IUT ANNECY
UNIVERSITÉ SAVOIE MONT BLANC

Cryptographie et sécurité

Cours 2: cryptographie asymétrique

Mickaël Bettinelli
(mickael.bettinelli@univ-smb.fr)

Prérequis et objectifs

Compétences nécessaire pour ce cours:

- Savoir utiliser l'opérateur modulo
- Effectuer des opérations sur $\mathbb{Z}/n\mathbb{Z}$ (exponentiations !)
- Décrire le fonctionnement du chiffrement symétrique

Compétences maîtrisées à la fin du cours:

- Utiliser l'algorithme d'Euclide étendu
- Nommer le théorème de Bézout
- Comprendre le fonctionnement du chiffrement asymétrique

Sommaire

- Echange de clés privées: Diffie-Hellman
- Les services
  - Confidentialité
  - Authentification
  - Intégrité
- Vue d'ensemble du chiffrement asymétrique
- Arithmétique modulaire
  - PGCD
  - Les nombres premiers
  - Bézout
  - Algorithme d'Euclide étendu
- RSA
- Conclusion

Cours interactif

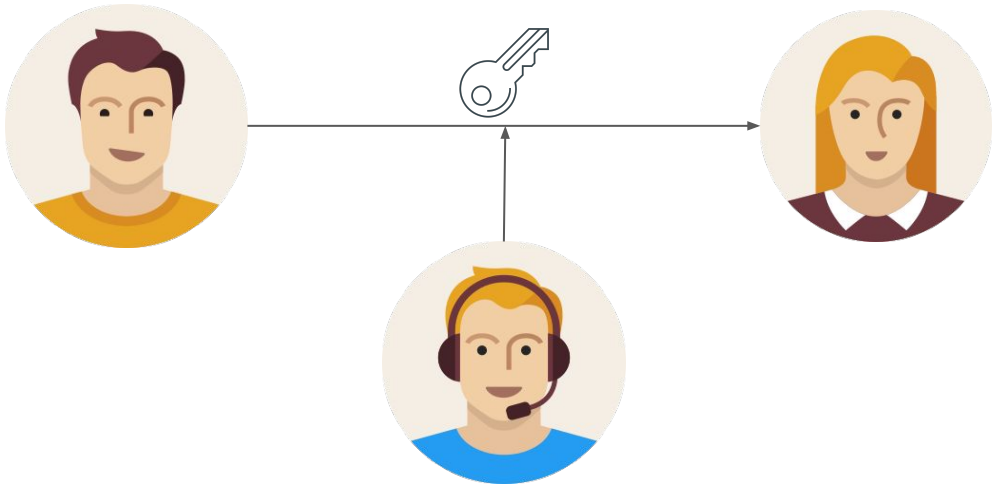
1. Connectez vous au quizz
2. Répondez aux questions
3. Chaque bonne réponse vous rapporte des points
4. Un classement des participants est effectué à la fin du questionnaire
5. Le premier du classement gagne ½ point en plus sur sa moyenne de crypto :)

Connectez vous avec votre téléphone sur:

https://quizizz.com/

Game code:

Problème des chiffrements symétriques



Bob souhaite envoyer la clé qu'il a utilisé pour chiffrer un message.

Problème: si Alice la reçoit, l'homme du milieu aussi

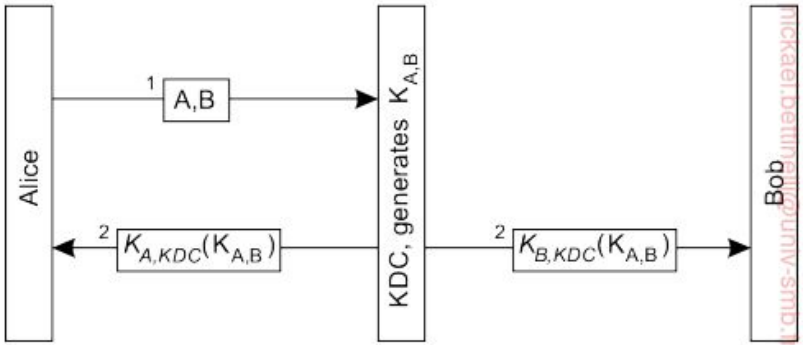
Mais comment transmettre la clé sans être écouté ?

1/ Key Distribution Center (KDC)

Un serveur central génère des clés partagées pour tout le monde.

Pré-requis: tout le monde partage une clé unique et secrète avec le KDC notée $K_{A, KDC}$.

1. Alice demande une clé au serveur
2. Le serveur envoie une clé à Alice encryptée avec la clé $K_{A, KDC}$
3. Le serveur envoie une clé à Bob encryptée avec la clé $K_{B, KDC}$



2/ Diffie-Hellman - introduction

Comment échanger des clés sur un réseau de manière sécurisée ?

- La méthode Diffie-Hellman génère des clés privées identiques chez plusieurs participants à la conversation

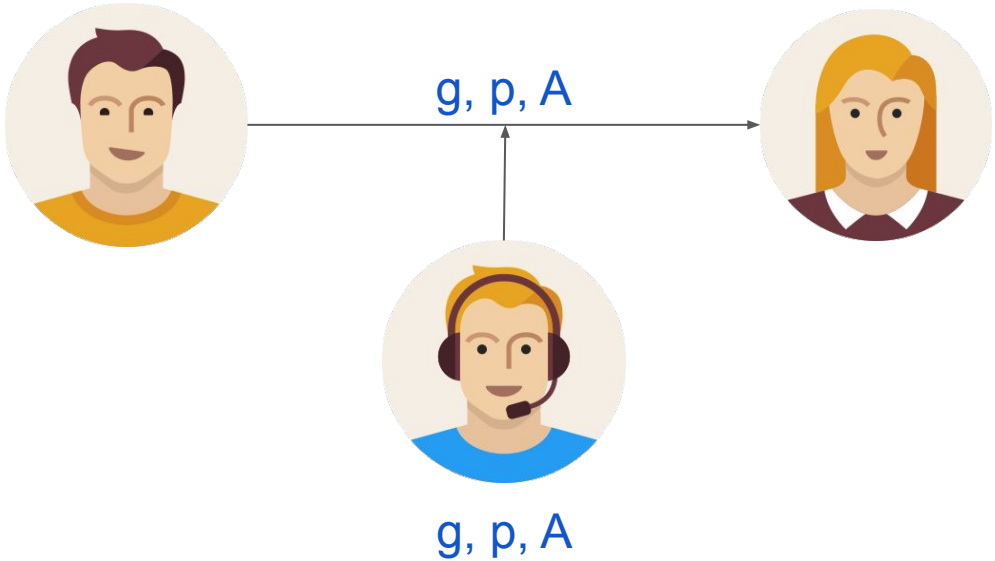
Qu’est ce que Diffie-Hellman ?

- Un algorithme de cryptographie inventé en 1976
- Récompensé par un prix Turing en 2015 (équivalent au prix Nobel pour l’informatique)

Diffie-Hellman - fonctionnement

a, g, p

$A = g^a \pmod p$

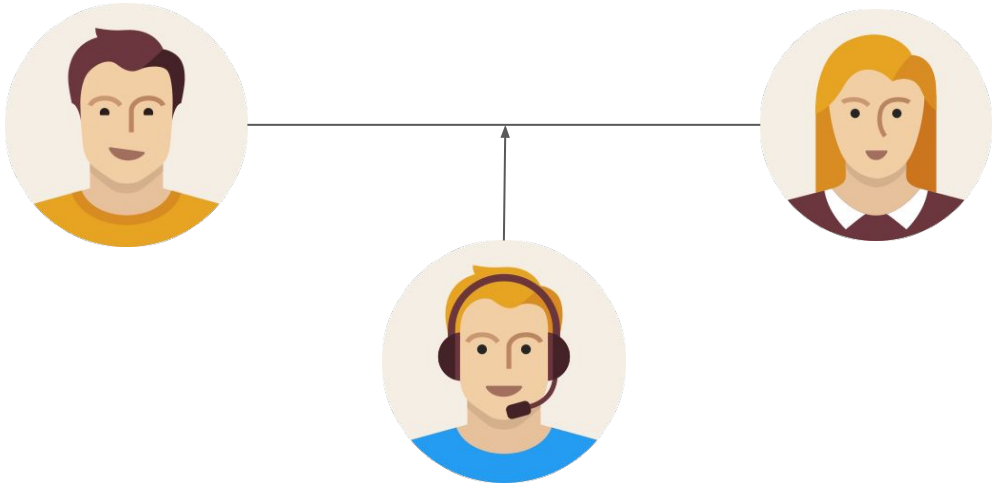


b

Diffie-Hellman - fonctionnement

g, p, a

$A = g^a \pmod p$



g, p, b

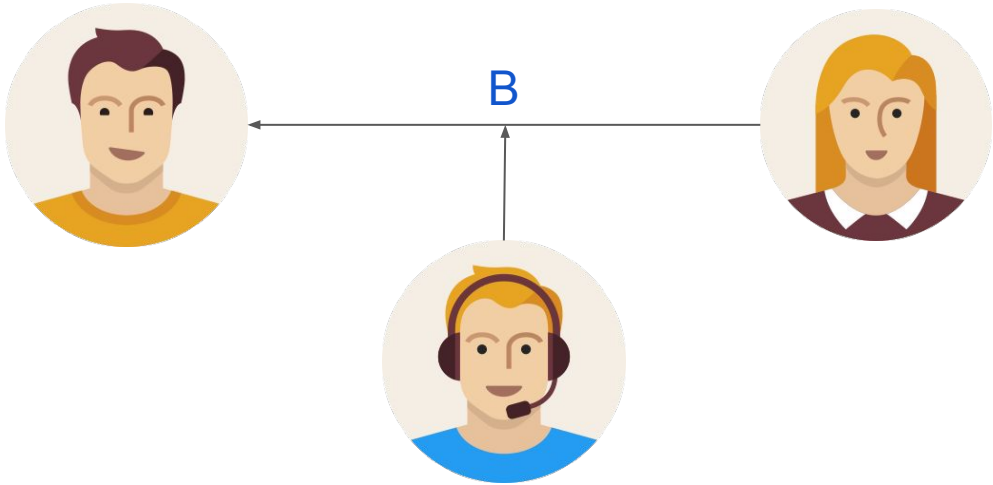
$B = g^b \pmod p$

g, p, A

Diffie-Hellman - fonctionnement

g, p, a

$A = g^a \pmod p$



g, p, b

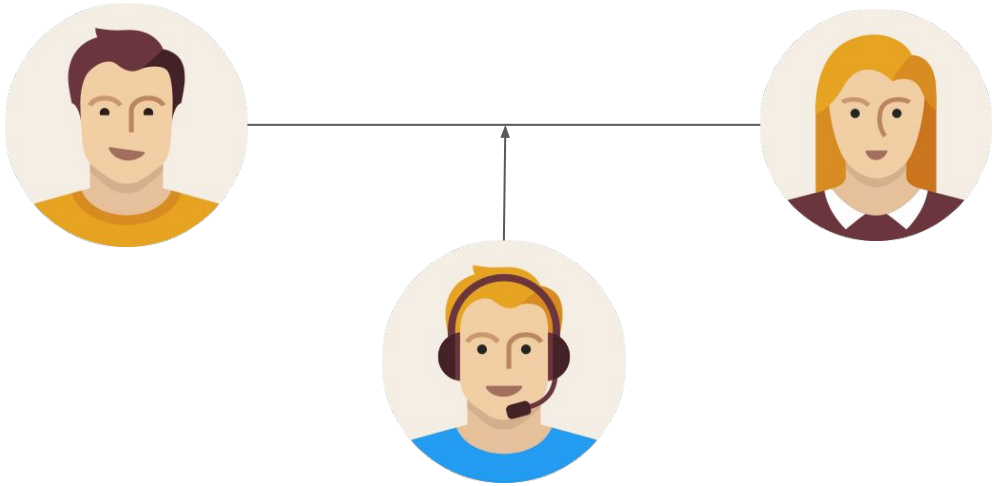
$B = g^b \pmod p$

Diffie-Hellman - fonctionnement

g, p, a

$A = g^a \pmod p$

$K = B^a \pmod p$



g, p, A, B

g, p, b

$B = g^b \pmod p$

$K = A^b \pmod p$

La clé K de Bob est identique à la clé K de Alice !

Diffie-Hellman - fonctionnement

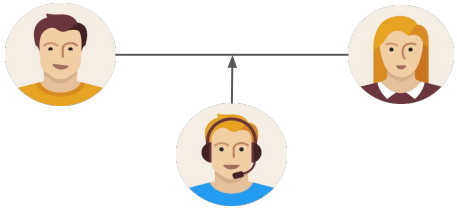
Pourquoi est-ce que ça fonctionne ?

- Alice et Bob calculent tous les deux la clé: **$K = g^{ab} \pmod p$**
- L'homme du milieu ne connaît que **$g^a \pmod p$ et $g^b \pmod p$** mais pas n'est pas capable de retrouver **g**, **a** et **b** (coût computationnel trop élevé)

g, p, a

$A = g^a \pmod p$

$K = B^a \pmod p$



g, p, b

$B = g^b \pmod p$

$K = A^b \pmod p$

g, p, A, B

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$
   a. $4^2 = 16 \pmod 5 = 1$

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$
   a. $4^2 = 16 \pmod 5 = 1$
   b. $1 * 4 = 4 \pmod 5 = 4$

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$
   a. $4^2 = 16 \pmod 5 = 1$
   b. $1 * 4 = 4 \pmod 5 = 4$
   c. A = 4

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$
   a. $4^2 = 16 \pmod 5 = 1$
   b. $1 * 4 = 4 \pmod 5 = 4$
   c. A = 4
3. Alice choisit un nombre aléatoire *b*

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$
   a. $4^2 = 16 \pmod 5 = 1$
   b. $1 * 4 = 4 \pmod 5 = 4$
   c. A = 4
3. Alice choisit un nombre aléatoire *b*
4. Alice envoit **$B = g^b \pmod p$** à Bob: **$B = 4^2 \pmod 5 = 1$**

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$
   a. $4^2 = 16 \pmod 5 = 1$
   b. $1 * 4 = 4 \pmod 5 = 4$
   c. A = 4
3. Alice choisit un nombre aléatoire *b*
4. Alice envoit **$B = g^b \pmod p$** à Bob: **$B = 4^2 \pmod 5 = 1$**
5. Elle calcule sa clé: **$K = A^b \pmod p = 4^2 \pmod 5 = 1$**

Exemple

Choix des paramètres:

p = 5 (doit être un nombre premier)

g = 4 (doit être plus petit que *p*)

a = 3

b = 2

1. Bob choisit un nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice: $4^3 \pmod 5$
   a. $4^2 = 16 \pmod 5 = 1$
   b. $1 * 4 = 4 \pmod 5 = 4$
   c. $A = 4$
3. Alice choisit un nombre aléatoire *b*
4. Alice envoit **$B = g^b \pmod p$** à Bob: **$B = 4^2 \pmod 5 = 1$**
5. Elle calcule sa clé: **$K = A^b \pmod p = 4^2 \pmod 5 = 1$**
6. Bob reçoit B et calcule sa clé: **$K = B^a \pmod p = 1^2 \pmod 5 = 1$**

La clé de Alice et Bob vaut 1

Diffie-Hellman - à vous

Le choix des paramètres est sur la fiche du quizz.

Les étapes de Diffie-Hellman:

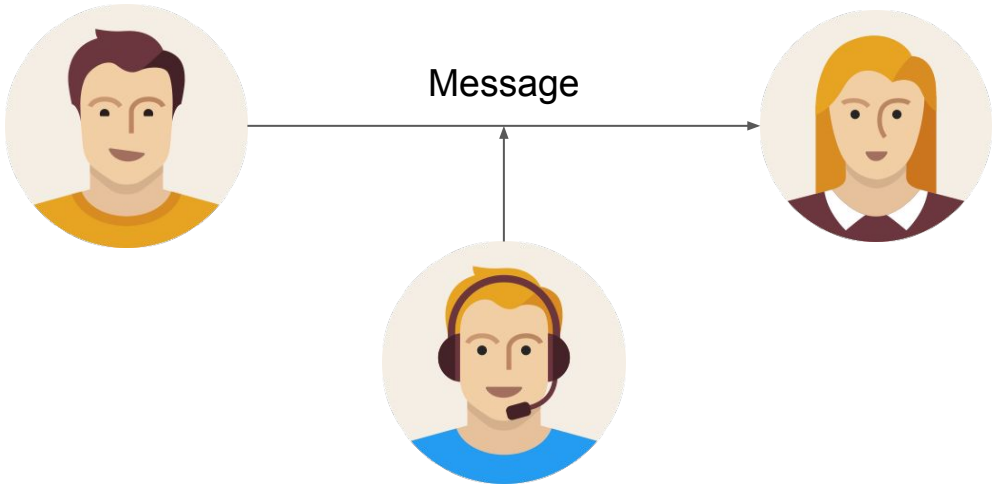
1. Bob choisit le nombre aléatoire *a*
2. Il envoit **$A = g^a \pmod p$** à Alice
3. Alice choisit le nombre aléatoire *b*
4. Alice envoit **$B = g^b \pmod p$** à Bob
5. Elle calcule sa clé: **$K = A^b \pmod p$**
6. Bob reçoit B et calcule sa clé: **$K = B^a \pmod p$**

Diffie-Hellman - attention

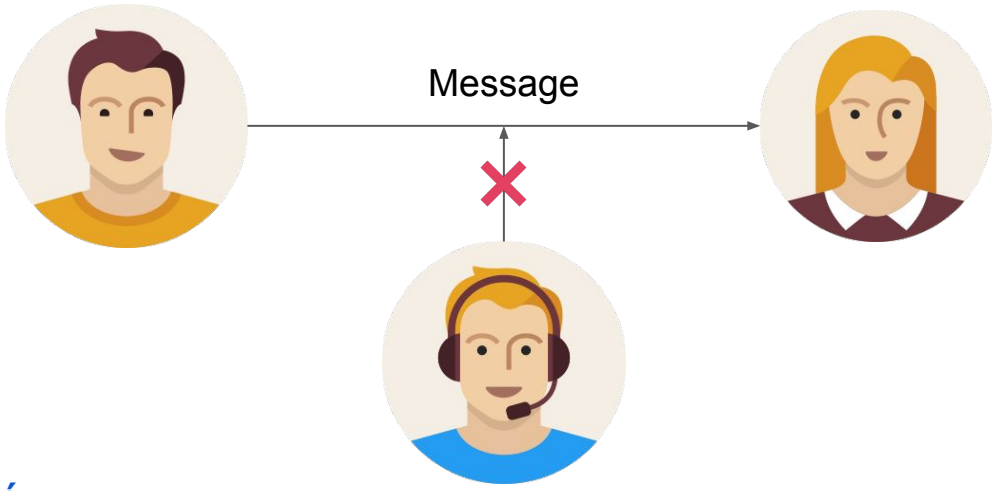
Ce que Diffie-Hellman ne fait pas:

- ne garantit pas l’authentification des intervenants
- ne garantit pas la “non-répudiation”
- ne garantit pas non plus l’intégrité des messages

Le chiffrement asymétrique: services souhaités

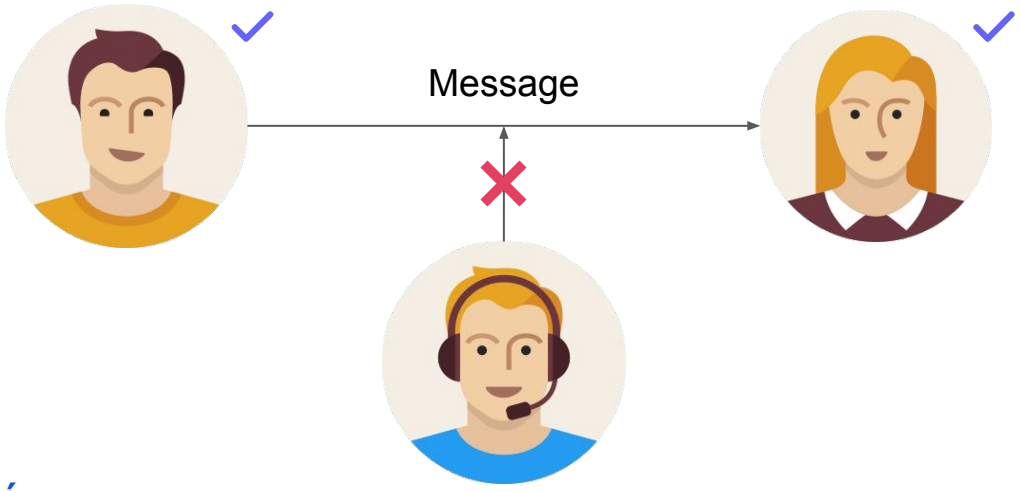


Le chiffrement asymétrique: services souhaités



Confidentialité

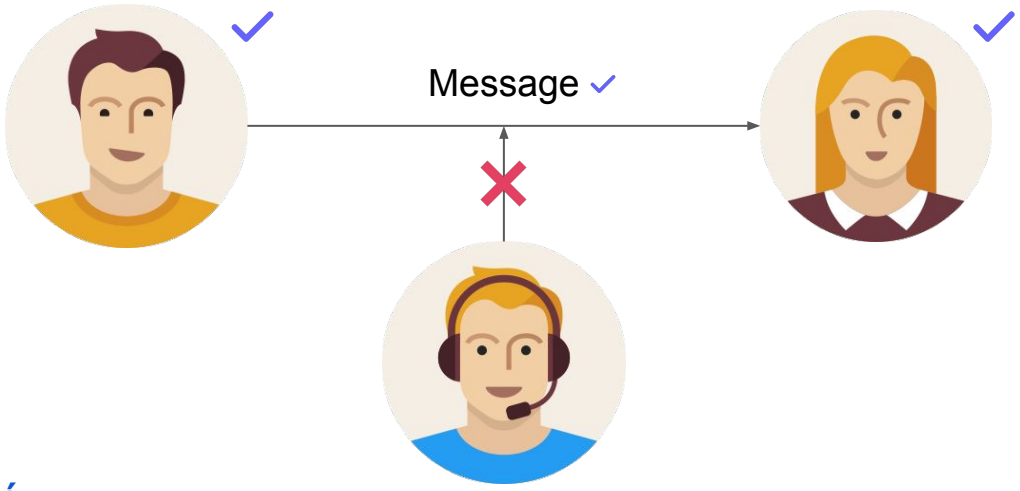
Le chiffrement asymétrique: services souhaités



Confidentialité

Authentification

Le chiffrement asymétrique: services souhaités

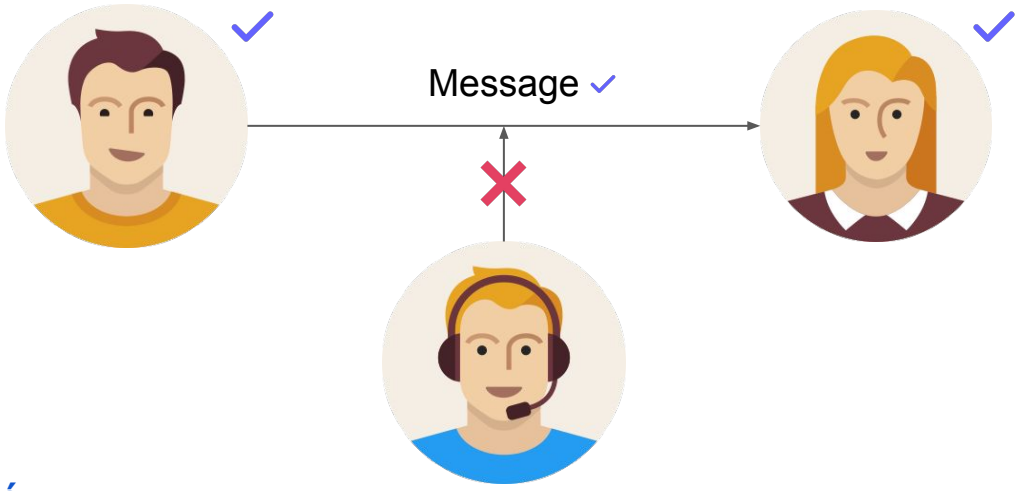


Confidentialité

Authentification

Intégrité

Le chiffrement asymétrique: services souhaités



Confidentialité

Authentification

Intégrité

Non-répudiation

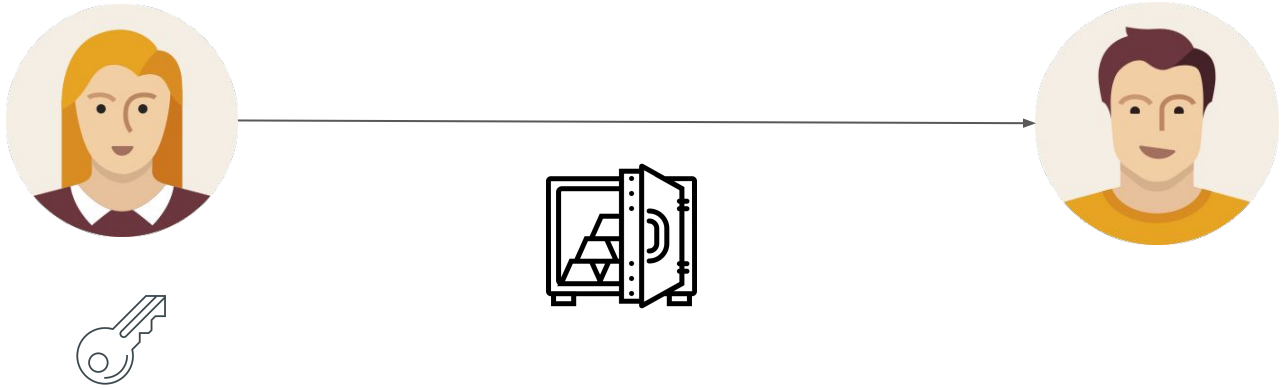
Vue d’ensemble

Vulgarisation de la confidentialité

- Bob veut communiquer de manière confidentielle avec Alice
- Alice et Bob ont à disposition un coffre fort et une clef pour sécuriser le message
- On suppose ce coffre inviolable et la clé impossible à copier

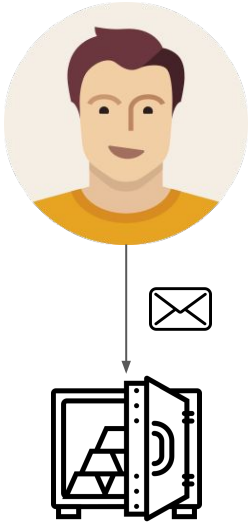
Vue d’ensemble - confidentialité

1) Alice envoie le coffre fort à Bob. Elle garde la clé pour elle



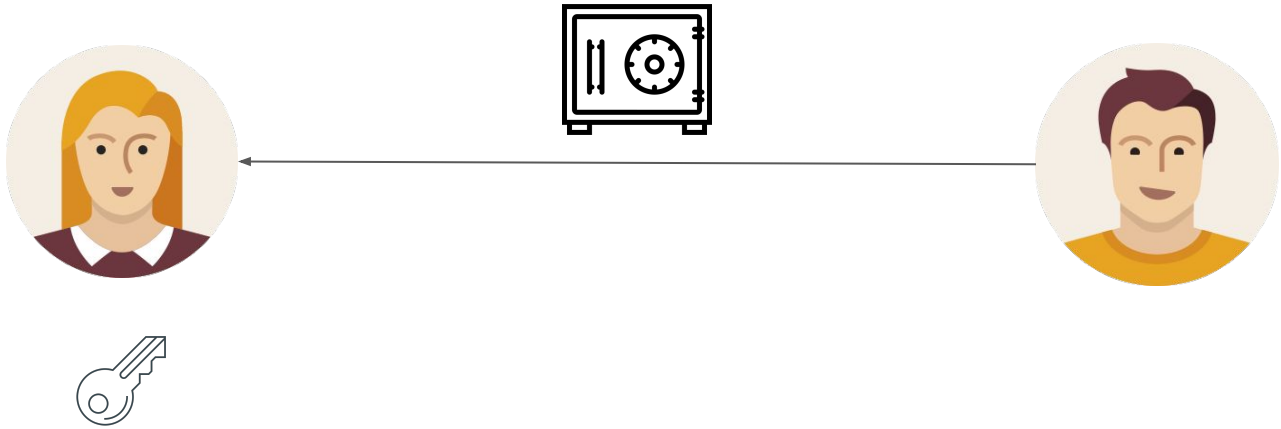
Vue d’ensemble - confidentialité

2) Bob dépose son message dans le coffre



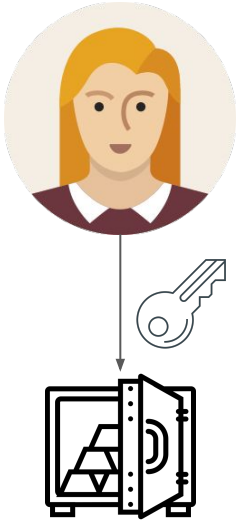
Vue d’ensemble - confidentialité

3) Bob ferme le coffre et l’envoie à Alice



Vue d’ensemble - confidentialité

4) Alice ouvre le coffre avec sa clé

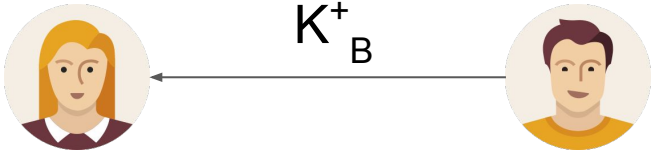


Vue d’ensemble - confidentialité

Du point de vue technique

Les termes:

- K^{+}_{a} : clé publique de Alice→le coffre fort
- K^{-}_{a} : clé privée de Alice→la clé du coffre
- m : message
- $K^{-}_{a}(m)$: de/encrytion de m avec la clé privée de Alice

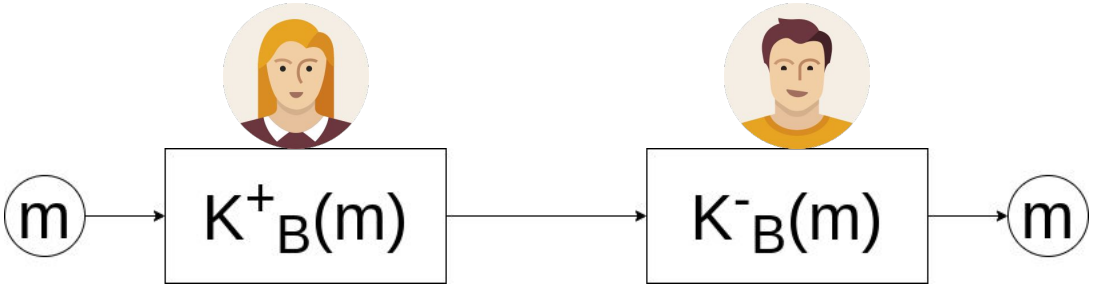


Vue d’ensemble - confidentialité

Du point de vue technique

Les termes:

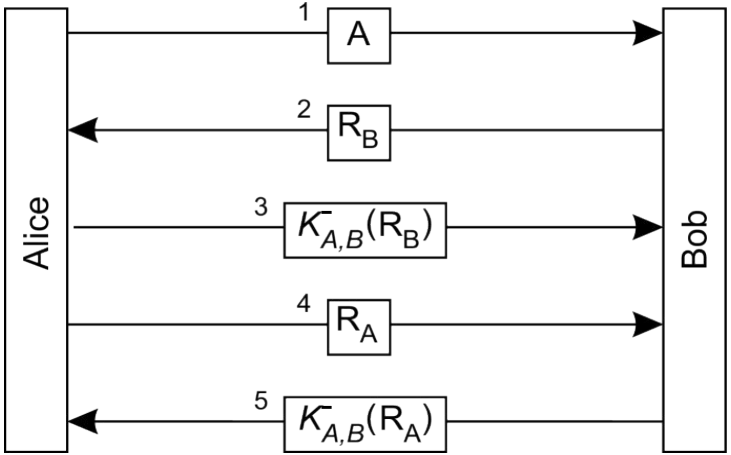
- K^{+}_{a} : clé publique de Alice→le coffre fort
- K^{-}_{a} : clé privée de Alice→la clé du coffre
- m : message
- $K^{-}_{a}(m)$: de/encrytion de m avec la clé privée de Alice



Vue d’ensemble - authentification

Bob attend un message de Alice, comment être sûr que le message vient bien de Alice ?

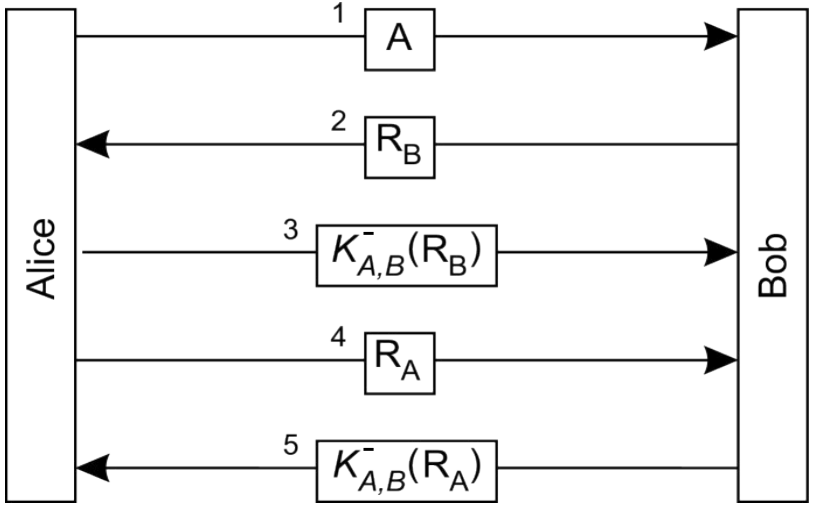
Le protocole défi-réponse permet l’authentification des interlocuteurs.



Vue d'ensemble - authentification

1. Alice envoie à Bob son identité
2. Bob envoie un défi à Alice (un message à chiffrer avec sa clé)
3. **Alice chiffre le message** avec sa clé privée et le renvoie à Bob
4. **Bob** vérifie que le message chiffré est correct en le **déchiffrant avec la clé publique de Alice**
5. Alice envoie un défi à Bob
6. Bob le chiffre et le renvoie à Alice
7. Alice vérifie le message chiffré

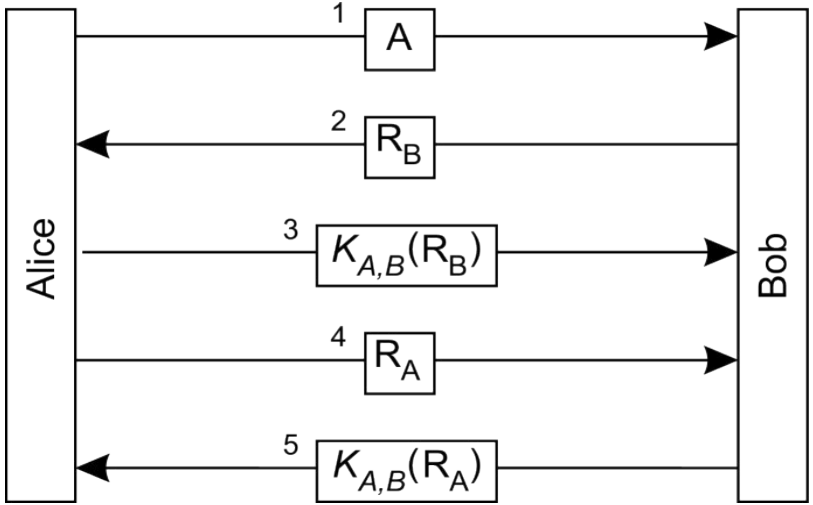
Si les messages chiffrés sont corrects, Alice et Bob savent qu'ils parlent à la bonne personne.



Vue d'ensemble - authentification

1. Alice envoie à Bob son identité
2. Bob envoie un défi à Alice (un message à chiffrer avec sa clé)
3. **Alice chiffre le message** avec sa clé privée et le renvoie à Bob
4. **Bob** vérifie que le message chiffré est correct en le **déchiffrant avec la clé publique de Alice**
5. Alice envoie un défi à Bob
6. Bob le chiffre et le renvoie à Alice
7. Alice vérifie le message chiffré

Cet algorithme peut être adapté avec Diffie-Hellman



Vue d’ensemble - intégrité, non-répudiation

Pourquoi l’intégrité ?

Exemple. Bob souhaite vendre un téléphone à Alice. Alice lui dit qu’elle s’engage à l’acheter pour 500€.

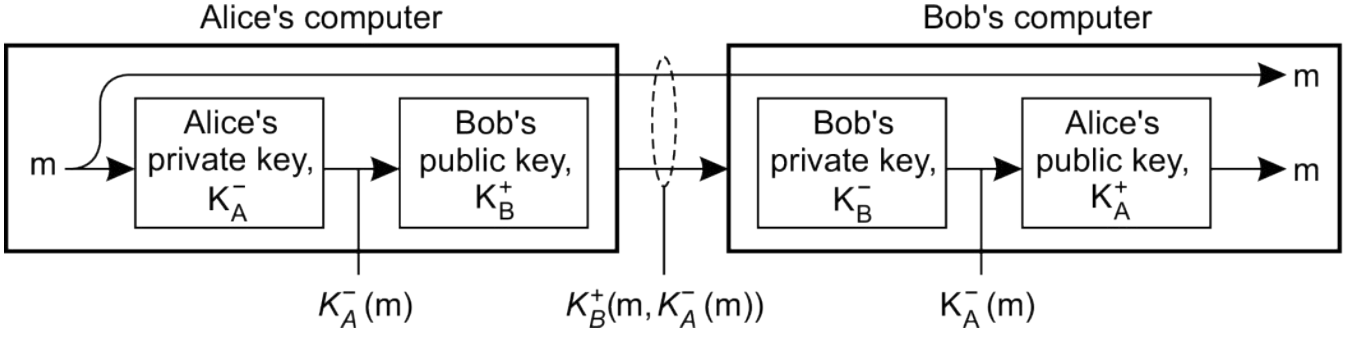
1. Alice veut être certaine que Bob ne puisse pas changer le 500€ en 800 €.
2. Bob ne veut pas qu’Alice nie avoir envoyé le message pour se désengager de la transaction.

Vue d’ensemble - intégrité

Comment vérifier que le message n’a pas été modifié ?

1. Alice signe (chiffre) son message avec sa clé privée
2. Elle envoie le message à Bob avec une copie non chiffrée du message
3. Bob déchiffre le message avec la clé publique de Alice
4. Bob compare les deux messages

- Bob ne peut modifier le message signé, car il ne pourra pas prouver que la version modifiée est également signée par Alice
- Alice ne peut nier avoir envoyé le message parce que sa clé publique le déchiffre



Comprendre les chiffrements asymétriques

Pour mieux comprendre le fonctionnement du chiffrement asymétrique nous allons :

- Faire un second tour dans l'arithmétique modulaire
- Voir dans le détail le fonctionnement de l'algorithme RSA
- S'exercer sur des exemples simple pour mieux appréhender les nouveaux concepts

Rappel - PGCD

PGCD = Plus grand dénominateur commun

Exemple: pgcd(15, 26)

Quels sont les diviseurs de 15 ?

- 5 * 3 = 15
- 3 * 5 = 15
- 1 * 15 = 15

Diviseurs de 15 = {1, 3, 15}

Diviseurs de 26 = {1, 2, 13, 26}

=> pgcd(15, 26) = 1

Nombres premiers

Définition. Un nombre premier est un entier naturel qui admet exactement deux diviseurs distincts entiers et positifs: 1 et lui-même.

Quelques exemples: {1, 5, 7, 11, 17, ...}

Nombres premiers entre eux

Définition. Deux nombres entiers sont dits premiers entre eux lorsqu'il n'admette aucun diviseur commun, sinon 1.

Deux nombres sont premiers entre eux si pgcd(a, b) = 1.

Quelques exemples:

- 15 et 26 sont premiers entre eux
- Est-ce que 38 et 89 sont premiers entre eux ?

Nombres premiers entre eux

Définition. Deux nombres entiers sont dits premiers entre eux lorsqu'il n'admette aucun diviseur commun, sinon 1.

Deux nombres sont premiers entre eux si pgcd(a, b) = 1.

Exemples:

- 15 et 26 sont premiers entre eux

Comment savoir lorsque l’on manipule de grands nombres ?

Nombres premiers entre eux

Algorithme d'Euclide. Permet de calculer le pgcd de deux nombres. Si le pgcd est de 1, alors les nombres sont premiers entre eux.

Principe. **L'algorithme d'Euclide** procède comme suit avec deux nombres entiers positifs a et b avec $a > b \geq 0$:

- si $r = 0$, l'algorithme termine et rend la valeur b
- sinon, l'algorithme calcule le reste r de la division euclidienne de a par b, puis recommence avec $a = b$ et $b = r$

Nombres premiers entre eux

Algorithme d'Euclide. Permet de calculer le pgcd de deux nombres. Si le pgcd est de 1, alors les nombres sont premiers entre eux.

Exemple. Avec $a = 39$ et $b = 16$

$39 = 2 * 16 + 7$

$16 = 2 * 7 + 2$

$7 = 3 * 2 + \mathbf{1}$

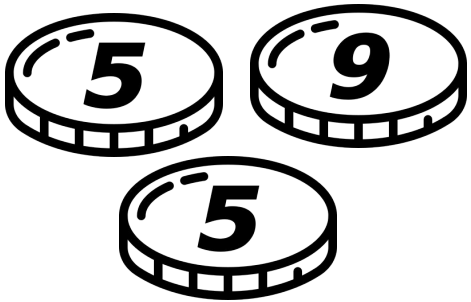
$2 = 2 * 1 + 0$

Le pgcd est le dernier reste non plus: **1**

39 et 16 sont premiers entre eux.

Théorème de Bachet-Bézout

Quelles sommes peut-on dépenser avec des pièces de 5€ et de 9€ ?



- 5€, 10€, … tous les multiples de 5
- 9€, 18€, … tous les multiples de 9

Mais on peut aussi donner 2 pièces de 5€ et rendre une pièce de 9€.

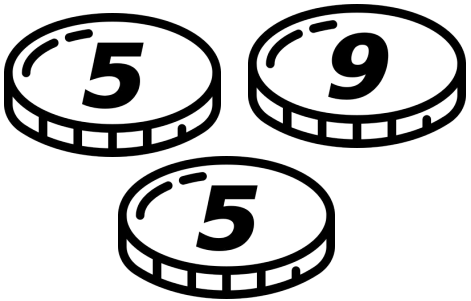
=> on donne 1€

Théorème de Bachet-Bézout

Mais on peut aussi donner 2 pièces de 5€ et rendre une pièce de 9€.

=> on donne 1€

Si on peut donner 1€, on peut aller donner n'importe quelle somme en répétant le processus.



Est-il possible de payer n'importe quelle somme avec des pièces de valeurs différentes ?

De 6€ et 18€ par exemple ?

Oui si les valeurs sont premières entre elles !

Théorème de Bachet-Bézout

Théorème (de Bézout). Soient a et b deux entiers relatifs. Si d est le PGCD de a et b, alors il existe deux entiers relatifs x et y tels que $ax + by = d$.

Exemple. Nous avons deux pièces de 7€ et 12€, est-il possible de payer n'importe quelle somme ?

Théorème de Bachet-Bézout

Théorème (de Bézout). Soient a et b deux entiers relatifs, on a a et b sont premiers entre eux $\Leftrightarrow \exists\ (x, y) \in \mathbb{Z}^2$, $xa + yb = 1$.

Exemple. Nous avons deux pièces de 7€ et 12€, est-il possible de payer n'importe quelle somme ?

$12 = 1 * 7 + 5$

$7 = 1 * 5 + 2$

$5 = 2 * 2 + \mathbf{1}$

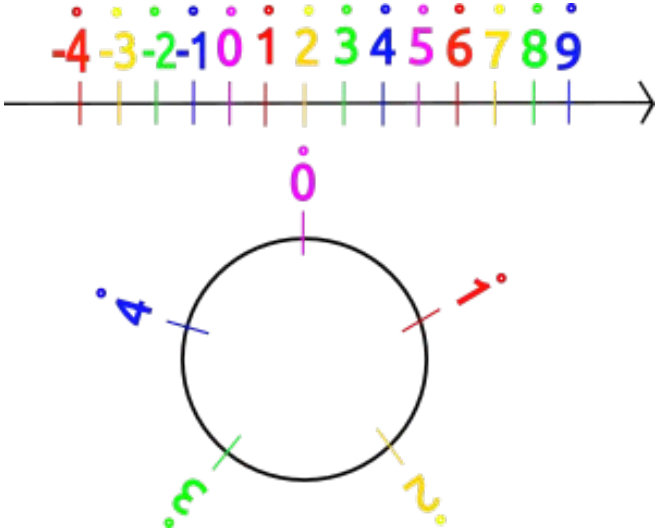
$2 = 2 * 1 + 0$

Le pgcd de 7 et 12 est 1, ils sont premiers entre eux. Il est possible de faire toutes les sommes avec des pièces de 7€ et 12€.

Les éléments inversibles

Soit a un élément de $\mathbb{Z}/n\mathbb{Z}$. On dit que a est inversible ssi il existe b $\in \mathbb{Z}/n\mathbb{Z}$ tel que $a \times b = \dot{1}$. On appelle $\dot{b}$ l'inverse de $\dot{a}$ et on le note $\dot{a}^{-1}$.

L'ensemble des éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$ est noté $(\mathbb{Z}/n\mathbb{Z})^x$



Exemple sur $\mathbb{Z}/20\mathbb{Z}$:

- Est-ce que $\dot{6}$ est l'inverse de $\dot{5}$?
- Est-ce que $\dot{3}$ est l'inverse de $\dot{7}$?

Reconnaître les éléments inversibles

Tous les éléments de $\mathbb{Z}/n\mathbb{Z}$ ont un opposé, mais pas forcément un inverse.

Soit a un élément de $\mathbb{Z}/n\mathbb{Z}$. On dit que a est inversible ssi il existe b $\in \mathbb{Z}/n\mathbb{Z}$ tel que $a \times b = \dot{1}$. On appelle b l'inverse de a et on le note a^{-1}.

L'ensemble des éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$ est noté $(\mathbb{Z}/n\mathbb{Z})^x$

Exemple sur $\mathbb{Z}/20\mathbb{Z}$:

- Est-ce que $\dot{6}$ est l'inverse de $\dot{5}$? $\dot{6} * \dot{5} = \dot{15}$ —> Non
- Est-ce que $\dot{3}$ est l'inverse de $\dot{7}$?

Reconnaître les éléments inversibles

Tous les éléments de $\mathbb{Z}/n\mathbb{Z}$ ont un opposé, mais pas forcément un inverse.

Soit a un élément de $\mathbb{Z}/n\mathbb{Z}$. On dit que a est inversible ssi il existe b $\in \mathbb{Z}/n\mathbb{Z}$ tel que $a \times b = \dot{1}$. On appelle b l'inverse de $\dot{a}$ et on le note $\dot{a}^{-1}$.

L'ensemble des éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$ est noté $(\mathbb{Z}/n\mathbb{Z})^x$

Exemple sur $\mathbb{Z}/20\mathbb{Z}$:

- Est-ce que $\dot{6}$ est l'inverse de $\dot{5}$? $\dot{6} * \dot{5} = \dot{15}$ —> Non
- Est-ce que $\dot{3}$ est l'inverse de $\dot{7}$? $\dot{3} * \dot{7} = \dot{1}$ —> Oui

Calcul de l'inverse

Principe.

1. Calculer l'inverse de m dans $\mathbb{Z}/n\mathbb{Z}$ nécessite de chercher la relation de Bézout entre le nombre et n.
2. L'algorithme d'Euclide étendu est utilisé pour chercher la relation de Bézout.

Exemple. Cherchons l'inverse de 37 dans $\mathbb{Z}/63\mathbb{Z}$.

Une relation de Bézout entre 37 et 63 doit s'exprimer sous la forme:

$x * 63 + y * 37 = 1$, x et y étant deux entiers.

Une possibilité est: $10 * 63 - 17 * 37 = 1$

Calcul de l'inverse

Principe.

1. Calculer l'inverse de m dans $\mathbb{Z}/n\mathbb{Z}$ nécessite de chercher la relation de Bézout entre le nombre et n.
2. L'algorithme d'Euclide étendu est utilisé pour chercher la relation de Bézout.

Une possibilité est: $10 * 63 - 17 * 37 = 1$

Or dans $\mathbb{Z}/63\mathbb{Z}$, $\dot{63} = \dot{0}$

On a donc $-\dot{17} * \dot{37} = \dot{1}$

L'inverse de $\dot{37}$ est donc $\dot{-17}$ (=46)

Calcul de l’inverse - Algorithme d’Euclide étendu

Comment trouver la relation de Bézout ? (10 * 63 - 17 * 37 = 1)

→Utilisation de l’algorithme d’Euclide étendu

Principe.

1. Même travail de décomposition qu’avec l’algorithme d’Euclide classique
2. “Remontée” pour trouver la relation de Bézout

Calcul de l’inverse - Algorithme d’Euclide étendu

Exemple. Pour 37 et 63.

63 = 1 * 37 + 26

37 = 1 * 26 + 11

26 = 2 * 11 + 4

11 = 2 * 4 + 3

4 = 1 * 3 + 1 ⟶ 1 = 4 - 1 * 3

Calcul de l'inverse - Algorithme d'Euclide étendu

Exemple. Pour 37 et 63.

63 = 1 * 37 + 26

37 = 1 * 26 + 11

26 = 2 * 11 + 4

11 = 2 * 4 + 3 → 1 = 4 - 1 * (11 - 2 * 4) = 3 * 4 - 1 * 11

4 = 1 * 3 + 1 → 1 = 4 - 1 * 3

Calcul de l'inverse - Algorithme d'Euclide étendu

Exemple. Pour 37 et 63.

63 = 1 * 37 + 26

37 = 1 * 26 + 11

26 = 2 * 11 + 4 $\quad\rightarrow\quad$ 1 = 3 * (26 - 2 * 11) - 1 * 11 = 3 * 26 - 7 * 11

11 = 2 * 4 + 3 $\quad\rightarrow\quad$ 1 = 4 - 1 * (11 - 2 * 4) = 3 * 4 - 1 * 11

4 = 1 * 3 + 1 $\quad\rightarrow\quad$ 1 = 4 - 1 * 3

Calcul de l'inverse - Algorithme d'Euclide étendu

Exemple. Pour 37 et 63.

$63 = 1 * 37 + 26$

$37 = 1 * 26 + 11$ $\quad\rightarrow\quad 1 = 3 * 26 - 7 * (37 - 1 * 26) = 10 * 26 - 7 * 37$

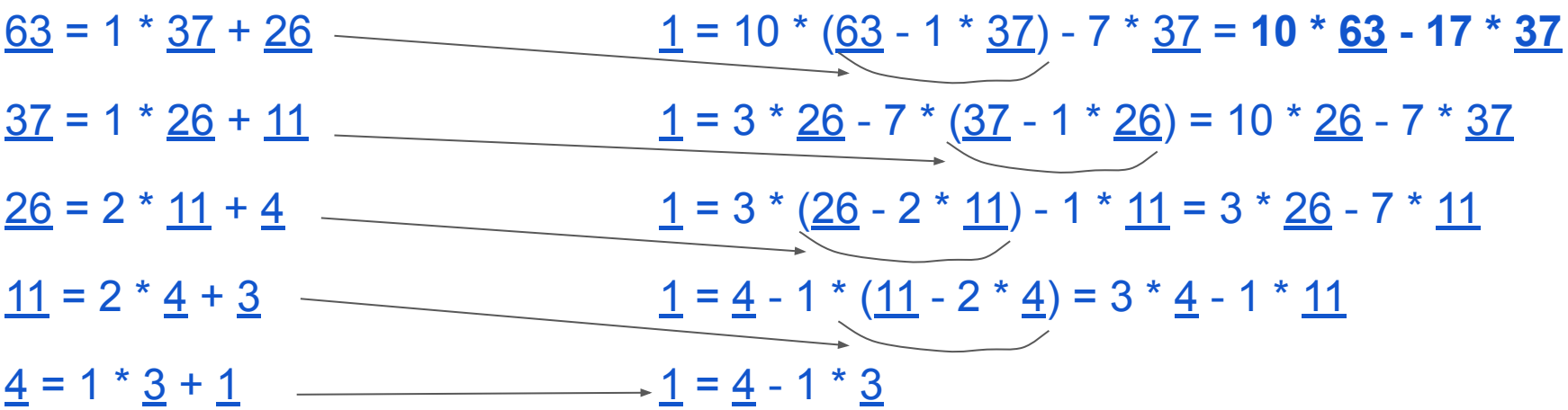
$26 = 2 * 11 + 4$ $\quad\rightarrow\quad 1 = 3 * (26 - 2 * 11) - 1 * 11 = 3 * 26 - 7 * 11$

$11 = 2 * 4 + 3$ $\quad\rightarrow\quad 1 = 4 - 1 * (11 - 2 * 4) = 3 * 4 - 1 * 11$

$4 = 1 * 3 + 1$ $\quad\rightarrow\quad 1 = 4 - 1 * 3$

Calcul de l'inverse - Algorithme d'Euclide étendu

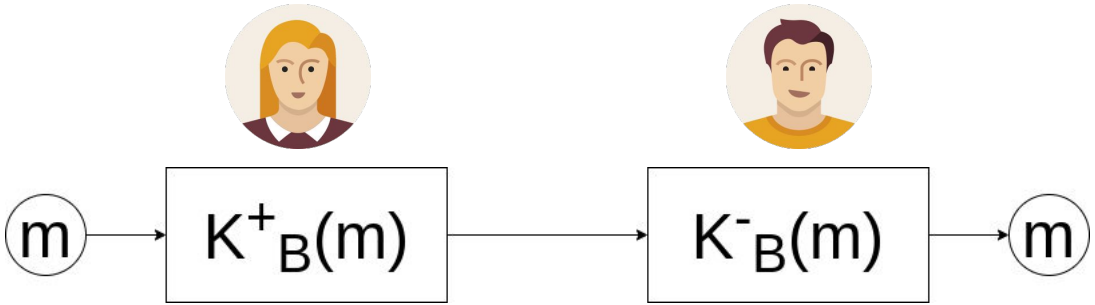
Exemple. Pour 37 et 63.



Calcul d’inverse

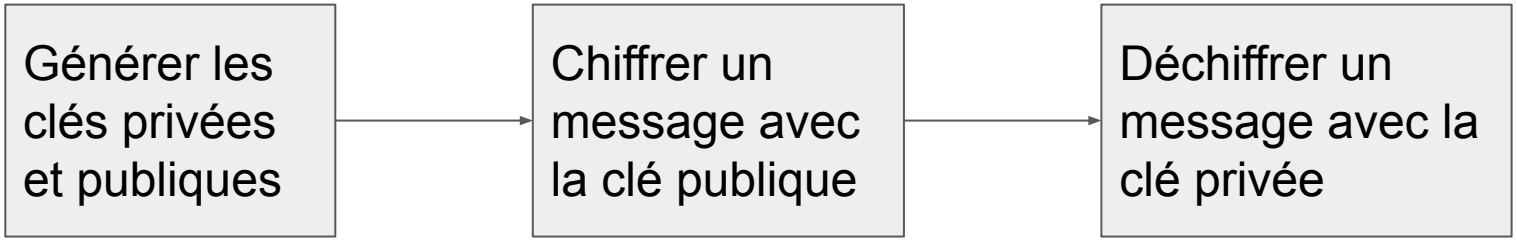
RSA - introduction

- Chiffrement asymétrique avec clés publiques et privées
- Inventé par Rivest, Shamir et Adleman en 1978
- Le fonctionnement de l'algorithme est connu mais le coût calculatoire pour déchiffrer un message sans les clefs est très élevé

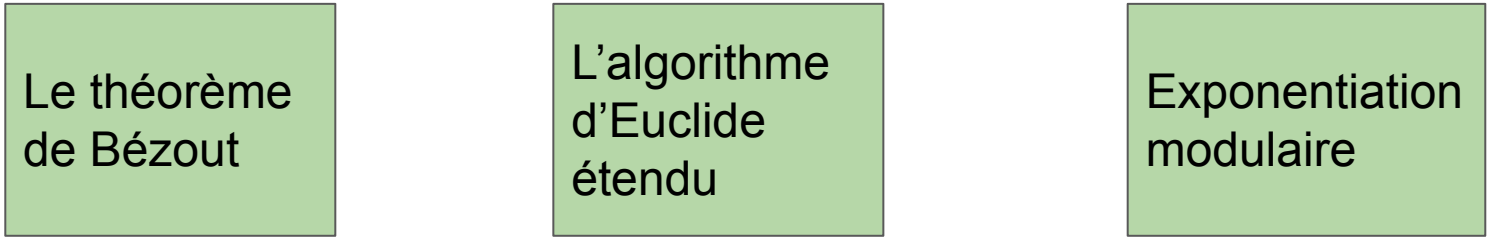


RSA

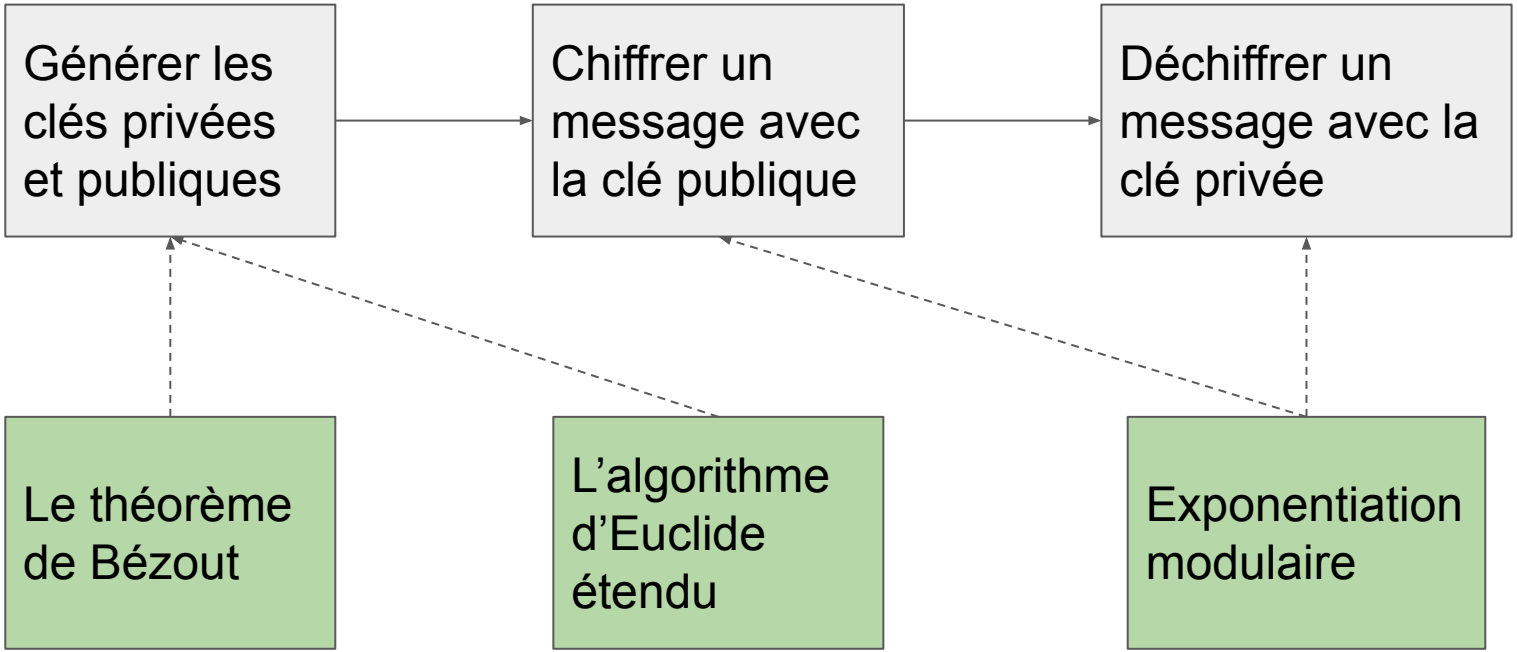
Fonctionnement de RSA étape par étape:



Outils mathématique à notre disposition:



RSA


Générer les clés privées et publiques
Chiffrer un message avec la clé publique
Déchiffrer un message avec la clé privée
Le théorème de Bézout
L'algorithme d'Euclide étendu
Exponentiation modulaire

Générer les clés publiques et privées

Comment générer les clés ?

1. Choix de deux nombres entiers p et q distincts
2. Calculer $n = p * q$
3. Calculer $\varphi(n) = (p - 1) * (q - 1)$ (privé car très complexe de connaître $\varphi(n)$ sans connaître p et q)
4. Choix d'un exposant e et calcul de son inverse d tel que pgcd(e, $\varphi(n)$) = 1 et $d * e \equiv 1 \mod \varphi(n)$
5. La clé publique est constituée de n et e, la clé privée est d

Exemple

p = 5, **q** = 17

n = p * q = 85

φ(n) = (p - 1) * (q - 1) = 64

Choix d'un exposant, tel que pgcd(*e*, φ(*n*)) = 1, **e** = 5

Calcul de l'inverse de *e* avec l'algorithme d'Euclide étendu: **d** = 13

Clés publiques: n = 85, e = 5

Clé privée: d = 13

Chiffrement du message

Comment chiffrer un message ?

1. Bob veut envoyer un message à Alice
2. Transformation du message en entier m tel que $0 \leq m < n$ (forcément inférieur puisque l'on travaille modulo n)
3. Bob demande la clé publique d'Alice (n et e)
4. Il calcule le message chiffré: $x \equiv m^e \pmod{n}$

Exemple

Clés publiques: n = 85, e = 5

Clé privée: d = 13

Message à envoyer: m = 10

Le message chiffré est $x \equiv m^e \pmod{n}$:

- $x \equiv 10^5 \pmod{85}$
- $10^2 = 100 \equiv 15 \pmod{85}$
- $10^3 = 15 * 10 \equiv 150 \pmod{85} \equiv 65 \pmod{85}$
- $10^4 = 650 \pmod{85} \equiv 55 \pmod{85}$
- $10^5 = 550 \pmod{85} \equiv \mathbf{40} \pmod{85}$
- $x = 40$

Déchiffrement du message

Comment chiffrer un message ?

1. Alice reçoit le message de Bob
2. Elle le décrypte en calculant: $m \equiv x^d \pmod{n}$ à l'aide de sa clé privée d.

Exemple

Clés publiques: n = 85, e = 5

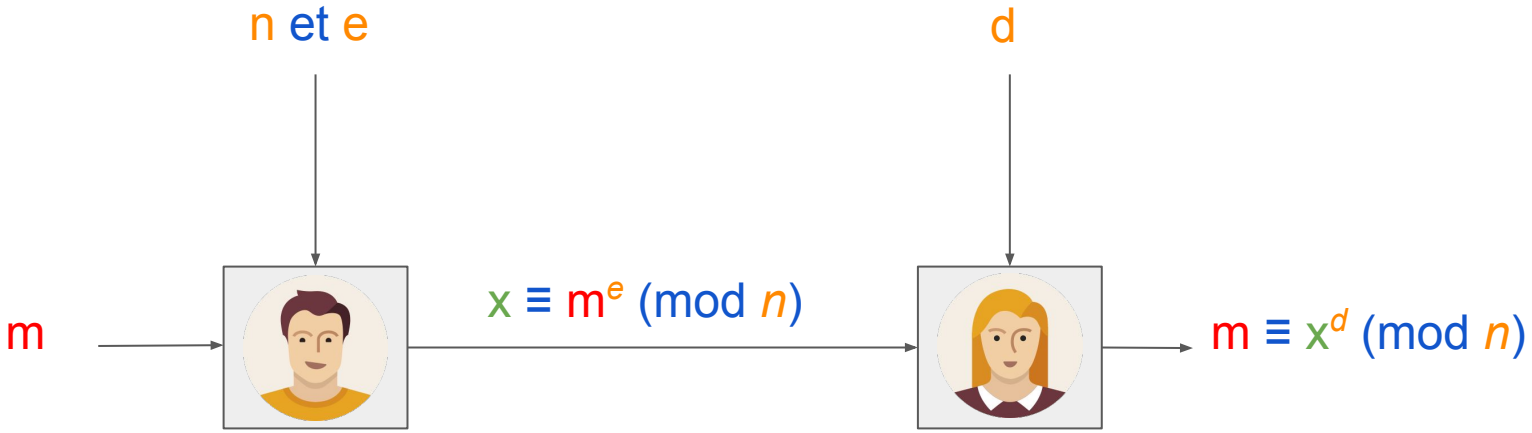
Clé privée: d = 13

Message à déchiffrer: x = 40

Le message déchiffré est $m \equiv x^d \pmod{n}$:

- $m \equiv 40^{13} \pmod{85}$
- $40^2 = 1600 \pmod{85} \equiv 70 \pmod{85}$
- $(40^2)^2 = 70^2 \pmod{85} \equiv 4900 \pmod{85} \equiv 55 \pmod{85}$
- $(40^4)^2 = 55^2 \pmod{85} \equiv 50 \pmod{85}$
- $40^{13} \equiv 40^8 * 40^4 * 40 \equiv 50 * 55 * 40 \pmod{85} \equiv 10 \pmod{85}$
- $m = 10$

Résumé



Avec:

- n et e les clés publiques de Alice
- d la clé privée de Alice
- m le message déchiffré, x le message chiffré

Conclusion sur les chiffrements symétriques

Ces méthodes sont utiles mais quelques inconvénients nuances leur utilisation:

- Le nombre de clés à transmettre est très grand:
  - Pour N personnes, il faudra transmettre N * (N-1) / 2 clés

Quand utiliser le chiffrement asymétrique ?

- Pour communiquer sur internet

Identifier les apprentissages

Quels sont les concepts à retenir de ce cours ?

Activité *1, 2, Tous*:

- Réfléchissez individuellement à cette question pendant - **3mn**
- Comparez vos idées avec l'un de vos voisins, convainquez-le que vous avez raison ! - **3mn**
- Mise en commun des réponses, des binômes sont interrogés - **2mn**

En résumé

Ce qu'il faut retenir:

- Le théorème de Bézout
- L'algorithme d'Euclide étendu
- Le fonctionnement de Diffie-Hellman et RSA

Le prochain cours:

- Comment gérer un grand nombre de clé
- Comment détecter des erreurs dans la transmission de messages

Quelques dernières questions sur le quizz

Ressources complémentaires

- Distributed Systems, Andrew S. Tanenbaum, 2017 (free ebook on https://www.distributed-systems.net/index.php/books/ds3/)
- https://www.tutorialspoint.com/cryptography_with_python/cryptography_with_python_understanding_rsa_algorithm.htm
- https://www.youtube.com/watch?v=XlaI_d4zyfo&ab_channel=Exo7Math
- https://www.youtube.com/watch?v=PTrRp-w4XHs&list=PLE8WtfrsTAinMMyQkK_CzXhXU_LHRNXy_&index=2&ab_channel=MathsAdultes
- https://www.youtube.com/watch?v=Ist_yFnhDBg&t=934s&ab_channel=MathsAdultes