sans posséder la clé qui a servi au chiffrement. Ce mot ne devrait donc être employé que dans le contexte de la cryptanalyse.
- **Crypter** : en relisant la définition du mot **décrypter**, on peut se rendre compte que le mot **crypter** n'a pas de sens et que son usage devrait être oublié. Le mot **cryptage** n'a pas plus de sens non plus.
- **Coder, décoder** : c'est une méthode ou un algorithme permettant de modifier la mise en forme d'un message sans introduire d'élément secret. Le Morse est donc un code puisqu'il transforme des lettres en trait et points sans notion de secret. L'ASCII est lui aussi un code puisqu'il permet de transformer une lettre en valeur binaire.

Comment fonctionne la cryptographie?

- Un **algorithme cryptographique**, ou **chiffre**, est une fonction mathématique utilisée dans le processus de chiffrement et de déchiffrement.
 - Un algorithme cryptographique fonctionne en combinaison avec une clé – un mot, un nombre, ou une phrase – pour chiffrer le texte clair. Le même texte clair se chiffre en un texte chiffré différent si l'on utilise des clés différentes.
 - La sécurité des données chiffrées est entièrement dépendante de deux choses: la force de l'algorithme cryptographique et le secret de la clé.



Définition d'un cryptosystème

Un cryptosystème est constitué :

d'un ensemble fini M de messages (textes) clairs

d'un ensemble fini C de cryptogrammes (messages cryptés)

d'un ensemble fini K de clefs

d'une application d'encryptage

$$E : M \times K \rightarrow C$$

et d'une application de décryptage

$$D : C \times K \rightarrow M$$

telles que, pour tout $M \in M$ et tout $K \in K$

$$D(E(M, K), K) = M$$

Différents types de cryptosystèmes

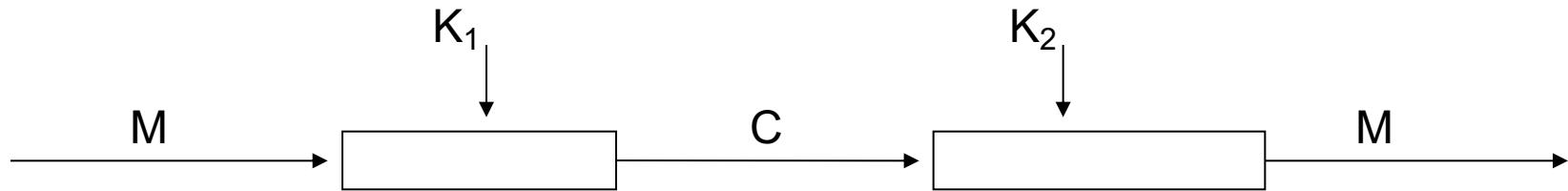
Cryptosystème symétrique

- Si $e == d$, la clef $K == e == d$ est dite symétrique de même que le cryptosystème
- et on dit aussi, dans ce cas, que la clef est secrète (ou bien privée).

Cryptosystème asymétrique

- Si e est publique et d est privée, on dit que le système (ou la clef) est asymétrique ou encore que c'est un système à clef publique.
- Dans un tel système asymétrique, le calcul de e (resp. d) en fonction de d (resp. e) doit être infaisable pratiquement.

Deux types d'algorithmes



Algorithmes à clef secrète

K_1 peut être calculé à partir K_2 et vice versa.

On a souvent $K_1 = K_2$

K_1 et K_2 doivent être secrètes

Algorithmes à clef publique

$K_1 \neq K_2$

K_2 ne peut pas être calculé à partir de K_1

K_1 peut être publique

K_2 doit être secrète (clef privée)

Exemple de cryptosystème

Considérons l'alphabet {A, B, C, , Z} et les deux permutations

$c =$  ABCDEFGHIJKLMNOPQRSTUVWXYZ
XYZABCDEFGHIJKLMNOPQRSTUVWXYZ

$d =$  ABCDEFGHIJKLMNOPQRSTUVWXYZ
DEFGHIJKLMNOPQRSTUVWXYZABC

Ici le message $m =$ CECI EST UN MESSAGE
est crypté en $m' = m_c =$ ZBF BPQ RK JBPPXDB
puis décrypté en $m = m'_d =$ CECI EST UN MESSAGE.

Chiffre de CESAR

Le Chiffre de César est considéré comme le plus ancien algorithme de chiffrage par substitution.

La technique est élémentaire : il suffit de remplacer chaque lettre du texte à chiffrer par la lettre qui se situe n places plus loin dans l'alphabet. Par exemple si $n=3$, on remplacera A par D, B par E, C par F etc.

Il existe de nombreuses variantes de cette technique. On peut, par exemple, remplacer chacune des lettres du message par un nombre correspondant à sa position dans l'alphabet.

On remplacera le "A" par "1", le "B" par "2", le "Z" par "26". On peut également compliquer la chose en faisant commencer l'alphabet par n'importe quelle lettre.

On peut également ne pas se contenter de faire glisser l'alphabet mais remplacer une lettre par n'importe quelle autre lettre de l'alphabet.

On arrive ainsi à plus de 400 000 000 000 000 000 000 000 combinaisons possibles, ce qui est déjà respectable.

Méthodes de Cryptage

Fondamentalement, il existe deux méthodes de cryptage, qui sont d'ailleurs toujours utilisées de nos jours.

Les confusions qui mélangent l'ordre des symboles contenus dans le message et les substitutions qui remplacent un symbole par un autre.

Sur la base de ces deux techniques, une multitudes de variantes, plus ou moins complexes, plus ou moins efficaces, ont été inventées.

Dans le cas du Chiffre de CESAR précédent, on parle de substitution mono-alphabétique car on remplace chaque lettre par une autre lettre de l'alphabet, toujours la même.

La substitution mono-alphabétique

Le premier usage révélé de chiffrement par substitution dans un usage militaire est rapporté par Jules César dans *La guerre des Gaules*. César utilisait fréquemment le chiffrement et en particulier le décalage de trois caractères.

La substitution mono-alphabétique fut la technique de chiffrement la plus utilisée durant le premier millénaire. Nombreux savants de l'antiquité tenaient cette technique pour inviolable.

Ce sont les Arabes qui réussirent à briser ce code et qui inventèrent la cryptanalyse au 9ième siècle.

Exemple

BQPSNRSJXJNJXLDPCLDLPQBE_QRKJXHNKPJSJPJIKSPUNBD
KIQRBKPQPBQPZITEJQDQBTSKPELNIUNPHNKPBKPCCKSSQW
KPSLXJPSNVVXSQCCJDJPBLDWPXBPSNVVXJPGKPJKDXIPZLC
EJKPGKSPSJQJXSJXHNKSPGPLZZNIIKDZKPGKSPGXVVKIKDJK
SPBKJJIKS

Hypothèses :

- Chaque lettre est chiffrée de la même façon...
- Certaines lettres sont utilisées plus souvent dans une langue donnée.

Technique de décryptage

Occurrence des lettres

En français

L	19.3	L	4.7	H	0.8
E	13.9	O	4.1	G	0.8
A	6.7	D	2.9	B	0.6
S	6.3	P	2.5	X	0.4
I	6.1	C	2.4	Y	0.3
T	6.1	M	2.1	J	0.3
N	5.6	V	1.3	Z	0.1
R	5.3	Q	1.3	K	0.0
U	5.2	F	0.9	W	0.0

Dans le cryptogramme

P	14.3	D	4.6	W	1.0
K	12.8	L	4.1	U	1.0
S	9.2	V	3.1	T	1.0
J	9.2	Z	2.6	-	0.5
X	5.6	G	2.6	O	0.0
Q	5.6	C	2.6	M	0.0
N	5.6	E	2.0	F	0.0
B	5.1	R	1.5	A	0.0
I	4.6	H	1.5	Y	0.0

Remplaçons **P** par _ et **K** par **E**

BQ_SNRSJXJNJXLD_CLDL_QBE_QREJXHNE_ESJ_JIES_UNBDE
IQRBE_Q_BQ_ZITEJQDQBTSE_ELNIUN_HNE_BE_CESSQWE_
SLXJ_SNVVXSQCCEDJ_BLDW_XB_SNVVXJ_GE_JEDXI_ZLCEJE
_GES_SJQJXSJXHNES_G_LZZNIIEDZE_GES_GXVVEIEDJES_BEJ
JIES

Remplaçons **Q** par **A** et **B** par **L**

LA_SNRSJXJNJXLD_CLDL_ALE_AREJXHNE_ESJ_JIES_UNLDEIA
RLE_A_LA_ZITEJADALTSE_ELNIUN_HNE_LE_CESSAWE_SLXJ_
SNVVXSACCEDJ_LLDW_XL_SNVVXJ_GE_JEDXI_ZLCEJE_GES_
SJAJXSJXHNES_G_LZZNIIEDZE_GES_GXVVEIEDJES_LEJJIES

Remplaçons **S** par S et **G** par D

LA_SNRSJXJNXLD_CLDL_ALE_AREJXHNE_ESJ_JIES_UNLDEIA
RLE_A_LA_ZITEJADALTSE_ELNIUN_HNE_LE_CESSAWE_SLXJ_
SNVVXSACCEDJ_LLDW_XL_SNVVXJ_DE_JEDXI_ZLCEJE_DES_S
JAJXSJXHNES_D_LZZNIIEDZE_DES_DXVVEIEDJES_LEJJIES

Remplaçons **J** par T et **I** par R

LA_SNRSTXTNTXLD_CLDL_ALE_ARETXHNE_EST_TRES_UNLDE
RARLE_A_LA_ZRTETADALTSE_ELNRUN_HNE_LE_CESSAWE_SL
XT_SNVVXSACCEDT_LLDW_XL_SNVVXT_DE_TEDXR_ZLCETE_
DES_STATXSTXHNES_D_LZZNRREDZE_DES_DXVVEREDTES_LE
TTRES

Remplaçons **X** par I, **H** par Q et **N** par U

LA_SURSTITUTILD_CLDL_ALE_ARETIQUE_EST_TRES_UULDERA
RLE_A_LA_ZRTETADALTSE_ELURUU_QUE_LE_CESSAWE_SLIT_
SUUVVISACCEDT_LLDW_IL_SUVVIT_DE_TEDIR_ZLCETE_DES_ST
ATISTIQUES_D_LZZURREDZE_DES_DIVEREDTES_LETTRES

Remplaçons **V** par F et **D** par N

LA_SURSTITUTILN_CLNL_ALE_ARETIQUE_EST_TRES_UULNERA
RLE_A_LA_ZRTETANALTSE_ELURUU_QUE_LE_CESSAWE_SLIT_
SUFFISACCENT_LLNW_IL_SUFFIT_DE_TENIR_ZLCETE_DES_STA
TISTIQUES_D_LZZURRENZE_DES_DIFFERENTES_LETTRES

Remplaçons **R** par B et **L** par O

LA_SUBSTITUTION_CONO_ALE_ARETIQUE_EST_TRES_UULNER
ABLE_A_LA_ZRTETANALTSE_EOURUU_QUE_LE_CESSAWE_SOI
T_SUFFISACCENT_LONW_IL_SUFFIT_DE_TENIR_ZOCETE_DES_
STATISTIQUES_D_OZZURRENZE DES_DIFFERENTES_LETTRES

Finallement

LA_SUBSTITUTION_MONO_ALPHABETIQUE_EST_TRES_VULNE
RABLE_A_LA_CRYPTANALYSE_POURVU_QUE_LE_MESSAGE_SO
IT_SUFFISAMMENT_LONG_IL_SUFFIT_DE_TENIR_COMPTE_DE
S_STATISTIQUES_D_OCCURRENCE DES_DIFFERENTES_LETTRES

Chiffre de Vigenère

Le **chiffre de Vigenère** est un algorithme de chiffrement établi par le cryptographe français Blaise de Vigenère.

Ce cryptosystème est de type **poly-alphabétique**, en opposition au mono-alphabétique que avons déjà vu, c'est-à-dire qu'il consiste à changer une lettre par une autre, mais cette dernière n'est pas toujours là même.

Cela permet une plus grande sécurité.

Cet algorithme utilise une clé, sous la forme d'un mot ou d'une phrase. Plus l'expression sera longue, plus le cryptogramme sera sécurisé.

Table de Vigenère

Le chiffrement se déroule en deux étapes.

Dans un premier temps, on choisit un message, par exemple « Bonne année » et une clé, par exemple « Dakar ».

En-dessous de chaque lettre du message, on écrit chaque lettre de la clé, en répétant le motif autant de fois que nécessaire.

**BONNE ANNEE
DAKAR DAKAR**

Dans un second temps, le message correspondant aux colonnes de la table, et la clé aux lignes.

Pour chaque lettre du message, la lettre chiffrée correspond au croisement entre la colonne correspondant à la lettre du message et la ligne correspondant à la lettre de la clé.

Par exemple, la lettre qui est au croisement entre la colonne 'B' et la ligne 'D' est 'E'. La première lettre du message chiffrée est donc 'E'. (et ainsi de suite ...)

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	Y	

**Trouvez le Principe de déchiffrement ? Décodez le message
« GEGEEDTS »**

Le masque jetable (ou chiffre de Vernam)

Le masque jetable est un cryptosystème établi par l'ingénieur Gilbert Vernam. *En théorie*, cet algorithme de chiffrement est réputé comme étant le seul à être **impossible à casser**.

Principe de chiffrement

Pour chiffrer un message, on doit prendre une clé, qui doit avoir les caractéristiques suivantes :

la clé doit avoir un nombre de caractères supérieur ou égal à celui du message ;

**les caractères de la clé doivent avoir été choisis de manière aléatoire ;
chaque clé ne doit être utilisée qu'une seule fois.**

Toutes ces propriétés dans le but final de garantir une sécurité optimale. Si elles sont respectées à la lettre, la sécurité garantie est **absolue**.

Le masque jetable (ou chiffre de Vernam)

Exemple :

Nous allons en prendre un message de quatre lettres : « ZERO ».

Nous tirons ensuite une chaîne de quatre lettres au hasard. Le résultat : « JRGV ». Ceci est la clé.

On attribue ensuite une valeur différente à chaque lettre de l'alphabet.

Pour faire simple, nous choisissons le même principe que pour le chiffre de César : 'A' vaut 0, 'B' vaut 1, 'C' vaut 2, etc. jusqu'à 'Z'=25

On additionne la valeur de chaque lettre du message avec la valeur de la lettre de la clé correspondante, puis on fait modulo 26.

Chiffrement :

On applique la formule $(x+y) \bmod 26 = z$

1. $Z + J = 25 + 9 = 34 - 26 = 8 = I$
2. $E + R = 4 + 17 = 21 = V$
3. $R + V = 17 + 21 = 38 - 26 = 12 = M$
4. $O + G = 14 + 6 = 20 = U$

Le déchiffrement s'effectue à peu près de la même manière, mis à part que, cette fois-ci, on soustrait la valeur de la lettre de la clé à la valeur de la lettre du cryptogramme correspondante et que l'on ajoute 26 lorsque le résultat est négatif.

$$\begin{aligned} I - J &= 8 - 9 = -1 + 26 = 25 = Z \\ V - R &= 21 - 17 = 4 = E \\ M - V &= 12 - 21 = -9 + 26 = 17 = R \\ U - G &= 20 - 6 = 14 = O \end{aligned}$$

Utilisation d'un «OU exclusif»

Le changement d'échelle que constitue le passage du niveau de codage "caractère" au niveau de codage "bit" a permis de manipuler véritablement les nombres,

La technique la plus simple est d'appliquer un « OU exclusif » (XOR) entre le texte à chiffrer et la clé. L'opération « OU exclusif », est une opération logique qui ne retourne la valeur 1 que si les deux bits comparés sont différents.

L'algorithme du « OU exclusif » simple n'est en fait rien d'autre qu'un chiffre de Vigenere, c'est à dire une substitution polyalphabétique utilisant une clé

Implémentation simple du cryptage par «OU exclusif».

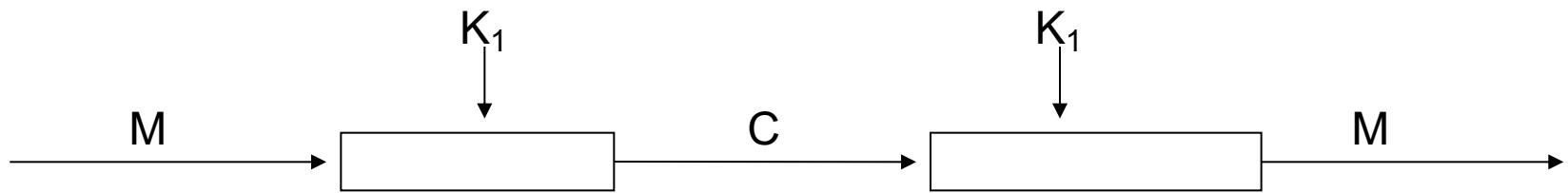
1. On procède caractère par caractère,
2. (clé + message) = cipher Ex : (Alex + Corenthin) = Message chiffré
3. Conversion en binaire : A = h43 : C = h41
4. (A = 01000011) + (C = 01000001) = (2 = 00000010)

Les caractères non affichables sont remplacés par leur code en hexadécimal.

Le processus est réversible : si on rechiffre un texte déjà chiffré, avec la même clé, on obtient le texte en clair.

La cryptographie à algorithmes symétriques

La cryptographie à algorithmes symétriques utilise la même clé pour les processus de codage et de décodage; cette clé est le plus souvent appelée "secrète" (en opposition à "privée") car toute la sécurité de l'ensemble est directement liée au fait que cette clé n'est connue que par l'expéditeur et le destinataire.



Ce type de cryptographie fonctionne habituellement suivant deux procédés différents, le cryptage par blocs et le cryptage par flot en continu ("stream").

Le chiffrement par flot

le cryptage est effectué bit-à-bit sans attendre la réception complète des données à crypter. C'est le principe des "stream-ciphers"

Une technique de chiffrement, du nom de "One-Time Pad" est utilisé pour chiffrer les flux. C'est le chiffrement inconditionnel le plus sûr.

Pour cela, on a besoin d'une chaîne aléatoire de la même longueur que le message d'origine, ce qui n'est pas pratique. Le but d'un stream cipher est de générer une chaîne aléatoire à partir d'une clé de longueur courte.

Le chiffrement par flot

Une autre technique consiste à "xorer", c'est-à-dire à appliquer un OU exclusif (XOR) au message avec un autre message prédéfini.

Bien entendu, cela nécessite que le destinataire (la personne qui décrypte) connaisse le message prédéfini et donc cela rajoute de la complexité au schéma général.

Les stream-ciphers sont utilisés aujourd'hui par différentes applications.

Pour chiffrer les flux, l'algorithme RC4 est très utilisé

Le chiffrement par bloc

Le cryptage en blocs (block-cipher) est au contraire beaucoup plus utilisé et permet une meilleure sécurité. Les algorithmes concernés sont également plus connus (DES, AES, Skipjack...); leur nom leur vient du fait qu'ils s'appliquent à des blocs de données et non à des flux de bits (cf. stream-ciphers).

Quatre modes de chiffrement par bloc sont utilisés :

Electronic CodeBook (ECB),

Cipher Block Chaining (CBC),

Cipher FeedBack (CFB) ou

Output FeedBack (OFB).

Ces blocs sont habituellement de 64 bits mais cela dépend entièrement de l'algorithme utilisé et de son implémentation.

De même, la taille de la clé varie suivant l'algorithme et suivant le niveau de sécurité requis;

Différents types de Chiffrement par blocs

Le mode **Electronic CodeBook (ECB)** est le plus simple des modes et s'applique aux block ciphers.

Il revient à crypter un bloc indépendamment des autres; cela permet entre autre de crypter suivant un ordre aléatoire (bases de données, etc...) mais en contre-partie, ce mode est très vulnérable aux attaques. Il demeure que si la clé fait 128 bits ou plus, cette attaque n'est pas exploitable en pratique de nos jours.

Cette technique est sensible à l'inversion ou la duplication de blocs sans que le destinataire s'en aperçoive. On peut l'utiliser pour pipeliner du hardware.

Le mode **Cipher Block Chaining (CBC)** est utilisé par les algorithmes en bloc.

C'est d'ailleurs le mode le plus courant. Il permet d'introduire une complexité supplémentaire dans le processus de cryptage en créant une dépendance entre les blocs successifs; autrement dit, le cryptage d'un bloc va être -d'une manière ou d'une autre- lié à ou aux blocs/chiffrés précédents.

Différents types de Chiffrement par blocs

Le mode **Cipher FeedBack (CFB)** est un mode destiné aux block ciphers dans le but d'en autoriser une utilisation plus souple, qui s'apparente plus à celle des algorithmes en continu.

On peut le considérer comme un intermédiaire entre les deux.

En effet, en partant d'un algorithme en bloc utilisant une longueur standard de n bits/blocs, le mode CFB va permettre de crypter des blocs dont la longueur pourra varier de n à 1 bits/blocs. Sachant que dans ce dernier cas, il serait plus économique en calculs d'utiliser directement un algorithme en continu.

Quant au cas où la longueur est celle de l'algorithme (à savoir n), le schéma de CFB se simplifie et ressemble quelque peu à celui de CBC (à quelques nuances près) :

Le mode **Output FeedBack (OFB)** est une variante de mode CFB précédemment abordé.

Il est d'ailleurs parfois appelé internal feedback. Il présente beaucoup de problèmes de sécurité et il est peu conseillé sauf dans le cas où sa longueur est égale à celle de l'algorithme utilisé.

algorithmes de cryptage par blocs : DES

L'algorithme symétrique « ***Data Encryption Standard*** » a été élaboré chez *IBM*, puis fut adopté comme norme de cryptage par l'administration américaine en 1977.

C'est à ce jour l'algorithme de cryptage le plus répandu. Il est employé par exemple pour crypter la transmission des NIP depuis les distributeurs bancaires.

Méthode :

Le cryptage DES de base utilise une clef de 56 bits.

Il s'effectue en 16 passes de rotation et 3 transpositions sur des mots de 64 bits.

Fiabilité :

En presque vingt ans, cet algorithme a fait l'objet de nombreuses tentatives de forçage.

On est parvenu à le percer en 1994 en partant d'un échantillon connu de 2^{43} mots (512 To. !!).

Performance :

Sur implantation matérielle, l'algorithme DES est capable de crypter ou décrypter entre 300 Mbits et 3 Gbits/sec. Il est donc éligible pour (dé)crypter sans surcoût des échanges permanents tels les échanges sur un réseau ou sur un bus.

On l'envisage pour le cryptage du téléphone ou d'un signal vidéo haute définition (1,5 Gbits/sec.).

Autres algorithmes de Cryptages par blocs

Triple DES :

On pratique de plus en plus souvent un « triple DES » pour acheminer une clef privée ou même pour un cryptage de données. Le triple DES est réalisé par trois cryptages successifs, employant deux clefs différentes: Cryptage1 -> Cryptage2 -> Cryptage1
Il est démontré que cela renforce le cryptage.

IDEA (*International Data Encryption Algorithm*) est une initiative de développement ouvert qui a abouti à un algorithme de cryptage (presque) symétrique, fiable et performant.

Les bases théoriques ont été largement diffusées et les développements ont été effectués au grand jour. Il est aussi performant que DES et s'implante bien sur des composants matériels.

SAFER (*Secure And Fast Encryption Routine*) est un algorithme du domaine public, qui opère des transformations sur des octets.

C'est une particularité qui le prédestine pour les implantations sur les cartes à puce. Il a l'inconvénient de ne pas être symétrique.

Autres algorithmes de Cryptages par blocs

Skipjack est un algorithme confidentiel mis au point par la NSA et implanté sur un composant, le *Clipper Chip*.

Étant confidentiel, on est pas en mesure de le mettre autant à l'épreuve qu'un algorithme public, ce qui fait dire aux mauvaises langues qu'il ne serait peut-être pas si sûr que cela.

Blowfish est un algorithme symétrique.

Il se décompose en opérations XOR et additions sur des mots de 32 bits, ce qui le rend bien plus rapide que DES sur des architectures 32 bits.

RC2, RC4 et RC5 (Confidentiels)

Ces algorithmes sont confidentiels et propriété de RSA Data Security <http://www.rsa.com/>. Ils sont très adaptables pour des compromis fiabilité / performance.

La Cryptographie à algorithmes asymétriques

Les algorithmes symétriques vus précédemment sont tous fiables mais ils posent un problème, c'est celui de l'échange de la clé : **comment transmettre de manière fiable à mon interlocuteur la clé de chiffrement utilisée pour chiffrer le message que je lui envoie ?**

Les algorithmes asymétriques ont été inventés pour pallier précisément le problème de transmission sécurisée de la clé.

On parle d'algorithmes asymétriques car ce n'est pas la même clé qui sert au chiffrement et au déchiffrement. Dans le cas de ces algorithmes, on parlera alors de clé privée et de clé publique. Ces deux clés, clé privée et clé publique, sont intimement liées par une fonction mathématique complexe.

La Cryptographie à algorithmes asymétriques

Les algorithmes asymétriques possèdent 2 modes de fonctionnement ;

- Le mode chiffrement dans lequel l'émetteur chiffre un fichier avec la clé publique du destinataire pour chiffrer. Le destinataire utilise sa clé privée pour déchiffrer le fichier. Dans ce mode, l'émetteur est sûr que seul le destinataire peut déchiffrer le fichier.
- Le mode signature dans lequel l'émetteur signe un fichier avec sa propre clé privée. Le destinataire utilise la clé publique de l'émetteur pour vérifier la signature du fichier. Dans ce mode, le destinataire est sûr que c'est bien l'émetteur qui a envoyé le fichier.