Pourquoi essaie-t-on de casser les fonctions cryptographiques?

María Naya-Plasencia
Inria, France

"Colloque Sécurité Informatique : mythes et réalité"
9 Dec. 2016

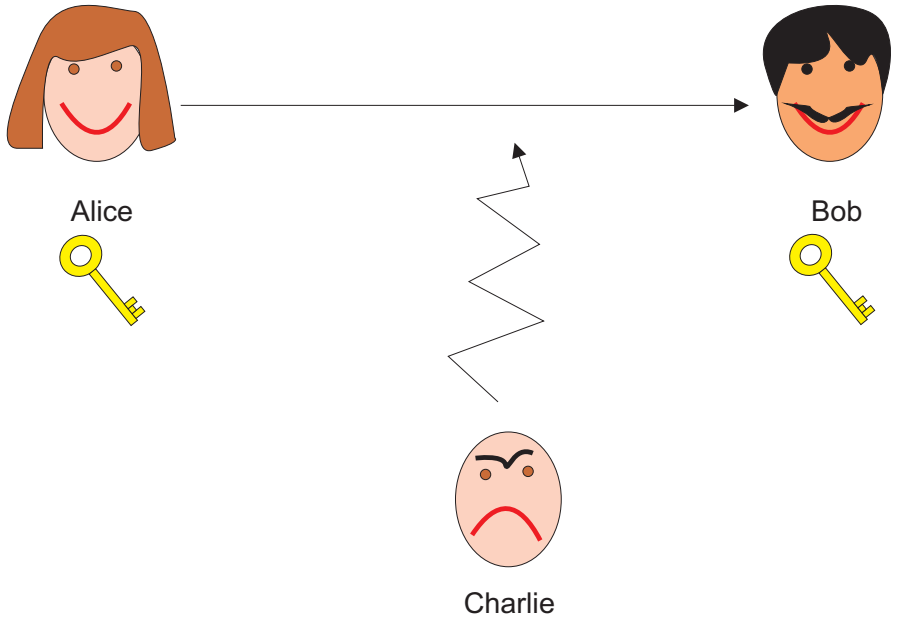
Plan

- Cryptographie
- Et du point de vue de la sécurité ?
  - Importance de la cryptanalyse **publique**
  - Exemple de scénario

Cryptographie

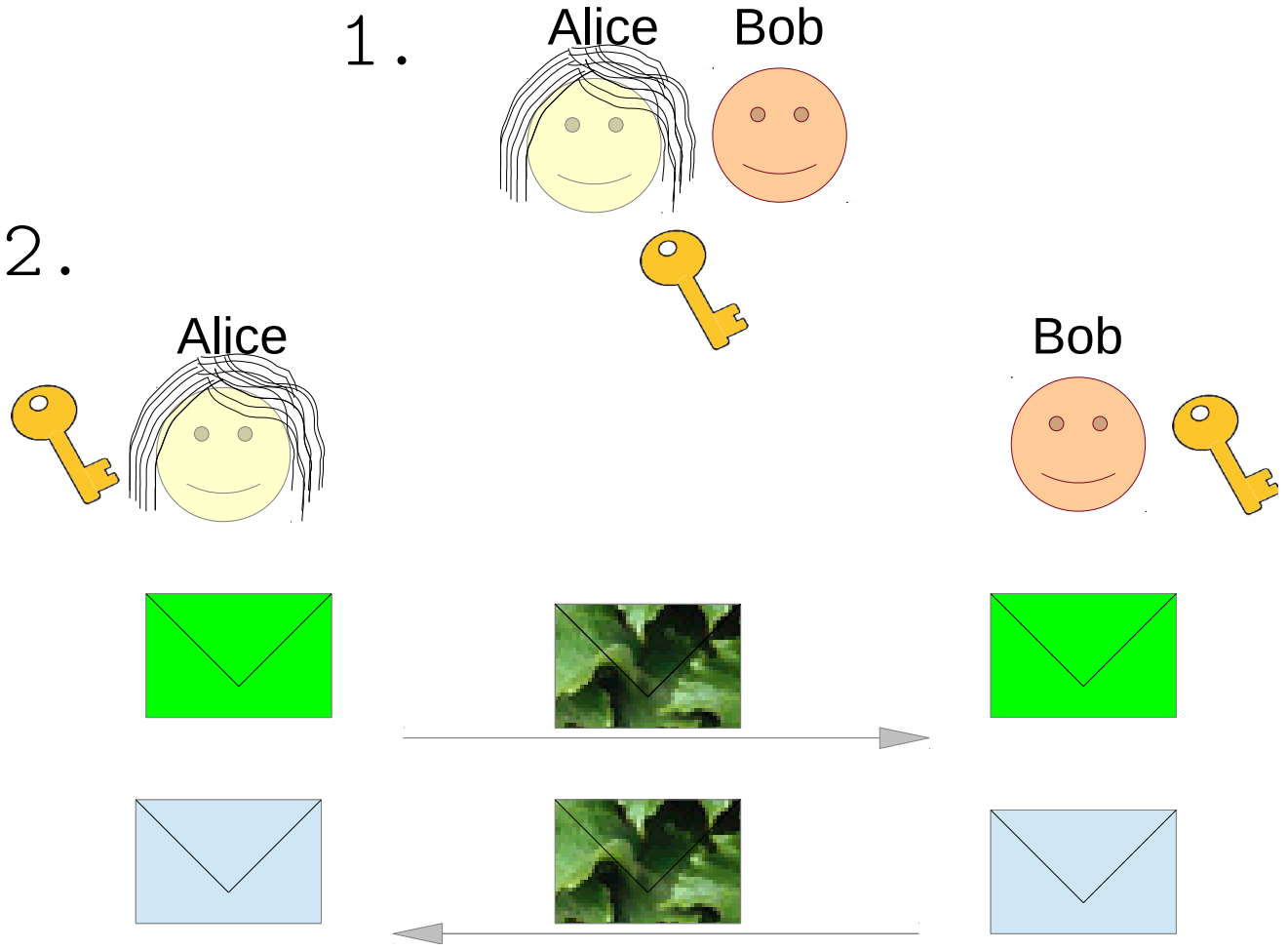
Cryptographie

- Cryptographie : cacher/protéger l'information, souvent à l'aide d'une clé.
  Cryptographie symétrique et cryptographie asymétrique.



Cryptographie Symétrique

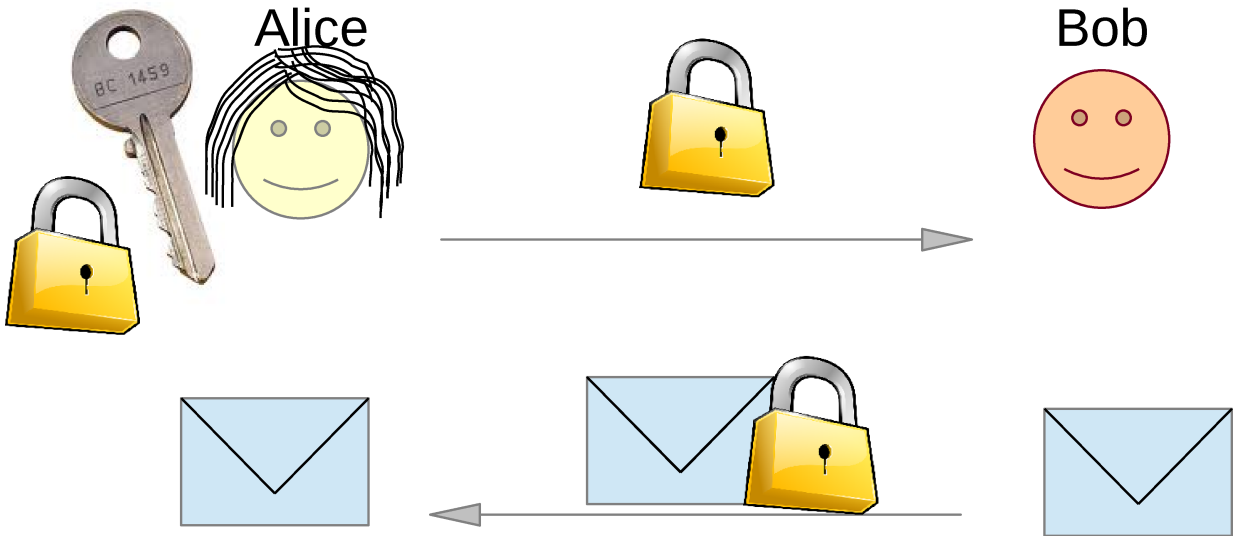
e.g. AES



Cryptographie Asymétrique

e.g. RSA

Sans besoin de se concerter au préalable :



Cryptographie Asymétrique vs Symétrique

Asymétrique:

- Avantage : pas besoin d'échange de clé.
- Inconvénient : "lent" et "coûteux".

Symétrique:

- Inconvénient: Besoin d'échange de clés.
- Avantage: Efficace, adapté à des environnements contraints.

$\Rightarrow$ Hybride : asymétrique pour échanger la clé, et ensuite symétrique !!.

Et la sécurité ?

Sécurité attendue

Primitives asymétriques : la sécurité répose souvent sur la difficulté d'un problème mathématique jugé difficile :

- e.g.: Casser RSA devrait être aussi difficile que résoudre le problème de la factorisation.

Primitives symétriques : doivent offrir une sécurité déterminée par les meilleures attaques génériques (attaques sur les primitives idéales correspondantes) :

- e.g.: retrouver la clé d'un chiffrement devrait coûter $2^{|K|}$ chiffrements (recherche exhaustive).
- e.g.: trouver une collision dans une fonction de hachage devrait coûter $2^{|h|/2}$ (collision aléatoire).

Cryptographie Asymétrique

D'où peuvent venir les faiblesses?

- Primitives sans réduction de sécurité (NTRU)
- Un problème moins difficile que prévu (OSS)
- Des paramètres spécifiques (Sac-à-dos, GGH)

Comment trouver les faiblesses ?
Et les faiblesses potentielles ?

Cryptographie Symétrique

D'où peuvent venir les faiblesses?

- Les composants idéaux n'existent pas, il faut bien les spécifier !
- Toute attaque meilleure qu'une attaque générique est une faille : primitive cassée
  - e.g. retrouver la clé avec un coût moindre que dans le cas idéal (Gost)
  - e.g. retrouver des collisions avec un coût moindre que dans le cas idéal (MD5, SHA1)

Comment trouver les faiblesses ?

Et les faiblesses potentielles ?

Comment trouver les faiblesses ?

Avec de la cryptanalyse !!!

Importance de la cryptanalyse

Cryptanalyse

Étude de la sécurité des primitives cryptographiques.

AKA: Essayer de casser les primitives, de trouver des attaques :

Mesure empirique de la sécurité.

Cryptanalyse et confiance

- Une primitive est considérée sûre si on ne trouve aucune attaque "efficace" sur elle.
- Donc plus on cryptanalyse une primitive sans trouver d'attaque, plus on a confiance en elle.
- Cryptanalyse : une tâche essentielle qui n'a pas de fin !

Cryptanalyse et confiance

- Nous cherchons la sécurité par connaissance, et non par obscurité: seule bonne voie.
- Les primitives sont publiques, et ses meilleures cryptanalyses doivent aussi l'être, pour que le processus fonctionne.
- $\Rightarrow$ grand besoin de cryptanalystes qui rendent leur travaux publiques (les gentils) !

Cryptanalyse et confiance

Cette mesure de la sécurité est empirique, et nous ne pouvons pas être sûrs de l'absence de cryptanalyses non-publiques réalisées...

Comment faire confiance à nos fonctions cryptographiques?

(Pas seulement parce qu'on n'a pas le choix !)

Illustration Récente: Scénario du Concours

Concours de primitives

Contexte de compétitions (AES, SHA-3, eSTREAM, CAESAR, post-quantum): grand nombre de candidats...

- Lequel choisir ?
- Comment anticiper de possibles faiblesses ?
- Comment continuer à faire confiance aux fonctions choisies?

Attaques sur des versions réduites

Si aucune attaque n'est trouvée sur une primitive, que peut on dire sur sa sécurité, ou sa marge de sécurité ? Comment comparer les primitives ?

La sécurité d'un chiffrement n'est pas une information binaire :

- Attaques sur des versions réduites.
- $\Rightarrow$ déterminer et adapter la marge de sécurité.

Remarques sur les Attaques Impraticables

Quand on considère des grandes clés, les attaques qui cassent les chiffrements peuvent avoir des complexités énormes très loin d'être implémentables :
e.g. 2^{120} opérations pour trouver une clé de 128 bits.

On considère la fonction comme cassée et dangereuse car :

- Propriétés faibles non-attendues par les concepteurs.
- Expérience nous montre que les attaques ne vont qu'en s'améliorant.
- Autres chiffrements sans ses propriétés indésirables existent: préferables.

Remarques sur les Attaques Très-Impraticables

Complexité de l'attaque inférieure d'un ou deux bits par rapport à la recherche exhaustive naïve :

- Quand il s'agit d'une recherche exhaustive accélérée (AES avec Bicliques) $\Rightarrow$ redéfinition de la sécurité (i.e. pas d'attaque mieux que l'attaque générique).

- Quand on détermine la marge de sécurité : le plus grand nombre de tours attaqué ? Pas attaques génériques, mais spécifiques.

Ex.: Advanced Encryption Standard

Le gagnant : AES, 10 tours.

- 1998: meilleure attaque par les auteurs: 6 tours
- 2001: nouvelle attaque sur 7 tours.
- 2001 à 2016: plus de 20 nouvelles attaques, améliorant la complexité.
- 2016: la meilleure attaque connue est toujours sur 7 tours. La meilleure complexité [DFJ12]: 2^{97} données, 2^{99} temps et 2^{98} mémoire.

Il semble très peu probable que des attaques sur la version complète apparaissent/existent dans l'état actuel.

Pas de surprise.

Quoi utiliser ?

Seulement des primitives recommandées par la communauté:

- Largement analysées, sans trouver de faiblesses, avec une grande marge de sécurité !

Pourquoi ? Quelques cas réels

Problèmes de sécurité quand on ne nous écoute pas:

- Flame[2012]: collisions sur MD5[WFL2004]
- Attaque sur TLS[ABP..13]: Biais de RC4[FMS01]
- FREAK[BB..15]: RSA-512 faible[90's]
- Sloth[BL16]: collisions sur MD5[WFL2004]

Problèmes qui avaient été prédits !!

Pourquoi essaie-t-on de casser les fonctions cryptographiques ?

Donc, Pourquoi ?

Pour être en avance sur les adversaires malveillants!

Les analyses publiques n'ont que des avantages:

- Attaque trouvée sur une primitive non encore utilisée

  $\rightarrow$ Ne jamais l'utiliser (très courant, risque nul).
- Attaque améliorée sur une version réduite

  $\rightarrow$ Réévaluation de la marge de sécurité (moyennement courant, risque pratiquement nul).
- Attaque théorique sur une primitive utilisée

  $\rightarrow$ Arrêt d'utilisation (très rare, risque très faible).
- Attaque pratique sur une primitive utilisée

  $\rightarrow$ Protocole à suivre + arrêt d'utilisation (très très rare, risque réduit par rapport à l'ignorance de l'attaque!).

Conclusion

Conclusion

- Cryptographie: vitale dans l'ère de l'Information.
- Cryptanalyse : essentielle pour faire confiance aux fonctions. Énorme besoin de cryptanalystes "gentils".
- Thématique en constante évolution, nécessaire pour les nouvelles constructions ainsi que pour la "maintenance".
- Better safe than sorry!