Cryptologie asymétrique

DUT S4

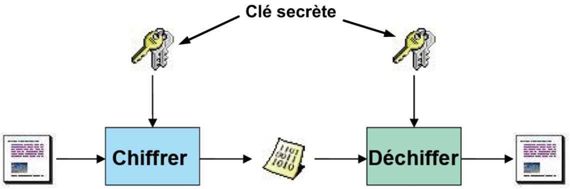
Pierre Ramet: ramet@labri.fr

2013-2014

Plan

Plan

La cryptgraphie à clé secrète



- **Une seule** clé pour chiffrer/déchiffrer
- La clé est connue des deux intervenants
- Si un attaquant intercepte la clé, fin de la confidentialité

Limites de la cryptographie à clé secrète

- Il faut pouvoir communiquer la clé secrète par un moyen sûr
  - Lettre, téléphone, malette diplomatique
  - Pas très pratique
- Nombre de clés à échanger pour communiquer avec plusieurs personnes

Nb personnes	Nb clés
2	1
5	10
100	4450
n	$n(n-1)$

- Plutôt contraignant

La cryptographie à clé publique

- Petite révolution dans les années 1970 (Diffie Hellman 1976)
- La sécurité ne repose désormais plus sur :
  - Un secret partagé (la clé secrète)
  - Des algorithmes obscurs
- Mais sur :
  - Des problèmes connus de tous (ex : factorisation)
  - Une information connue de tous (la clé publique)

La cryptographie à clé publique

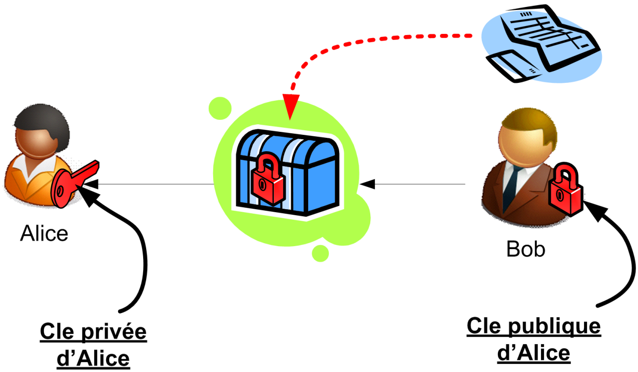
- La cryptographie à clé publique ou asymétrique est basée sur un concept très différent de la cryptographie symétrique
- Chaque intervenant possède une **clé publique**
  - Cette clé peut être connue de tous. Par exemple, disponible dans un répertoire accessible publiquement, sur internet
  - Toute personne connaissant cette clé peut envoyer un message chiffré au propriétaire de cette clé
- Chaque intervenant possède une **clé privée**
  - Cette clé doit demeurer confidentielle
  - Cette clé est liée (mathématiquement) à la clé publique correspondante
  - Cette clé permet de déchiffrer tout message chiffré avec la clé publique correspondante

Le principe du coffre fort

On peut assimiler la cryptographie à clé publique au protocole suivant :

- Bob veut envoyer un message à Alice de manière confidentielle
- Alice fournit un coffre fort à Bob, ainsi qu'un cadenas
  - Alice conserve la clé du cadenas
- Bob met ses documents dans le coffre d'Alice et le cadenasse
  - Le cadenas est la **clé publique** d'Alice
  - Il permet de mettre des informations dans le coffre
  - **Difficile** d'ouvrir le coffre juste avec le cadenas
- Alice récupère le coffre, et l'ouvre avec la clé du cadenas
  - C'est la **clé privée** d'Alice

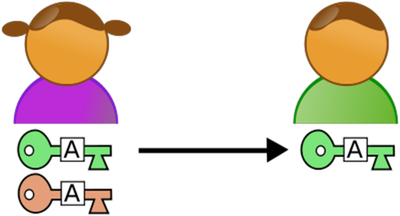
Exemple



Exemple plus réaliste

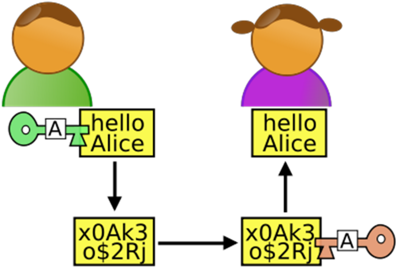
Un exemple plus réaliste :

- Bob veut envoyer un message à Alice de manière confidentielle
- Alice possède un couple (clé privée,clé publique)
- Bob récupère la clé publique d'Alice (disponible publiquement)



Exemple plus réaliste (2)

- Bob **chiffre** son message avec la **clé publique** d'Alice
- Il l'envoie à Alice
- Alice déchiffre le message avec sa clé privée



Avantages

Avantages :

- Si N intervenants veulent s'échanger des informations sans l'aide d'un tiers, chaque intervenant doit avoir une clé publique unique connue de tous
  - Donc, N clés sont suffisantes
- Les clés publiques doivent être distribuées de façon authentifiée, mais non confidentielle
  - Seule la clé publique est divulgée
  - Connaître la clé publique d'un intervenant **ne permet pas de déchiffrer** ses messages

Comment est-ce possible ?

- La cryptographie à clé publique est basée sur des problèmes mathématiques
- Utilisation de fonction **à sens unique à brèche secrète**
  - Métaphore du cadenas
    - Facile à fermer
    - Nécessite une clé pour ouvrir

Plan

Rappel de la complexité

Théorie de la complexité :

- On dira qu'un problème est **complexe** si il appartient à la classe NP (*non-determinist polynomial*)
  - C'est à dire que trouver une solution au problème se fait en $O(2^{n^k})$
  - Vérifier la solution se fait en temps polynomial
  - n étant la longueur de l'entrée (en bits)
- On dirat qu'un problème est **facile** si il existe un algorithme le résolvant appartenant à P
  - Trouver une solution se fait en $O(n^k)$
  - Facile ... si k reste petit
- Donnez des exemples

Rappel de la complexité (2)

- Un ordinateur peut résoudre des problèmes appartenant à la classe P
  - Dans la plupart des cas, c'est à dire si k pas trop grand
  - Un ordinateur peut difficilement résoudre des problèmes NP-complexes
    - Dès que n devient un peu grand, le temps nécessaire devient prohibitif
    - Exemple : factoriser un nombre de 1024 bits
  - Conjecture $P \neq NP$?
    - Pas prouvé !
    - Mais on l'espère

Fonction à sens unique

Definition

Une fonction à sens unique est une fonction f telle que $f(x)$ est facile à calculer et $f^{-1}(x)$ est difficile à calculer

- Exemple :
  - casser un oeuf
  - mélanger un pot de peinture rouge et un pot de peinture blanche

Factorisation

- Quelle est la complexité de factorisation ?
- Trouver les deux facteurs premiers de :
  - $35 = 5 * 7$
  - $221 = 13 * 17$
  - 50123093 ?
- Comment calculer le dernier exemple ?
- Quelle complexité ?
  - ici $n = \mid log_2(50123093) \mid +1 = 26$

Plan

Fonction à sens unique à brêche secrète

Definition

Une fonction à sens unique et à brèche secrète est une fonction f telle que

- $f(x)$ est facile à calculer
- $f^{-1}(x)$ est difficile à calculer
- $f^{-1}(x)$ sachant k est facile à calculer
  - k est la **brêche secrête**

Récapitulatif

- La cryptographie asymétrique : Chaque utilisateur possède **deux** clés :
  - Une **clé publique** qui permet de **chiffrer** des messages pour l'utilisateur
  - Une **clé privée** qui permet à l'utilisateur de **déchiffrer** les messages chiffrés avec sa clé publique
- La clé publique est **diffusée** à tout le monde
  - La connaître ne permet pas de déchiffrer les messages
- La clé privée est gardée **secrète** par l'utilisateur
  - La seule qui permette de déchiffrer les messages

Récapitulatif (2)

- La cryptographie à clé publique est basée sur des problèmes mathématiques **difficiles à résoudre**
  - Factorisation
  - Logarithme discret
- De ces problèmes, on extrait des **fonction à sens unique à brêche secrète**
  - Calculer $f(x)$ est **facile** (f=clé publique, x=message)
  - Calculer $f^{-1}(x)$ est **difficile**
  - Calculer $f^{-1}(x)$ sachant k est **facile** (k=clé privée)
- Les deux problèmes les plus célèbres :
  - Le problème **RSA**
  - Le problème **Diffie Hellman**

Plan

Calcul modulaire

- $37 \equiv 2 \ mod \ 5$:
  - $37 = 2 + k * 5$
  - Reste de la division Euclidienne
- Addition, multiplication, exponentiation modulaire
  - Opérations peu coûteuses
- Z_n= ensemble des résidus modulo n muni des opérations modulaires
- Inversion modulaire :
  - Trouver b tel que $ab \equiv 1 \ mod \ n$
    - Si $pgcd(a, n) = 1$, une solution unique (algorithme d'Euclide étendu)
    - Sinon pas de solution

Calcul modulaire (2)

- Petit théorème de Fermat :

Theorem

Si m premier, et pgcd$(a, m) = 1$, $a^{m-1} \equiv 1 \ mod \ m$

- Fonction d'Euler :

Definition

- $\varphi(n)$ est le nombre de résidus premiers avec n
- Si n est premier, $\varphi(n) = n - 1$
- Si $n = p * q$, alors $\varphi(n) = (p - 1)(q - 1)$

Calcul modulaire (3)

- Petit théorème de Fermat généralisé par Euler :

Theorem

Si $pgcd(a, n) = 1$, $a^{\varphi(n)} \equiv 1 \; mod \; n$

- Inverse modulaire :

Theorem

Si $pgcd(a, n) = 1$, *l'inverse de a est* $a^{\varphi(n)-1} \; mod \; n$

Plan

Quelques exemples

Deux principales fonction à sens unique et à brêche secrète en cryptographie asymétrique :

- Basé sur le problème `factorisation` :
  - Le problème **RSA**
- Basé sur le problème `logarithme discret`
  - Le problème **Diffie Hellman**
- Abordons tout d'abord RSA

Le problème RSA

- Le problème **factorisation** :
  - Entrée : $n = p * q$ produit de deux nombres premiers
  - Sortie : p et q
- Fournit une fonction à sens unique, mais pas de brêche secrète
- Le problème `RacineIemeModulaire` ou problème **RSA** :
  - Entrées :
    - Un entier $n = p * q$ produit de deux nombres premiers
    - Un entier $e > 0$ premier avec $(p - 1) * (q - 1)$
    - Un entier c
  - Sortie : m tel que $c = m^e \ mod n$
- Fonction à sens unique et à brêche secrète (p, q)
- le cryptosystème RSA est basé sur les problèmes `RacineIemeModulaire` **et** `factorisation`

Plan

RSA

- Chiffrement à clé publique le plus utilisé
- Créé en 1977 par Rivest, Shamir et Adleman
- Breveté par le MIT en 1983 aux États-Unis. Le brevet a expiré le 21 septembre 2000
- Utilisé dans :
  - Les banques
  - Les cartes à puce
  - Les site webs commerciaux

Protocole

Trois étapes :

1. Création d'une clé publique et d'une clé privée pour Bob (la clé publique est diffusée à tout le monde, par exemple à Alice)
2. A chaque fois qu'Alice veut envoyer un message confidentiel à Bob, elle utilise la clé publique de Bob pour chiffrer le message
3. Bob utilise sa clé privée pour déchiffrer le message envoyé par Alice

Création des clés

1. Choisir deux grand nombres p et q premiers
2. $n = p * q$ et $\varphi(n) = (p-1)(q-1)$
3. e un entier tel que $1 < e < \varphi(n)$ et e premier avec $\varphi(n)$
   - i.e $pgcd(e, \varphi(n)) = 1$
4. Calculer d tel que $ed = 1 \ mod \ \varphi(n)$
5. **Clé publique** : (e, n)
6. **Clé privée** : d (ou (p, q))

Chiffrement

1. Obtenir la clé publique (e, n) du destinataire
2. Représenter le message comme un entier m tel que $1 < m < n$
3. Calculer $c = m^e \ mod \ n$: texte chiffré
   - Relation avec le problème `RSA` ?

Déchiffrement

1. A l'aide de la clé privée d, calculer :

$$m = c^d \ mod \ n$$

2. Et c'est tout !

Exemple

- $p = 31$ et $q = 137$
- $n = 4247$ et $\varphi(n) = 4080$
- $e = 967$ $(1 < e < \varphi(n)$ et $pgcd(e, \varphi(n)) = 1)$
- $d = 2983$ $(1 < d < \varphi(n)$ et $ed = 707x4080 + 1 = 1 \ mod \ \varphi(n))$
- Clé publique : (e, n)
- Clé privée : d

Exemple Chiffrement/Déchiffrement

- Message en clair $m = 3333$
- Chiffrement :

$$c = m^e \ mod \ n = 3333^{967} \ mod \ 4247 = 3790$$

- Déchiffrement :

$$m = c^d \ mod \ n = 3790^{2983} \ mod \ 4247 = 3333$$

Preuve formelle

1. On rappelle :

$$m = c^d \ mod \ n$$

2. Exercice :
   - Démontrez que $c^d \ mod \ n$ permet bien de retrouver le message en clair. On s'aidera de :
     - Des propriétés de l'arithmétique modulaire
     - Du petit théorème de Fermat

Plan

RSA pourquoi ça marche ?

- Attaque à texte chiffré : revient à résoudre le problème `RSA` qui est **difficile**
  - C'est à dire difficile de calculer la solution de manière efficace
  - Problème supposé dans NP
  - S'assurer quand même que n est grand
- Retrouver la clé privée à partir de la clé publique : revient à résoudre le problème `factorisation` qui est **difficile**
  - Opération mathématiquement impossible si n est grand
  - Et heureusement RSA utilise de grand nombres (plus de 1024bits conseillé)
  - Record actuel : 512bits (Anciennes cartes à puces : 320bits !)
  - Combien de temps cela prendrait-il pour un ordinateur à 4Ghz si n fait 1024 bits ?

Confiance dans RSA

- Utilisé depuis 25 ans
  - Quelques défauts mineurs ont été corrigés
- **La confiance dans la sécurité de RSA est calculatoire** : difficulté de factoriser un grand nombre en facteurs premiers
- Mais il n'existe pas de démonstration que RSA ne puisse pas être un jour pris en défaut

Inconvénients

- RSA est très lent
  - 1000 fois plus que DES
  - Clé de grande taille
- Souvent RSA+chiffrement symétrique :
  1. D'abord l'expéditeur d'un message choisit une clé secrète symétrique
  2. Il chiffre son message avec cette clé secrète
  3. Il envoie au destinataire ce message chiffré et ainsi que la clé secrète chiffrée avec la clé publique du destinataire
  4. Le destinataire déchiffre avec sa clé privée la clé secrète chiffrée
  5. Avec le clé secrète déchiffrée, il déchiffre le message

Plan

Le problème Diffie Hellman

- Le problème **Logarithme Discret** :
  - Entrée : un entier premier p, un générateur g de Z_p^* et $y \in Z_p^*$
  - Sortie : en entier e tel que $g^e \ mod \ p = y$
- Fournit une fonction à sens unique, mais pas de brêche secrète
- Le problème **Diffie Hellman** :
  - Entrées :
    - Un entier premier p
    - Un générateur g de Z_p^*
    - Deux entiers $g^a \ mod \ p$ et $g^b \ mod \ p$
  - Sortie : l'entier $g^{ab} \ mod \ p$
- Fonction à sens unique et à brêche secrète (a, b)
- Le cryptosystème Diffie-Hellman est basé sur les problèmes `Logarithme Discret` **et** `Diffie Hellman`

Plan

Diffie Hellman

- Pas un protocole de chiffrement, mais un protocole d'**échange de clé**
- Basé sur les problèmes **Logarithme Discret** et **Diffie Hellman**
- Objectif :
  - Alice et Bob veulent s'échanger une information connue d'eux seuls

Protocole

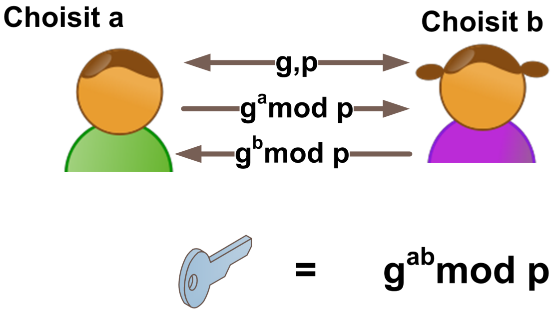
1. Soit p un grand nombre premier et g un générateur de Z_p^*
2. Alice et Bob se mettent d'accord sur p et g
3. Alice choisit un entier a et calcule $g^a \ mod \ p$
4. Alice envoie $g^a \ mod \ p$ à Bob
5. Bob choisit un entier b et calcule $g^b \ mod \ p$
6. Bob envoie $g^b \ mod \ p$ à Alice

Protocole (2)

- Alice calcule $(g^b \ mod \ p)^a modp = g^{ab} \ mod \ p$
- Bob calcule $(g^a \ mod \ p)^b modp = g^{ab} \ mod \ p$
- La clé échangée est :

$$k = g^{ab} \ mod \ p$$

Diffie Hellman



Plan

Diffie Hellman : Pourquoi ça marche ?

- Un attaquant peut observer p, g, $g^b \ mod \ p$ et $g^a \ mod \ p$
- Pour déterminer k il peut :
  - Essayer de déterminer a ou b
    - Problème du Logarithme Discret $\Rightarrow$ **difficle**
  - Essayer de déterminer directement $g^{ab} \ mod \ p$
    - Problème dit de Diffie Hellman $\Rightarrow$ **difficle**
- L'algorithme **El Gamal** est basé sur les mêmes problèmes

Inconvénients

- Comme RSA, très lent
  - Diffie-Hellman+chiffrement symétrique
- Pas d'authentification

Récapitulatif

Récapitulatif :

- Le protocole RSA :
  - Protocole de **chiffrement**
  - Le plus utilisé
  - Repose sur **factorisation** et **RacinelemeModulaire** (difficiles)
- Le protocole Diffie Hellman :
  - Protocole d'**échange de clés**
  - Repose sur le problème `Logarithme Discret` (difficile)
- Bien d'autres protocoles
  - El Gamal
  - Courbes Elliptiques
  - etc

Où est utilisée la cryptographie asymétrique ?

Partout !

- IPSEC
  - Authentification du serveur plus échange de clés : Signature RSA, DSA ..
  - Chiffrement de la communication (AES, TDES, DES ...)
- SSL/TLS
  - Authentification du serveur plus échange de clés : RSA + DH
  - Chiffrement de la communication (AES, TDES, DES ...)
- SSH
  - Authentification du serveur plus échange de clés : DH
  - Authentification du client (facultatif)
  - Chiffrement de la communication (AES, TDES, DES ...)

Où est utilisée la cryptographie asymétrique ?

- Client mail
  - PGP, Outlook
  - Signature des mails : RSA, DSA
  - Chiffrement des mails RSA + AES, TDES, DES ...
- Essayez !
  - GPG : GNU Privacy Guard
  - Plugin Thunderbird : Enigmail

Conclusion

- La cryptographie est un outil essentiel de la politique de sécurité de l'entreprise
  - **Confidentialité**
  - **Intégrité**
  - **Authenticité**
- Cryptographie à clé secrète
  - Rapide, mais comment s'échanger la clé
- Cryptographie à clé publique
  - Plus lente, mais plus pratique
  - Permet notamment d'authentifier grâce aux **signatures**