Les protocoles cryptographiques: chiffrer c'est bien mais cela ne suffit pas !

Stéphanie Delaune

Chargée de recherche CNRS

18 Mai 2016

ENS Cachan

- 12 départements d'enseignement: mathématiques, informatique, chimie, génie mécanique, sciences sociales, ...
- 13 laboratoires de recherche dont

Laboratoire Spécification & Vérification

L'enseignement en informatique à l'ENS Cachan

$\longrightarrow$ formation tournée vers les **fondements de l'informatique**

L'enseignement en informatique à l'ENS Cachan

$\longrightarrow$ formation tournée vers les **fondements de l'informatique**

- Les bases: calculabilité et logique, algorithmique, complexité, langages formels, programmation, ...
- Des enseignements plus spécifiques: initiation à la cryptologie, preuves assistées par ordinateur, démonstration automatique, ...

L'enseignement en informatique à l'ENS Cachan

$\longrightarrow$ formation tournée vers les **fondements de l'informatique**

- Les bases: calculabilité et logique, algorithmique, complexité, langages formels, programmation, ...
- Des enseignements plus spécifiques: initiation à la cryptologie, preuves assistées par ordinateur, démonstration automatique, ...

« La science informatique n'est pas plus la science des ordinateurs que l'astronomie n'est celle des téléscopes »

E. Dijkstra

La recherche au LSV

$\longrightarrow$ accroître notre confiance dans les logiciels critiques

La recherche au LSV

$\longrightarrow$ accroître notre confiance dans les logiciels critiques

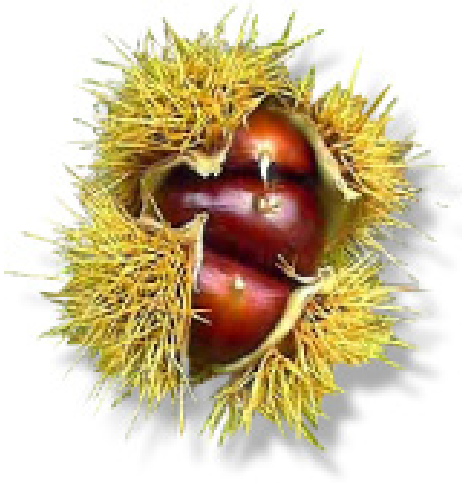
- logiciel: texte relativement long écrit dans un langage spécifique et qui sera exécuté par un ordinateur
- critique: une défaillance peut avoir des conséquences désastreuses en termes humains ou économiques

La recherche au LSV

Ennemi public numéro 1: le bug ...

La recherche au LSV

Ennemi public numéro 1: le bug ...



... aussi connu sous le nom de bogue !

Dans la vie quotidienne !

Ariane V - 4 juin 1996

Un crash après 40 secondes de vol dû ...

Ariane V - 4 juin 1996

Un crash après 40 secondes de vol
dû ...
à un bug logiciel !

1. 189 vols réussis pour Ariane IV,
2. réutilisation du logiciel de lancement d'Ariane IV,
3. ajout du nécessaire pour la nouvelle fusée.

$\longrightarrow$ Le logiciel d'Ariane IV contenait un bug !

Sonde Mars Climate Orbiter - 26 septembre 1999

Perte de la sonde due ...

Sonde Mars Climate Orbiter - 26 septembre 1999

Perte de la sonde due ... à un problème d'unité de mesure !

Carte bancaire

La carte bleue est protégée par un grand nombre public dont on ne connait pas la factorisation.

Nombre de 96 chiffres

213598703592091008239502270499962879705109534182641740644252416500858395774644508840500943086599

Carte bancaire

La carte bleue est protégée par un grand nombre public dont on ne connait pas la factorisation.

Nombre de 96 chiffres

213598703592091008239502270499628797051095341826417406442524165008583957746445088405009430865999

Affaire Serge Humpich (1997)

il factorise ce nombre de 96 chiffres et conçoit de fausses cartes bleues (les « YesCard »).

Carte bancaire

La carte bleue est protégée par un grand nombre public dont on ne connait pas la factorisation.

Nombre de 96 chiffres

213598703592091008239502270499628797051095341826417406442524165008583957746445088405009430865999

Affaire Serge Humpich (1997)

il factorise ce nombre de 96 chiffres et conçoit de fausses cartes bleues (les « YesCard »).

$\longrightarrow$ Depuis, le nombre utilisé pour sécuriser les cartes bancaires comportent 232 chiffres.

Comment assurer le bon fonctionnement de ces logiciels ?

Tests

- à la main ou génération automatique;
- vérification d'un nombre fini de cas.

Comment assurer le bon fonctionnement de ces logiciels ?

Tests

- à la main ou génération automatique;
- vérification d'un nombre fini de cas.

$\infty \infty \infty$ Accéder à l'infini: un rêve impossible ? $\infty \infty \infty$

Comment assurer le bon fonctionnement de ces logiciels ?

Tests

- à la main ou génération automatique;
- vérification d'un nombre fini de cas.

$\infty \infty \infty$ Accéder à l'infini: un rêve impossible ? $\infty \infty \infty$

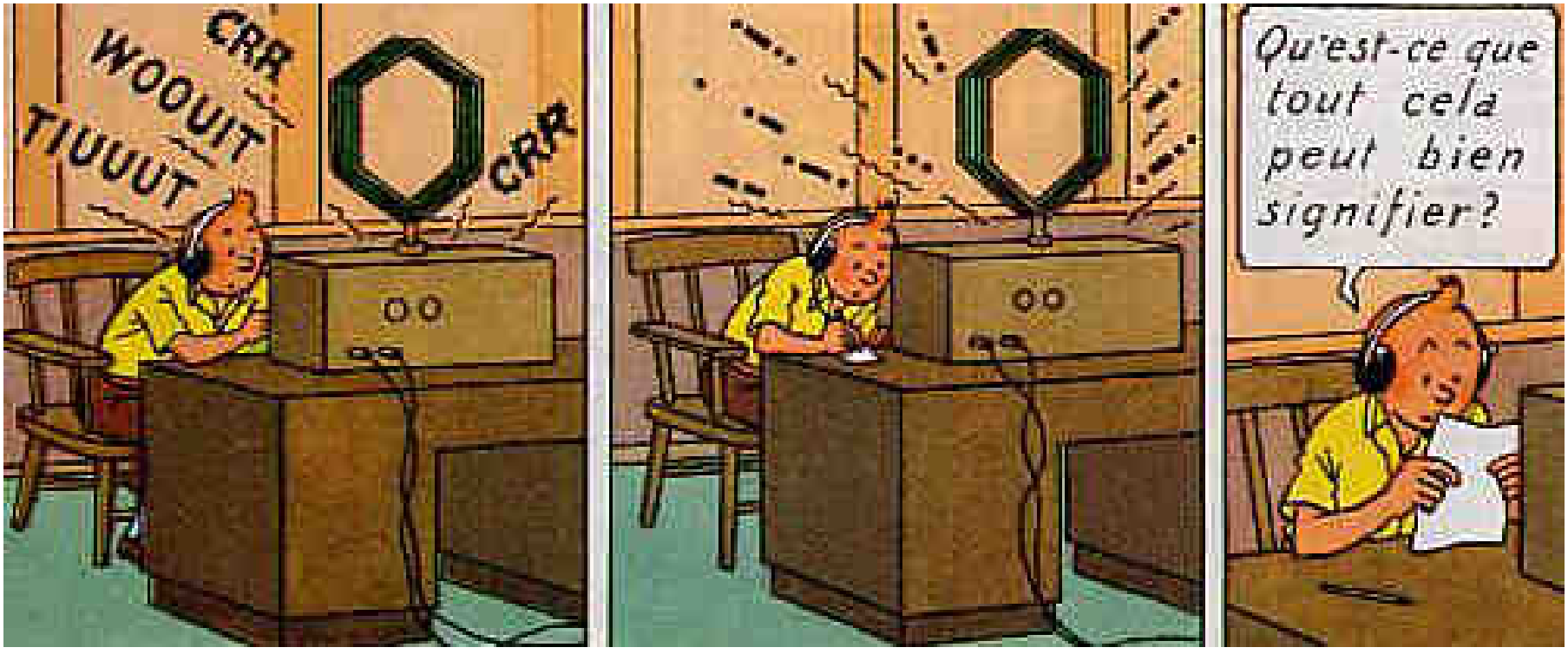
Vérification (preuves formelles)

$\longrightarrow$ preuves mathématiques

- à la main ou à l'aide d'ordinateur;
- vérification de tous les cas possibles;
- plus difficile.

Les protocoles cryptographiques:

comment sécuriser nos communications ?



Équipe SECSI

Sécurité des Systèmes d'Information

- 4 permanents: David Baelde, H. Comon-Lundh, S. Delaune, et J. Goubault-Larrecq.




- 1 post-doctorant
- 3 doctorants

Protocoles cryptographiques

PayPal™

- petits programmes destinés à sécuriser nos communications (*e.g.* confidentialité, authentification)
- omniprésents dans notre vie quotidienne.

Protocoles cryptographiques

PayPal™

- petits programmes destinés à sécuriser nos communications (*e.g.* confidentialité, authentification)
- omniprésents dans notre vie quotidienne.

Protocoles cryptographiques

PayPal™

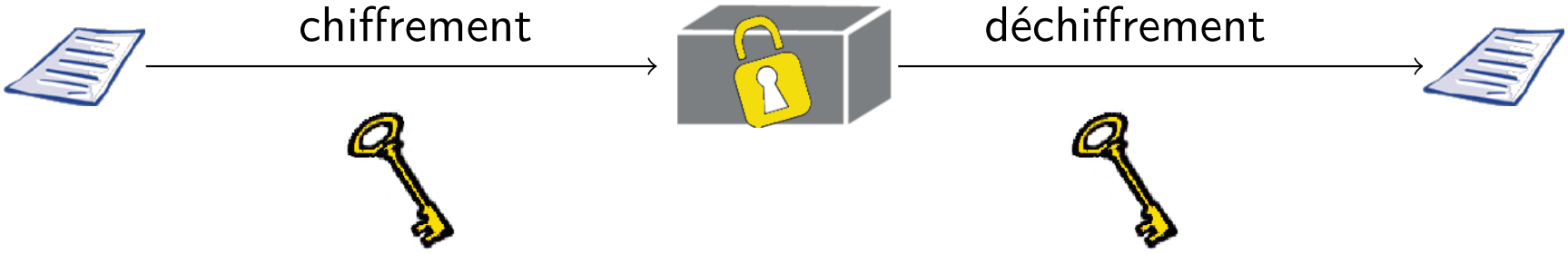
- petits programmes destinés à sécuriser nos communications (*e.g.* confidentialité, authentification)
- omniprésents dans notre vie quotidienne.

Nos informations personnelles sont-elles en danger ?

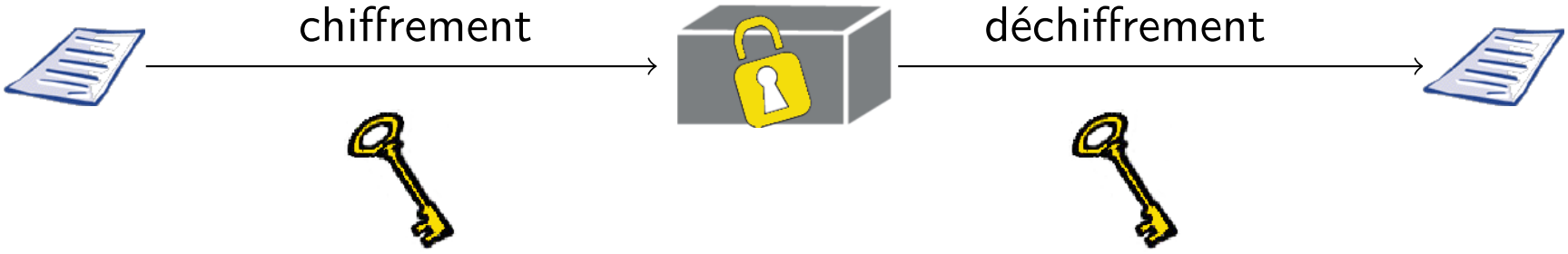
Le chiffrement

« DONLQ GL »

Chiffrement symétrique

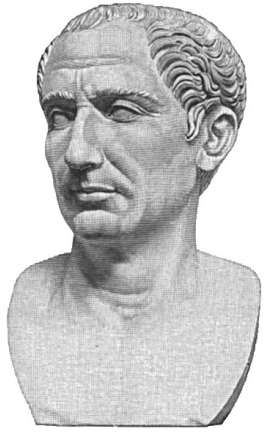


Chiffrement symétrique



scytale (400 av. JC)

César (50 av. JC)



Quelques algorithmes plus récents:

- Data Encryption Standard (1977);
- Advanced Encryption Standard (2000).

Un exemple célèbre

Machine Enigma (1918-1945)

- machine electro-mécanique portable utilisée par les Allemands pendant la 2nd Guerre mondiale;
- permutations et substitutions.

Un exemple célèbre

Machine Enigma (1918-1945)

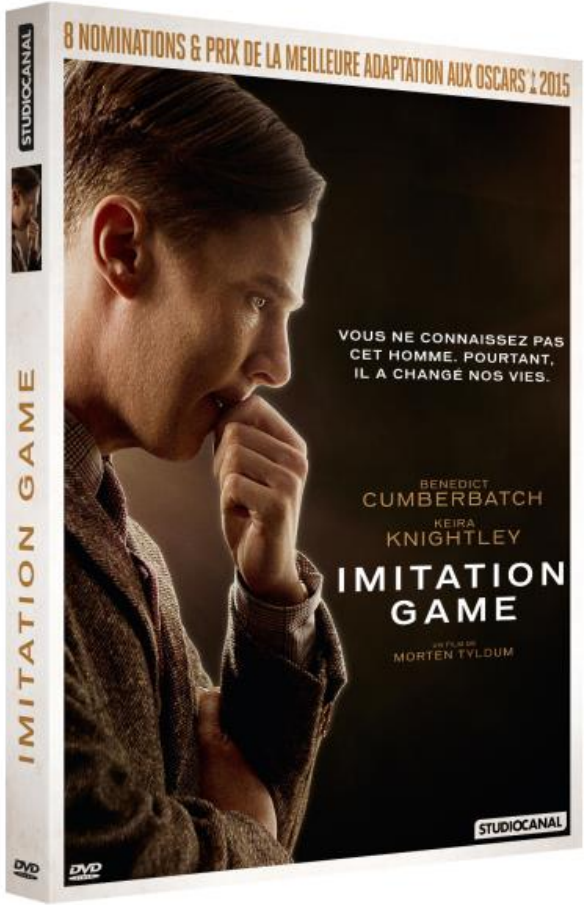
- machine electro-mécanique portable utilisée par les Allemands pendant la 2nd Guerre mondiale;
- permutations et substitutions.

Un peu d'histoire

- 1918: Invention de la machine Enigma
- 1940: Bataille de l'Atlantique pendant laquelle la Bombe d'Alan Turing a été utilisée pour décrypter les messages
- 2016: La DGSE déclassifie ses secrets sur la machine Enigma

Un film à voir

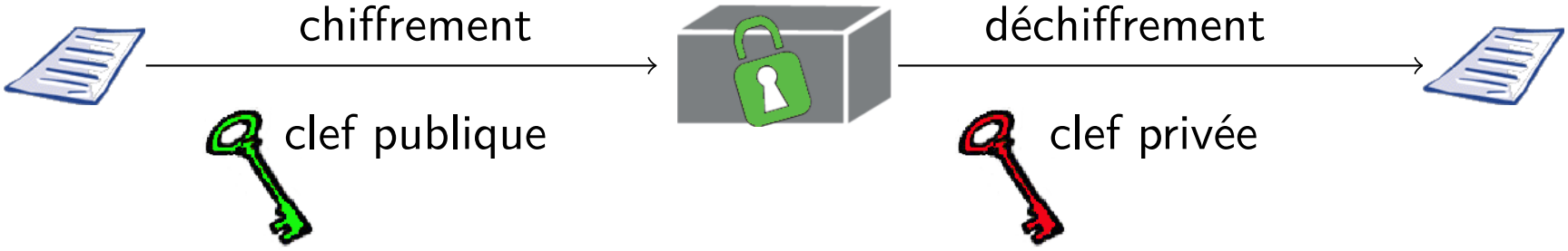
L'incroyable histoire d'Alan Turing qui a contribué à la victoire des Alliés lors de la 2nd Guerre mondiale, en perçant les secrets de la machine Enigma.



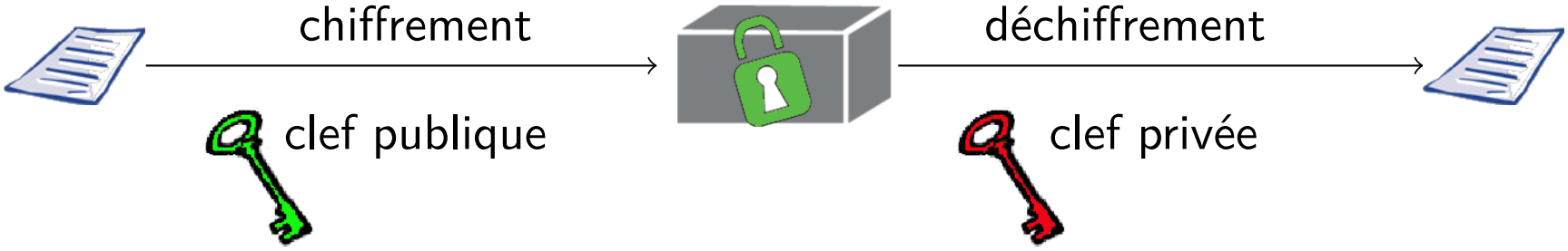
Imitation Game

Disponible
en DVD et Blu-Ray

Chiffrement asymétrique (ou à clefs publiques)

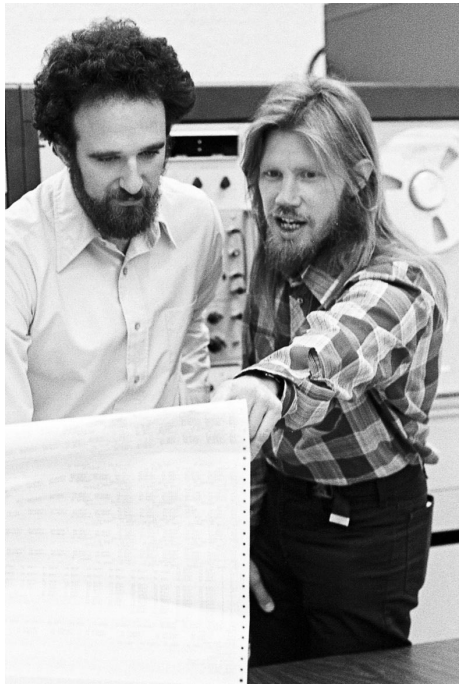


Chiffrement asymétrique (ou à clefs publiques)



Quelques exemples:

- 1976: 1er système
  W. Diffie et M. Hellman
  $\longrightarrow$ Prix Turing 2016
- 1977: système RSA
  R. Rivest, A. Shamir, et L. Adleman



$\longrightarrow$ Ces systèmes sont toujours utilisés de nos jours.

Mais chiffrer ne suffit pas toujours !

La carte bancaire

Le vote électronique

Le passeport électronique

Retour sur le protocole de carte bancaire

- Le client Cl insère sa carte C dans le terminal T.
- Le marchand saisit le montant M de la transaction.

- Le terminal vérifie qu'il s'agit d'une « vraie carte ».
- Le client entre son code.
  Si $M \geq €100$, alors dans 20% des cas,
  - Le terminal contacte la banque B.
  - La banque donne (ou pas) son autorisation.

En détails (1/2)

4 acteurs: la Banque B , le Client Cl, la Carte C et le Terminal T

En détails (1/2)

4 acteurs: la Banque B , le Client Cl, la Carte C et le Terminal T

La Banque possède

- une clef privée – priv(B)
- une clef publique – pub(B)
- une clef symétrique secrète partagée avec la carte – K_{CB}

En détails (1/2)

4 acteurs: la Banque B , le Client Cl, la Carte C et le Terminal T

La Banque possède

- une clef privée – $\mathsf{priv}(B)$
- une clef publique – $\mathsf{pub}(B)$
- une clef symétrique secrète partagée avec la carte – K_{CB}

La Carte possède

- des données $Data$: nom du propriétaire, date d'expiration, ...
- la signature de ces données – $\{Data\}_{\mathsf{priv}(B)}$
- la clef K_{CB}, clef secrète partagée avec la banque.

En détails (1/2)

4 acteurs: la Banque B , le Client Cl, la Carte C et le Terminal T

La Banque possède

- une clef privée – $\mathrm{priv}(B)$
- une clef publique – $\mathrm{pub}(B)$
- une clef symétrique secrète partagée avec la carte – K_{CB}

La Carte possède

- des données $Data$: nom du propriétaire, date d'expiration, ...
- la signature de ces données – $\{Data\}_{\mathrm{priv}(B)}$
- la clef K_{CB}, clef secrète partagée avec la banque.

Le Terminal possède

- la clef publique de la banque – $\mathrm{pub}(B)$

En détails (2/2)

Le terminal T lit la carte C:

1. $C \rightarrow T : Data, \{Data\}_{\mathsf{priv}(B)}$

En détails (2/2)

Le terminal T lit la carte C:

1. $C \rightarrow T$: $Data, \{Data\}_{\mathsf{priv}(B)}$

Le terminal T demande le code:

2. $T \rightarrow Cl$: $code?$
3. $Cl \rightarrow C$: 1234
4. $C \rightarrow T$: code bon

En détails (2/2)

Le terminal T lit la carte C:

1. $C \rightarrow T$: $Data, \{Data\}_{\mathsf{priv}(B)}$

Le terminal T demande le code:

2. $T \rightarrow CI$: *code?*
3. $CI \rightarrow C$: 1234
4. $C \rightarrow T$: code bon

Le terminal T demande l'autorisation à la banque B:

5. $T \rightarrow B$: *autorisation?*
6. $B \rightarrow T$: 45289
7. $T \rightarrow C$: 45289
8. $C \rightarrow T$: $\{45289\}_{K_{CB}}$
9. $T \rightarrow B$: $\{45289\}_{K_{CB}}$
10. $B \rightarrow T$: *ok*

Attaques sur la carte bleue

Initialement la sécurité été assurée par :

- cartes difficilement réplicables,
- secret des clefs et du protocole.

Attaques sur la carte bleue

Initialement la sécurité été assurée par :

- cartes difficilement réplicables,
- secret des clefs et du protocole.

Mais il y a des failles !

- le chiffrement n'est pas sûr (les clefs de 320 bits ne sont plus sûres);
- on peut faire des fausses cartes.

$\rightarrow$ "YesCard" fabriquées par Serge Humpich (1997).

La « YesCard »: Comment ca marche ?

Faille logique

$$
\begin{aligned}
1.C &\rightarrow T &&: \text{Data}, \{\text{Data}\}_{\text{priv}(B)} \\
2.T &\rightarrow CI &&: code? \\
3.CI &\rightarrow C &&: 1234 \\
4.C &\rightarrow T &&: ok
\end{aligned}
$$

La « YesCard »: Comment ca marche ?

Faille logique

$$
\begin{aligned}
1.C &\rightarrow T &&: \text{Data}, \{\text{Data}\}_{\text{priv}(B)} \\
2.T &\rightarrow CI &&: code? \\
3.CI &\rightarrow C' &&: 2345 \\
4.C' &\rightarrow T &&: ok
\end{aligned}
$$

La « YesCard »: Comment ca marche ?

Faille logique

$$
\begin{aligned}
1.C &\rightarrow T &&: \text{Data}, \{\text{Data}\}_{\text{priv}(B)} \\
2.T &\rightarrow CI &&: code? \\
3.CI &\rightarrow C' &&: 2345 \\
4.C' &\rightarrow T &&: ok
\end{aligned}
$$

Remarque : il y a toujours quelqu'un à débiter.
$\rightarrow$ ajout d'un faux chiffrement sur une fausse carte (Serge Humpich).

La « YesCard »: Comment ca marche ?

Faille logique

$$
\begin{aligned}
1.C &\rightarrow T &&: \text{Data}, \{\text{Data}\}_{\text{priv}(B)} \\
2.T &\rightarrow CI &&: code? \\
3.CI &\rightarrow C' &&: 2345 \\
4.C' &\rightarrow T &&: ok
\end{aligned}
$$

Remarque : il y a toujours quelqu'un à débiter.
$\rightarrow$ ajout d'un faux chiffrement sur une fausse carte (Serge Humpich).

$$
\begin{aligned}
1.C' &\rightarrow T &&: \text{XXX}, \{\text{XXX}\}_{\text{priv}(B)} \\
2.T &\rightarrow CI &&: code? \\
3.CI &\rightarrow C' &&: 0000 \\
4.C' &\rightarrow T &&: ok
\end{aligned}
$$

Le passeport électronique

Passeport électronique

C'est un passeport contenant une puce RFID.

$\longrightarrow$ ils sont délivrés en France depuis l'été 2006.

Passeport électronique

C'est un passeport contenant une puce RFID.

$\longrightarrow$ ils sont délivrés en France depuis l'été 2006.

La puce RFID permet de stocker:

- les informations écrites sur le passeport,
- votre photo numérisée.

Passeport électronique

C'est un passeport contenant une puce RFID.

$\longrightarrow$ ils sont délivrés en France depuis l'été 2006.

La puce RFID permet de stocker:

- les informations écrites sur le passeport,
- votre photo numérisée.

Il est interrogeable à distance à l'insu de son propriétaire !

Passeport électronique

$1^{ère}$ Génération

Aucun mécanisme de sécurité pour protéger les informations personnelles

Passeport électronique

$1^{ère}$ Génération

Aucun mécanisme de sécurité pour protéger les informations personnelles

$\longrightarrow$ possibilité de récupérer la signature du porteur en interrogeant le passeport à distance

Passeport électronique — 1ère Génération

Aucun mécanisme de sécurité pour protéger les informations personnelles

$\longrightarrow$ possibilité de récupérer la signature du porteur en interrogeant le passeport à distance

“Faille” découverte sur les passeports belges

Passeport émis entre 2004 et 2006 en Belgique

Passeport électronique

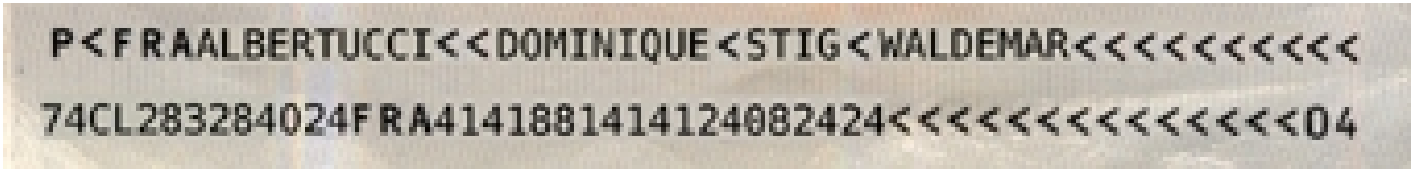
2ème Génération

Passeport émis à partir de 2006 en France, en Belgique, ...

Passeport électronique — 2ème Génération

Passeport émis à partir de 2006 en France, en Belgique, . . .

$\longrightarrow$ mis en place d'un mécanisme (protocole BAC) qui permet de protéger nos informations personnelles



Avec ces deux lignes, on peut interroger la puce électronique.

Protocole BAC

C'est un protocole d'établissement de clef qui doit assurer:

1. la protection de nos données personnelles; et
2. la non traçabilité du porteur du passeport.

Protocole BAC

C'est un protocole d'établissement de clef qui doit assurer:

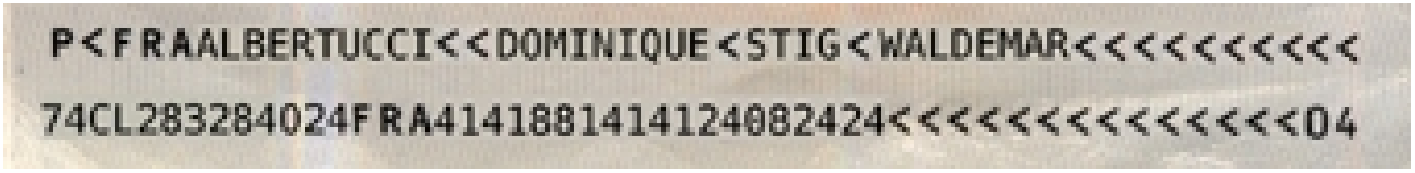
1. la protection de nos données personnelles; et
2. la non traçabilité du porteur du passeport.

Non traçabilité - ISO/IEC standard 15408

Un utilisateur doit pouvoir utiliser plusieurs fois un service ou une ressource sans permettre à un tiers de faire un lien entre ces différentes utilisations.

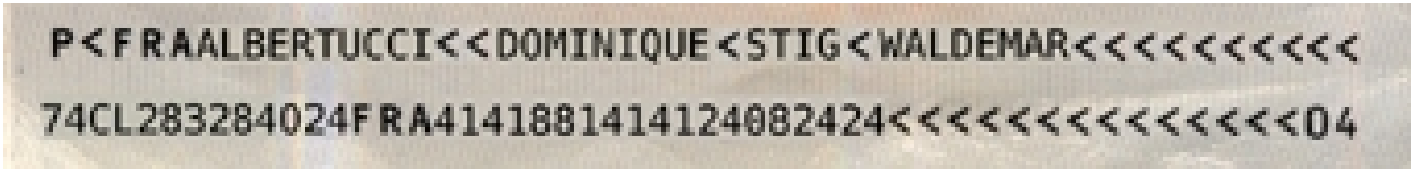
Protocole BAC - étape par étape

1. le lecteur dérive deux clefs K_E et K_M par lecture des lignes codées



Protocole BAC - étape par étape

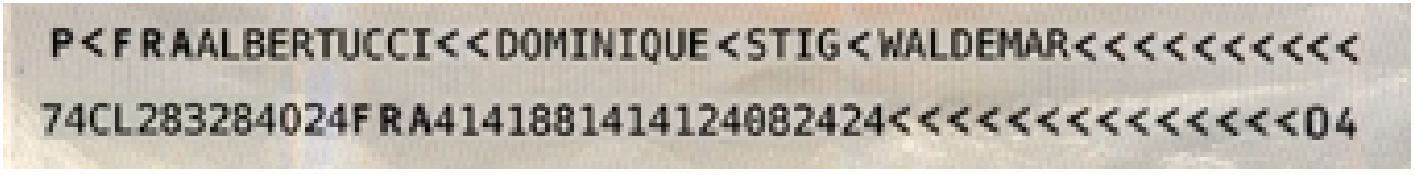
1. le lecteur dérive deux clefs K_E et K_M par lecture des lignes codées



2. Ensuite le passeport et le lecteur s'envoient des messages chiffrés pour établir une nouvelle clef symétrique K (utilisable qu'une seule fois);

Protocole BAC - étape par étape

1. le lecteur dérive deux clefs K_E et K_M par lecture des lignes codées



2. Ensuite le passeport et le lecteur s'envoient des messages chiffrés pour établir une nouvelle clef symétrique K (utilisable qu'une seule fois);

3. Ensuite, nos données personnelles sont transmises chiffrées avec cette clef K.

Protocole BAC - version française

Dans la description du protocole:

- il est mentionné que le passeport doit répondre à tous les messages qu'il reçoit (éventuellement avec un message d'erreur;
- ces messages d'erreurs ne sont pas précisés.

Protocole BAC - version française

Dans la description du protocole:

- il est mentionné que le passeport doit répondre à tous les messages qu'il reçoit (éventuellement avec un message d'erreur;
- ces messages d'erreurs ne sont pas précisés.

Il en résulte une implémentation différente selon les nations, et ...

une attaque sur le passeport Français !!

Vote électronique

⟶ Le vote électronique, pour quelles élections ?

Journal du CNRS - Déc. 2015

Comment vérifier ces protocoles ?

Les mathématiques et l'informatique à la rescousse !

Notre but:

- faire des preuves mathématiques rigoureuses;
- d'une façon automatique.

Construire une machine à détecter les bugs !

Comment vérifier ces protocoles ?

Les mathématiques et l'informatique à la rescousse !

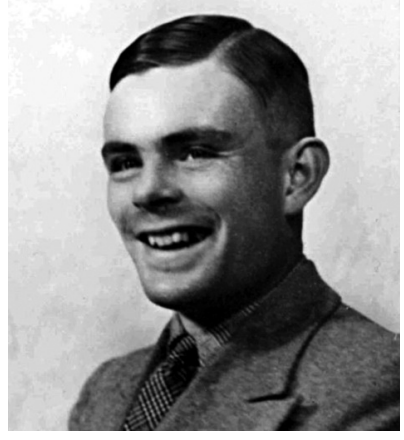
Notre but:

- faire des preuves mathématiques rigoureuses;
- d'une façon automatique.

Construire une machine à détecter les bugs !

A. Turing (1936)

Une telle machine n'existe pas ...



... même dans le cas particulier des protocoles cryptographiques.

Mais alors, que faisons nous ?

Le problème n'a pas de solution ....

Mais alors, que faisons nous ?

Le problème n'a pas de solution ....

mais seulement dans le cas général

Mais alors, que faisons nous ?

Le problème n'a pas de solution ....

mais seulement dans le cas général

Différentes pistes:

- résoudre le problème dans de nombreux cas intéressants,

Mais alors, que faisons nous ?

Le problème n'a pas de solution ....

mais seulement dans le cas général

Différentes pistes:

- résoudre le problème dans de nombreux cas intéressants,
- proposer des procédures approchées,

  Exemple: si le vérificateur répond « oui » alors le logiciel est sûr, sinon on ne peut rien dire

Un exemple pour illustrer

$$\leftarrow \{secret\}_{pub(A)}$$

$$\rightarrow \{\{secret\}_{pub(A)}\}_{pub(B)}$$

$$\leftarrow \{secret\}_{pub(B)}$$

$$\longrightarrow \{\{m\}_{pub(A)}\}_{pub(B)} = \{\{m\}_{pub(B)}\}_{pub(A)}.$$

Un exemple pour illustrer

$$\{secret\}_{pub(A)} \leftarrow$$

$$\{\{secret\}_{pub(A)}\}_{pub(B)} \rightarrow$$

$$\{secret\}_{pub(B)} \leftarrow$$

Question

Alice souhaite envoyer un secret à Bob. Peut-elle utiliser ce protocole ?

Un exemple pour illustrer

$\{secret\}_{pub(A)}$ ←

$\{\{secret\}_{pub(A)}\}_{pub(B)}$ →

$\{secret\}_{pub(B)}$ ←

Question

Alice souhaite envoyer un secret à Bob. Peut-elle utiliser ce protocole ? **Non !** (problème d'authentification)

$\{secret\}_{pub(A)}$ ←

$\{\{secret\}_{pub(A)}\}_{pub(I)}$ →

$\{secret\}_{pub(I)}$ ←

Les mathématiques et la logique à la rescousse !

- les messages sont représentés par des termes, e.g. $\{s\}_{\mathsf{pub}(A)}$;
- l'atome $\mathcal{K}(m)$ signifie que l'attaquant connaît le message m.

Les mathématiques et la logique à la rescousse !

- les messages sont représentés par des termes, e.g. $\{s\}_{\mathsf{pub}(A)}$;
- l'atome $\mathcal{K}(m)$ signifie que l'attaquant connaît le message m.

Des formules logiques pour représenter protocole et attaquant:

$$H_1, \ldots, H_n \Rightarrow C \quad \text{signifie} \quad \text{“si } H_1, \ldots, H_n \text{ alors } C\text{”}$$

Les mathématiques et la logique à la rescousse !

- les messages sont représentés par des termes, e.g. $\{s\}_{\mathsf{pub}(A)}$;
- l'atome $\mathcal{K}(m)$ signifie que l'attaquant connaît le message m.

Des formules logiques pour représenter protocole et attaquant:

$$H_1, \ldots, H_n \Rightarrow C \quad \text{signifie} \quad \text{“si } H_1, \ldots, H_n \text{ alors } C\text{”}$$

Exemple

$$\Rightarrow \mathcal{K}(\{s\}_{\mathsf{pub}(A)}) \qquad \Rightarrow \mathcal{K}(\mathsf{priv}(A))$$

$$\mathcal{K}(\{x\}_{\mathsf{pub}(y)}),\ \mathcal{K}(\mathsf{priv}(y)) \ \Rightarrow\ \mathcal{K}(x)$$

Les mathématiques et la logique à la rescousse !

- les messages sont représentés par des termes, e.g. $\{s\}_{\mathsf{pub}(A)}$;
- l'atome $\mathcal{K}(m)$ signifie que l'attaquant connaît le message m.

Des formules logiques pour représenter protocole et attaquant:

$$H_1, \ldots, H_n \Rightarrow C \quad \text{signifie} \quad \text{“si } H_1, \ldots, H_n \text{ alors } C\text{”}$$

Exemple

$$\Rightarrow \mathcal{K}(\{s\}_{\mathsf{pub}(A)}) \qquad \Rightarrow \mathcal{K}(\mathsf{priv}(A))$$

$$\mathcal{K}(\{x\}_{\mathsf{pub}(y)}),\ \mathcal{K}(\mathsf{priv}(y)) \ \Rightarrow\ \mathcal{K}(x)$$

Théorème

Soit P un protocole. Si $\mathcal{K}(s)$ n'est pas une conséquence logique de l'ensemble des formules $\mathcal{C}_P \cup \mathcal{C}_A$ alors le protocole est sûr.

Retour sur l'exemple

Formules pour le protocole:

$$
\begin{array}{rcl}
 & \Rightarrow & \mathcal{K}(\{s\}_{\mathsf{pub}(A)}) \\
\mathcal{K}(x) & \Rightarrow & \mathcal{K}(\{x\}_{\mathsf{pub}(B)}) \\
\mathcal{K}(\{x\}_{\mathsf{pub}(A)}) & \Rightarrow & \mathcal{K}(x)
\end{array}
$$

$$
\begin{array}{l}
A \rightarrow B : \{s\}_{\mathsf{pub}(A)} \\
B \rightarrow A : \{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)} \\
A \rightarrow B : \{s\}_{\mathsf{pub}(B)}
\end{array}
$$

Retour sur l'exemple

Formules pour le protocole:

$$\begin{aligned} & \Rightarrow \quad \mathcal{K}(\{s\}_{\mathsf{pub}(A)}) \\ \mathcal{K}(x) & \Rightarrow \quad \mathcal{K}(\{x\}_{\mathsf{pub}(B)}) \\ \mathcal{K}(\{x\}_{\mathsf{pub}(A)}) & \Rightarrow \quad \mathcal{K}(x) \end{aligned}$$

$$\begin{aligned} & A \to B : \{s\}_{\mathsf{pub}(A)} \\ & B \to A : \{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)} \\ & A \to B : \{s\}_{\mathsf{pub}(B)} \end{aligned}$$

Formules pour l'attaquant:

$$\begin{aligned} & \Rightarrow \quad \mathcal{K}(\mathsf{pub}(A)) \\ & \Rightarrow \quad \mathcal{K}(\mathsf{pub}(B)) \\ & \Rightarrow \quad \mathcal{K}(\mathsf{pub}(I)) \\ & \Rightarrow \quad \mathcal{K}(\mathsf{priv}(I)) \\ \mathcal{K}(x),\ \mathcal{K}(\mathsf{pub}(y)) & \Rightarrow \quad \mathcal{K}(\{x\}_{\mathsf{pub}(y)}) \\ \mathcal{K}(\{x\}_{\mathsf{pub}(y)}),\ \mathcal{K}(\mathsf{priv}(y)) & \Rightarrow \quad \mathcal{K}(x) \end{aligned}$$

Retour sur l'exemple

Formules pour le protocole:

$$\begin{array}{rcl} & \Rightarrow & \mathcal{K}(\{s\}_{\mathsf{pub}(A)}) \\ \mathcal{K}(x) & \Rightarrow & \mathcal{K}(\{x\}_{\mathsf{pub}(B)}) \\ \mathcal{K}(\{x\}_{\mathsf{pub}(A)}) & \Rightarrow & \mathcal{K}(x) \end{array}$$

$$\begin{array}{l} A \to B : \{s\}_{\mathsf{pub}(A)} \\ B \to A : \{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)} \\ A \to B : \{s\}_{\mathsf{pub}(B)} \end{array}$$

Formules pour l'attaquant:

$$\begin{array}{rcl} & \Rightarrow & \mathcal{K}(\mathsf{pub}(A)) \\ & \Rightarrow & \mathcal{K}(\mathsf{pub}(B)) \\ & \Rightarrow & \mathcal{K}(\mathsf{pub}(I)) \\ & \Rightarrow & \mathcal{K}(\mathsf{priv}(I)) \\ \mathcal{K}(x),\ \mathcal{K}(\mathsf{pub}(y)) & \Rightarrow & \mathcal{K}(\{x\}_{\mathsf{pub}(y)}) \\ \mathcal{K}(\{x\}_{\mathsf{pub}(y)}),\ \mathcal{K}(\mathsf{priv}(y)) & \Rightarrow & \mathcal{K}(x) \end{array}$$

Est-ce que $\mathcal{K}(s)$ est une conséqence logique ?

Retour sur l'exemple

Formules pour le protocole:

$$\begin{aligned} & \Rightarrow & \mathcal{K}(\{s\}_{\mathsf{pub}(A)}) \\ \mathcal{K}(x) & \Rightarrow & \mathcal{K}(\{x\}_{\mathsf{pub}(B)}) \\ \mathcal{K}(\{x\}_{\mathsf{pub}(A)}) & \Rightarrow & \mathcal{K}(x) \end{aligned}$$

$$\begin{aligned} A &\to B : \{s\}_{\mathsf{pub}(A)} \\ B &\to A : \{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)} \\ A &\to B : \{s\}_{\mathsf{pub}(B)} \end{aligned}$$

Formules pour l'attaquant:

$$\begin{aligned} & \Rightarrow & \mathcal{K}(\mathsf{pub}(A)) \\ & \Rightarrow & \mathcal{K}(\mathsf{pub}(B)) \\ & \Rightarrow & \mathcal{K}(\mathsf{pub}(I)) \\ & \Rightarrow & \mathcal{K}(\mathsf{priv}(I)) \\ \mathcal{K}(x),\ \mathcal{K}(\mathsf{pub}(y)) & \Rightarrow & \mathcal{K}(\{x\}_{\mathsf{pub}(y)}) \\ \mathcal{K}(\{x\}_{\mathsf{pub}(y)}),\ \mathcal{K}(\mathsf{priv}(y)) & \Rightarrow & \mathcal{K}(x) \end{aligned}$$

Est-ce que $\mathcal{K}(s)$ est une conséqence logique ? **Oui !**

Comment fait-on en général ?

Résolution

$\longrightarrow$ saturer l'ensemble de clauses avec toutes ses conséquences logiques

Comment fait-on en général ?

Résolution

$\longrightarrow$ saturer l'ensemble de clauses avec toutes ses conséquences logiques

$\mathcal{K}(\{s\}_{\mathsf{pub}(A)})$;

Comment fait-on en général ?

Résolution

$\longrightarrow$ saturer l'ensemble de clauses avec toutes ses conséquences logiques

$\mathcal{K}(\{s\}_{\mathsf{pub}(A)})$; $\mathcal{K}(\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)})$;

Comment fait-on en général ?

Résolution

$\longrightarrow$ saturer l'ensemble de clauses avec toutes ses conséquences logiques

$\mathcal{K}(\{s\}_{\mathsf{pub}(A)})$; $\mathcal{K}(\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)})$; $\mathcal{K}(\{\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)}\}_{\mathsf{pub}(B)})$;

Comment fait-on en général ?

Résolution

$\longrightarrow$ saturer l'ensemble de clauses avec toutes ses conséquences logiques

$\mathcal{K}(\{s\}_{\mathsf{pub}(A)})$; $\mathcal{K}(\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)})$; $\mathcal{K}(\{\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)}\}_{\mathsf{pub}(B)})$; ...

... mais cela ne termine pas (même sur des exemples très simples)

Comment fait-on en général ?

Résolution

$\longrightarrow$ saturer l'ensemble de clauses avec toutes ses conséquences logiques

$\mathcal{K}(\{s\}_{\mathsf{pub}(A)})$; $\mathcal{K}(\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)})$; $\mathcal{K}(\{\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)}\}_{\mathsf{pub}(B)})$; $\ldots$

... mais cela ne termine pas (même sur des exemples très simples)

Résolution ordonnée avec sélection

$\longrightarrow$ choisir « soigneusement » le litéral sur lequel on va faire la résolution

Comment fait-on en général ?

Résolution

$\longrightarrow$ saturer l'ensemble de clauses avec toutes ses conséquences logiques

$\mathcal{K}(\{s\}_{\mathsf{pub}(A)})$; $\mathcal{K}(\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)})$; $\mathcal{K}(\{\{\{s\}_{\mathsf{pub}(A)}\}_{\mathsf{pub}(B)}\}_{\mathsf{pub}(B)})$; ...

... mais cela ne termine pas (même sur des exemples très simples)

Résolution ordonnée avec sélection

$\longrightarrow$ choisir « soigneusement » le litéral sur lequel on va faire la résolution

On ne peut toujours pas assurer la terminaison en général mais cela fonctionne en pratique !

$\longrightarrow$ outil de vérification ProVerif développé par B. Blanchet

`http://proverif.rocq.inria.fr`

À retenir

Les protocoles cryptographiques sont:

- difficiles à concevoir et à analyser;
- vulnérables aux attaques logiques.

Des primitives robustes, c'est bien . . .

. . . **mais ce n'est pas suffisant !**

À retenir

Les protocoles cryptographiques sont:

- difficiles à concevoir et à analyser;
- vulnérables aux attaques logiques.

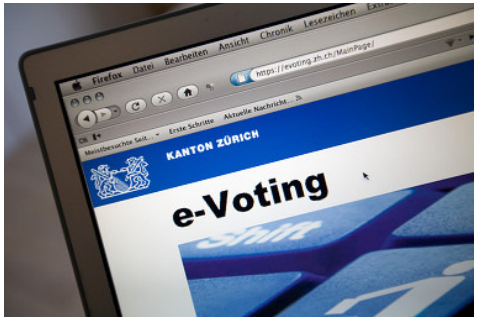
Il est important de s'assurer du bon fonctionnement de ces protocoles.

Ce que l'on sait bien faire:

- les propriétés de sécurité les plus classiques;
- l'analyse de protocoles plutôt petits;
- les primitives cryptographiques standard.

De nombreuses pistes à explorer

Au vu des applications qui voient le jour, **ce n'est pas suffisant !**



- nouveaux objectifs de sécurité
  $\longrightarrow$ anonymat, non traçabilité, proximité physique, ...
- propriétés algébriques
  $\longrightarrow$ chiffrement homomorphe, ou exclusif, ...
- modularité
  $\longrightarrow$ une même application est généralement composée de plusieurs protocoles