Cryptologie : Tour d'horizon

Christophe Guyeux

IUT de Belfort-Montbéliard, Université de Franche-Comté

Cours de Mathématiques et Informatique, 2008

Plan

Introduction à la cryptologie

Définitions

Définitions

Définitions

Cryptographie : étude des systèmes mathématiques propres à résoudre les problèmes de confidentialité et d'authentification.

Définitions

Cryptographie : étude des systèmes mathématiques propres à résoudre les problèmes de confidentialité et d'authentification.
S'occupe de communication en présence d'adversaires.

Définitions

Cryptographie : étude des systèmes mathématiques propres à résoudre les problèmes de confidentialité et d'authentification.
S'occupe de communication en présence d'adversaires.

Cryptanalyse : consiste à s'attaquer aux outils assurant des fonctionnalités de sécurité, pour en trouver les faiblesses.

Définitions

Cryptographie : étude des systèmes mathématiques propres à résoudre les problèmes de confidentialité et d'authentification.
S'occupe de communication en présence d'adversaires.

Cryptanalyse : consiste à s'attaquer aux outils assurant des fonctionnalités de sécurité, pour en trouver les faiblesses.

Cryptologie : partie de la sécurité des systèmes d'information qui s'occupe d'assurer, ou au contraire de compromettre, les grandes fonctions de sécurité.

Introduction à la cryptologie

Transmission de l'information

description

Trois grands problèmes de la transmission de l'information :

description

Trois grands problèmes de la transmission de l'information :

1. la représentation de l'information, son codage, sa compression ;

description

Trois grands problèmes de la transmission de l'information :

1. la représentation de l'information, son codage, sa compression ;
2. l'intégrité des données, la détection et la correction des erreurs ;

description

Trois grands problèmes de la transmission de l'information :

1. la représentation de l'information, son codage, sa compression ;
2. l'intégrité des données, la détection et la correction des erreurs ;
3. la sécurité de l'information, sa confidentialité.

Outils mathématiques

Cette théorie de l'information se base sur une panoplie d'outils mathématiques :

Outils mathématiques

Cette théorie de l'information se base sur une panoplie d'outils mathématiques :

1. théorie de la complexité des algorithmes ;

Outils mathématiques

Cette théorie de l'information se base sur une panoplie d'outils mathématiques :

1. théorie de la complexité des algorithmes ;
2. probabilités, statistiques, entropie, suites pseudo-aléatoires ;

Outils mathématiques

Cette théorie de l'information se base sur une panoplie d'outils mathématiques :

1. théorie de la complexité des algorithmes ;
2. probabilités, statistiques, entropie, suites pseudo-aléatoires ;
3. arithmétique ;

Outils mathématiques

Cette théorie de l'information se base sur une panoplie d'outils mathématiques :

1. théorie de la complexité des algorithmes ;
2. probabilités, statistiques, entropie, suites pseudo-aléatoires ;
3. arithmétique ;
4. algèbres de Boole ;
5. corps finis, géométrie affine et projective, géométrie combinatoire ;
6. algèbre commutative ;
7. géométrie algébrique.

Introduction à la cryptologie

Sécurité de l'information

Buts

Sécurité et confidentialité de l'information : buts à atteindre.

Buts

Sécurité et confidentialité de l'information : buts à atteindre.

secret, confidentialité, chiffrement : l'information n'est ni disponible, ni divulguée aux personnes, entités ou processus non autorisés ;

Buts

Sécurité et confidentialité de l'information : buts à atteindre.

secret, confidentialité, chiffrement : l'information n'est ni disponible, ni divulguée aux personnes, entités ou processus non autorisés ;

l'intégrité des données : prévention d'une modification non autorisée de l'information (attaque : *substitution*) ;

Buts

Sécurité et confidentialité de l'information : buts à atteindre.

secret, confidentialité, chiffrement : l'information n'est ni disponible, ni divulguée aux personnes, entités ou processus non autorisés ;

l'intégrité des données : prévention d'une modification non autorisée de l'information (attaque : *substitution*) ;

l'authentification : vérifier l'identité des différents éléments (personne, machine, document, etc.) impliqués dans un dialogue (attaques : *mascarades*).

Buts

Sécurité et confidentialité de l'information : buts à atteindre.

secret, confidentialité, chiffrement : l'information n'est ni disponible, ni divulguée aux personnes, entités ou processus non autorisés ;

l'intégrité des données : prévention d'une modification non autorisée de l'information (attaque : *substitution*) ;

l'authentification : vérifier l'identité des différents éléments (personne, machine, document, etc.) impliqués dans un dialogue (attaques : *mascarades*).

non-répudiation : impossibilité de nier un contrat.

Buts

Buts

signature : mécanisme garantissant l'authentification de l'expéditeur, l'intégrité des données et la non-répudiation ;

Buts

signature : mécanisme garantissant l'authentification de l'expéditeur, l'intégrité des données et la non-répudiation ;

certification : une entité connue, digne de confiance, valide une certaine information ;

Buts

signature : mécanisme garantissant l'authentification de l'expéditeur, l'intégrité des données et la non-répudiation ;

certification : une entité connue, digne de confiance, valide une certaine information ;

contrôle d'accès : l'accès à certaines ressources est limité aux personnes autorisées ;

Buts

signature : mécanisme garantissant l'authentification de l'expéditeur, l'intégrité des données et la non-répudiation ;

certification : une entité connue, digne de confiance, valide une certaine information ;

contrôle d'accès : l'accès à certaines ressources est limité aux personnes autorisées ;

gestion des clés : distribution, intégrité, recouvrement ;

Buts

signature : mécanisme garantissant l'authentification de l'expéditeur, l'intégrité des données et la non-répudiation ;

certification : une entité connue, digne de confiance, valide une certaine information ;

contrôle d'accès : l'accès à certaines ressources est limité aux personnes autorisées ;

gestion des clés : distribution, intégrité, recouvrement ;

preuve : une entité (appelée *prouveur*) souhaite démontrer à une autre entité (appelée *vérificateur*) qu'elle détient un certain secret (mot de passe, clé secrète, etc.)

Techniques cryptographiques

Techniques cryptographiques

Chiffrement à clé secrète

Terminologie

Un peu de vocabulaire...

Terminologie

Un peu de vocabulaire...

Chiffrement : consiste à transformer un texte claire en texte chiffré.

Terminologie

Un peu de vocabulaire...

Chiffrement : consiste à transformer un texte claire en texte chiffré.

Cryptogramme : le texte chiffré.

Terminologie

Un peu de vocabulaire...

Chiffrement : consiste à transformer un texte claire en texte chiffré.

Cryptogramme : le texte chiffré.

Décriffrement : opération inverse du chiffrement.

Terminologie

Un peu de vocabulaire...

Chiffrement : consiste à transformer un texte claire en texte chiffré.

Cryptogramme : le texte chiffré.

Décriffrement : opération inverse du chiffrement.

Décryptage : consiste à retrouver le texte clair à partir du cryptogramme lorqu'on ne connaît pas la clé.

Principe

Dans un système à clé secrète (ou *symétrique*) :

Principe

Dans un système à clé secrète (ou *symétrique*) :

- Un expéditeur et un destinataire partagent une même clé secrète.

Principe

Dans un système à clé secrète (ou *symétrique*) :

- Un expéditeur et un destinataire partagent une même clé secrète.
- Cette clé est utilisée à la fois pour le chiffrement et le déchiffrement.

Principe

Dans un système à clé secrète (ou *symétrique*) :

- Un expéditeur et un destinataire partagent une même clé secrète.
- Cette clé est utilisée à la fois pour le chiffrement et le déchiffrement.
- À chaque clé K sont associées :
  - une fonction de chiffrement $\mathcal{C}_K$,
  - une fonction de déchiffrement $\mathcal{D}_K$,

Principe

Dans un système à clé secrète (ou *symétrique*) :

- Un expéditeur et un destinataire partagent une même clé secrète.
- Cette clé est utilisée à la fois pour le chiffrement et le déchiffrement.
- À chaque clé K sont associées :
  - une fonction de chiffrement $\mathcal{C}_K$,
  - une fonction de déchiffrement $\mathcal{D}_K$,
- Le schéma classique d'un chiffrement à clé secrète est :
  1. L'expéditeur chiffre le texte clair m pour obtenir le texte chiffré $c = \mathcal{C}_K(m)$.
  2. L'expéditeur envoie c au destinataire.
  3. Le destinataire rétablit le texte clair en calculant $m = \mathcal{D}_K(c)$

Principe

Dans un système à clé secrète (ou *symétrique*) :

- Un expéditeur et un destinataire partagent une même clé secrète.
- Cette clé est utilisée à la fois pour le chiffrement et le déchiffrement.
- À chaque clé K sont associées :
  - une fonction de chiffrement $\mathcal{C}_K$,
  - une fonction de déchiffrement $\mathcal{D}_K$,
- Le schéma classique d'un chiffrement à clé secrète est :
  1. L'expéditeur chiffre le texte clair m pour obtenir le texte chiffré $c = \mathcal{C}_K(m)$.
  2. L'expéditeur envoie c au destinataire.
  3. Le destinataire rétablit le texte clair en calculant $m = \mathcal{D}_K(c)$
- Autrement dit : $\mathcal{D}_K \circ \mathcal{C}_K =$ Identité.

Qualités d'un système à clé secrète

Dans un système à clé secrète :

Qualités d'un système à clé secrète

Dans un système à clé secrète :

- La sécurité ne doit pas reposer sur le fait que le système de chiffrement est inconnu de l'ennemi...

Qualités d'un système à clé secrète

Dans un système à clé secrète :

- La sécurité ne doit pas reposer sur le fait que le système de chiffrement est inconnu de l'ennemi...
- ...mais sur la robustesse de la clé (principe de Kerckhoffs).

Problèmes rencontrés

Les problèmes principaux sont ici :

Problèmes rencontrés

Les problèmes principaux sont ici :

- Engendrer la clé secrète : nécessite l'utilisation d'un générateur aléatoire dont le comportement ne doit pas être prévisible.

Problèmes rencontrés

Les problèmes principaux sont ici :

- Engendrer la clé secrète : nécessite l'utilisation d'un générateur aléatoire dont le comportement ne doit pas être prévisible.
- L'échanger : problème spécifique de la cryptographie à clé secrète. Nécessité d'avoir un canal sûr, une sphère de confiance.

Problèmes rencontrés

Les problèmes principaux sont ici :

- Engendrer la clé secrète : nécessite l'utilisation d'un générateur aléatoire dont le comportement ne doit pas être prévisible.
- L'échanger : problème spécifique de la cryptographie à clé secrète. Nécessité d'avoir un canal sûr, une sphère de confiance.
- Stocker la clé secrète.

Problèmes rencontrés

Les problèmes principaux sont ici :

- Engendrer la clé secrète : nécessite l'utilisation d'un générateur aléatoire dont le comportement ne doit pas être prévisible.
- L'échanger : problème spécifique de la cryptographie à clé secrète. Nécessité d'avoir un canal sûr, une sphère de confiance.
- Stocker la clé secrète.
- Éviter de l'exposer lors de son utilisation pour chiffrer ou déchiffrer.

Pour ces raisons, la cryptographie à clé secrète, qui a une longue histoire, reste confinée aux applications militaires.

Types de chiffrement à clé secrète

Il existe deux types de chiffrement à clé secrète :

Types de chiffrement à clé secrète

Il existe deux types de chiffrement à clé secrète :

- Le chiffrement à flot : cryptosystème de Vernam...

Types de chiffrement à clé secrète

Il existe deux types de chiffrement à clé secrète :

- Le chiffrement à flot : cryptosystème de Vernam...
- Le chiffrement par bloc : DES, AES...

Chiffrement à flot

Une méthode de chiffrement à flot...

Chiffrement à flot

Une méthode de chiffrement à flot...

- Opère individuellement sur chaque bit de texte clair.

Chiffrement à flot

Une méthode de chiffrement à flot...

- Opère individuellement sur chaque bit de texte clair.
- Utilise une transformation qui varie en fonction de la place du bit d'entrée.

Chiffrement à flot

Une méthode de chiffrement à flot...

- Opère individuellement sur chaque bit de texte clair.
- Utilise une transformation qui varie en fonction de la place du bit d'entrée.
- Le prototype : cryptosystème de Vernam (ou masque jetable)...

Chiffrement à flot : cryptosystème de Vernam

Dans le cryptosystème de Vernam :

Chiffrement à flot : cryptosystème de Vernam

Dans le cryptosystème de Vernam :

- La clé secrète est une très longue suite aléatoire de bits, que possède expéditeur et destinataire.

Chiffrement à flot : cryptosystème de Vernam

Dans le cryptosystème de Vernam :

- La clé secrète est une très longue suite aléatoire de bits, que possède expéditeur et destinataire.
- Si l'on a un message de n bits à chiffrer :

Chiffrement à flot : cryptosystème de Vernam

Dans le cryptosystème de Vernam :

- La clé secrète est une très longue suite aléatoire de bits, que possède expéditeur et destinataire.
- Si l'on a un message de n bits à chiffrer :
  - on prend les n premiers bits de notre clé,

Chiffrement à flot : cryptosystème de Vernam

Dans le cryptosystème de Vernam :

- La clé secrète est une très longue suite aléatoire de bits, que possède expéditeur et destinataire.
- Si l'on a un message de n bits à chiffrer :
  - on prend les n premiers bits de notre clé,
  - on fait le « ou exclusif bit à bit » entre le message m et cette partie K de la clé : $c = m \oplus K$

Chiffrement à flot : cryptosystème de Vernam

Dans le cryptosystème de Vernam :

- La clé secrète est une très longue suite aléatoire de bits, que possède expéditeur et destinataire.
- Si l'on a un message de n bits à chiffrer :
  - on prend les n premiers bits de notre clé,
  - on fait le « ou exclusif bit à bit » entre le message m et cette partie K de la clé : $c = m \oplus K$
  - le destinataire calcule $c \oplus K$, et retrouve m.
- Cette « très longue suite aléatoire de bits » est difficilement réalisable. En pratique : générateur pseudo-aléatoire avec germe.

Chiffrement par bloc

Chiffrement par bloc

- Un chiffrement par bloc opère avec une transformation fixe qui s'applique sur des blocs de texte clair, de taille fixe.

Chiffrement par bloc

- Un chiffrement par bloc opère avec une transformation fixe qui s'applique sur des blocs de texte clair, de taille fixe.
- Exemples : DES, Triple DES, MISTY1, IDEA, AES, Camellia, SHACAL-2, etc.

Techniques cryptographiques

Techniques cryptographiques

Chiffrement à clé publique

Historique

Bref historique du chiffrement à clé publique :

1976, Diffie et Hellman : Notion de couple de clés, l'une servant au chiffrement, l'autre au déchiffrement.

Historique

Bref historique du chiffrement à clé publique :

1976, Diffie et Hellman : Notion de couple de clés, l'une servant au chiffrement, l'autre au déchiffrement.

1977, Rivest, Shamir et Adleman : système RSA.

Description

Dans un cryptosystème à clé publique...

Description

Dans un cryptosystème à clé publique...

- Chaque utilisateur A dispose d'une paire de clés : une clé privée d_A et une clé publique e_A.

Description

Dans un cryptosystème à clé publique...

- Chaque utilisateur A dispose d'une paire de clés : une clé privée d_A et une clé publique e_A.
- d_A n'est connue que de A (seul élément à rester secret), e_A est connue de tous. On ne peut pas retrouver d_A à partir de e_A.

Description

Dans un cryptosystème à clé publique...

- Chaque utilisateur A dispose d'une paire de clés : une clé privée d_A et une clé publique e_A.
- d_A n'est connue que de A (seul élément à rester secret), e_A est connue de tous. On ne peut pas retrouver d_A à partir de e_A.
- Sont également rendu publique :

Description

Dans un cryptosystème à clé publique...

- Chaque utilisateur A dispose d'une paire de clés : une clé privée d_A et une clé publique e_A.
- d_A n'est connue que de A (seul élément à rester secret), e_A est connue de tous. On ne peut pas retrouver d_A à partir de e_A.
- Sont également rendu publique :

  Le chiffrement $\mathcal{E}$, fonction qui à une clé publique e_A et un texte clair x fait correspondre le chiffré $y = \mathcal{E}(e_A, x)$ de x à destination de A.

Description

Dans un cryptosystème à clé publique...

- Chaque utilisateur A dispose d'une paire de clés : une clé privée d_A et une clé publique e_A.
- d_A n'est connue que de A (seul élément à rester secret), e_A est connue de tous. On ne peut pas retrouver d_A à partir de e_A.
- Sont également rendu publique :

  Le chiffrement $\mathcal{E}$, fonction qui à une clé publique e_A et un texte clair x fait correspondre le chiffré $y = \mathcal{E}(e_A, x)$ de x à destination de A.

  Le déchiffrement $\mathcal{D}$, fonction qui à une clé privée d_A et un chiffré y fait correspondre le texte clair $x = \mathcal{D}(d_A, y)$ associé à y.

Description

Dans un cryptosystème à clé publique...

- Chaque utilisateur A dispose d'une paire de clés : une clé privée d_A et une clé publique e_A.
- d_A n'est connue que de A (seul élément à rester secret), e_A est connue de tous. On ne peut pas retrouver d_A à partir de e_A.
- Sont également rendu publique :

  Le chiffrement $\mathcal{E}$, fonction qui à une clé publique e_A et un texte clair x fait correspondre le chiffré $y = \mathcal{E}(e_A, x)$ de x à destination de A.

  Le déchiffrement $\mathcal{D}$, fonction qui à une clé privée d_A et un chiffré y fait correspondre le texte clair $x = \mathcal{D}(d_A, y)$ associé à y.
- Si $E_A(x) = \mathcal{E}(e_A, x)$ et $D_A(y) = \mathcal{D}(d_A, y)$, alors $D_A \circ E_A = Id$

Schéma d'un système de chiffrement à clé publique

Si Bob veut communiquer le texte clair m à Alice :

Schéma d'un système de chiffrement à clé publique

Si Bob veut communiquer le texte clair m à Alice :

1. Bob calcule le texte chiffré $c = E_A(m)$ avec la clé publique de Alice.

Schéma d'un système de chiffrement à clé publique

Si Bob veut communiquer le texte clair m à Alice :

1. Bob calcule le texte chiffré $c = E_A(m)$ avec la clé publique de Alice.
2. c est envoyé.

Schéma d'un système de chiffrement à clé publique

Si Bob veut communiquer le texte clair m à Alice :

1. Bob calcule le texte chiffré $c = E_A(m)$ avec la clé publique de Alice.
2. c est envoyé.
3. Alice retrouve le texte clair en calculant $m = D_A(c)$.

Discution

Un tel cryptosystème...

Discution

Un tel cryptosystème...

- Repose sur la possibilité de construire des paires de fonctions réciproques (E_A et D_A) :

Discution

Un tel cryptosystème...

- Repose sur la possibilité de construire des paires de fonctions réciproques (E_A et D_A) :
  - faciles à calculer,

Discution

Un tel cryptosystème...

- Repose sur la possibilité de construire des paires de fonctions réciproques (E_A et D_A) :
  - faciles à calculer,
  - où E_A est très dure à inverser (en pratique).

Discution

Un tel cryptosystème...

- Repose sur la possibilité de construire des paires de fonctions réciproques (E_A et D_A) :
  - faciles à calculer,
  - où E_A est très dure à inverser (en pratique).
- Ne nécessite pas l'échange de clé secrète.

Discution

Un tel cryptosystème...

- Repose sur la possibilité de construire des paires de fonctions réciproques (E_A et D_A) :
  - faciles à calculer,
  - où E_A est très dure à inverser (en pratique).
- Ne nécessite pas l'échange de clé secrète.
- Pose certains problèmes :

Discution

Un tel cryptosystème...

- Repose sur la possibilité de construire des paires de fonctions réciproques (E_A et D_A) :
  - faciles à calculer,
  - où E_A est très dure à inverser (en pratique).
- Ne nécessite pas l'échange de clé secrète.
- Pose certains problèmes :
  - Comment éviter que la clé publique ne soit corrompue (attaque du serveur de clés, nouvelle clé forgée) ?

Discution

Un tel cryptosystème...

- Repose sur la possibilité de construire des paires de fonctions réciproques (E_A et D_A) :
  - faciles à calculer,
  - où E_A est très dure à inverser (en pratique).
- Ne nécessite pas l'échange de clé secrète.
- Pose certains problèmes :
  - Comment éviter que la clé publique ne soit corrompue (attaque du serveur de clés, nouvelle clé forgée) ?
  - Comment éviter que la clé privée ne soit exposée lorqu'elle est stockée, ou lors de son utilisation (inspection de la mémoire de la machine utilisant la clé, virus, etc.) ?

Exemples

Exemples de cryptosystèmes à clé publique, et difficulté sur laquelle ils se basent...

Exemples

Exemples de cryptosystèmes à clé publique, et difficulté sur laquelle ils se basent...

RSA : factoriser un produit de deux nombres premiers.

Exemples

Exemples de cryptosystèmes à clé publique, et difficulté sur laquelle ils se basent...

RSA : factoriser un produit de deux nombres premiers.

ElGamal : logarithme discret dans certains groupes (protocole de Diffie-Hellman).

Exemples

Exemples de cryptosystèmes à clé publique, et difficulté sur laquelle ils se basent...

RSA : factoriser un produit de deux nombres premiers.

ElGamal : logarithme discret dans certains groupes (protocole de Diffie-Hellman).

PSEC : utilise le calcul sur le groupe des points d'une courbe elliptique.

Exemples

Exemples de cryptosystèmes à clé publique, et difficulté sur laquelle ils se basent...

RSA : factoriser un produit de deux nombres premiers.

ElGamal : logarithme discret dans certains groupes (protocole de Diffie-Hellman).

PSEC : utilise le calcul sur le groupe des points d'une courbe elliptique.

Rabin : extraire une racine carrée modulo un produit n de deux grands nombres premiers (la décomposition de n étant secrète).

Techniques cryptographiques

Fonctions de hachage

Présentation

Présentons maintenant la notion de fonction de hachage :

- Une fonction de hachage transforme un message long en un résumé court, de taille fixe : l'*empreinte*.

Présentation

Présentons maintenant la notion de fonction de hachage :

- Une fonction de hachage transforme un message long en un résumé court, de taille fixe : l'*empreinte*.
- Une fonction de hachage doit résister :

Présentation

Présentons maintenant la notion de fonction de hachage :

- Une fonction de hachage transforme un message long en un résumé court, de taille fixe : l'*empreinte*.
- Une fonction de hachage doit résister :
  - À la détermination d'une préimage : impossibilité pratique de retrouver le message à partir de son empreinte.

Présentation

Présentons maintenant la notion de fonction de hachage :

- Une fonction de hachage transforme un message long en un résumé court, de taille fixe : l'*empreinte*.
- Une fonction de hachage doit résister :
  - À la détermination d'une préimage : impossibilité pratique de retrouver le message à partir de son empreinte.
  - Aux collisions : impossibilité pratique de construire deux messages ayant le même résumé.

Remarques et exemples

Remarquons que...

- Ces résistances sont pratiques : liées à l'ampleur des calculs.

Remarques et exemples

Remarquons que...

- Ces résistances sont pratiques : liées à l'ampleur des calculs.
- D'ailleurs, la résistance théorique aux collisions est impossible : il n'existe pas d'applications injective de A dans B, quand le cardinal de B est inférieur à celui de A

Remarques et exemples

Remarquons que...

- Ces résistances sont pratiques : liées à l'ampleur des calculs.
- D'ailleurs, la résistance théorique aux collisions est impossible : il n'existe pas d'applications injective de A dans B, quand le cardinal de B est inférieur à celui de A
- Exemples de fonctions de hachage : SHA1, MD5, SHA-512, Whirlpool, etc.

Techniques cryptographiques

Techniques cryptographiques

Signature et authentification

Principe

Algorithme de signature numérique...

Principe

Algorithme de signature numérique...

- L'utilisateur A qui veut signer un message M commence par en faire un condensé $m = h(M)$, grâce à une fonction publique de hachage.

Principe

Algorithme de signature numérique...

- L'utilisateur A qui veut signer un message M commence par en faire un condensé $m = h(M)$, grâce à une fonction publique de hachage.
- A dispose d'une clé publique e_A et d'une clé privée d_A.

Principe

Algorithme de signature numérique...

- L'utilisateur A qui veut signer un message M commence par en faire un condensé $m = h(M)$, grâce à une fonction publique de hachage.
- A dispose d'une clé publique e_A et d'une clé privée d_A.
- A calcule $s = E_A(m)$, cryptogramme de M, et envoie (M, s).

Principe

Algorithme de signature numérique...

- L'utilisateur A qui veut signer un message M commence par en faire un condensé $m = h(M)$, grâce à une fonction publique de hachage.
- A dispose d'une clé publique e_A et d'une clé privée d_A.
- A calcule $s = E_A(m)$, cryptogramme de M, et envoie (M, s).
- À partir de la clé publique e_A, on peut calculer $m = D_A(s)$, et s'assurer (puisque h aussi est publique) que m est bien le condensé de M.

Principe

Algorithme de signature numérique...

- L'utilisateur A qui veut signer un message M commence par en faire un condensé $m = h(M)$, grâce à une fonction publique de hachage.
- A dispose d'une clé publique e_A et d'une clé privée d_A.
- A calcule $s = E_A(m)$, cryptogramme de M, et envoie (M, s).
- À partir de la clé publique e_A, on peut calculer $m = D_A(s)$, et s'assurer (puisque h aussi est publique) que m est bien le condensé de M.
- Comme A est le seul à avoir accès à E_A, tout le monde peut s'assurer que c'est bien A qui a signé M (ce qui implique la non-répudiation).

Systèmes de signature numérique

Voici quelques exemples de systèmes de signature numérique :

Systèmes de signature numérique

Voici quelques exemples de systèmes de signature numérique :

RSA : Le système cryptographique utilisé est alors RSA.

Systèmes de signature numérique

Voici quelques exemples de systèmes de signature numérique :

RSA : Le système cryptographique utilisé est alors RSA.

DSS : Digital Signature Standard, utilise DSA basé sur la difficulté du problème du logarithme discret sur le groupe multiplicatif $\mathbb{Z}/p\mathbb{Z}$ (p : grand nombre premier).

Principe de l'authentification

- Le mécanisme d'authentification recouvre différentes fonctionnalités, comme :

Principe de l'authentification

- Le mécanisme d'authentification recouvre différentes fonctionnalités, comme :
  - garantir que l'expéditeur d'un message est bien celui qu'il prétend être,

Principe de l'authentification

- Le mécanisme d'authentification recouvre différentes fonctionnalités, comme :
  - garantir que l'expéditeur d'un message est bien celui qu'il prétend être,
  - garantir que le message n'est ni corrompu, ni forgé,

Principe de l'authentification

- Le mécanisme d'authentification recouvre différentes fonctionnalités, comme :
  - garantir que l'expéditeur d'un message est bien celui qu'il prétend être,
  - garantir que le message n'est ni corrompu, ni forgé,
- Un processus de signature peut servir d'authentification.

Principe de l'authentification

- Tout le monde peut s'assurer de la conformité de la signature, ce qui peut être indésirable dans un processus d'authentification.

Principe de l'authentification

- Tout le monde peut s'assurer de la conformité de la signature, ce qui peut être indésirable dans un processus d'authentification.
- On peut encore souhaiter empêcher la réutilisation des données échangées lors d'une précédente identification...

Principe de l'authentification

- Tout le monde peut s'assurer de la conformité de la signature, ce qui peut être indésirable dans un processus d'authentification.
- On peut encore souhaiter empêcher la réutilisation des données échangées lors d'une précédente identification...
- Les MAC (Message Authentification Code), construits par exemple en utilisant des systèmes de chiffrement par bloc à clé secrète (DES, AES), sont un exemple de systèmes d'authentification.

Techniques mathématiques

Arithmétique et algèbre

Problème de la factorisation

- Soit un entier n, produit de deux grands nombres premiers p et q. Déterminer p et q, connaissant n.

Problème de la factorisation

- Soit un entier n, produit de deux grands nombres premiers p et q. Déterminer p et q, connaissant n.
- La difficulté de ce problème est utilisé dans le chiffrement et la signature RSA, et dans la méthode d'identification de Feige-Fiat-Shamir.

Problème de la factorisation

- Soit un entier n, produit de deux grands nombres premiers p et q. Déterminer p et q, connaissant n.
- La difficulté de ce problème est utilisé dans le chiffrement et la signature RSA, et dans la méthode d'identification de Feige-Fiat-Shamir.
- On dispose de plusieurs algorithmes (non polynomiaux) de factorisation : méthodes ρ et $p-1$ de Pollard, méthode des fractions continues, des courbes elliptiques, etc.

Racines carrées

- Soit $p > 2$ un nombre premier, et x un carré dans $\mathbb{Z}/p\mathbb{Z}$. Il est possible de calculer en pratique les racines carrées de x.

Racines carrées

- Soit $p > 2$ un nombre premier, et x un carré dans $\mathbb{Z}/p\mathbb{Z}$. Il est possible de calculer en pratique les racines carrées de x.
- Par contre, dans $\mathbb{Z}/n\mathbb{Z}$ (n étant le produit de deux grands nombres premiers, que l'on ne connaît pas), le problème devient aussi difficile que le précédent.

Racines carrées

- Soit $p > 2$ un nombre premier, et x un carré dans $\mathbb{Z}/p\mathbb{Z}$. Il est possible de calculer en pratique les racines carrées de x.
- Par contre, dans $\mathbb{Z}/n\mathbb{Z}$ (n étant le produit de deux grands nombres premiers, que l'on ne connaît pas), le problème devient aussi difficile que le précédent.
- Ce problème est par exemple utilisé dans la méthode d'identification de Feige-Fiat-Shamir.

Logarithme discret

- Connaissant a^n, retrouver $n = log_a a^n$ peut s'avérer très difficile, comme dans $\mathbb{Z}/p\mathbb{Z}$: les meilleurs algorithmes connus sont en temps non polynomial.

Logarithme discret

- Connaissant a^n, retrouver $n = log_a a^n$ peut s'avérer très difficile, comme dans $\mathbb{Z}/p\mathbb{Z}$: les meilleurs algorithmes connus sont en temps non polynomial.
- Ce problème est par exemple utilisé dans l'échange de clé de Diffie et Hellman, dans les cryptosystèmes d'ElGamal, et la signature DSA.

Techniques mathématiques

Générateurs pseudo-aléatoires

Description

- Un générateur pseudo-aléatoire produit à partir d'un nombre initial appelé *germe* une suite de nombres ayant de bonnes propriétés statistiques de répartition.

Description

- Un générateur pseudo-aléatoire produit à partir d'un nombre initial appelé *germe* une suite de nombres ayant de bonnes propriétés statistiques de répartition.
- Pour un usage cryptographique, un tel générateur doit posséder également de bonnes propriétés d'imprévisibilité.

Description

- Un générateur pseudo-aléatoire produit à partir d'un nombre initial appelé *germe* une suite de nombres ayant de bonnes propriétés statistiques de répartition.
- Pour un usage cryptographique, un tel générateur doit posséder également de bonnes propriétés d'imprévisibilité.
- Les générateurs pseudo-aléatoires sont en général construits à l'aide de suites récurrentes ; le germe fournit le premier terme de la suite.

Description

- Un générateur pseudo-aléatoire produit à partir d'un nombre initial appelé *germe* une suite de nombres ayant de bonnes propriétés statistiques de répartition.
- Pour un usage cryptographique, un tel générateur doit posséder également de bonnes propriétés d'imprévisibilité.
- Les générateurs pseudo-aléatoires sont en général construits à l'aide de suites récurrentes ; le germe fournit le premier terme de la suite.
- Avec un bon générateur pseudo-aléatoire, on peut émuler un système à masque jetable.

Techniques mathématiques

Techniques mathématiques

Théorie de la complexité des algorithmes

Présentation

Complexité de la sécurité parfaite.

- Dans une sécurité parfaite, il est nécessaire de disposer d'une clé aussi longue que la somme des longueurs de tous les textes à chiffrer.

Présentation

Complexité de la sécurité parfaite.

- Dans une sécurité parfaite, il est nécessaire de disposer d'une clé aussi longue que la somme des longueurs de tous les textes à chiffrer.
- Elle doit être tirée au hasard dans un ensemble de clés équirépartis.

Présentation

Complexité de la sécurité parfaite.

- Dans une sécurité parfaite, il est nécessaire de disposer d'une clé aussi longue que la somme des longueurs de tous les textes à chiffrer.
- Elle doit être tirée au hasard dans un ensemble de clés équirépartis.
- Cette sécurité parfaite est appliquée dans le téléphone rouge.

Présentation

Complexité dans les cas les plus fréquents.

- Les contraintes sont trop fortes pour les systèmes civils concrets : on dispose d'une plus faible sécurité, basées sur l'impossibilité de réaliser en pratique certains calculs.

Présentation

Complexité dans les cas les plus fréquents.

- Les contraintes sont trop fortes pour les systèmes civils concrets : on dispose d'une plus faible sécurité, basées sur l'impossibilité de réaliser en pratique certains calculs.
- La théorie de la complexité nous permet d'évaluer le temps d'exécution de certains algorithmes, la résistance de certaines fonctions cryptographiques.

Présentation

Complexité dans les cas les plus fréquents.

- Les contraintes sont trop fortes pour les systèmes civils concrets : on dispose d'une plus faible sécurité, basées sur l'impossibilité de réaliser en pratique certains calculs.
- La théorie de la complexité nous permet d'évaluer le temps d'exécution de certains algorithmes, la résistance de certaines fonctions cryptographiques.
- Certaines notions cryptographiques importantes sont définies à l'aide des classes de complexité, elles-mêmes définies à partir des machines de Turing.

Cryptanalyse

Cryptanalyse
Introduction

Buts de l'adversaire

Les buts de l'adversaire...

Buts de l'adversaire

Les buts de l'adversaire...

- Déterminer des informations (récupérer le texte clair, la clé).

Buts de l'adversaire

Les buts de l'adversaire...

- Déterminer des informations (récupérer le texte clair, la clé).
- Interagir avec le système (forger une signature, modifier un message), sans avoir le droit de le faire (notamment, sans connaître la clé).

Buts de l'adversaire

Les buts de l'adversaire...

- Déterminer des informations (récupérer le texte clair, la clé).
- Interagir avec le système (forger une signature, modifier un message), sans avoir le droit de le faire (notamment, sans connaître la clé).
- etc.

Afin de se protéger...

Pour se protéger, on doit connaître de l'adversaire :

Afin de se protéger...

Pour se protéger, on doit connaître de l'adversaire :

- Sa puissance de calcul.

Afin de se protéger...

Pour se protéger, on doit connaître de l'adversaire :

- Sa puissance de calcul.
- Est-il passif (observateur) ou actif (capable de modifier, voire de forger des messages).

Afin de se protéger...

Pour se protéger, on doit connaître de l'adversaire :

- Sa puissance de calcul.
- Est-il passif (observateur) ou actif (capable de modifier, voire de forger des messages).
- De quelles informations dispose-t-il ?

Afin de se protéger...

Pour se protéger, on doit connaître de l'adversaire :

- Sa puissance de calcul.
- Est-il passif (observateur) ou actif (capable de modifier, voire de forger des messages).
- De quelles informations dispose-t-il ?

...et nous demander exactement ce que l'on souhaite le plus protéger.

Cryptanalyse

Cryptanalyse

Classification des attaques

Classification des attaques

Classification des attaques

- Le cryptanalyste connaît tout sur le fonctionnement du système, sauf la clé secrète (principe de Kerckhoffs).

Classification des attaques

- Le cryptanalyste connaît tout sur le fonctionnement du système, sauf la clé secrète (principe de Kerckhoffs).
- Il existe diverses attaques, suivant les informations dont peut disposer le cryptanalyste :

Classification des attaques

- Le cryptanalyste connaît tout sur le fonctionnement du système, sauf la clé secrète (principe de Kerckhoffs).
- Il existe diverses attaques, suivant les informations dont peut disposer le cryptanalyste :

  Attaque à chiffré seul : le cryptanalyste ne connaît que le chiffré.

Classification des attaques

- Le cryptanalyste connaît tout sur le fonctionnement du système, sauf la clé secrète (principe de Kerckhoffs).
- Il existe diverses attaques, suivant les informations dont peut disposer le cryptanalyste :

  Attaque à chiffré seul : le cryptanalyste ne connaît que le chiffré.

  Attaque à textes clairs connus : le cryptanalyste connaît des paires de textes clairs et leurs chiffrés.

Classification des attaques

- Le cryptanalyste connaît tout sur le fonctionnement du système, sauf la clé secrète (principe de Kerckhoffs).
- Il existe diverses attaques, suivant les informations dont peut disposer le cryptanalyste :

  Attaque à chiffré seul : le cryptanalyste ne connaît que le chiffré.

  Attaque à textes clairs connus : le cryptanalyste connaît des paires de textes clairs et leurs chiffrés.

  Attaque à textes clairs choisis : le cryptanalyste peut obtenir le chiffrement des textes clairs de son choix, et ceci seulement avant la donné du chiffré à attaquer (appelé le *challenge*).

Classification des attaques

Classification des attaques

Attaque à textes chiffrés choisis : le cryptanalyste peut obtenir le déchiffrement des textes chiffrés de son choix, et ceci seulement avant la donnée du challenge.

Classification des attaques

Attaque à textes chiffrés choisis : le cryptanalyste peut obtenir le déchiffrement des textes chiffrés de son choix, et ceci seulement avant la donnée du challenge.

Attaque adaptative à textes chiffrés choisis : le cryptanalyste peut obtenir le déchiffrement des textes chiffrés de son choix, et ceci avant et après la donnée du challenge.

Classification des attaques

Attaque à textes chiffrés choisis : le cryptanalyste peut obtenir le déchiffrement des textes chiffrés de son choix, et ceci seulement avant la donnée du challenge.

Attaque adaptative à textes chiffrés choisis : le cryptanalyste peut obtenir le déchiffrement des textes chiffrés de son choix, et ceci avant et après la donnée du challenge.

Dans un système à clé publique, l'adversaire dispose de la clé de chiffrement (qui est publique). Il peut donc toujours porter au moins une attaque à textes clairs choisis (et même adaptative).

Quelques attaques attaques

Quelques attaques attaques

Cryptographie à clé publique : exemples d'attaques du RSA

Quelques attaques attaques

Cryptographie à clé publique : exemples d'attaques du RSA

- module commun ;
- petit exposant privé ;
- petit exposant public, attaque par « broadcast » ;
- attaque par analyse de temps, par analyse de courant.

Quelques attaques attaques

Cryptographie à clé publique : exemples d'attaques du RSA

- module commun ;
- petit exposant privé ;
- petit exposant public, attaque par « broadcast » ;
- attaque par analyse de temps, par analyse de courant.

Cryptographie à clé secrète : exemples d'attaques d'un chiffrement par bloc

Quelques attaques attaques

Cryptographie à clé publique : exemples d'attaques du RSA

- module commun ;
- petit exposant privé ;
- petit exposant public, attaque par « broadcast » ;
- attaque par analyse de temps, par analyse de courant.

Cryptographie à clé secrète : exemples d'attaques d'un chiffrement par bloc

- cryptanalyse différentielle ;
- cryptanalyse linéaire ;
- attaque par interpolation.