

# Cryptologie

Master Informatique :  
Enjeux juridiques et déontologiques de l'informatique

Fabien LAGUILLAUMIE

Professeur à l'Université Claude Bernard Lyon 1

`fabien.laguillaumie@ens-lyon.fr`  
`http://perso.ens-lyon.fr/fabien.laguillaumie`





## Introduction

Risques informatiques

Recommandations sur la taille des clés

## Un peu d'histoire et de principes

L'âge artisanal

L'âge technique

L'âge paradoxal

La confidentialité et un chiffrement parfait

## Focus : Cryptographie à clé publique

RSA

Le paradoxe : Algorithmique efficace vs. algorithmique non-efficace

## Échange de clé

# Risques informatiques

Quels risques pour quels besoin ?

gestion de la comptabilité, cloud, développement, production logicielle, échanges bancaires, prévisions climatiques, secrétariat, défense,...

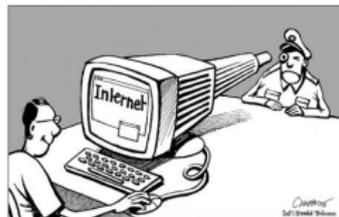
- ▶ Risques humains
- ▶ Risques techniques
- ▶ Risques juridiques

# Risques informatiques

Quels risques pour quels besoins ?

- ▶ Risques humains

- ▶ maladresse
- ▶ inconscience
- ▶ malveillance  $\rightsquigarrow$  ingénierie sociale, espionnage

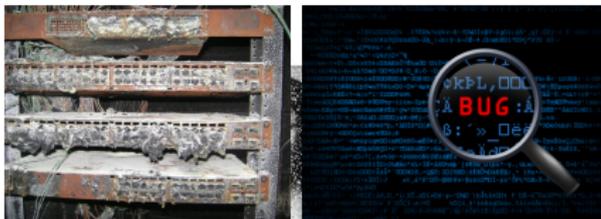


- ▶ Risques techniques
- ▶ Risques juridiques

# Risques informatiques

Quels risques pour quels besoin ?

- ▶ Risques humains
- ▶ Risques techniques
  - ▶ incidents liés au matériel
  - ▶ incidents liés au logiciel
  - ▶ incidents liés à l'environnement



## ARIANE 5 FAILURE

- ▶ BACKGROUND:-
  - ▶ European space agency's re-useable launch vehicle.
  - ▶ Ariane-4 was a major success
  - ▶ Ariane -5 was developed for the larger payloads
- ▶ LAUNCHED:-on June 4 1996
- ▶ MISSION was to delivered \$500 million payloads to the orbit
- ▶ THE MAIDEN FLIGHT OF THE ARIANE 5 ENDED IN A FAILURE.
- ▶ ONLY AFTER 40 SECONDS THE FLIGHT VEERED OFF ITS PATH AND BROKE UP AND EXPLODED
- ▶ CAUSE: Unhandled floating point exception in code
- ▶ ENGINEERS FROM THE ARIANE PROJECT STARTED TO INVESTIGATE THE CAUSES OF LAUNCH FAILURE.

# Risques informatiques

Quels risques pour quels besoin ?

- ▶ Risques humains
- ▶ Risques techniques
- ▶ Risques juridiques
  - ▶ non-respect de la législation relative à la signature numérique
  - ▶ protection du patrimoine informationnel
  - ▶ non-respect de la vie privée
  - ▶ droit de la preuve



**CNIL**  
Commission Nationale de l'Informatique et des Libertés



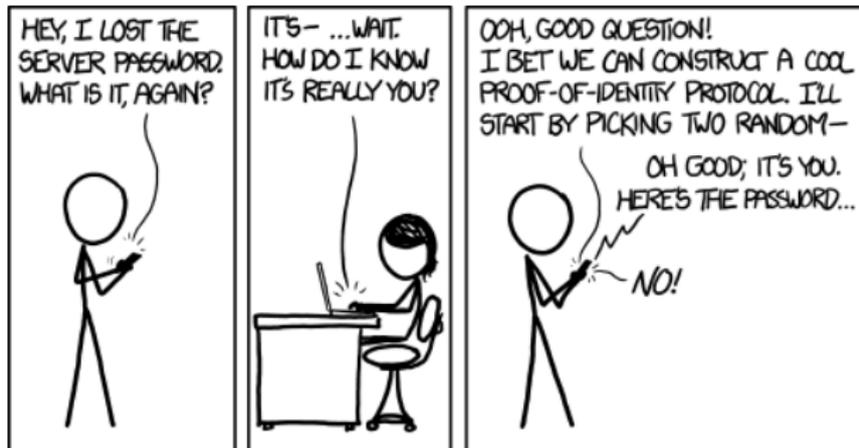
# Risques informatiques

## Conséquences :

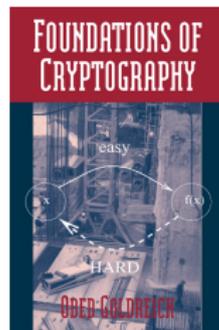
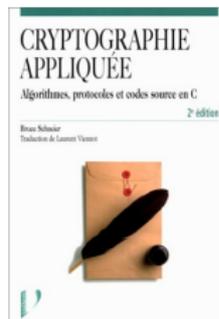
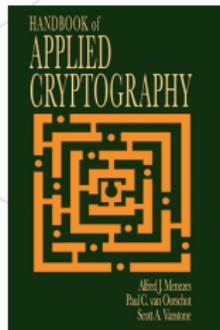
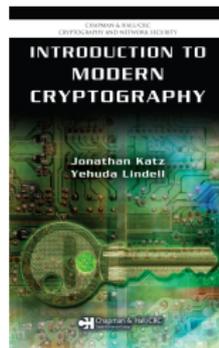
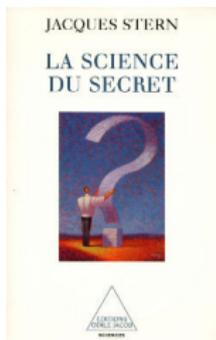
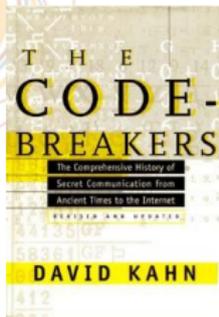
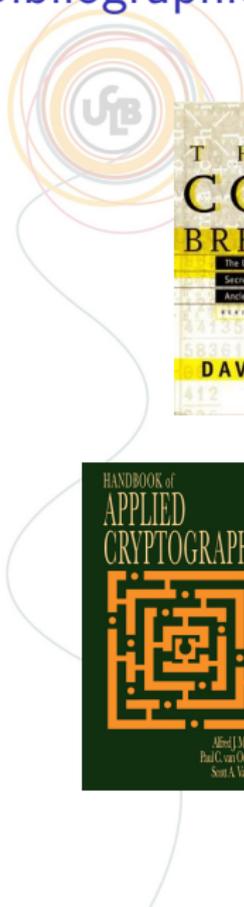
- ▶ données perdues ou altérées, bref inexploitable
- ▶ données ou traitements durablement indisponibles  $\rightsquigarrow$  arrêt d'une production ou d'un service
- ▶ divulgation d'informations confidentielles ou erronées  $\rightsquigarrow$  profits à des sociétés concurrentes ou nuisance à l'image
- ▶ déclenchement d'actions pouvant provoquer des accidents physiques



## LA CRYPTOGRAPHIE



# Bibliographie



# Introduction

Cryptologie = science du secret et de la confiance

▶ Oded Goldreich (Weizmann Institute of Science) :

« Cryptography is concerned with the construction of schemes that withstand any abuse : Such schemes are constructed so to maintain a desired functionality, even under malicious attempts aimed at making them deviate from their prescribed functionality. »



# Introduction

Cryptologie = science du secret et de la confiance

Dans la vraie vie :

- ▶ Internet :
  - ▶ sites bancaires
  - ▶ sites de vente en ligne
  - ▶ site d'enchères
  - ▶ ...



# Introduction

Dans la vraie vie :

- ▶ Carte à puce
- ▶ cartes de paiements
- ▶ carte vitale



# Introduction

Dans la vraie vie :

- ▶ Signature électronique (<http://www.ssi.gouv.fr>)



La signature électronique permet, à l'aide d'un procédé cryptographique, de garantir l'intégrité du document signé et l'identité du signataire.

L'écrit électronique signé électroniquement peut être reconnu comme preuve en justice. L'ANSSI a publié un mémento visant à dresser le cadre juridique autour de la signature électronique. Partant d'un rappel sur le contexte législatif, il expose, au jour d'aujourd'hui, le cadre technique défini pour la mise en œuvre d'une signature électronique présumée fiable au sens du décret 2001-272 sur la signature électronique.

Pour l'ensemble du vocabulaire utilisé dans ce document il est conseillé de se référer à la FAQ « Décret n° 2001-272 du 30 mars 2001 pris pour l'application de l'article 1316-4 du code civil et relatif à la signature électronique ».

Le procédé de signature électronique est présumé fiable, au sens du décret 2001-272 sur la signature électronique, si :

- ▶ la signature électronique est sécurisée ;
- ▶ elle est créée par un dispositif sécurisé de création de signature, c'est à dire par un dispositif certifié conforme aux exigences de l'article 3. I du décret conformément à la procédure de "Certification de conformité des dispositifs de création de signature électronique" ;
- ▶ et la vérification de cette signature repose sur l'utilisation d'un certificat électronique qualifié. Les certificats délivrés par des "prestataires de services de certification électronique qualifiés" sont présumés qualifiés.

# Introduction

Dans la vraie vie :



PREMIER MINISTRE  
Secrétariat général de la défense nationale  
Direction centrale de la sécurité des systèmes d'information  
Sous-direction des opérations  
Bureau conseil

Signature électronique  
Point de situation

---

**MEMENTO**

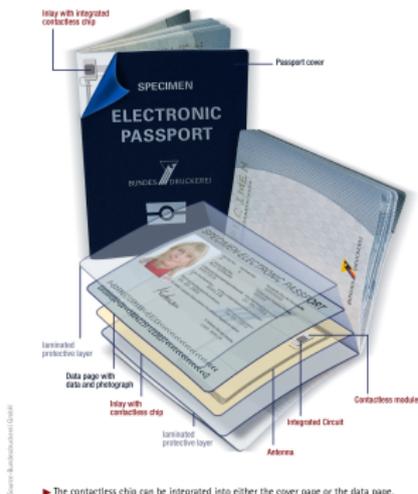
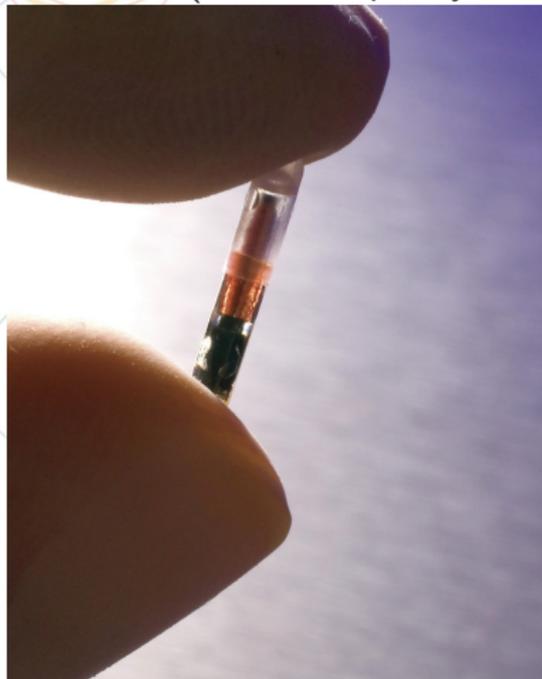
Version 0.94  
25.08.04

# Introduction

Cryptologie = science du secret et de la confiance

Dans la vraie vie :

- ▶ RFID (**R**adio-**F**requency **I**Dentification)



- ▶ RFID Security & Privacy Lounge <http://www.avoine.net/rfid/>

# Introduction

Cryptologie = science du secret et de la confiance

Dans la vraie vie :



# Introduction

Cryptologie = science du secret et de la confiance

Dans la vraie vie :

▶ identification animale

▶ identification VIP



# Introduction

Cryptologie = science du secret et de la confiance

Dans la vraie vie :



# Introduction

Cryptologie = science du secret et de la confiance

Dans la vraie vie :



## Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars

Aurélien Francillon, Boris Danev, Srdjan Čapkun  
Department of Computer Science  
ETH Zurich

8092 Zurich, Switzerland

{aurelien.francillon, boris.danev, srdjan.capkun}@inf.ethz.ch

# Introduction

Dans la vraie vie :

- ▶ Télé payante
- ▶ décodeur
- ▶ pay-tv



# Introduction

Dans la vraie vie :

- ▶ Télécommunications

- ▶ GSM

- ▶ Wifi



- ▶ Hybrid fixed/mobile phone enabling communications both over fixed (PSTN, ISDN, VoIP) and mobile (Quadri-Band GSM, GPRS Class 10, EDGE, UMTS) telecom networks
- ▶ Vocoder ensuring secure and high-quality speech : STANAG 4591 (2.4 kbps) and G.728 (16 kbps)
- ▶ Security level : High Grade (up to French « SECRET DÉFENSE »)

# Introduction

## ▶ Mail à la liste Crypto de l'ÉNS (16 août 2010) : Vodafone Mobile Algorithms

New Mobile Phone Security Algorithms - Public Evaluation Invited

A new set of cryptographic algorithms is being proposed for inclusion in the "4G" mobile standard called LTE (Long Term Evolution).

The algorithms are :

- \* a stream cipher called ZUC, which is the core of both new LTE algorithms;
- \* the LTE encryption algorithm called 128-EEA3, defined straightforwardly using ZUC;
- \* the LTE integrity algorithm called 128-EIA3, designed as a Universal Hash Function using ZUC as its core.

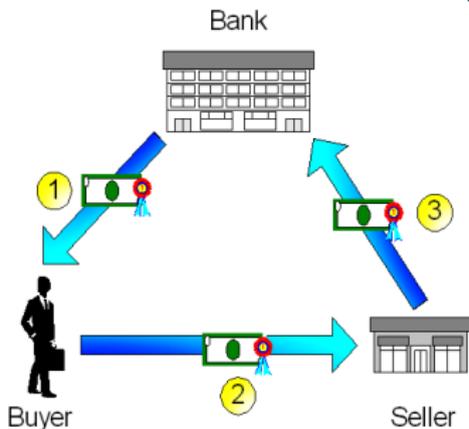
The algorithms are here : [http://gsmworld.com/our-work/programmes-and-initiatives/fraud-and-security/gsm\\_security\\_algorithms.htm](http://gsmworld.com/our-work/programmes-and-initiatives/fraud-and-security/gsm_security_algorithms.htm). All of the algorithms were designed by DACAS, the Data Assurance and Communication Security Research Center of the Chinese Academy of Sciences. They have been evaluated by the algorithm standardisation group ETSI SAGE, and also by two other teams of well known cryptologists, and are believed to be strong and suitable for LTE.

Now the algorithms are open for public evaluation. Comments and analysis are invited, before a final decision is taken in (probably) January 2011 as to whether to include the new algorithms in the LTE standard. A discussion forum <http://zucalg.forumotion.net/> has been created for this - please post any evaluation results there.

# Introduction

Dans la vraie vie :

- ▶ Paiement
  - ▶ porte-monnaie électronique
  - ▶ cryptocurrency
  - ▶ e-cash



# Introduction

Récemment :

## Affaire Snowden : comment la NSA déjoue le chiffrement des communications

Le Monde.fr | 05.09.2013 à 23h28 • Mis à jour le 06.09.2013 à 19h14

Abonnez-vous  
à partir de 1 €



Réagir



Classer



Partager



Recommander

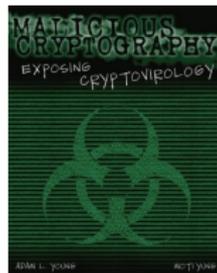
Envoyer



1 695 personnes le recommandent.



Les désormais célèbres documents d'Edward Snowden, l'ancien consultant de l'Agence de sécurité nationale (NSA) viennent d'éclaircir une facette encore





EDITORIAL  
Leaving the E.U. Would Hurt Britain's Economy



CHARLES M. BLOW  
The End of American Idealism

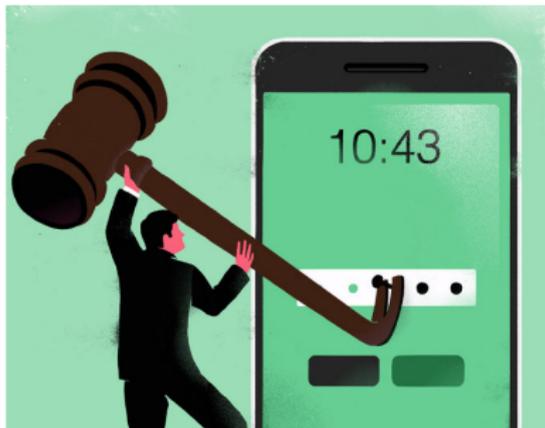


PAUL KRUGMAN  
When Fallacies Coll

The Opinion Pages OP-ED CONTRIBUTORS

## When Phone Encryption Blocks Justice

By CYRUS R. VANCE Jr., FRANÇOIS MOLINS, ADRIAN LEPPARD and JAVIER ZARAGOZA AUG. 11, 2015



## François Molins: "Les nouveaux téléphones rendent la justice aveugle"

Actualité | Société | Propos recueillis par Emmanuel Piquette et Etic-Pelletier, publiés le 02/09/2015 à 08:57

521 commentaires

Partager Partager Partager



www.lesdevoir.com



## TRIBUNE Sécurité informatique : tous connectés, tous responsables

Par Guillaume Poupard, Directeur général de l'Agence nationale de la sécurité des systèmes d'information (ANSSI) — 21 janvier 2019 à 09:59 (mis à jour le 22 janvier 2019 à 12:31)



Partager Tweeter

# Introduction

Google News : cryptograph(y/ie)



9T05Mac

Exchanges Codes Mac iPad iPhone Watch TV Music

JANUARY 2

USB-C upgrade allows cryptography to authenticate connected devices

Ben Long - Jan 28 2019 7:00am PT @bentlongy



ORIGINE ET PROTECTION DES DONNÉES : DÉBATTRE POUR RÉSOUDRE LA «CRISE DE CONFIANCE»

Par Amanda Dubois - 20 Janvier 2018 à 11:40

f FB TW

Forbes

Billionaires Innovation Leadership Money Consumer Industry

3,900 views | Jan 17, 2019, 04:34pm

## IBM Lattice Cryptography Is Needed Now To Defend Against Quantum Computing Future

Kevin Krawell Contributor  
Texas Research Contributor Group @  
Enterprise & Cloud

When it comes to securing data, it is not too early to start anticipating the future threat of quantum computing. Today's cryptographic

InformationWeek Dark Reading Network Computing About Us Adve

IT NETWORK

DARKReading

Join us live at  
Interop

Authors Slideshows Video Tech Library University Radio Calendar Black Hat News

ANALYTICS ATTACKS/BREACHES APP SEC CAREERS & PEOPLE CLOUD ENDPOINT IoT MOBILE OPERATIONS

ATTACKS/BREACHES

1/22/2019  
02:30 PM



## The Fact and Fiction of Homomorphic Encryption

The approach's promise continues to entice cryptographers and academics. But don't expect it to help in the real world anytime soon.

The history of homomorphic encryption stretches back to the late 1970s. Just

# Introduction

Le cœur de la crypto :

- ▶ échange de clés
- ▶ sécurité des communications (confidentialité, intégrité)

# Introduction

Le cœur de la crypto :

- ▶ échange de clés
- ▶ sécurité des communications (confidentialité, intégrité)

mais encore

- ▶ signatures numériques
- ▶ communications anonymes
- ▶ protocoles : vote, e-cash, enchères, interrogation anonyme de BD
- ▶ **multi-party computation** (thm : c'est possible !)



# Introduction

Le cœur de la crypto :

- ▶ échange de clés
- ▶ sécurité des communications (confidentialité, intégrité)

mais encore

- ▶ signatures numériques
- ▶ communications anonymes
- ▶ protocoles : vote, e-cash, enchères, interrogation anonyme de BD
- ▶ **multi-party computation** (thm : c'est possible !)

et la magie :

- ▶ preuves à divulgation nulle de connaissance
- ▶ calculs secrets délégués

outsourcing computation

search  
query



$E[\text{query}]$

$E[\text{results}]$

results

What did she  
search for?



Google





## Zoom : Vote électronique

# Introduction

Google query :



YOU ARE HERE: [HOME](#) / / / [ELECTRONIC VOTING LIKELY TO BECOME WIDESPREAD IN KOREA](#)

## Enhancement Of Stockholder Rights Electronic Voting Likely to Become Widespread in Korea



ZDNet.fr > Blogs > L'esprit libre > Vote électronique: Axelle Lemaire trolle >

## Vote électronique: Axelle Lemaire trolle

**Société** : La secrétaire d'État au Numérique a exposé vendredi sa philosophie en matière de logiciel libre, qui "petit à petit s'impose", et d'autres points de son projet de loi numérique. Elle a aussi lancé un tweet en faveur du vote par Internet, avant de

# Introduction

[http://www.coe.int/t/DEMOCRACY/ELECTORAL-ASSISTANCE/themes/evoting/default\\_en.asp](http://www.coe.int/t/DEMOCRACY/ELECTORAL-ASSISTANCE/themes/evoting/default_en.asp)

« The Council of Europe continues to be the only organisation that has set intergovernmental standards in the field of e-voting. »

**E-voting refers to an election or referendum that involves the use of electronic means in at least the casting of the vote.**

- ▶ vote *off-line* : machines à voter, non connectées, vote supervisé (par des autorités électorales)
- ▶ vote *mixte* : machines connectées et vote supervisé
- ▶ vote *on-line* : par internet, sur son ordinateur/mobile, non supervisé



# Introduction

## Avantages :

- ▶ réduction des coûts d'une élection
- ▶ résultats rapides et sûrs
- ▶ permettre à plus de gens de voter
- ▶ facilité de mise en œuvre
- ▶ plusieurs types de votes supportés

## Scrutins uninominaux

- ▶ majoritaire à un tour

1993 - Vénézuela - Rafael Caldera élu à la présidence avec 30,5 % des suffrages exprimés

- ▶ majoritaire à deux tours

surreprésente la majorité parlementaire et sanctionne les minorités

- ▶ vote alternatif : classement par ordre de préférence, puis élimination du dernier et reclassement jusqu'à majorité absolue

dépouillement coûteux

# Vote électronique



## Propriétés de sécurité :

- ▶ Éligibilité
- ▶ Confidentialité du bulletin
- ▶ Absence de reçu
- ▶ Résistance à la coercition
- ▶ Vérifiabilité individuelle
- ▶ Vérifiabilité universelle
- ▶ Équité

Eligibility

Ballot secrecy

Receipt-freeness

Coercion-resistance

Fairness

# Vote électronique



## Propriétés de sécurité :

- ▶ **Éligibilité** Eligibility  
seuls les électeurs légitimes peuvent voter, et une seule fois
- ▶ **Confidentialité du bulletin** Ballot secrecy
- ▶ **Absence de reçu** Receipt-freeness
- ▶ **Résistance à la coercition** Coercion-resistance
- ▶ **Vérifiabilité individuelle**
- ▶ **Vérifiabilité universelle**
- ▶ **Équité** Fairness

# Vote électronique



## Propriétés de sécurité :

- ▶ **Éligibilité** Eligibility
- ▶ **Confidentialité du bulletin** Ballot secrecy  
impossible de connaître le vote d'un électeur  
long terme vs éternel
- ▶ **Absence de reçu** Receipt-freeness
- ▶ **Résistance à la coercition** Coercion-resistance
- ▶ **Vérifiabilité individuelle**
- ▶ **Vérifiabilité universelle**
- ▶ **Équité** Fairness

# Vote électronique



## Propriétés de sécurité :

- ▶ **Éligibilité** Eligibility
- ▶ **Confidentialité du bulletin** Ballot secrecy
- ▶ **Absence de reçu** Receipt-freeness  
impossible de prouver pour qui un électeur a voté *après* le vote
- ▶ **Résistance à la coercition** Coercion-resistance
- ▶ **Vérifiabilité individuelle**
- ▶ **Vérifiabilité universelle**
- ▶ **Équité** Fairness

# Vote électronique



## Propriétés de sécurité :

- ▶ **Éligibilité** Eligibility
- ▶ **Confidentialité du bulletin** Ballot secrecy
- ▶ **Absence de reçu** Receipt-freeness
- ▶ **Résistance à la coercition** Coercion-resistance  
impossible de forcer à voter pour quelqu'un ou à s'abstenir
- ▶ **Vérifiabilité individuelle**
- ▶ **Vérifiabilité universelle**
- ▶ **Équité** Fairness

# Vote électronique



## Propriétés de sécurité :

- ▶ **Éligibilité** Eligibility
- ▶ **Confidentialité du bulletin** Ballot secrecy
- ▶ **Absence de reçu** Receipt-freeness
- ▶ **Résistance à la coercition** Coercion-resistance
- ▶ **Vérifiabilité individuelle**  
Un électeur peut s'assurer que son vote a bien été pris en compte
- ▶ **Vérifiabilité universelle**
- ▶ **Équité** Fairness

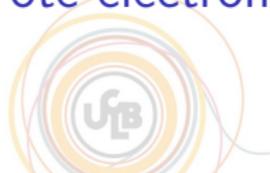
# Vote électronique



## Propriétés de sécurité :

- ▶ **Éligibilité** Eligibility
- ▶ **Confidentialité du bulletin** Ballot secrecy
- ▶ **Absence de reçu** Receipt-freeness
- ▶ **Résistance à la coercition** Coercion-resistance
- ▶ **Vérifiabilité individuelle**
- ▶ **Vérifiabilité universelle**  
Quiconque peut vérifier que le décompte est fait à partir de bulletins légitimes et qu'il est correct
- ▶ **Équité** Fairness

# Vote électronique



Propriétés de sécurité :

- ▶ Éligibilité
  - ▶ Confidentialité du bulletin
  - ▶ Absence de reçu
  - ▶ Résistance à la coercition
  - ▶ Vérifiabilité individuelle
  - ▶ Vérifiabilité universelle
  - ▶ Équité
- Pas de résultats intermédiaires

Eligibility

Ballot secrecy

Receipt-freeness

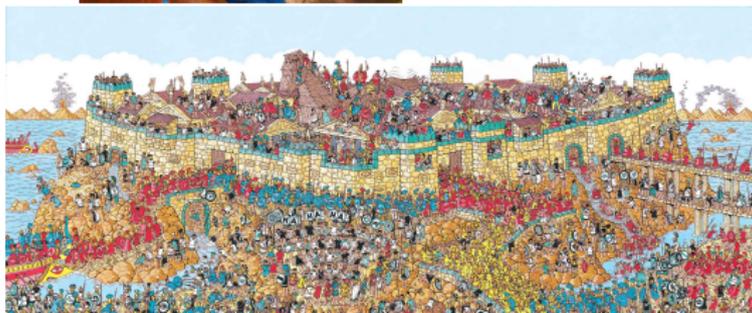
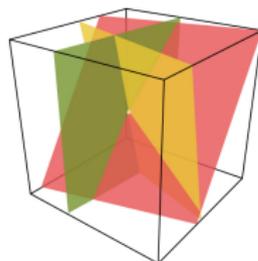
Coercion-resistance

Fairness

# Introduction

## Outils cryptographiques :

- ▶ Signatures
- ▶ Chiffrement homomorphe
- ▶ Déchiffrement à seuil
- ▶ Preuve à divulgation nulle de connaissance



# Partage de secret de Shamir

- ▶ Un secret est partagé entre  $n$  entités. part
- ▶ Une coopération entre au moins  $t$  d'entre elles est nécessaire pour le reconstruire.  $2 \leq t \leq n$

1. Le *dealer*  $\mathcal{D}$  choisit un secret  $s \in \mathbb{F}_p$   $p \geq n + 1$
2.  $\mathcal{D}$  associe à chaque participant  $\mathcal{P}_i$  un identifiant public  $x_i$
3.  $\mathcal{D}$  tire uniformément  $t - 1$  éléments secrets  $a_1, \dots, a_{t-1}$  dans  $\mathbb{F}_p$
4. Pour tout  $1 \leq i \leq n$ ,  $\mathcal{D}$  calcule  $y_i = P(x_i)$  avec

$$P(X) = s + \sum_{i=1}^{t-1} a_i X^i.$$

5.  $\mathcal{D}$  donne secrètement à l'entité  $i$  la *part*  $y_i$ .

# Partage de secret de Shamir

**Complétude** : Tout groupe de  $t$  entités (ou plus) peut retrouver  $s$ .

$\rightsquigarrow$  retrouver  $P(X)$

- ▶  $U_{i_1}, \dots, U_{i_t}$  les  $t$  entités
- ▶ Pour tout  $1 \leq j \leq t$ ,  $y_{ij} = P(x_{ij})$
- ▶ Système linéaire :

$$\begin{pmatrix} 1 & x_{i_1} & x_{i_1}^2 & \dots & x_{i_1}^{t-1} \\ 1 & x_{i_2} & x_{i_2}^2 & \dots & x_{i_2}^{t-1} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & x_{i_t} & x_{i_t}^2 & \dots & x_{i_t}^{t-1} \end{pmatrix} \begin{pmatrix} s \\ a_1 \\ \vdots \\ a_t \end{pmatrix} = \begin{pmatrix} y_{i_1} \\ y_{i_2} \\ \vdots \\ y_{i_t} \end{pmatrix}$$

Matrice de Vandermonde de déterminant  $\prod_{1 \leq j < k \leq t} (x_{ij} - x_{ik})$ .

$\rightsquigarrow$  unique solution dans  $\mathbb{F}_p \rightsquigarrow s$

*interpolation de Lagrange*

# Partage de secret de Shamir

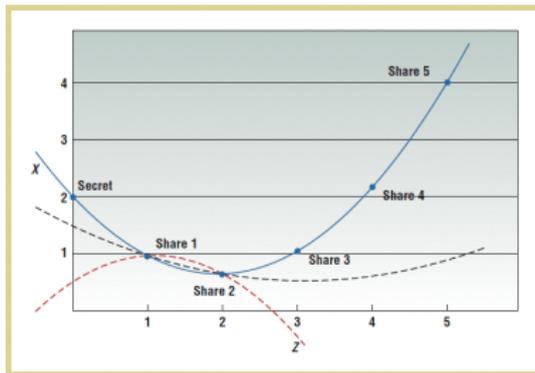
## Protection du secret :

- ▶  $P(X)$  est un polynôme aléatoire de degré  $t - 1$
- ▶  $P(0) = s$

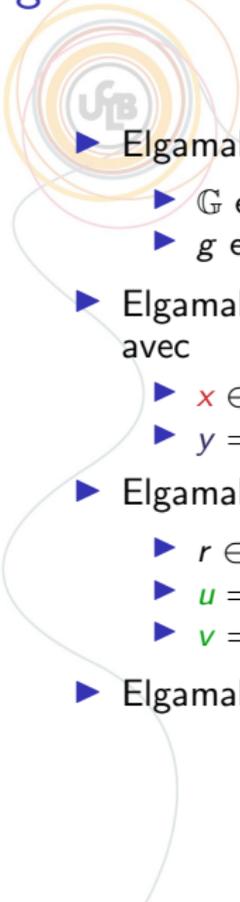
$t - 1$  personnes ne peuvent retrouver le secret !

$\forall k \in \mathbb{F}_p, \exists ! P_k(X)$  de degré  $t - 1$  :

$$P_k(0) = k \quad \text{et} \quad \begin{cases} P_k(x_{i_1}) = y_{i_1} \\ \vdots \\ P_k(x_{i_{t-1}}) = y_{i_{t-1}} \end{cases}$$



# Elgamal avec déchiffrement à seuil



▶  $\text{Elgamal.Setup}(k) \rightsquigarrow \{k, \mathbb{G}, g, q\}$

- ▶  $\mathbb{G}$  est un groupe cyclique d'ordre  $2^{k-1} < q < 2^k$
- ▶  $g$  est un générateur de  $\mathbb{G}$ .

▶  $\text{Elgamal.KeyGen}(\mathcal{P}) \rightsquigarrow (x, y)$

avec

- ▶  $x \in \llbracket 1, q-1 \rrbracket$
- ▶  $y = g^x$

▶  $\text{Elgamal.Encrypt}(\mathcal{P}, y, m) \rightsquigarrow c = (u, v)$

- ▶  $r \in_R \llbracket 1, q-1 \rrbracket$
- ▶  $u = g^r$
- ▶  $v = my^r$

▶  $\text{Elgamal.Decrypt}(\mathcal{P}, x, (u, v)) \rightsquigarrow m$

$$m = \frac{v}{u^x}$$

# Elgamal avec déchiffrement à seuil

- Un *dealer* de confiance génère  $(x, y)$  en utilisant Elgamal.KeyGen
- Il partage  $x$  en utilisant le partage de secret de Shamir
- Il envoie une part  $(x_i, y_i)$  de  $x$  à chaque entité de déchiffrement

[ Le déchiffrement se déroule normalement en utilisant Elgamal.Encrypt ]

Considérons  $t$  entités souhaitant déchiffrer  $c = (u, v)$

$$\mathcal{S} \subset \{1, \dots, n\}$$

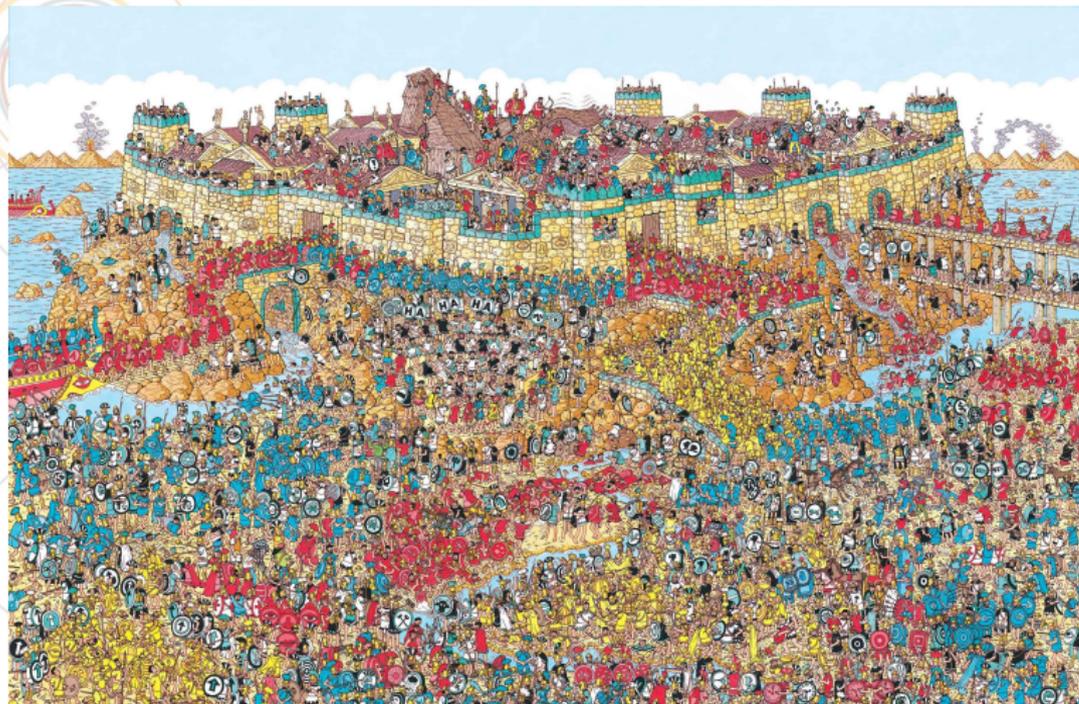
- ▶ Elles calculent  $d_i = u^{y_i}$
- ▶ Elles envoient  $d_i$  au *dealer*

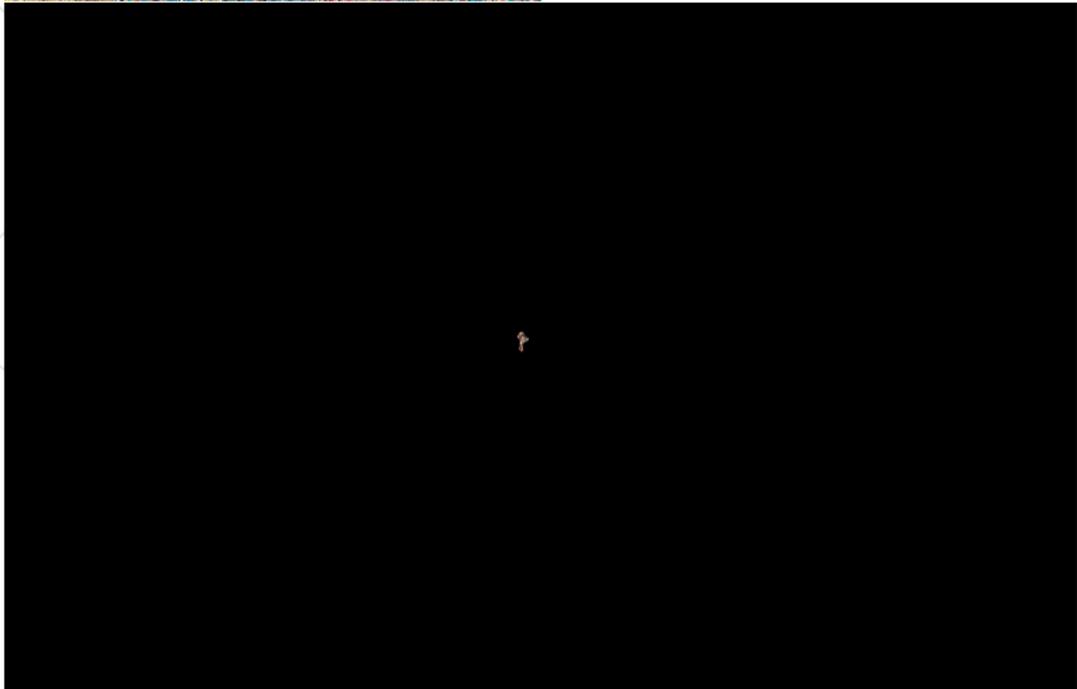
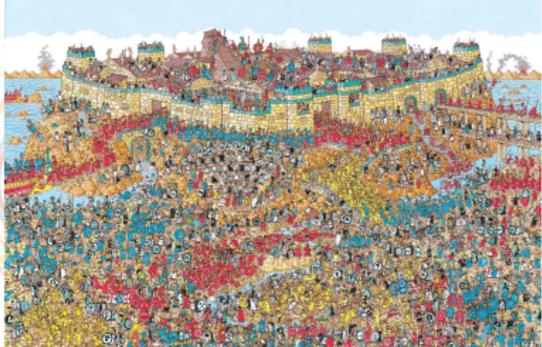
Le *dealer* calcule

$$\left( \prod_{i \in \mathcal{S}} d_i^{\ell_{i,0,\mathcal{S}}} \right)^{-1} v = m$$

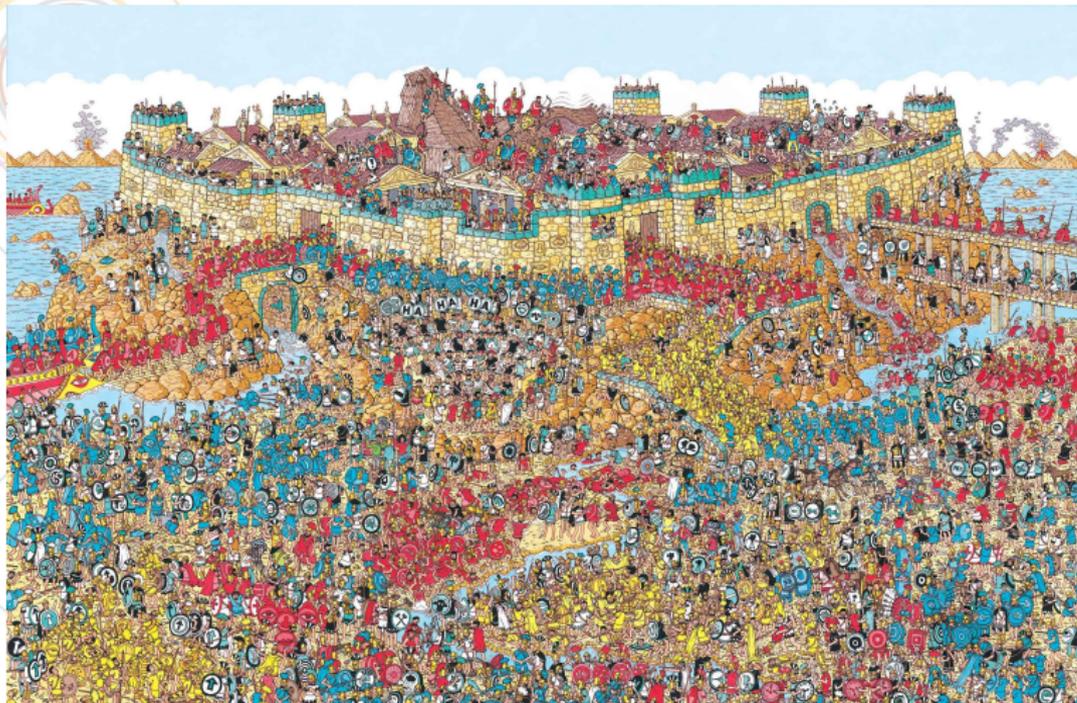
Avec  $\ell_{i,0,\mathcal{S}} = \prod_{j \in \mathcal{S}, j \neq i} \frac{-x_j}{x_i - x_j}$  tels que  $x = \sum_{i \in \mathcal{S}} y_i \ell_{i,0,\mathcal{S}}$ .

# Introduction

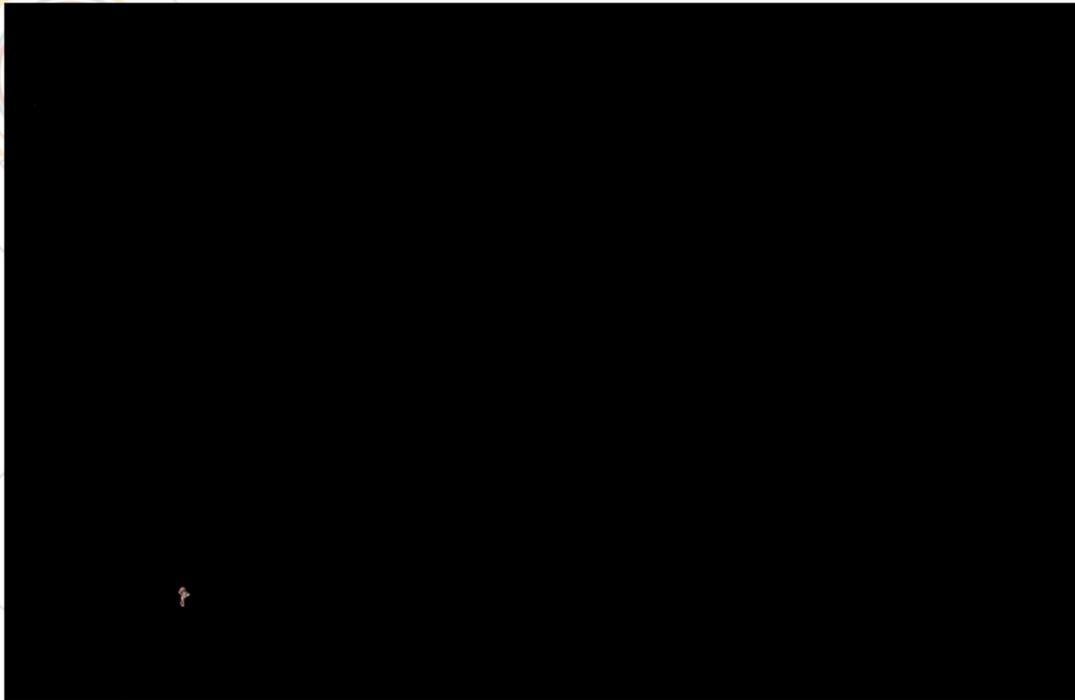




# Introduction



# Introduction



# Introduction

## Quelques grandeurs



B. Schneier. Cryptographie appliquée.

Probabilité de mourir foudroyé (par jour)	1 chance sur 9 milliards ( $2^{33}$ )
Probabilité de gagner le gros lot à la loterie américaine	1 chance sur 4 000 000 ( $2^{22}$ )
Probabilité de gagner le gros lot à la loterie américaine et de mourir le même jour	1 chance sur $2^{61}$
Probabilité d'être tué dans un accident automobile (aux États-Unis sur toute une vie)	1 chance sur 88 ( $2^7$ )
Âge de la Terre	$10^9$ années ( $2^{30}$ )
Âge de l'Univers	$10^{10}$ années ( $2^{34}$ )
Nombre d'atomes constituant l'Univers	$10^{77}$ ( $2^{265}$ )

# Introduction

Recommandations ANSSI

Mécanismes cryptographiques - Règles et recommandations,  
Rev. 1.20, ANSSI , 01/2010.

RègleCléSym-1. La taille minimale des clés symétriques utilisées jusqu'en 2020 est de 100 bits.

RègleCléSym-2. La taille minimale des clés symétriques devant être utilisées au-delà de 2020 est de 128 bits.

RecomCléSym-1. La taille minimale recommandée des clés symétriques est de 128 bits.

# Introduction

## Recommandations ANSSI



RègleAlgoBloc-1. Pour un algorithme de chiffrement ne devant pas être utilisé après 2020, aucune attaque nécessitant moins de  $Nop = 2^{100}$  opérations de calcul doit être connue.

RègleAlgoBloc-2. Pour un algorithme de chiffrement utilisé au-delà de 2020, aucune attaque nécessitant moins de  $Nop = 2^{128}$  opérations de calcul doit être connue.

RecomAlgoBloc-1. Il est recommandé d'employer des algorithmes de chiffrement par bloc largement éprouvés dans le milieu académique.



## Factorisation

RègleFact-1. La taille minimale du module est de 2048 bits, pour une utilisation ne devant pas dépasser l'année 2020.

RègleFact-2. Pour une utilisation au-delà de 2020, la taille minimale du module est de 4096 bits.

RègleFact-3. Les exposants secrets doivent être de même taille que le module.

RègleFact-4. Pour les applications de chiffrement, les exposants publics doivent être strictement supérieurs à  $2^{16} = 65536$ .

# Introduction

Recommandations ANSSI



RecomFact-1. Il est recommandé, pour toute application, d'employer des exposants publics strictement supérieurs à  $2^{16} = 65536$ .

RecomFact-2. Il est recommandé que les deux nombres premiers  $p$  et  $q$  constitutifs du module soient de même taille et choisis aléatoirement uniformément.

# Introduction

Cryptologie :

▶ **Cryptographie :**

- ▶ conception de systèmes cryptographiques
- ▶ étude (preuve) de leur sécurité
- ▶ amélioration des performances

▶ **Cryptanalyse :**

- ▶ mise en défaut des systèmes cryptographiques
- ▶ attaque des problèmes algorithmiques sous-jacents
- ▶ observation des “canaux auxiliaires”

# Introduction

## Objectifs :

- ▶ **confidentialité** : garantir que le contenu d'une communication (ou d'un fichier) n'est pas accessible à tout autre personne que le destinataire légitime
- ▶ **authenticité** : s'assurer de l'identité d'une entité donnée ou de l'origine d'une communication (ou d'un fichier)
- ▶ **intégrité** : s'assurer que le contenu d'une communication (ou d'un fichier) n'a pas été modifié de façon malveillante

# Introduction

## Objectifs :

- ▶ **confidentialité** : garantir que le contenu d'une communication (ou d'un fichier) n'est pas accessible à tout autre personne que le destinataire légitime

↪ chiffrement

- ▶ **authenticité** : s'assurer de l'identité d'une entité donnée ou de l'origine d'une communication (ou d'un fichier)
- ▶ **intégrité** : s'assurer que le contenu d'une communication (ou d'un fichier) n'a pas été modifié de façon malveillante

# Introduction

## Objectifs :

- ▶ **confidentialité** : garantir que le contenu d'une communication (ou d'un fichier) n'est pas accessible à tout autre personne que le destinataire légitime

~> chiffrement

- ▶ **authenticité** : s'assurer de l'identité d'une entité donnée ou de l'origine d'une communication (ou d'un fichier)

~> identification/signature

- ▶ **intégrité** : s'assurer que le contenu d'une communication (ou d'un fichier) n'a pas été modifié de façon malveillante

# Introduction

## Objectifs :

- ▶ **confidentialité** : garantir que le contenu d'une communication (ou d'un fichier) n'est pas accessible à tout autre personne que le destinataire légitime

~> chiffrement

- ▶ **authenticité** : s'assurer de l'identité d'une entité donnée ou de l'origine d'une communication (ou d'un fichier)

~> identification/signature

- ▶ **intégrité** : s'assurer que le contenu d'une communication (ou d'un fichier) n'a pas été modifié de façon malveillante

~> hachage/signature

# Introduction

La cryptologie n'est pas la **stéganographie**.



# Introduction

La cryptologie n'est pas la **stéganographie**.



Je suis très émue de vous dire que j'ai bien compris l'autre soir que vous aviez toujours une envie folle de me faire danser. Je garde le souvenir de votre baiser et je voudrais bien que ce soit là une preuve que je puisse être aimée par vous. Je suis prête à vous montrer mon affection toute désintéressée et sans calcul, et si vous voulez me voir aussi vous dévoiler sans artifice mon âme toute nue, venez me faire une visite.

[...]

# Introduction

La cryptologie n'est pas la **stéganographie**.



Je suis très émue de vous dire que j'ai  
toujours une envie folle de me faire  
baiser et je voudrais bien que ce soit  
par vous. Je suis prête à vous montrer mon  
cul, et si vous voulez me voir aussi  
toute nue, venez me faire une visite.  
[...]



## Un peu d'histoire et de principes

# Principes de Kerchoffs (La Cryptographie militaire – 1883)

Auguste Kerckhoffs von Nieuwenhof (19 janvier 1835 - 1903) est un cryptologue militaire néerlandais.

1. *Le système doit être matériellement, sinon mathématiquement indéchiffrable ;*
2. *Il faut qu'il n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi ;*
3. *La clé doit pouvoir en être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants ;*
4. *Il faut qu'il soit applicable à la correspondance télégraphique ;*
5. *Il faut qu'il soit portatif, et que son maniement ou son fonctionnement n'exige pas le concours de plusieurs personnes ;*
6. *Enfin, il est nécessaire, vu les circonstances qui en commandent l'application, que le système soit d'un usage facile, ne demandant ni tension d'esprit, ni la connaissance d'une longue série de règles à observer.*



# L'âge technique



- ▶ machine à chiffrer (Enigma)  
~> automatisation
- ▶ naissance de l'informatique  
~> Turing, Colossus à Bletchley Park
- ▶ **Data Encryption Standard**  
~> du militaire au civil, prémices de la théorie



# L'âge paradoxal

► naissance de la

cryptographie à clé publique

on ne s'échange plus de clé : on la publie !

↪ chaque utilisateur possède un couple

$(sk, pk)$

où  $pk$  est publique et  $sk$  est gardée secrète

$sk \Rightarrow pk$

**il est « difficile » de retrouver  $sk$  à partir de  $pk$ .**



*New Directions in Cryptography.* W. Diffie and M. E. Hellman,  
IEEE Transactions on Information Theory, vol. IT-22, Nov. 1976, pp 644–654.

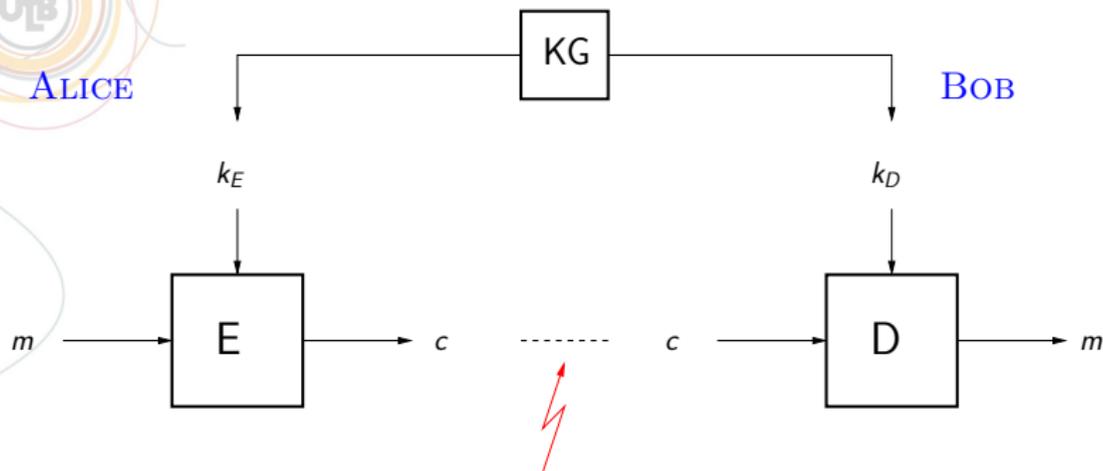


Dans tous les cas, un **secret**, partagé ou non, est nécessaire pour mettre en place un système cryptographique.

- ▶ cryptographie à clé secrète / cryptographie symétrique  
clé secrète partagée
- ▶ cryptographie à clé publique / cryptographie asymétrique  
clé secrète non divulguée

# Confidentialité

Chiffrement



- ▶ Encryption
- ▶ Decryption
- ▶ Key Generation

# Confidentialité

Chiffrement à clé secrète

Cryptographie à clé secrète :

$$k_E = k_D$$

Un exemple fondamental : le *one-time pad*

Chiffrement de Vernam (1917) - masque jetable



# Confidentialité

Chiffrement à clé secrète

Cryptographie à clé secrète :

$$k_E = k_D$$

Un exemple fondamental : le *one-time pad*

Chiffrement de Vernam (1917) - masque jetable

Le message :  $m \in \{0, 1\}^\ell$

$$m = m_1 m_2 \dots m_\ell$$

avec  $m_i \in \{0, 1\}$  pour  $1 \leq i \leq \ell$ .



# Confidentialité

Chiffrement à clé secrète

Cryptographie à clé secrète :

$$k_E = k_D$$

Un exemple fondamental : le *one-time pad*

Chiffrement de Vernam (1917) - masque jetable

Le message :  $m \in \{0, 1\}^\ell$

$$m = m_1 m_2 \dots m_\ell$$

avec  $m_i \in \{0, 1\}$  pour  $1 \leq i \leq \ell$ .

La clé :  $k \in_R \{0, 1\}^\ell$ ,

$$k = k_1 k_2 \dots k_\ell$$

avec  $k_i \in \{0, 1\}$  pour  $1 \leq i \leq \ell$ .



# Confidentialité

## Chiffrement à clé secrète

Le chiffrement :  $c \in \{0, 1\}^{\ell}$  :

$$\begin{array}{rcccccc} m & = & m_1 & m_2 & \dots & m_{\ell} \\ \oplus & k & = & k_1 & k_2 & \dots & k_{\ell} \\ \hline c & = & c_1 & c_2 & \dots & c_{\ell} \end{array}$$

soit

$$c_i = m_i \oplus k_i \quad \forall 1 \leq i \leq \ell.$$

# Confidentialité

## Chiffrement à clé secrète

Le chiffrement :  $c \in \{0, 1\}^{\ell}$  :

$$\begin{array}{r} m = m_1 \quad m_2 \quad \dots \quad m_{\ell} \\ \oplus \quad k = k_1 \quad k_2 \quad \dots \quad k_{\ell} \\ \hline c = c_1 \quad c_2 \quad \dots \quad c_{\ell} \end{array}$$

soit

$$c_i = m_i \oplus k_i \quad \forall 1 \leq i \leq \ell.$$

Le déchiffrement :

$$\begin{array}{r} c = c_1 \quad c_2 \quad \dots \quad c_{\ell} \\ \oplus \quad k = k_1 \quad k_2 \quad \dots \quad k_{\ell} \\ \hline m = m_1 \quad m_2 \quad \dots \quad m_{\ell} \end{array}$$

soit

$$m_i = c_i \oplus k_i \quad \forall 1 \leq i \leq \ell.$$

# Confidentialité

Chiffrement à clé secrète

En effet :

$$c \oplus k = m \oplus k \oplus k = m$$

## Définition

*Un chiffrement est dit parfait si l'on a*

$$\Pr(M = m_0 \mid C = c_0) = \Pr(M = m_0).$$

## Théorème

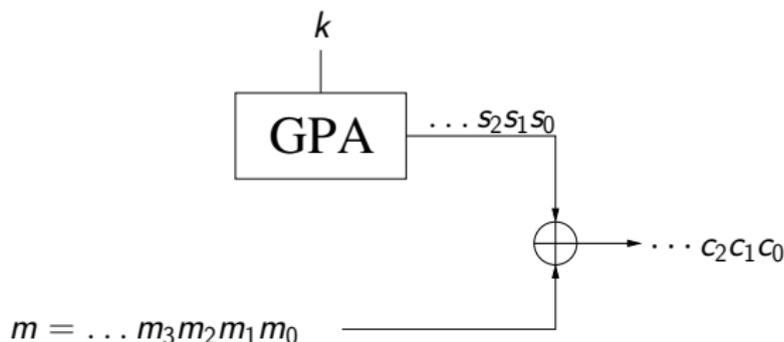
*Si la clé  $k$  est tirée aléatoirement et uniformément parmi les chaînes binaires de longueur  $\ell$  et n'est utilisée qu'une seule fois, le chiffrement de Vernam assure une confidentialité parfaite.*

# Confidentialité

## Chiffrement à clé secrète

### Deux grandes familles

- ▶ chiffrement par blocs (*block cipher*)
  - ▶ DES
  - ▶ 3-DES
  - ▶ AES
- ▶ chiffrement à flot (*stream cipher*)
  - ▶ A5/1 - GSM
  - ▶ E0 - Bluetooth
  - ▶ très durs à concevoir (eSTREAM – The ECRYPT Stream Cipher Project)





## Focus : Cryptographie à clé publique



## Méthodologie :

- ▶ Définition d'un **modèle de sécurité**
- ▶ **Conception** d'un protocole atteignant la fonctionnalité
- ▶ **Preuve** de la sécurité

# Cryptographie à clé publique

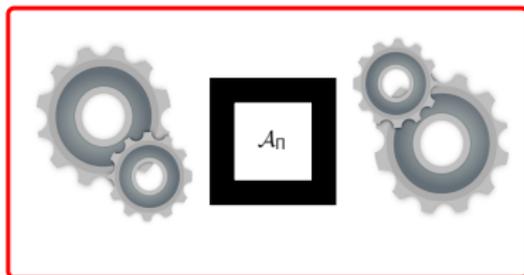
Que signifie « sûrs » ?

dépend de l'application

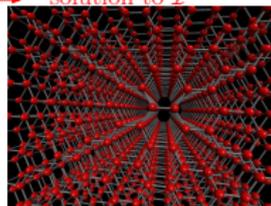
- ▶  $\rightsquigarrow$  modèle de sécurité d'une primitive cryptographique
- ▶  $\rightsquigarrow$  preuve de sa sécurité (insécurité?)

prouver = réduire un problème difficile  $\mathbf{P}$  à une attaque contre le schéma  $\mathbf{\Pi}$

instance  $\mathcal{I}$  of  $\mathbf{P}$



solution to  $\mathcal{I}$



- ▶ Exhiber des problèmes "difficiles" :
  - ▶ problèmes NP difficiles (e. g., euclidean lattices)
  - ▶ problèmes arithmétiques : logarithme discret, factorisation

$$N = p \times q$$

(record in January 2010 : 768 bits)

# Confidentialité

## Chiffrement

Cryptographie à clé publique :

$$k_E \neq k_D \text{ et } \begin{cases} k_E = pk_{Bob} \\ k_D = sk_{Bob} \end{cases}$$

- ▶ Alice a obtenu la clé publique de Bob sur sa page web
- ▶ Bob et lui seul, grâce à sa clé secrète, peut déchiffrer

*Remarque :*

comment Alice est-elle sure que la clé publique de Bob est bien la sienne ?

↪ certification des clés publiques par une autorité (ex. :  VeriSign)

# Confidentialité

## Chiffrement

### Cryptographie à clé publique :

$$k_E \neq k_D \text{ et } \begin{cases} k_E = pk_{Bob} \\ k_D = sk_{Bob} \end{cases}$$

- ▶ Alice a obtenu la clé publique de Bob sur sa page web
- ▶ Bob et lui seul, grâce à sa clé secrète, peut déchiffrer

### Remarque :

comment Alice est-elle sure que la clé publique de Bob est bien la sienne ?

↪ certification des clés publiques par une autorité (ex. :  VeriSign)

↪ **Public Key Infrastructure** (enregistrement des utilisateurs, génération de certificats, renouvellement, révocation, séquestre...)

# RSA : petit rappel d'arithmétique

$$a, b, N \in \mathbb{Z}$$

▶ division euclidienne :  $\exists!(q, r) \in \mathbb{N}^2$  tel que  $a = bq + r$  avec  $0 \leq r < b$

$$\begin{array}{r|l} 405 & 17 \\ 14 & 23 \end{array}$$

▶  $a \equiv b \pmod{N} \iff N \mid b - a$

▶  $\mathbb{Z}/N\mathbb{Z} = \{0, 1, \dots, N - 1\}$ .

▶  $N = p \times q$  avec  $p$  et  $q$  deux entiers premiers

$$\varphi(N) = (p - 1) \times (q - 1)$$

# RSA : petit rappel d'arithmétique

Le théorème au cœur de RSA :

Theorem (Euler)

$a, N \in \mathbb{Z}$

$$\text{pgcd}(a, N) = 1 \implies a^{\varphi(N)} \equiv 1 \pmod{N}$$

# Introduction

1er exemple de cryptosystème : le chiffrement **RSA**



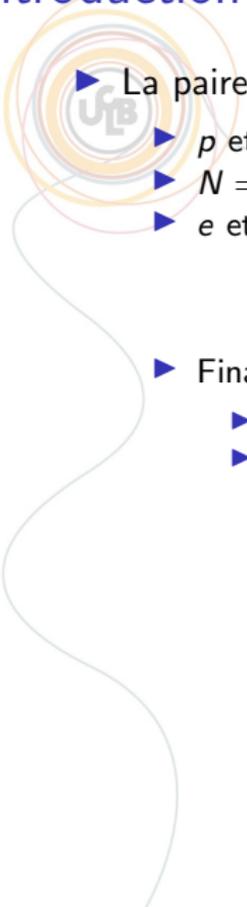
# Introduction

1er exemple de cryptosystème : le chiffrement **RSA**



*A Method for Obtaining Digital Signatures and Public-Key Cryptosystems.* R. Rivest, A. Shamir, L. Adleman. Communications of the ACM, Vol. 21 (2), pp. 120–126 (1978)

# Introduction



▶ La paire de clés :

▶  $p$  et  $q$  sont deux grands premiers (1024 bits)

▶  $N = pq$

▶  $e$  et  $d$  sont deux entiers premiers à  $\varphi(N) = (p - 1)(q - 1)$  tels que

$$ed \equiv 1 \pmod{\varphi(N)}$$

▶ Finalement

▶  $(N, e)$  est la clé publique ( $pk$ )

▶  $(d, p, q)$  est la clé secrète ( $sk$ )

# Introduction



▶ La paire de clés :

▶  $p$  et  $q$  sont deux grands premiers (1024 bits)

▶  $N = pq$

▶  $e$  et  $d$  sont deux entiers premiers à  $\varphi(N) = (p - 1)(q - 1)$  tels que

$$ed \equiv 1 \pmod{\varphi(N)}$$

▶ Finalement

▶  $(N, e)$  est la clé publique ( $pk$ )

▶  $(d, p, q)$  est la clé secrète ( $sk$ )

▶ Pour chiffrer  $m \in \mathbb{Z}/N\mathbb{Z}$  :

# Introduction

▶ La paire de clés :

▶  $p$  et  $q$  sont deux grands premiers (1024 bits)

▶  $N = pq$

▶  $e$  et  $d$  sont deux entiers premiers à  $\varphi(N) = (p - 1)(q - 1)$  tels que

$$ed \equiv 1 \pmod{\varphi(N)}$$

▶ Finalement

▶  $(N, e)$  est la clé publique ( $pk$ )

▶  $(d, p, q)$  est la clé secrète ( $sk$ )

▶ Pour chiffrer  $m \in \mathbb{Z}/N\mathbb{Z}$  :

$$c \equiv m^e \pmod{N}$$

# Introduction

► Pour déchiffrer  $c \in \mathbb{Z}/N\mathbb{Z}$

$$m \equiv c^d \pmod{N}$$

En effet :

$$c^d \pmod{N} \equiv m^{ed} \pmod{N}$$

# Introduction

► Pour déchiffrer  $c \in \mathbb{Z}/N\mathbb{Z}$

$$m \equiv c^d \pmod{N}$$

En effet :

$$\begin{aligned} c^d \pmod{N} &\equiv m^{ed} \pmod{N} \\ &\equiv m^{1+k\varphi(N)} \pmod{N} \end{aligned}$$

# Introduction

► Pour déchiffrer  $c \in \mathbb{Z}/N\mathbb{Z}$

$$m \equiv c^d \pmod{N}$$

En effet :

$$\begin{aligned} c^d \pmod{N} &\equiv m^{ed} \pmod{N} \\ &\equiv m^{1+k\varphi(N)} \pmod{N} \\ &\equiv m \times (m^{\varphi(N)})^k \pmod{N} \end{aligned}$$

# Introduction

► Pour déchiffrer  $c \in \mathbb{Z}/N\mathbb{Z}$

$$m \equiv c^d \pmod{N}$$

En effet :

$$\begin{aligned} c^d \pmod{N} &\equiv m^{ed} \pmod{N} \\ &\equiv m^{1+k\varphi(N)} \pmod{N} \\ &\equiv m \times (m^{\varphi(N)})^k \pmod{N} \\ &\equiv m \end{aligned}$$

# Confidentialité

## Certificat X.509

Certificate:

Data:

Version: 1 (0x0)

Serial Number: 7829 (0x1e95)

Signature Algorithm: md5WithRSAEncryption

Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,  
OU=Certification Services Division,  
CN=Thawte Server CA/emailAddress=server-certs@thawte.com

Validity

Not Before: Jul 9 16:04:02 1998 GMT

Not After : Jul 9 16:04:02 1999 GMT

Subject: C=US, ST=Maryland, L=Pasadena, O=Brent Baccala,  
OU=FreeSoft, CN=www.freesoft.org/emailAddress=baccala@freesoft.org

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public Key: (1024 bit)

Modulus (1024 bit):

00:b4:31:98:0a:c4:bc:62:c1:88:aa:dc:b0:c8:bb:  
33:35:19:d5:0c:64:b9:3d:41:b2:96:fc:f3:31:e1:  
66:36:d0:8e:56:12:44:ba:75:eb:e8:1c:9c:5b:66:  
70:33:52:14:c9:ec:4f:91:51:70:39:de:53:85:17:  
16:94:6e:ee:f4:d5:6f:d5:ca:b3:47:5e:1b:0c:7b:  
c5:cc:2b:6b:c1:90:c3:16:31:0d:bf:7a:c7:47:77:  
8f:a0:21:c7:4c:d0:16:65:00:c1:0f:d7:b8:80:e3:  
d2:75:6b:c1:ea:9e:5c:5c:ea:7d:c1:a1:10:bc:b8:  
e8:35:1c:9e:27:52:7e:41:8f

Exponent: 65537 (0x10001)



Le paradoxe :  
Algorithmique efficace  
vs.  
algorithmique non-efficace

## Génération des clés :

Soit  $k \in \mathbb{N}$  le *paramètre de sécurité*

- Construire 2 nombres premiers  $p$  et  $q$  tels que  $2^{k-1} \leq p, q \leq 2^k - 1$
- $N = p \times q$
- Choisir  $e \in (\mathbb{Z}/\varphi(N)\mathbb{Z})^*$  et calculer  $d$  tel que

$$ed \equiv 1 \pmod{\varphi(N)}.$$

clé publique	$(N, e)$
clé privée	$(d, p, q)$

# Génération des clés :

Soit  $k \in \mathbb{N}$  le *paramètre de sécurité*

- Construire 2 nombres premiers  $p$  et  $q$  tels que  $2^{k-1} \leq p, q \leq 2^k - 1$   
Primalité ( $\in \mathcal{P}$  depuis 2002)

- $N = p \times q$

Multiplication

- Choisir  $e \in (\mathbb{Z}/\varphi(N)\mathbb{Z})^*$  et calculer  $d$  tel que

$$ed \equiv 1 \pmod{\varphi(N)}.$$

Euclide étendu

clé publique	$(N, e)$
clé privée	$(d, p, q)$

# Arithmétique

## Rappel sur RSA

### ► Chiffrement :

Un message  $m$  est un élément de  $\mathbb{Z}/N\mathbb{Z}$ .

- Alice obtient  $(N_B, e_b)$ .

# Arithmétique

## Rappel sur RSA

### ► Chiffrement :

Un message  $m$  est un élément de  $\mathbb{Z}/N\mathbb{Z}$ .

- Alice obtient  $(N_B, e_b)$ .
- $c = m^{e_b} \pmod{N_B}$ .

# Arithmétique

## Rappel sur RSA

### ► Chiffrement :

Un message  $m$  est un élément de  $\mathbb{Z}/N\mathbb{Z}$ .

- Alice obtient  $(N_B, e_b)$ .
- $c = m^{e_b} \pmod{N_B}$ .

### ► Déchiffrement :

- Bob utilise sa clé secrète  $(d_B, p_B, q_B)$ .

# Arithmétique

## Rappel sur RSA

### ► Chiffrement :

Un message  $m$  est un élément de  $\mathbb{Z}/N\mathbb{Z}$ .

- Alice obtient  $(N_B, e_b)$ .
- $c = m^{e_b} \pmod{N_B}$ .

### ► Déchiffrement :

- Bob utilise sa clé secrète  $(d_B, p_B, q_B)$ .
- $c^{d_B} \pmod{N_B} = m$ .

# Arithmétique

## Rappel sur RSA

### ► Chiffrement :

Un message  $m$  est un élément de  $\mathbb{Z}/N\mathbb{Z}$ .

- Alice obtient  $(N_B, e_b)$ .
- $c = m^{e_b} \pmod{N_B}$ .

Exponentiation Modulaire

### ► Déchiffrement :

- Bob utilise sa clé secrète  $(d_B, p_B, q_B)$ .
- $c^{d_B} \pmod{N_B} = m$ .

Exponentiation Modulaire

# Multiplication vs. Factorisation



**Multiplication** :  $(p, q) \mapsto p \times q$

**Factorisation** :  $N = p \times q \mapsto (p, q)$

**Complexité :**

► **Multiplication** :

$O(n^2)$ $O(n^{1,585})$ $O(n^{1,465})$ $O(n \log n \log \log n)$ $O(n \log n 2^{O(\log^* n)})$	scolaire Karatsuba Toom-Cook Schönhage–Strassen (Fürer)
--	---

► **Factorisation** :

$O(2^{n/2})$ $O(2^{n/4})$ $L_{1/2,1}(N)$ $L_{1/3,(64/9)^{1/3}}(N)$	division successive Pollard crible quadratique Number Field Sieve
---	--

$$L_{t,c}(N) = e^{c(\log N)^t (\log \log N)^{1-t}}$$

# Exponentiation modulaire



$$m^e \bmod N$$

## ► ExpModN( $m, e, N$ )

```
x=m
for i from 1 to e-1
  x = x*m mod N
return x
```

Complexité :

## ► ExpBinMod( $m, e, N$ )

```
b=m
for i from t-1 to 0
  b = b2 mod N
  if (e[i] == 1) then
    b=b*m mod N
return b
```

Complexité :

# Exponentiation modulaire



$$m^e \bmod N$$

## ► ExpModN( $m, e, N$ )

```
x=m
for i from 1 to e-1
  x = x*m mod N
return x
```

Complexité :

## ► ExpBinMod( $m, e, N$ )

```
b=m
for i from t-1 to 0
  b = b2 mod N
  if (e[i] == 1) then
    b=b*m mod N
return b
```

Complexité :

# Exponentiation modulaire



$$m^e \bmod N$$

► ExpModN( $m, e, N$ )

```
x=m
for i from 1 to e-1
  x = x*m mod N
return x
```

Complexité :  $O(e \times \log(N)^2)$

► ExpBinMod( $m, e, N$ )

```
b=m
for i from t-1 to 0
  b = b^2 mod N
  if (e[i] == 1) then
    b=b*m mod N
return b
```

Complexité :  $O(\log(e) \times \log(N)^2)$

# Confidentialité

## Chiffrement à clé publique

Les systèmes les plus classiques :

- ▶ RSA
  - ▶ ElGamal
  - ▶ NTRU
  - ▶ McEliece
- 
- ▶ Boneh-Franklin : chiffrement basé sur l'identité



# Échange de clé

## Échange de clé (Diffie-Hellman).

- ▶  $G$  est un groupe cyclique d'ordre un grand premier  $q$ , et  $g$  est un générateur.
- ▶ Alice tire au hasard  $a \in \llbracket 1, q - 1 \rrbracket$  et pose  $y_A = g^a$
- ▶ Bob tire au hasard  $b \in \llbracket 1, q - 1 \rrbracket$  et pose  $y_B = g^b$

# Échange de clé

## Échange de clé (Diffie-Hellman).

- ▶  $G$  est un groupe cyclique d'ordre un grand premier  $q$ , et  $g$  est un générateur.
- ▶ Alice tire au hasard  $a \in \llbracket 1, q - 1 \rrbracket$  et pose  $y_A = g^a$
- ▶ Bob tire au hasard  $b \in \llbracket 1, q - 1 \rrbracket$  et pose  $y_B = g^b$

couple de clés d'Alice	couple de clés de Bob
$sk_A = a$	$sk_B = b$
$pk_A = y_A$	$pk_B = y_B$

# Échange de clé

## Échange de clé (Diffie-Hellman).

- ▶  $\mathbb{G}$  est un groupe cyclique d'ordre un grand premier  $q$ , et  $g$  est un générateur.
- ▶ Alice tire au hasard  $a \in \llbracket 1, q-1 \rrbracket$  et pose  $y_A = g^a$
- ▶ Bob tire au hasard  $b \in \llbracket 1, q-1 \rrbracket$  et pose  $y_B = g^b$

couple de clés d'Alice	couple de clés de Bob
$sk_A = a$	$sk_B = b$
$pk_A = y_A$	$pk_B = y_B$

Alice et Bob peuvent obtenir la clé partagée  $K = g^{ab} \in \mathbb{G}$

Alice calcule	Bob calcule
$y_B^a$	$y_A^b$
$\parallel$	$\parallel$
$(g^b)^a$	$(g^a)^b$
$=$	$=$
$g^{ab}$	$g^{ab}$

# Conclusion



- ▶ De nouveaux défis : protection des données **et** de leur traitement
- ▶ L'arrivée de l'ordinateur quantique
- ▶ Nouveaux paradigmes : chiffrement fonctionnel
- ▶ De nouvelles applications : obfuscation

# Multiparty computation



- ▶ Alice et Bob ont eu un premier rendez-vous
- ▶ Ils veulent savoir s'il y en aura un second  
mais...

**ils ne veulent pas se prendre une veste en direct !**

- ▶ Ils vont jouer à un jeu à l'issue duquel, la seule information connue sera la possibilité d'un second rendez-vous ou pas.

# Multiparty computation



- ▶ Alice et Bob ont eu un premier rendez-vous
- ▶ Ils veulent savoir s'il y en aura un second  
mais...

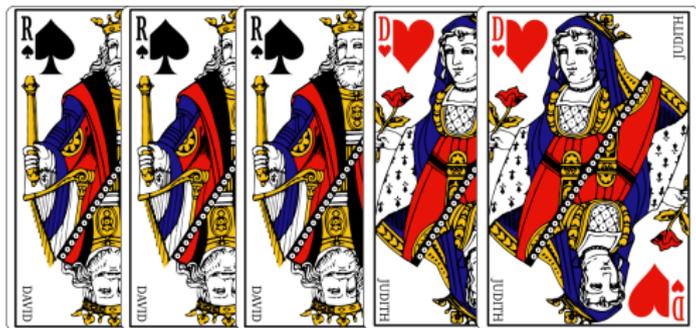
**ils ne veulent pas se prendre une veste en direct !**

- ▶ Ils vont jouer à un jeu à l'issue duquel, la seule information connue sera la possibilité d'un second rendez-vous ou pas.

**Après le premier rendez-vous :**

- ▶ Alice sait si elle veut un second rendez-vous
- ▶ Bob sait si il veut un second rendez-vous
- ▶ et c'est tout !

# Multiparty computation



# Multiparty computation

Dans ce jeu :

- ▶ Un **roi** de ♠ est face cachée sur la table
- ▶ Alice et Bob reçoivent un **roi** et une **reine**
- ▶ Bob pose ses cartes face cachée par dessus le **roi** de ♠
  - ▶ si il veut un second rendez-vous : **reine** au dessus
- ▶ Alice pose ses cartes face cachée sur le dessus du paquet
  - ▶ si elle veut un second rendez-vous : **roi** sur le dessus



# Multiparty computation

## Dans ce jeu :

- ▶ Un **roi** de ♠ est face cachée sur la table
- ▶ Alice et Bob reçoivent un **roi** et une **reine**
- ▶ Bob pose ses cartes face cachée par dessus le **roi** de ♠
  - ▶ si il veut un second rendez-vous : **reine** au dessus
- ▶ Alice pose ses cartes face cachée sur le dessus du paquet
  - ▶ si elle veut un second rendez-vous : **roi** sur le dessus



# Multiparty computation

## Dans ce jeu :

- ▶ Un **roi** de ♠ est face cachée sur la table
- ▶ Alice et Bob reçoivent un **roi** et une **reine**
- ▶ Bob pose ses cartes face cachée par dessus le **roi** de ♠
  - ▶ si il veut un second rendez-vous : **reine** au dessus
- ▶ Alice pose ses cartes face cachée sur le dessus du paquet
  - ▶ si elle veut un second rendez-vous : **roi** sur le dessus



# Multiparty computation

Dans ce jeu :

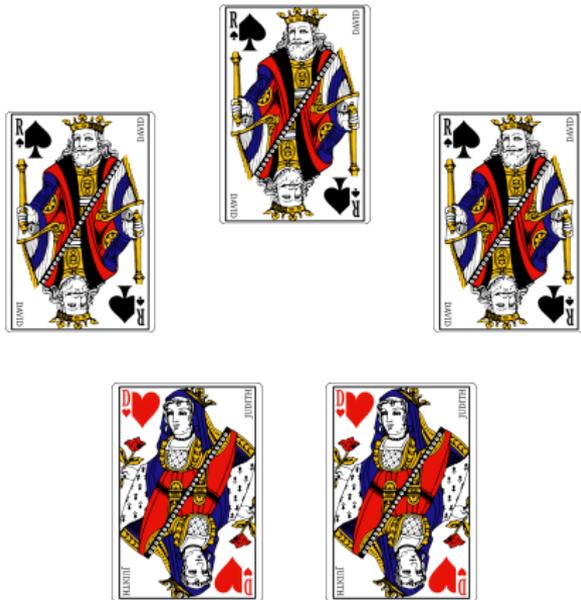
- ▶ Un **roi** de ♠ est face cachée sur la table
- ▶ Alice et Bob reçoivent un **roi** et une **reine**
- ▶ Bob pose ses cartes face cachée par dessus le **roi** de ♠
  - ▶ si il veut un second rendez-vous : **reine** au dessus
- ▶ Alice pose ses cartes face cachée sur le dessus du paquet
  - ▶ si elle veut un second rendez-vous : **roi** sur le dessus



- ▶ Alice et Bob coupent

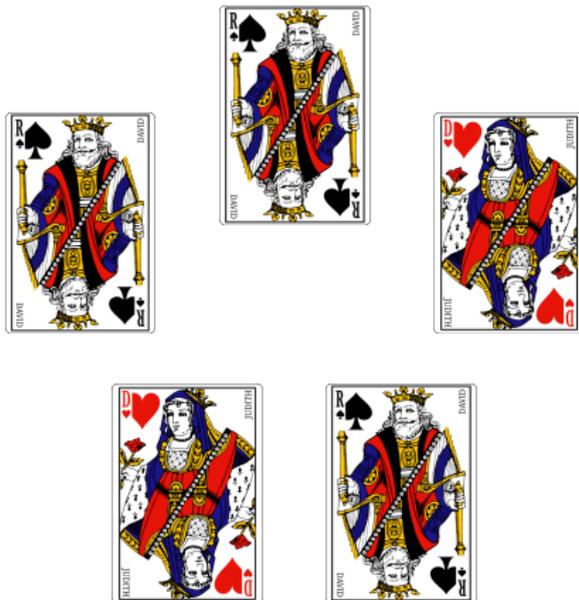
# Multiparty computation

- ▶ Si les reines sont côte à côte :  
Alice et Bob sont amoureux !



# Multiparty computation

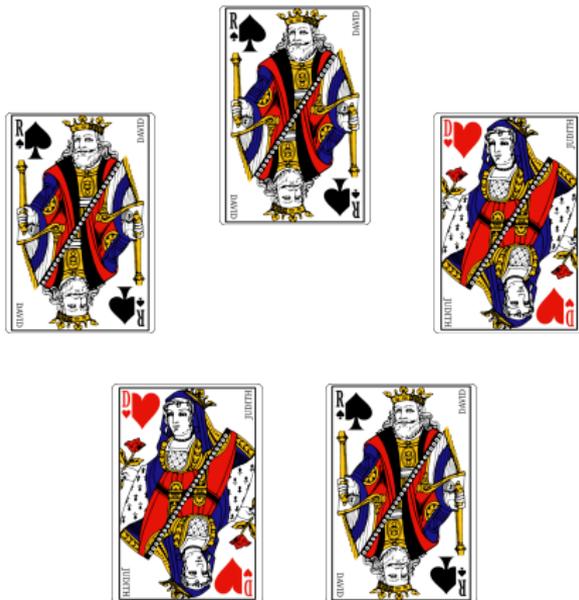
- ▶ Si les reines sont côte à côte :  
Alice et Bob sont amoureux !
- ▶ Sinon :  
**Rien n'est révélé si les reines ne sont pas côte à côte**



# Multiparty computation

- ▶ Si les reines sont côte à côte :  
Alice et Bob sont amoureux !
- ▶ Sinon :  
**Rien n'est révélé si les reines  
ne sont pas côte à côte**

*i.e.*, on ne sait pas si seul l'un  
des deux n'aime pas l'autre, ou  
aucun ne s'aime

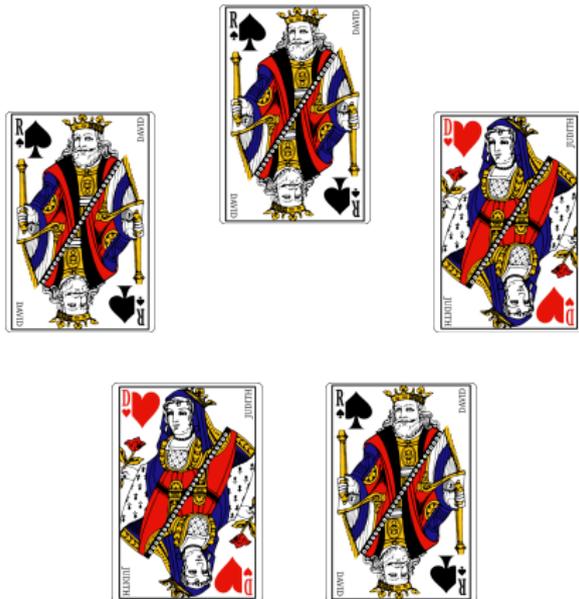


# Multiparty computation

- ▶ Si les reines sont côte à côte :  
Alice et Bob sont amoureux !
- ▶ Sinon :  
**Rien n'est révélé si les reines  
ne sont pas côte à côte**

*i.e.*, on ne sait pas si seul l'un  
des deux n'aime pas l'autre, ou  
aucun ne s'aime

- ▶ fonction “et”



# Multiparty computation

▶ Si les reines sont côte à côte :  
Alice et Bob sont amoureux !

▶ Sinon :

**Rien n'est révélé si les reines  
ne sont pas côte à côte**

*i.e.*, on ne sait pas si seul l'un  
des deux n'aime pas l'autre, ou  
aucun ne s'aime

▶ fonction "et"

**Multiparty computation** : calcule une fonction de sorte à ce qu'une entrée  
secrète ne soit pas révélée aux autres parties

(**attention** : de l'information peut se déduire du résultat de la fonction)

