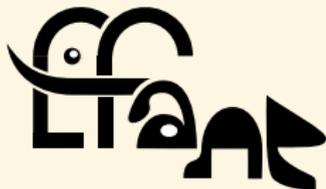


À quoi sert la cryptologie? – Petit panorama des mathématiques de la cryptologie

2019/05/22 – Concours Alkindi, Inria Bordeaux

Damien Robert

Équipe LFANT, Inria Bordeaux Sud-Ouest



université
de **BORDEAUX**

Inria
informatics mathematics

- Ecole préparatoire MPSI/MP à Lyon ;
 - Normale sup Paris ;
 - Thèse à Inria Nancy ;
 - Postdoc à Microsoft Research, Redmond ;
 - Chercheur à Inria Bordeaux.
- ⇒ Algorithmes en théorie des nombres et géométrie algébrique, applications en cryptologie.



Cryptologie à clé publique

Cryptologie :

- Chiffrement;
- Authenticité;
- Intégrité.

Applications :

- Militaires;
- Vie privée;
- Communications (internet, téléphones...)
- Commerce électronique...



Contexte Historique

- Riche histoire ; chiffrement de messages depuis l'antiquité au moins ;
- Principale application auparavant **militaire** ;
- Dorénavant la cryptologie joue un rôle essentiel pour garantir la **sécurité des communications** ;
- **Cryptanalyse** : déchiffrement d'Énigma par le groupe Ultra (Secret) à Bletchley Park lors de la seconde guerre mondiale ;
- À la fin de la guerre, vente des machines Énigma capturées par les alliés à d'autres pays.

Exemple (Cryptanalyse d'Énigma)

- $\text{enigma}(c) = \text{plug}^{-1} \circ \text{rotor}_1^{-1} \circ \text{rotor}_2^{-1} \circ \text{rotor}_3^{-1} \circ \text{reflector} \circ \text{rotor}_3 \circ \text{rotor}_2 \circ \text{rotor}_1 \circ \text{plug}(c)$.
 - Les rotors bougent à chaque étape.
 - Le réflecteur transpose une lettre avec une lettre distincte.
 - Le plug est constitué de six cables qui transposent les lettres.
 - Au total
 $3! \times 26^3 \times 26 \cdot 25 \cdot 24 \cdot 23 \cdot 22 \cdot 21 \cdot 20 \cdot 19 \cdot 18 \cdot 17 \cdot 16 \cdot 15 / 2^6 = 914709608446233600000 \approx 2^{70}$
possibilités!
- ⇒ Permutation sans point fixe!

Contexte Historique

- Riche histoire ; chiffrement de messages depuis l'antiquité au moins ;
- Principale application auparavant **militaire** ;
- Dorénavant la cryptologie joue un rôle essentiel pour garantir la **sécurité des communications** ;
- **Cryptanalyse** : déchiffrement d'Énigma par le groupe Ultra (Secret) à Bletchley Park lors de la seconde guerre mondiale ;
- À la fin de la guerre, vente des machines Énigma capturées par les alliés à d'autres pays.

Exemple (Cryptanalyse d'Énigma)

- $\text{enigma}(c) = \text{plug}^{-1} \circ \text{rotor}_1^{-1} \circ \text{rotor}_2^{-1} \circ \text{rotor}_3^{-1} \circ \text{reflector} \circ \text{rotor}_3 \circ \text{rotor}_2 \circ \text{rotor}_1 \circ \text{plug}(c)$.
 - Les rotors bougent à chaque étape.
 - Le réflecteur transpose une lettre avec une lettre distincte.
 - Le plug est constitué de six cables qui transposent les lettres.
 - Au total
 $3! \times 26^3 \times 26 \cdot 25 \cdot 24 \cdot 23 \cdot 22 \cdot 21 \cdot 20 \cdot 19 \cdot 18 \cdot 17 \cdot 16 \cdot 15 / 2^6 = 914709608446233600000 \approx 2^{70}$
possibilités!
- ⇒ Permutation sans point fixe!

Protocoles cryptographiques

- Briques de base (primitives), s'appuyant sur des objets mathématiques : chiffrer un message de longueur fixé.
- Ces primitives sont combinées pour former des algorithmes/modes opératoires : algorithme de chiffrement, algorithme de signature
- Ces modes opératoires sont combinés pour former des protocoles : protocole de session TLS (chiffrement + authentification), protocole de vote
- Ces protocoles sont implémentés en logiciel ou matériel
- Puis ils sont utilisés.



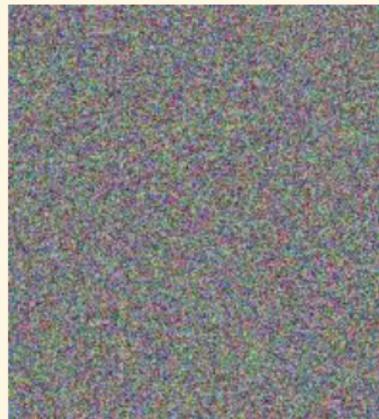
Exemple : De AES à un algorithme de chiffrement

- AES permet de chiffrer un message m d'une certaine taille k ($k = 128$ bits) :
 $m \mapsto E(m)$;
 - Comment chiffrer un message de longueur arbitraire?
 - Idée naturelle : découper m en messages $m_1 \parallel m_2 \dots \parallel m_d$ de taille k , et poser
 $E(m) = E(m_1) \parallel E(m_2) \dots \parallel E(m_d)$;
- ⇒ Le même sous bloc est chiffré de la même manière.



Exemple : De AES à un algorithme de chiffrement

- AES permet de chiffrer un message m d'une certaine taille k ($k = 128$ bits) :
 $m \mapsto E(m)$;
 - Comment chiffrer un message de longueur arbitraire?
 - Idée naturelle : découper m en messages $m_1 \parallel m_2 \dots \parallel m_d$ de taille k , et poser
 $E(m) = E(m_1) \parallel E(m_2) \dots \parallel E(m_d)$;
- ⇒ Le même sous bloc est chiffré de la même manière.



Exemple : intégrité et chiffrement

- Comment combiner les deux briques de base que sont le chiffrement (Encrypt) et l'intégrité (MAC Message Authentication Code);
- Encrypt then MAC?
- MAC then Encrypt?
- Encrypt + Mac?



Exemple : intégrité et chiffrement

- Comment combiner les deux briques de base que sont le chiffrement (Encrypt) et l'intégrité (MAC Message Authentication Code);
- Encrypt then MAC?
- MAC then Encrypt?
- Encrypt + Mac?



Attaques et sécurité

Attaques :

- Attaques sur les briques de base (très rare) ;
- Attaques sur l'empilement des briques en algorithmes ou protocoles ;
- Attaques sur l'implémentation ;
- Attaques sur l'exécution.

Sécurité

- Briques de base : repose sur des problèmes mathématiques bien identifiés et très étudiés (difficulté de la factorisation, logarithme discret dans les courbes elliptiques)
- Preuves de sécurité sur les algorithmes et protocoles : si un attaquant peut attaquer le protocole (avec une certaine probabilité p en temps T), alors il peut attaquer une brique de base (avec une certaine probabilité p' en temps T')



Sécurité?

- Erreur dans les preuves
- Preuves justes mais modèle incorrect
- Modèle correct mais utilisé dans un autre contexte
- Réductions de sécurités inefficaces
- Bugs dans les programmes
- Erreurs ou backdoors matérielles
- Attaques physiques (par canaux cachés) : mesure des impulsions électromagnétiques, du bruit, du temps de calcul, des cache miss. L'attaquant a plus d'informations que juste le message chiffré!

Exemple (Attaques sur TLS)

- Protocole : Renegotiation attack / Version rollback attack
- BEAST (attaque sur le mode Cipher Block Chaining)
- CRIME and BREACH (attaque sur la compression)
- Downgrade attack : FREAK (export grade cryptography), Logjam (gros précalculs)
- Bugs : Heartbleed (buffer overflow), BERserk, goto fail
- Certificats mal formés

Mitiger la perte des clés privées : **perfect forward secrecy** via un échange de clés éphémères par Diffie-Hellman.

Quelques applications cryptographiques modernes

- Chiffrement de groupe ;
- Mise en gage ;
- Partage de secret ;
- Preuves sans divulgation de connaissance ;
- Certificats anonymes ;
- Transfert inconscient ;
- Signature de cercle ;
- Calcul multipartite sécurisé ;
- Chiffrement fonctionnel ;
- Obfuscation.



Applications cryptographiques : Bitcoin

- Monnaie électronique décentralisée
- Fichier de transaction public (blockchain)
- Signature des transactions par une courbe elliptique
- La vérification de la blockchain (et validation des nouvelles transactions) fabrication de nouveaux bitcoins.



Applications cryptographiques : Vote électronique Belenios

- Confidentialité du vote (partage de secret)
- Addition des votes chiffrés (chiffrement homomorphe)
- Résultats corrects (preuves Zero-Knowledge)
- Validation (et confidentialité) de la liste des électeurs



L'essor du cloud computing

- Chiffrement homomorphe : l'utilisateur fournit au nuage un message chiffré $f_K(m)$ et un programme P , et le nuage renvoie $f_K(P(m))$.
Le nuage n'a rien appris sur la donnée m , ni sur le résultat!
 - L'utilisateur fournit au nuage un message chiffré $f_K(m)$ et le chiffrement $f_K(P)$ d'un programme P , et le nuage renvoie $f_K(P(m))$.
Le nuage ne sait pas ce qu'il a calculé!
- ⇒ Une version faible utilise les couplages de courbes elliptiques ;
- ⇒ La version complète utilise des réseaux, en particulier des réseaux d'idéaux dans des corps de nombres ;
- ☹ Encore très lent.



Cryptographie post-quantique

- RSA (basé sur la factorisation) et les courbes elliptiques sont vulnérables aux ordinateurs quantiques ;
- Problématique pour des secrets à très long terme (50 ans) : secrets défense
- Conception de nouveaux protocoles résistant aux ordinateurs quantiques
- Diffie-Hellmann : échange de clé par des opérations dans un groupe.
Diffie-Hellmann post-quantique : échange de clé par des opérations dans un graphe.



Chiffrement

Alice (Sophie Germain)



veut écrire

à Bob (Carl Friedrich Gauss)



Chiffrement à clé secrète



Alice

$m = \text{QUARTIQUE}$

clé secrète de chiffrement $K = +3$

$$c = f_K(m)$$

$c = \text{TXDUWLTXH}$ est envoyé



à Bob

$c = \text{TXDUWLTXH}$

clé secrète de déchiffrement $K' = -K = -3$

$$m = f_K^{(-1)}(c) = f_{K'}(c)$$



Chiffrement à clé secrète



Alice

$m = \text{QUARTIQUE}$

clé secrète de chiffrement $K = +3$

$$c = f_K(m)$$

$c = \text{TXDUWLTXH}$ est envoyé



à Bob

$c = \text{TXDUWLTXH}$

clé secrète de déchiffrement $K' = -K = -3$

$$m = f_K^{(-1)}(c) = f_{K'}(c)$$



Chiffrement à clé secrète



Alice

$m = \text{QUARTIQUE}$

clé secrète de chiffrement $K = +3$

$$c = f_K(m)$$

$c = \text{TXDUWLTXH}$ est envoyé



à Bob

$c = \text{TXDUWLTXH}$

clé secrète de déchiffrement $K' = -K = -3$

$$m = f_{K'}^{-1}(c) = f_{K'}(c)$$



Chiffrement à clé secrète

- Trois étapes : création et distribution de clés, chiffrement, déchiffrement
- Avantages : simple, rapide, bien connu
- Fragilités : attaques statistiques, gestion de clés
- Le chiffrement de Vernam (One Time Pad) est inconditionnellement sûr, quelle que soit la puissance de calcul de l'adversaire (Attention : nécessite une clé secrète de même taille que le message envoyé);
- Notion de théorie de l'information (Shannon);
- Très compliqué et coûteux à mettre en place correctement.

Comment transmettre la clé secrète de manière sécurisée?

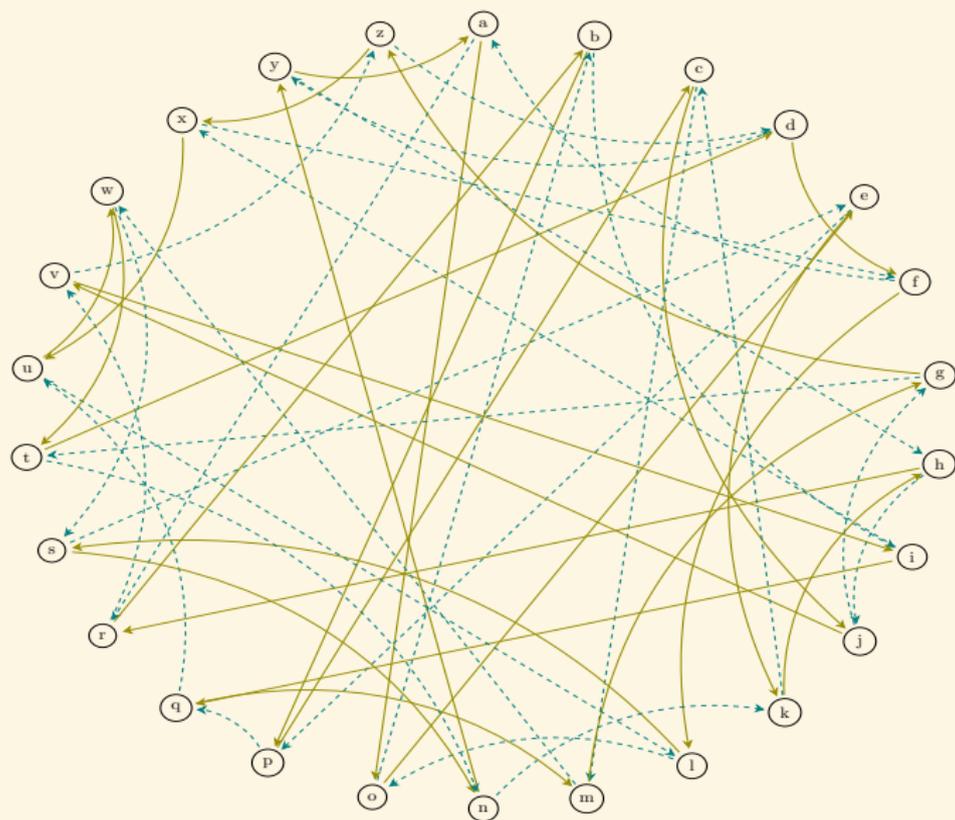


Échange de clé

- Échanger une clé secrète commune à travers un canal public ;
- Proposé par Diffie et Hellman en 1976 ;
- Utilise des groupes ;
- Version moderne post-quantique : utilise des graphes.

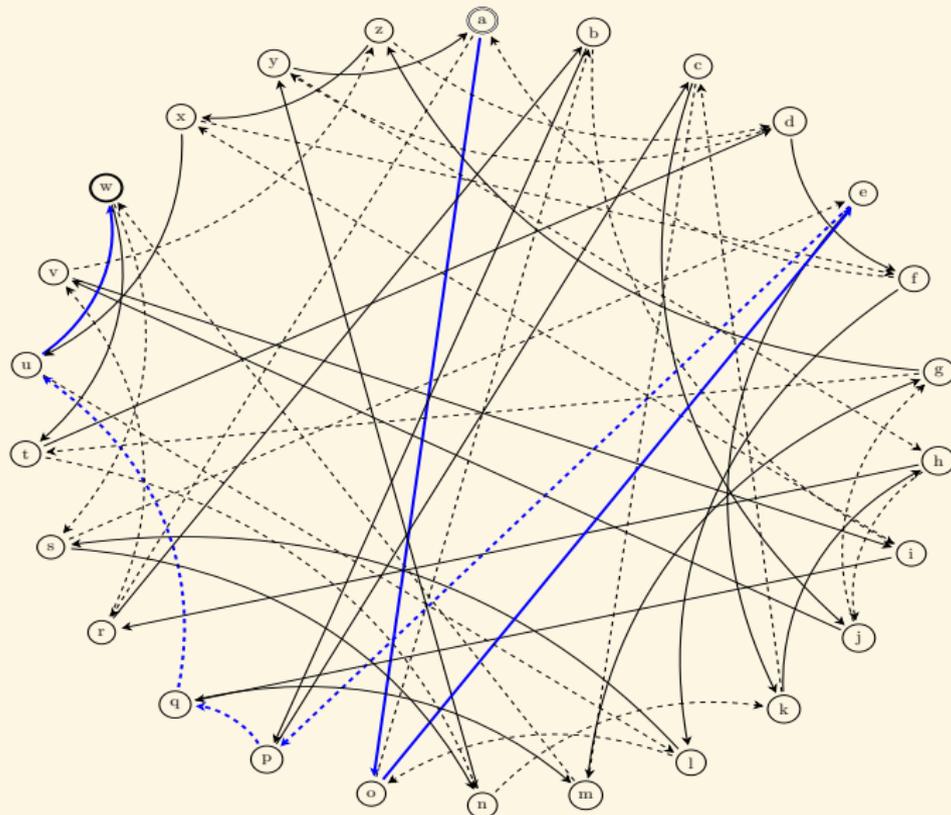


Échange de clé par graphe



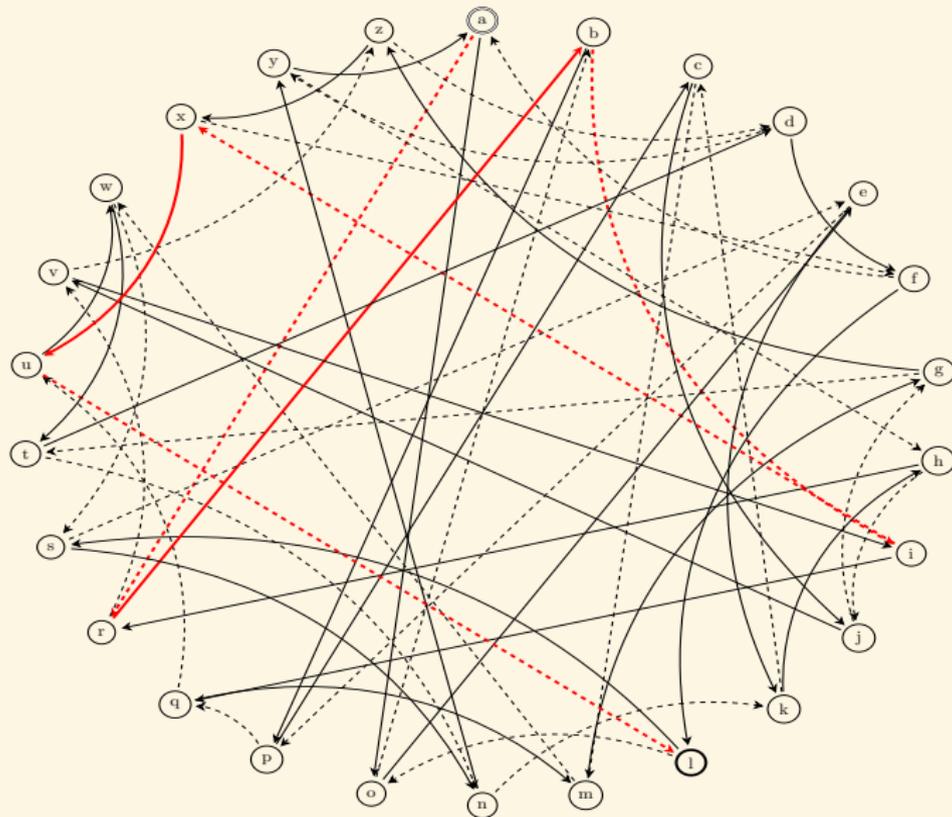
Échange de clé par graphe

Alice part de 'a', suit le chemin 001110, et tombe sur 'w'.



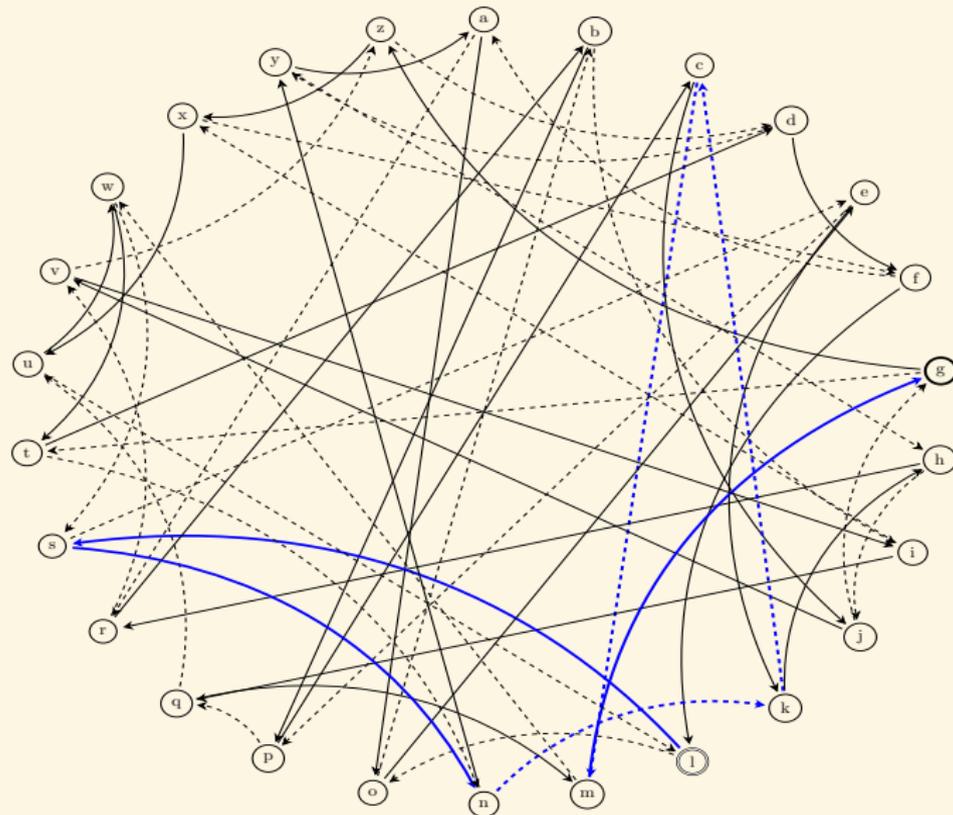
Échange de clé par graphe

Bob part de 'a', suit le chemin 101101, et tombe sur 'l'.



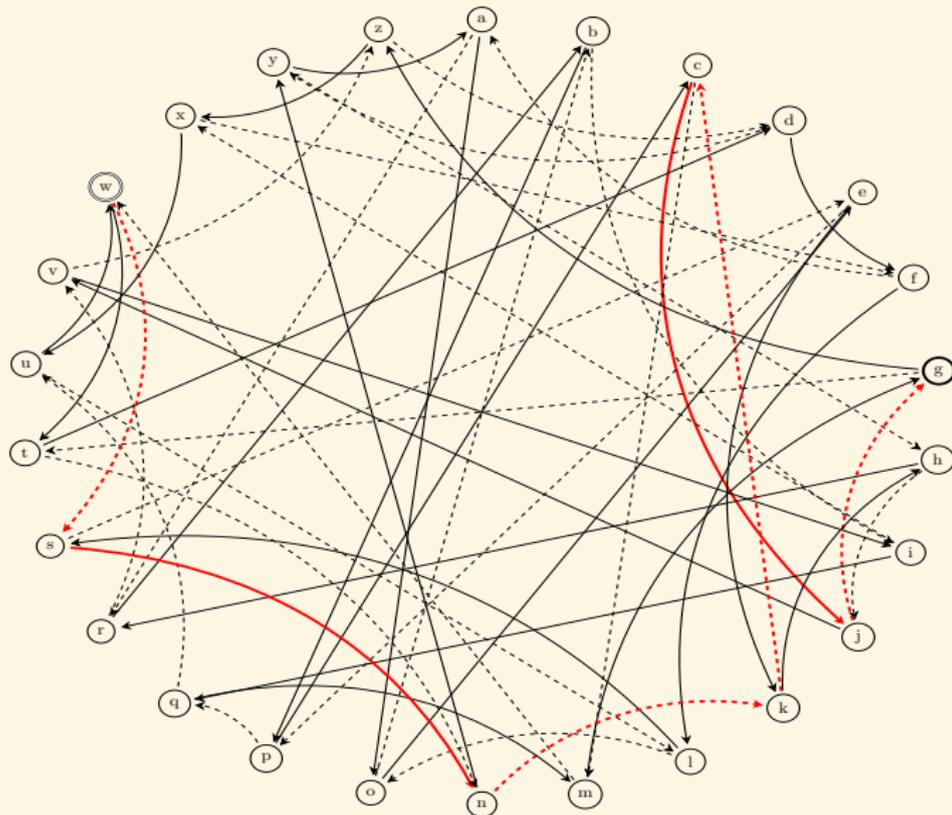
Échange de clé par graphe

Alice part de 'l', suit le chemin 001110, et obtient 'g'.



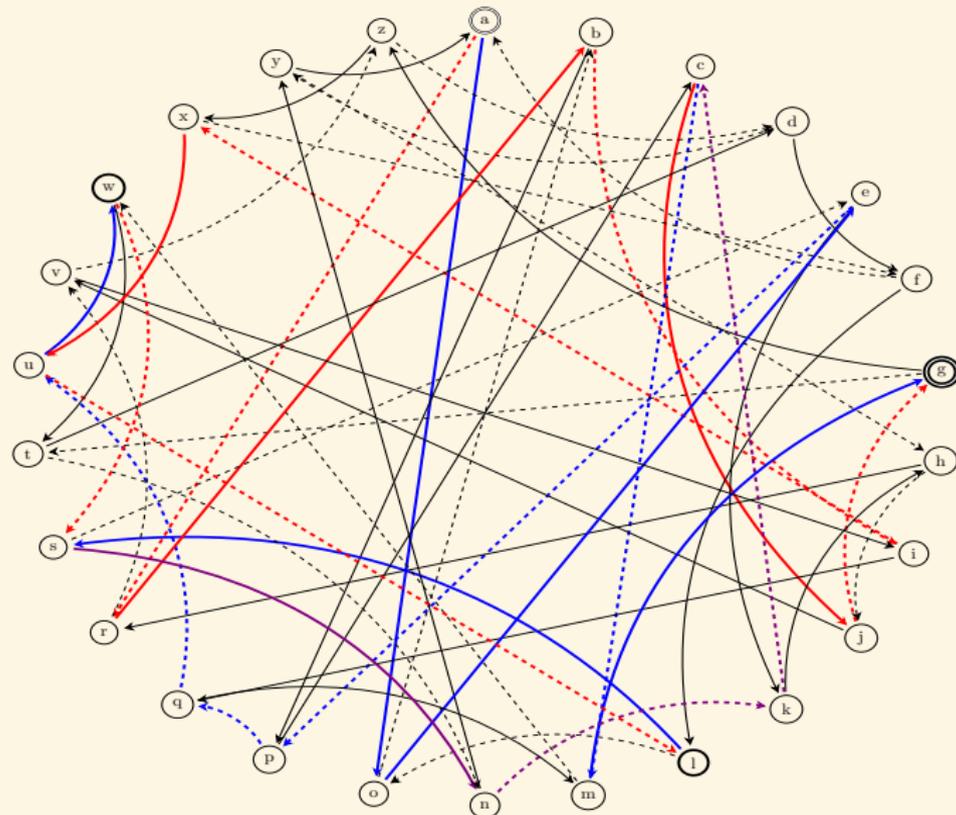
Échange de clé par graphe

Bob part de 'w', suit le chemin 101101, et obtient 'g'.



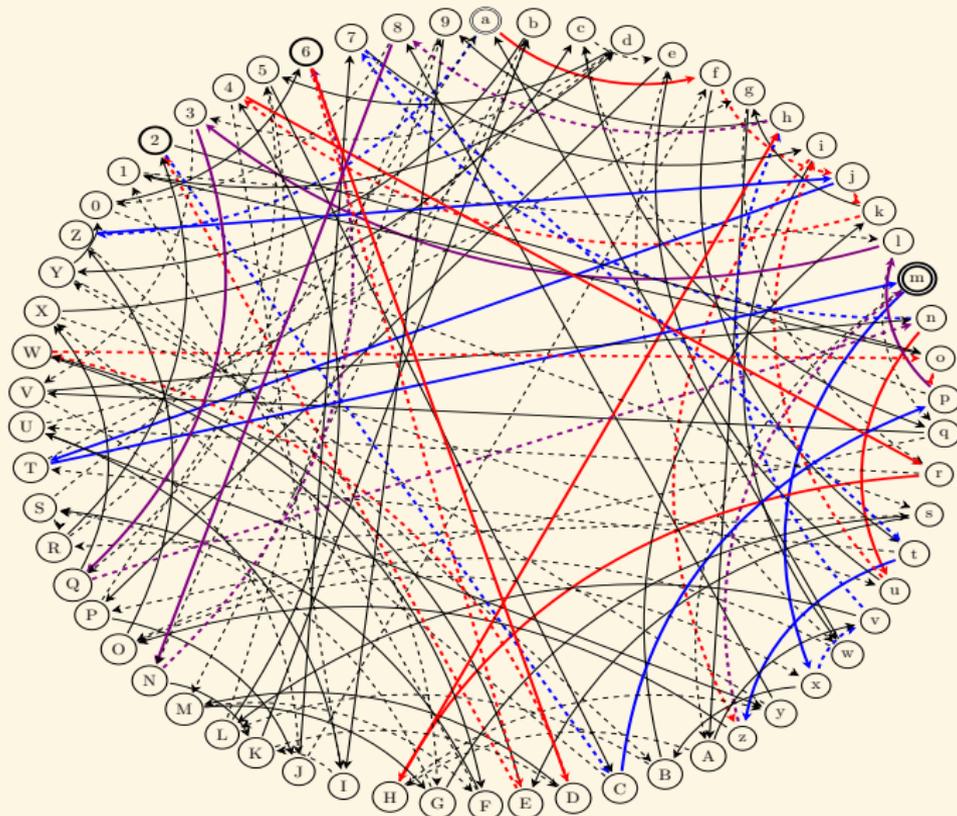
Échange de clé par graphe

L'échange de clé complet



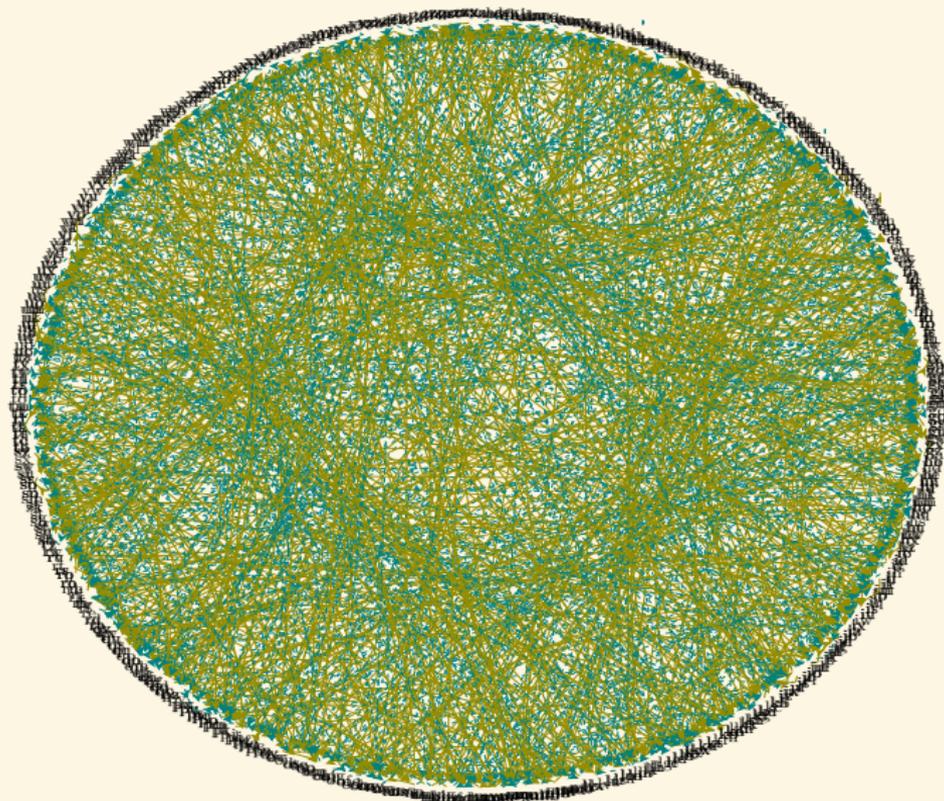
Échange de clé par graphe

Graphe plug grand (62 noeuds)



Échange de clé par graphe

Graphe encore plus grand (676 noeuds)



Échange de clé par graphe

Taille cryptographique :

- Pour une sécurité de 2^{128} bits, il faut un arbre avec $n = 2^{256}$ noeuds (attaque en \sqrt{n} : on cherche un chemin en partant du point de départ et d'arrivée à la fois.)
- Il faut aussi que les arrêtes mélanges bien le graphe : en $\log(n)$ étapes on arrive sur un noeud « uniforme » (graphe de Ramanujan).
- Le graphe ne tient pas en mémoire...Il faut un algorithme qui à partir d'un noeud donne ses voisins.



Chiffrement à clé publique



Alice
veut envoyer m à Bob.

Elle trouve la clé publique de chiffrement K_{Bob}^{pub} dans l'annuaire.

Elle calcule $c = f_{K_{Bob}^{pub}}(m)$

c est envoyé



à Bob

qui utilise sa clé secrète de déchiffrement K_{Bob}^{sec}

$$m = f_{K_{Bob}^{pub}}^{(-1)}(c) = g_{K_{Bob}^{sec}}(c)$$


Chiffrement à clé publique



Alice
veut envoyer m à Bob.

Elle trouve la clé publique de chiffrement K_{Bob}^{pub} dans l'annuaire.

Elle calcule $c = f_{K_{Bob}^{pub}}(m)$

c est envoyé



à Bob

qui utilise sa clé secrète de déchiffrement K_{Bob}^{sec}

$$m = f_{K_{Bob}^{pub}}^{(-1)}(c) = g_{K_{Bob}^{sec}}(c)$$



Chiffrement à clé publique



Alice
veut envoyer m à Bob.

Elle trouve la clé publique de chiffrement K_{Bob}^{pub} dans l'annuaire.

Elle calcule $c = f_{K_{Bob}^{pub}}(m)$

c est envoyé

à Bob



qui utilise sa clé secrète de déchiffrement K_{Bob}^{sec}

$$m = f_{K_{Bob}^{pub}}^{(-1)}(c) = g_{K_{Bob}^{sec}}(c)$$



Chiffrement à clé publique

- Trois étapes : création et publication de clés, chiffrement, déchiffrement
- Avantages : gestion de clé simplifiée, solidité mathématique
- Permet de faire des signatures, du chiffrement de groupe...
- Fragilités : plus lent, plus compliqué à implémenter

En pratique on combine les deux chiffrements : clé publique pour échanger une clé de session (secrète) qui servira à chiffrer à la volée.

Comment savoir quelle clé publique utiliser pour communiquer avec un utilisateur donné?

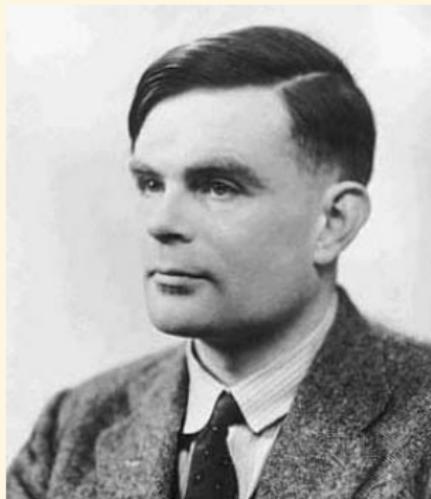


Comment faire ?

- Situations asymétriques : l'un sait l'autre pas.
- Celui qui connaît le secret a un avantage (il peut déchiffrer, il peut se prouver).
- Mesurer cet avantage : **théorie de la complexité algorithmique**.
- S'appuyer sur des problèmes difficiles.



La thèse de Turing-Church



Alan Turing



Alonzo Church

Tests de primalité

Savoir si un entier P est premier.



Pierre de Fermat



Agrawal, Kayal et Saxena

$$T = n^{6+\epsilon(n)}$$

où n est le nombre de chiffres décimaux de P .

Factorisation

Théorème fondamental de l'arithmétique.



Euclide



Carl Friedrich Gauss

$$N = \prod_{1 \leq i \leq l} p_i^{e_i}.$$

Factorisation



Hendrik Lenstra



Brigitte Vallée

Factoriser un entier N prend un temps $T = \exp(\sqrt{n})$ où n est le nombre de chiffres décimaux de n . (Algorithme heuristique : $T = \exp(n^{1/3})$)



$$(p, q) \xrightarrow{\text{green}} N = pq$$

$$(p, q) \xleftarrow{\text{red}} N = pq$$

- En décembre 2009, Thorsten Kleinjung et une dizaine de collègues ont factorisé un nombre de 232 décimales.
- *The sieving, which was done on many hundreds of machines, took almost two years.*
- Calculer le produit de deux nombres de 116 décimales prend 8 millièmes de secondes sur mon ordinateur portable.



Protocole RSA



Rivest, Shamir et Adleman

Protocole RSA

- Soit $N = pq$ un produit de deux grands nombres premiers ;
- Soit e premier à $\phi(N) = (p - 1)(q - 1)$ et d l'inverse de e modulo $\phi(N)$;
- **Chiffrement** : $x \mapsto x^e \pmod N$;
- **Déchiffrement** : $x \mapsto x^d \pmod N$;

Théorème (Petit théorème de Fermat)

$$x^{\#(\mathbb{Z}/N\mathbb{Z})^\times} = 1 \pmod N.$$

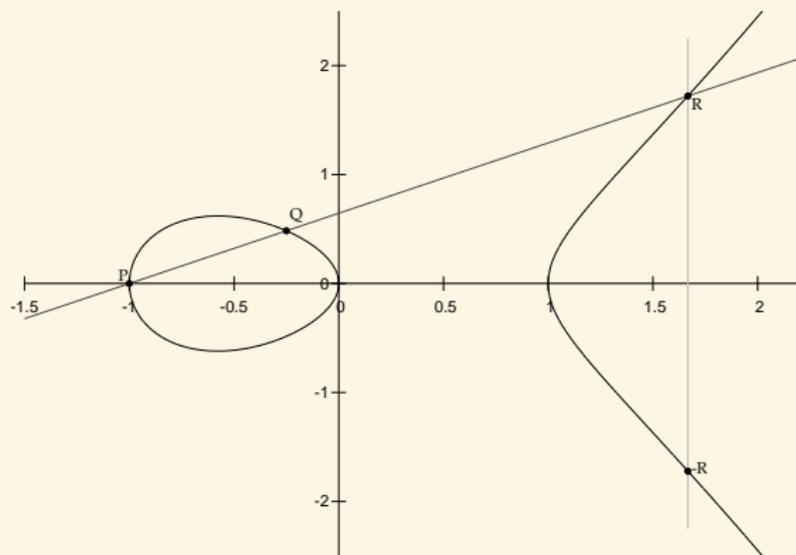


Les courbes elliptiques

Définition (char $k \neq 2, 3$)

Une courbe elliptique est une courbe plane d'équation

$$y^2 = x^3 + ax + b \quad 4a^3 + 27b^2 \neq 0.$$



Exponentiation :

$$(\ell, P) \mapsto \ell P$$

Logarithme discret :

$$(P, \ell P) \mapsto \ell$$

ECC contre RSA pour 128 bits de sécurité

- ECC (Curve25519) 256 bits :

AAAC3NzaC11ZDI1NTE5AAAAIMoNrNYhU7CY1Xs6v4Nm1V6oRHs/FEe8P+XaZ0PcxPzz

- RSA 3248 bits :

MIHRgIBAAKCAZcAv1GW+b5L2tmqb5bUJMrfLHgr2jga/Q/8IJ5QJqeSsB7xLVT/
ODN3KNSPxyjaHmDNDtWgSikZvPYeyZWfLp0B0vGwDqQugUGHvfg473Z0lqZk6
1nA45XZGHUPT98p4+ghPag5JyvAVs1fCf/Vl1tBHbu/NOYIAC4F3tHP81nn+10nB
e1lEALbdmVgTTZ5jcrRrt4IDT5a4IeI9yTe0aVdTsuJ6990hpkRvZyTou1eoxp5eV
KQ7aIX6es9Xjnr8widZunM8rqhBw9EMmLqabnXZITPQoV3rUAnwKzDLV7E56viJk
S2xU5+95IctYU/RTTbf3wTxnkdOqxId0MONHyBjsukXgYKxvB1fwhBKZ4tWu1iGw
UCIiK7qLm12zJhLn4WovaxrvvtX008250xncEfYDXyU4xbRnJn+ZsTTzququfC1M
U4MYRDwy7uj+H1EmIGul69Fw9NkuCItWi9dFpcDtSP+/1eEN7wc2F1xhDIRwer0F
6I1P4S2wn1uQyHzsTLVdcp+rqa1AsvbwBCKL4ravEO2CEQIDAQABAOIB11wt5YoJ
YZzk4RcnbSX/LvmwICfdmkjTKW6F1w+P4TnotCr0wPG00bDoANJoUrcXnsSqNGMgC
015F8q9+UuDwZx4KBZm0j8IPOPzJ2nYck5dYDhyMHZDq1LJ4zJfgPQGQ5Wwq2Bwm
2RHDhAddTth6YZArs/z9hAqtA9gqMPnMPcdQpIv1sHS0n06zBJD8sJQA+k0XG+Y2
GS8NakLcUV1DpNd/Q+QHkv4Aw1ge2EF8QvmKtu/9reK0BqWnm2Tpad6RtAhZwPJX
Uhd9yiesTF6rjZ1ZcMGXUaNSRt0zD3D4zowRz2JLTCe4GkiJmtc3waN6hu1IaIqz
boI11evqnbatqnc4rCq8sf21yZqaLUIbwH41W2G3K8xMJN3iY8cgHTYneNYa+/d
7xyNW1M09SK1HsyaPcWv98BdD+At0x/6R6YPykeR+qXJ9ETGFKW4U6iNbBQXOMB
kZb1Ry8vFMH8vsYIzh8Edg6aq00S5cU57KiD5/Gc8KuqI6vmf21cCdCa487kCgW6
cGXQ2bLZGYBiMZfFO01pCQECgcwA5ZU3/8yS0duNhsDz3sgC2u40HwHUBxu50Ua
a5t4CoUY9iuf7b7qhbEcdvLgI01XA5xo+r4p0xgbLVdUTsRR1mrDM2+wrCjjwXCw
pAFMR12Rr72yLUC7N0WnNcoUshrNL4x/1j8T4WLRcannpXcor+/kn1r2rdLEBRcc+
zRTAdJlMPT4kwJehTE9Mzw2/03GX3MeLzVzJk1zvpCGw20N/2Yqjs++v5hXoHPs
21y6y6/FV097dvFctf7NahS04Jsjubfnj0Mx89AUNZsCgcwA1DfabCGJ5SckmQ+mg
2q91DPjz6r29wmBtYyT20oZ2kd4QBHR0p0t59yG4bvdRqZG/Dr5LJdVdVFRKroh1
/JJ7rLz/ZBQCLRS5t7/G2B0kBDOMMM+02wR60CTmxUhmGvs0ZWRp5Kkha5PSvZa
Wau2CN3mXNK72RL3RFUvuhNynkOEj50au1RaGgpZ0B0JTKYI9nffbe8SxU+DV8MC
gcwA18be28T15Fxyg+/IGQ3EBHFcuctITDQaA2Ew/8pTfK+z0kr9yYISsKXUuaSk
+skghkPcruGw8LgabH4GT/zGu+1H4btyekSBxeCtFqTtpED1WJOWD2ozi7NXSjd
YrhF+VCcMCA7ekOqS5HjkmT4XMO/wPab4VFEKzLnHzQ1cZB3ke7/4/0HnD5cIE7
vWNeRcDvdYRggT+wBX+Y6bXP142Smj8uyu1oDmpmR5ZUCnTdqT408K/RT0x4jCeC
CUH6v5rVil107bS4CdkCgctXvnQwCzmmvVrV744Tftuh81TwHnqGwaA/LKU3wW9
T/x9ba1uHFHXkaWvRba61LiCDGpsYM4hwTYokqYnfbC2rvOWO6rtnX1P1An3y61V
ovQfgDeNiFmIyvniPEEM0JZA+QnburLYwOx4DgwYvyBnpa18WP08c3L/J4hkWlM
Pc30Dj0XUumLevAnCv0jvgSfW8NenSVfzw+KToIEKaP0rWfJZUWDA479vY6D
UNwRjPntYIwtSAV+FPvRvInk0ZeHamW9H+D1cwkBy2euc93gruYdtFej/biGASD
+H... (truncated)



Identification par mot de passe



Alice
mot de passe d'Alice **BELOTE**

BELOTE est envoyé



à Bob
mot de passe de Bob **REBELOTE**

REBELOTE est envoyé à Alice



Identification par mot de passe

- Alice et Bob doivent convenir d'un mot de passe secret partagé (question secrète)
- Avantage : simple
- Fragilités : risque de réutilisation e.g. par un tiers, gestion de mots de passe



Identification sans divulgation de connaissance



Alice connaît un secret S_{Alice}



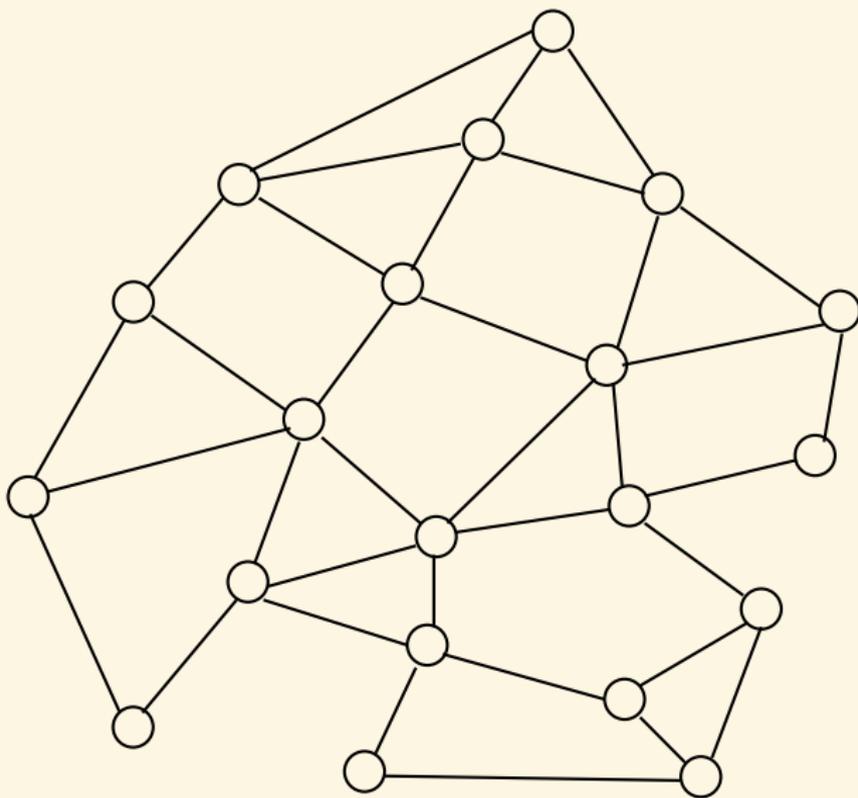
Bob

interroge Alice et se convainc qu'elle connaît bien le secret.

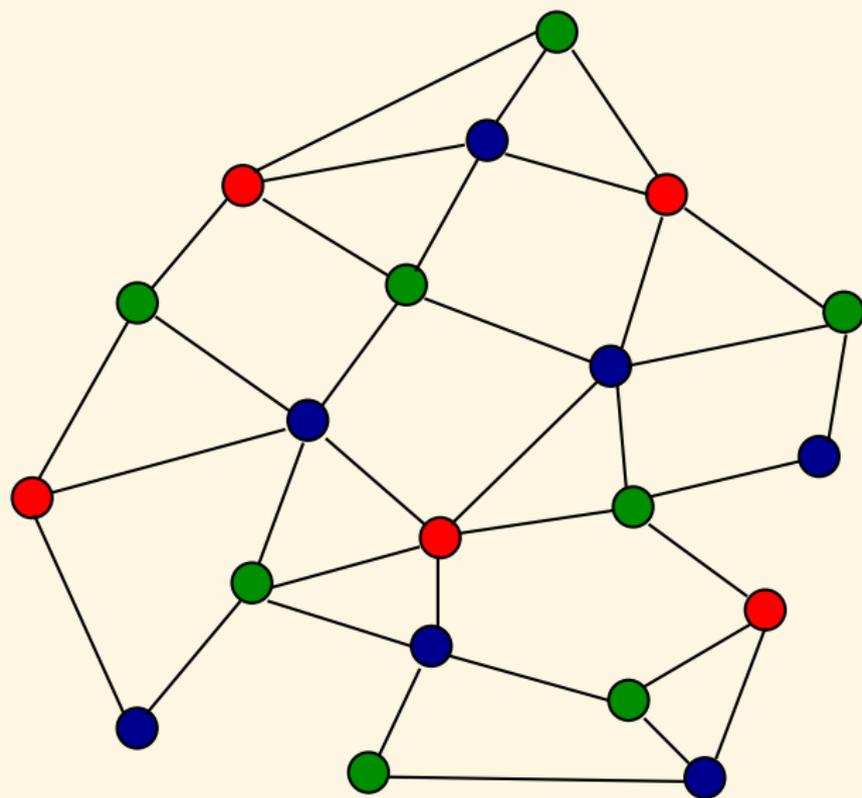
À la fin de l'échange, Bob n'a rien appris sur ce secret!



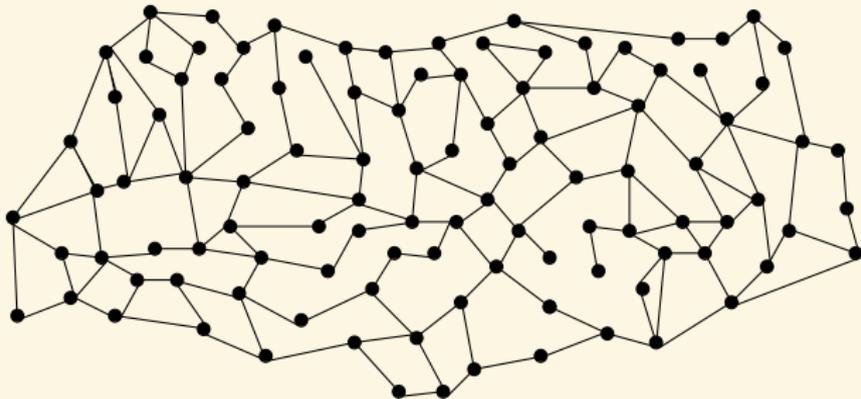
Coloriage de graphe



Coloriage de graphe



Coloriage de graphe



Zero Knowledge Proofs

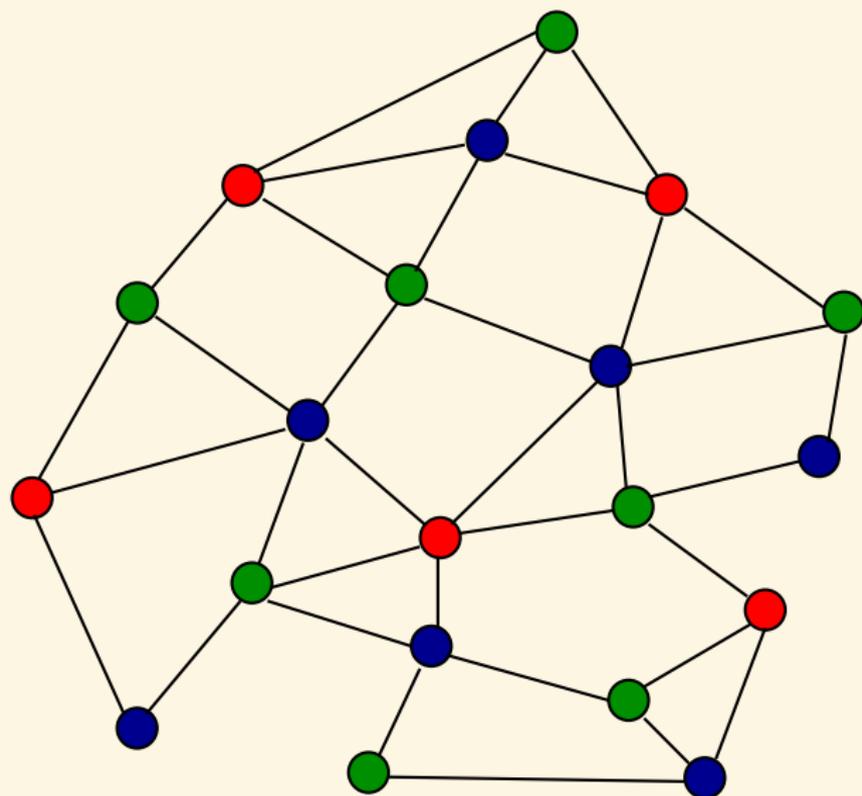


Shafi Goldwasser (1981)

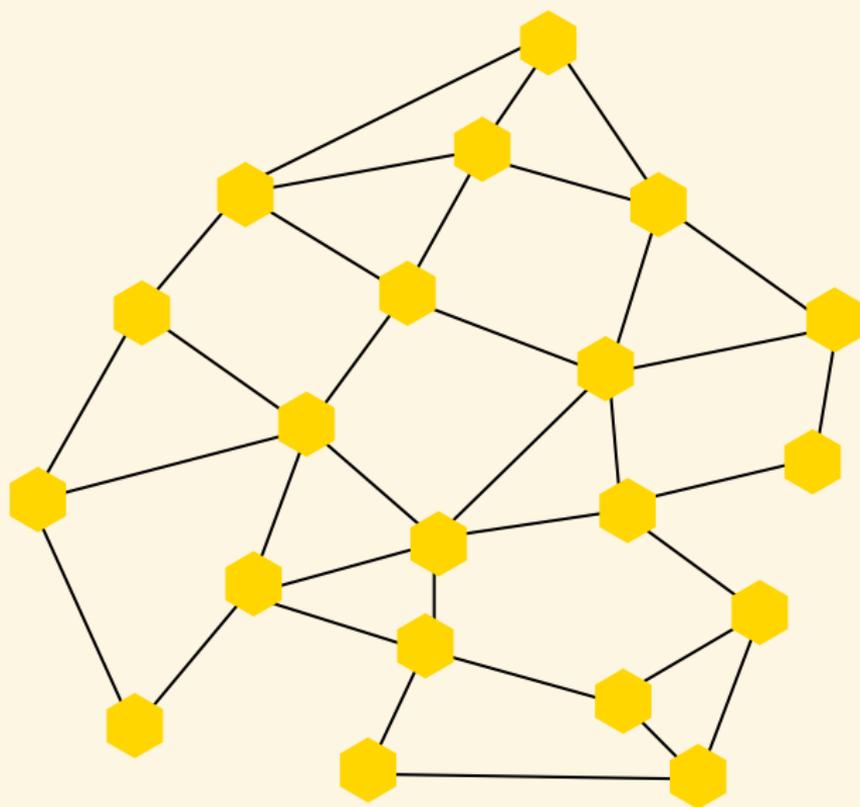


Oded Goldreich (1991)

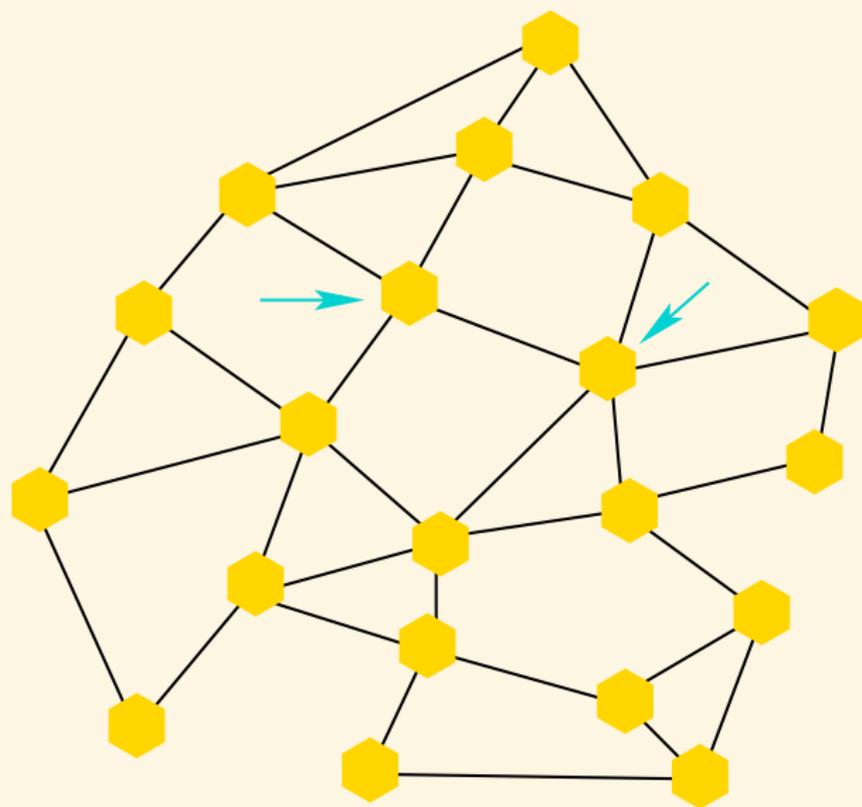
Le coloriage d'Alice (secret)



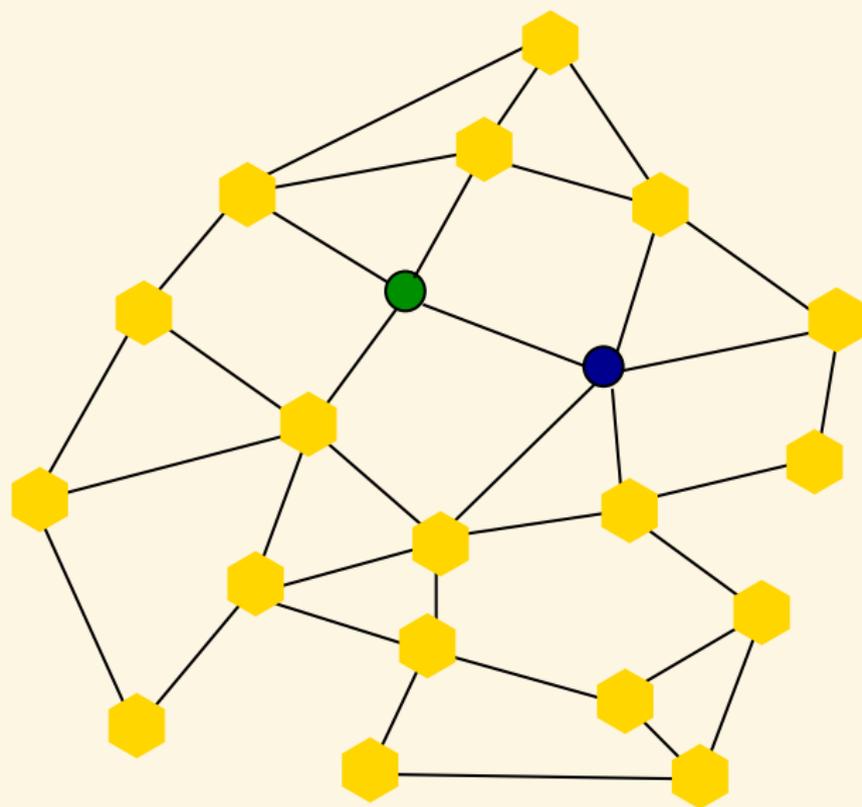
Le coloriage d'Alice caché



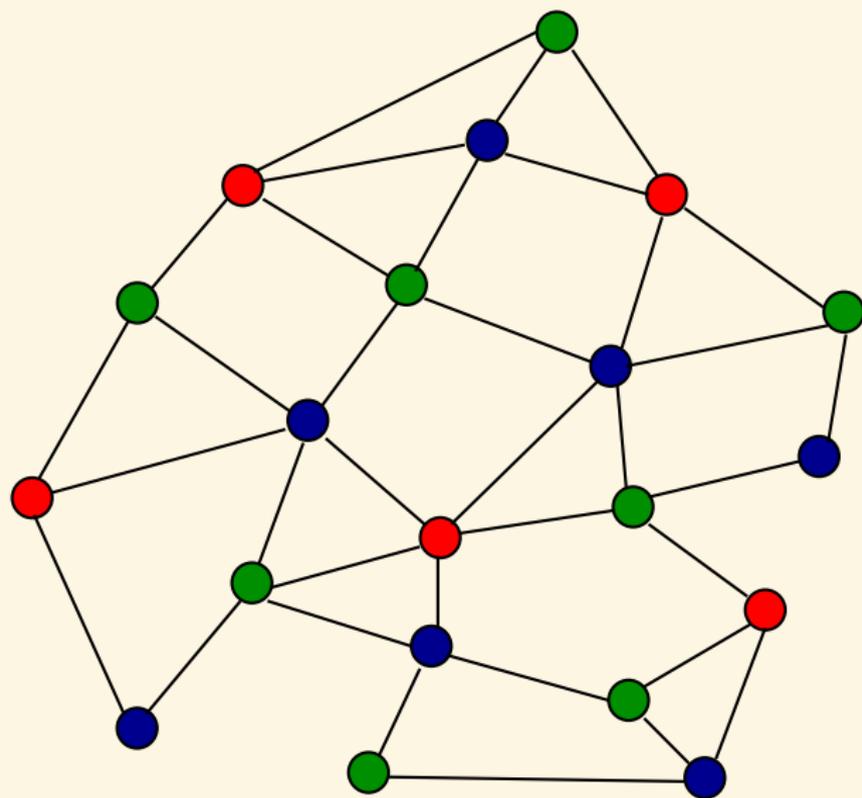
La question de Bob



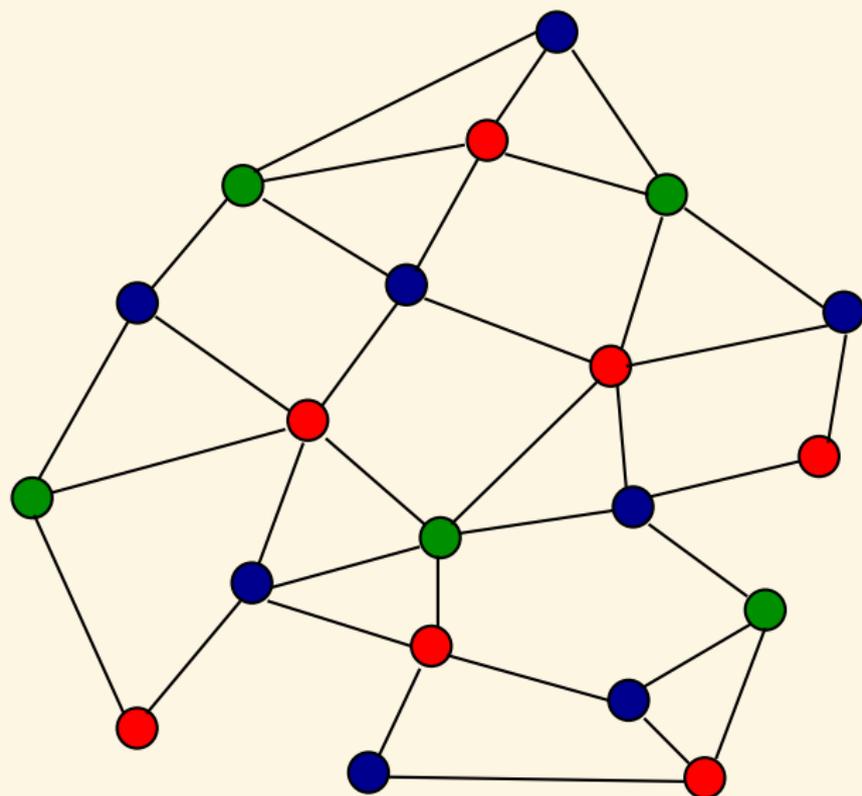
Dévoilement



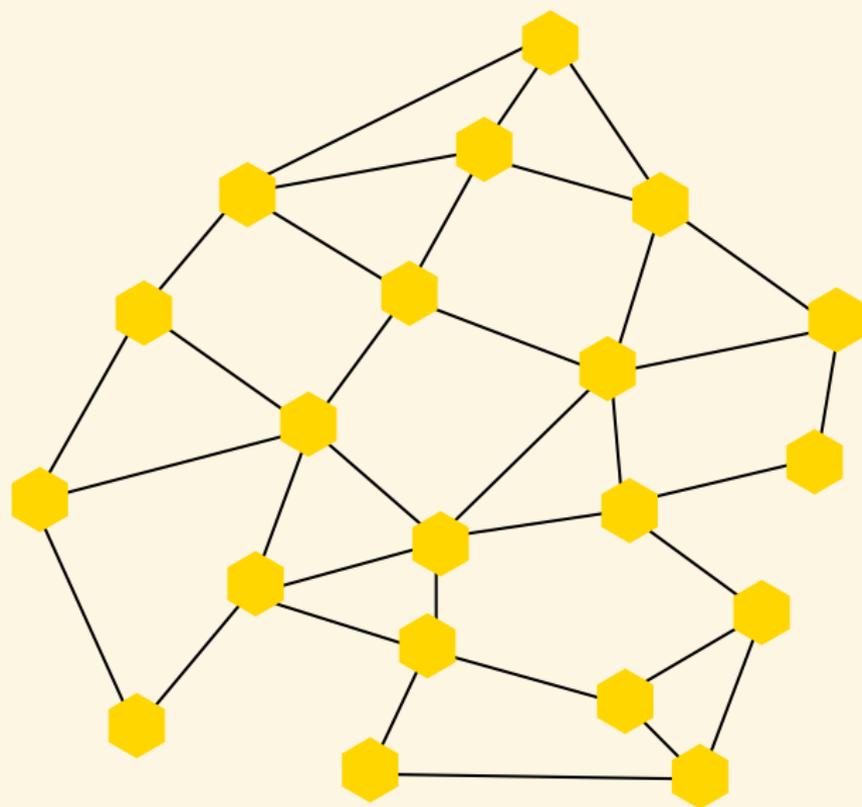
Le coloriage d'Alice (secret)



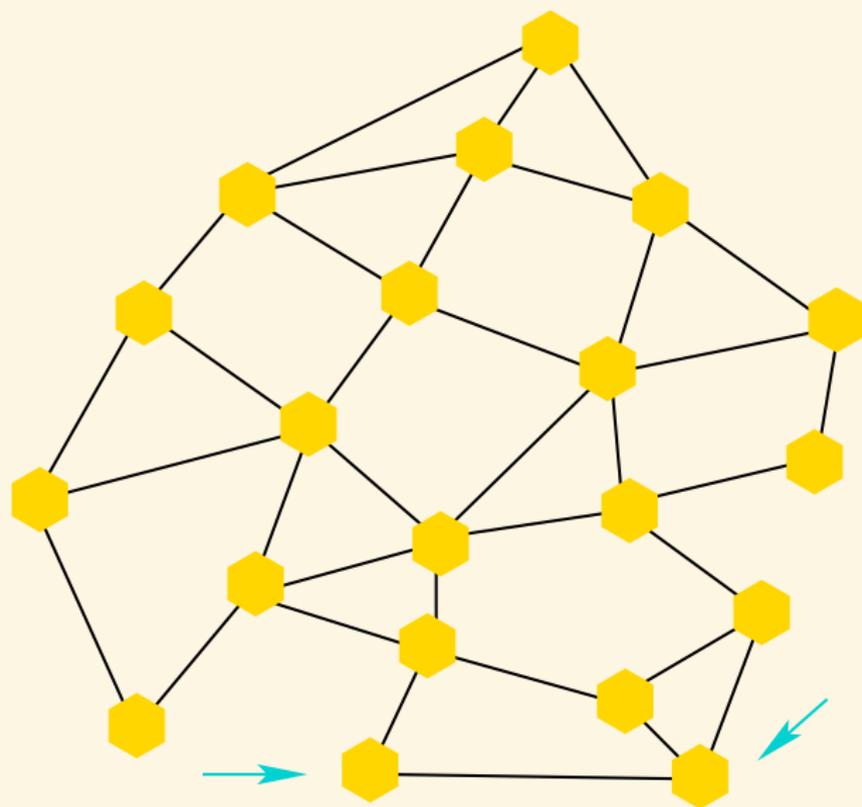
Le coloriage d'Alice permuté



Le coloriage d'Alice caché



La deuxième question de Bob



Dévoilement

