D31: Protocoles Cryptographiques

Cryptographie à clé publique

Nicolas Méloni

Master 2: 1er semestre
(2014/2015)

Prérequis et objectifs

Objectifs

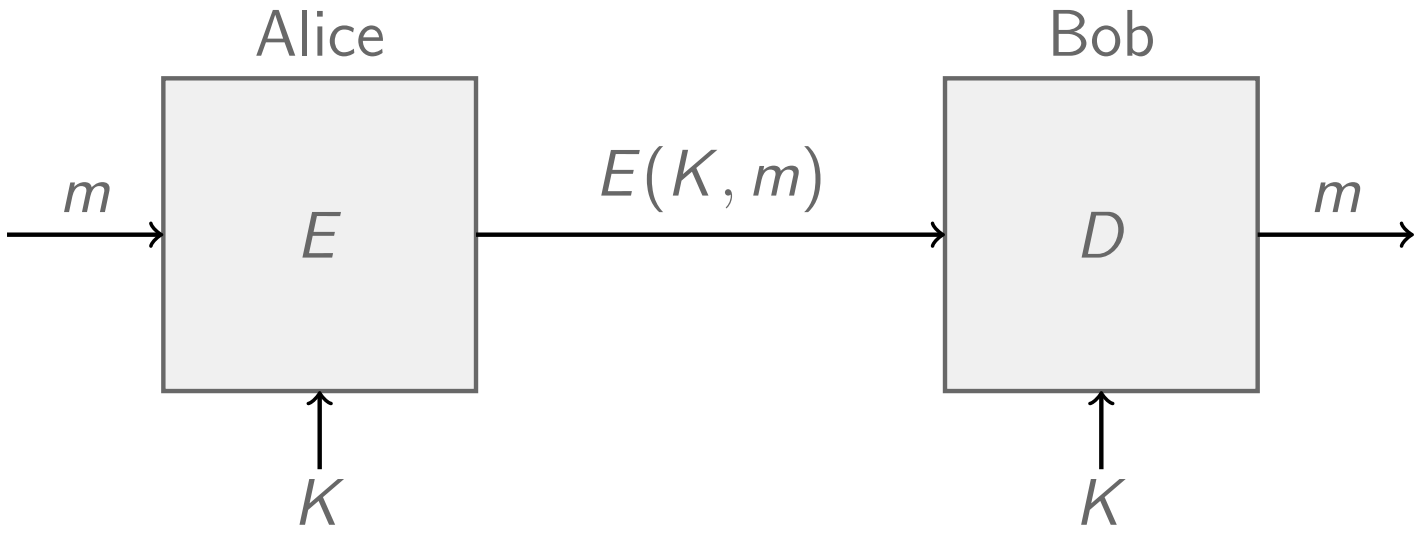
- Comprendre la différence entre crypto à clé secrête et à clé publique
- Découvrir les principales primitives cryptographiques utilisées
- Comprendre la notion de sécurité cryptographique
- Se familiariser avec quelques protocoles classiques

Prérequis

- Le cours de P. Véron
- Un peu de mathématiques (calcul modulaire, algo d'Euclide étendu etc)
- Des restes du cours de compléxité (D22)

Notion de cryptographie à clé publique

- Schéma de chiffrement à clé privée:



La même clé est utilisée pour le chiffrement et le déchiffrement.

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

Notion de cryptographie à clé publique

Problème

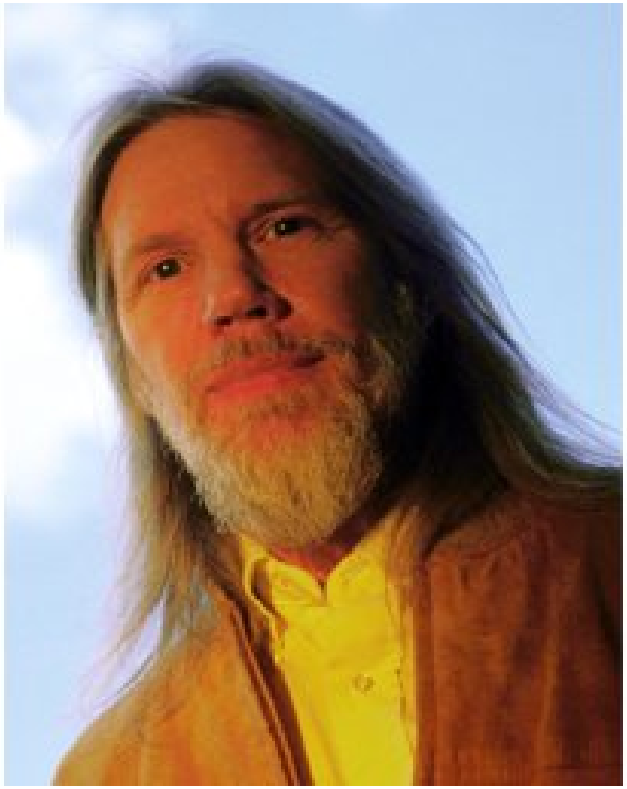
Comment transmettre la clé de chiffrement?

Problème difficile notament si:

- les participants sont éloignés physiquement,
- la clé doit être changée fréquemment,
- le nombre de participants augmentent.

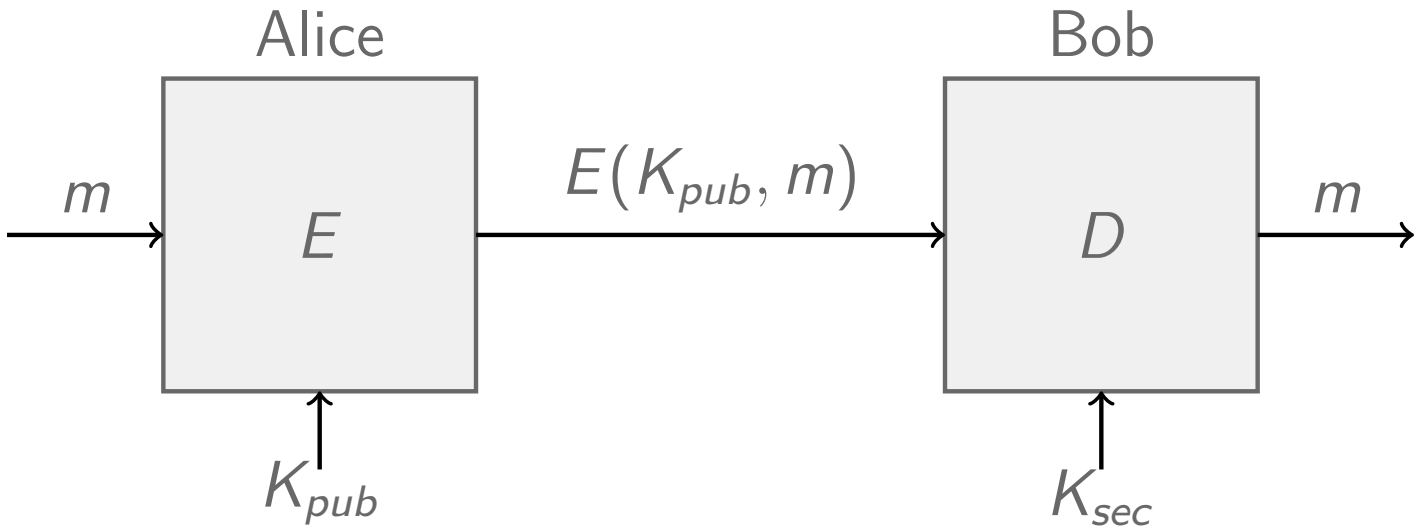
Notion de cryptographie à clé publique

- Solution théorique proposée en 1976 par Whitfield Diffie et Martin E. Hellman.



Notion de cryptographie à clé publique

- Bob crée deux clés: une publique qu'il donne à Alice et une privée qu'il garde secrête.



- Il doit être calculatoirement impossible d'obtenir K_{sec} à partir de K_{pub}.

Définitions

Cryptosystème à clé publique (PKC en anglais)

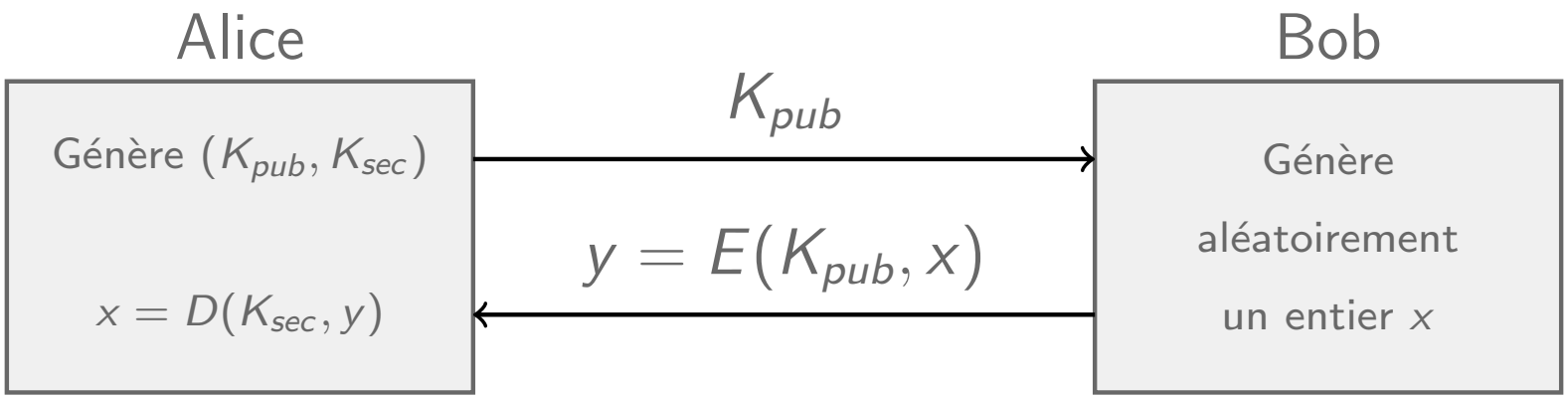
Soit $\mathcal{M}$ l'ensemble des messages et $\mathcal{C}$ l'ensemble des chiffrés. Un cryptosystème à clé publique est un triplet (G, E, D) où:

- G est un algorithme générant les couples de clés (K_{pub}, K_{sec}),
- $E(K_{pub}, .)$ est un algorithme prenant un message $m \in \mathcal{M}$ en entrée et renvoyant un chiffré $c \in \mathcal{C}$ en sortie,
- $D(K_{sec}, .)$ est un algorithme prenant un chiffré $c \in \mathcal{C}$ en entrée et renvoyant un message $m \in \mathcal{M}$ en sortie,
- $\forall (K_{pub}, K_{sec})$ généré par G

$$D(K_{sec}, E(K_{pub}, m)) = m.$$

Un premier exemple

- Mise en place d'une clé de session:



Alice et Bob peuvent maintenant utiliser x comme secret commun.

Sécurité des systèmes à clé publique

- Objectifs de l'attaquant:

Cassage total

L'attaquant est capable de retrouver la clé privé du système.

Cassage partiel

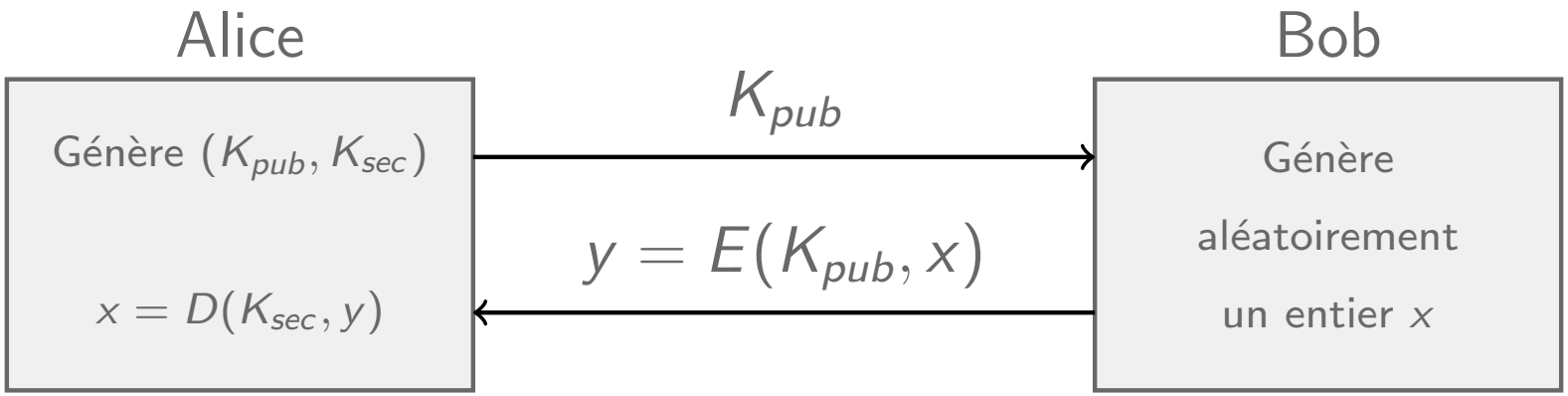
L'attaquant est capable, avec probabilité non négligeable, de déchiffrer un message chiffré qu'il n'a jamais vu.

Distingabilité des chiffrés

L'attaquant est capable, avec probabilité supérieure à $1/2$, de distinguer les chiffrés de deux messages clairs donnés.

Adversaire passif

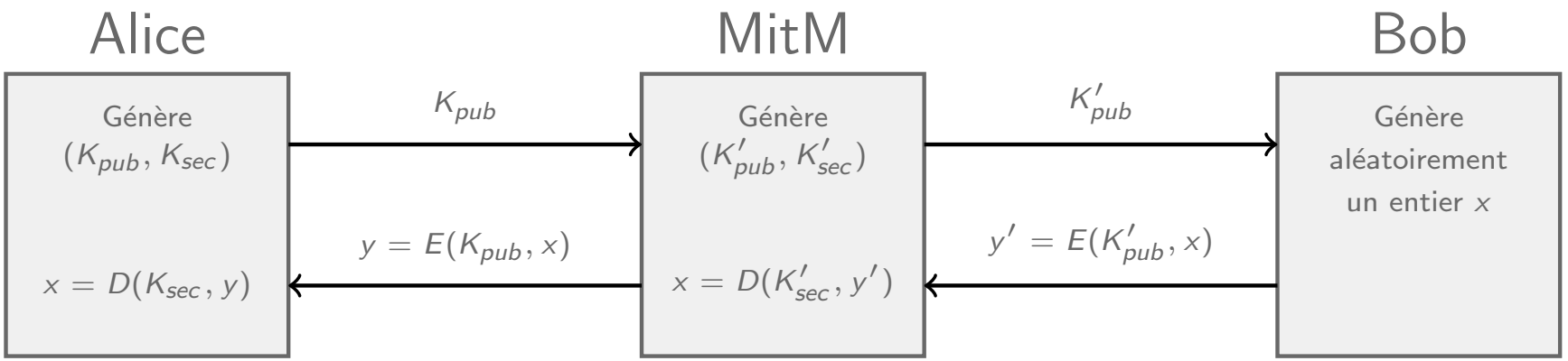
- Mise en place d'une clé de session:



- Sémantiquement sûr contre un adversaire passif
- Vulnérable à l'attaque *man in the middle*

Adversaire actif

- L'attaque Man-in-the-Middle:



- Pour Alice et Bob le protocole s'est déroulé normalement
- L'attaquant peut maintenant déchiffrer tous les messages qu'ils s'envoient

Le cryptosystème Merkle-Hellman

- Cryptosystème obsolète
- Repose sur le problème $\mathcal{NP}$-complet de la somme de sous-ensemble (SSP).

Problème SSP

- **Instance:** k un entier et S un ensemble de n entiers relatifs.
- **Question:** Existe-t-il un sous ensemble $S' \subset S$ sont la somme des éléments vaut k?

Le cryptosystème Merkle-Hellman

Suite super-croissante

On dit qu'une suite d'entier (a_n) est super-croissante si chaque élément est plus grand que la somme des éléments qui le précèdent, i.e.

$$\forall n \geq 0,\ a_n \geq \sum_{i=0}^{n-1} a_i.$$

- Si l'ensemble S peut être ordonné en une suite super-croissante, alors le problème SSP devient facile.

Le cryptosystème Merkle-Hellman

Idée du cryptosystème:

- Transformer un problème SSP facile en un problème SSP difficile
- La transformation servira de clé secrête, le problème difficile de clé publique.

Génération de clés:

1. générer $A = \{a_1, \ldots, a_n\}$ une suite super-croisssante,
2. générer $p > \sum_{i=1}^{n} a_i$ et q premier avec p,
3. calculer $B = \{b_1, \ldots, b_n\}$ où $b_i \equiv a_i q \mod p$,
4. B est la clé publique, (A, p, q) est la clé privée.

Le cryptosystème Merkle-Hellman

Chiffrement:

Soit $m = m_1 \dots m_n$ un message de n bits:

1. calculer $c = \sum_{i=1}^{n} m_i b_i$,
2. retourner le chiffré c.

Déchiffrement:

Soit c un chiffré:

1. calculer $c' \equiv cq^{-1} \mod p$,
2. résoudre le problème SSP $c' = \sum_{i=1}^{n} m_i a_i$,
3. renvoyé le message $m = m_1 \dots m_n$.