Protocoles cryptographiques Modélisation et Vérification de propriétés

Marie-Laure Potet

VERIMAG - Cours Ensimag ISI

November 8, 2022

Introduction

Protocoles cryptographiques : échanges de données impliquant de la cryptographie pour s'exécuter dans un environnement hostile (réseau, participant . . .) Ex : TLS, SSH, Kerberos

- Code sensible
- La robustesse dépend de ce que peut faire un attaquant
- Propriétés " non classiques " (authentification, intégrité, anonymat . . .)

Objectifs :

- savoir modéliser les protocoles
- savoir énoncer des propriétés de sécurité
- TP : outils de vérification

Notations

Soit :

- m, m_1 et m_2 des messages
- K une clé
- A et B des agents

On note :

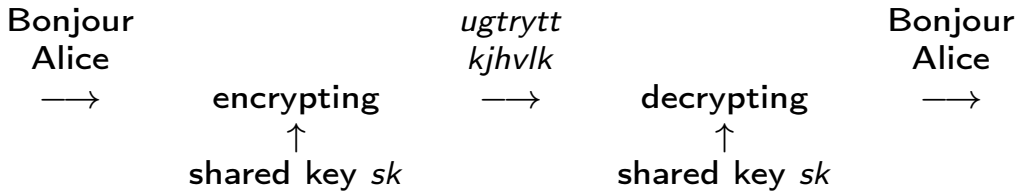
- m_1, m_2 (ou (m_1, m_2)) le message constitué de m_1 et m_2
- $\{m\}_K$ le message m chiffré avec la clé K
- $A \rightarrow B : m$ l'envoi par A du message m à B
- $I[A] \rightarrow B : m$ l'envoi du message m par I se faisant passer pour A

Primitives cryptographiques

Les ingrédients :

- Chiffrement : protection en lecture/modification
- Signature : authentification
- hash : intégrité
- Nonce (nombre aléatoire unique) : fraicheur du message

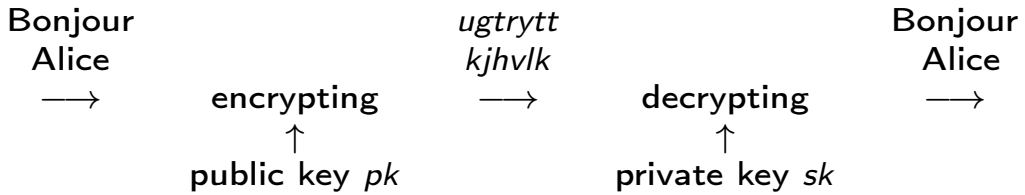
Symmetric encryption



- **Notation** K_{AB}
- **Decrypting:** $\{\{m\}_{K_{AB}}\}_{K_{AB}} = m$
- **Security property:** Informally, a ciphertext cannot be decrypted without knowing the key.
- It is fast, useful to encrypt large messages
- shared keys must be securely distributed
- **Examples:** DES, 3DES, AES, RC4, RC5, RC6 ...
- chiffrement par blocs (ECB, CBC, . . .) ou par flot

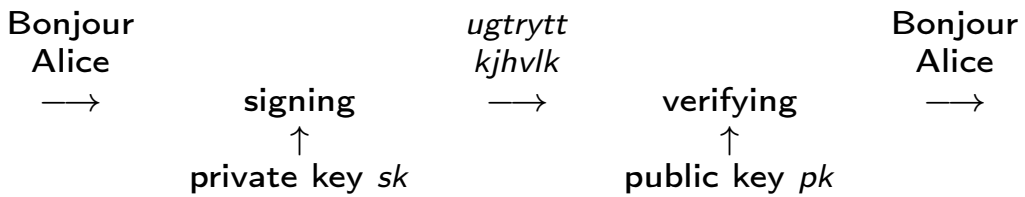
`https://www.ssi.gouv.fr/uploads/2015/01/rgs_v-2-0_b1.pdf`

Asymmetric encryption



- **Notation :** K_A public key, K_A^{-1} private key
- **Decrypting:** $\{\{m\}_{K_A}\}_{K_A^{-1}} = \{\{m\}_{K_A^{-1}}\}_{K_A} = m$
- **Security property:** $\{m\}_{K_A}$ can be decrypted only knowing K_A^{-1}
- It is slower, but allows to communicate with an unknown participant (by means of a public key infrastructure PKI)
- **Examples:** RSA, ElGamal, Cramer-Shoup, OAEP+,...

Signature



- **Notation :** K_A public key, K_A^{-1} private key
- **Decrypting:** $verif(\{m\}_{K_A^{-1}}, K_A) = true$
- **Security property:** Informally, a signature that can be verified using a public key *pk*, cannot be created without knowing the inverse private key *sk*.
- **Examples:** RSA, DSA, ECDSA,...

Other cryptographic primitives

- A hashing function h produces for a large message m, a small message $h(m)$ called hash. **Informal security properties:**
  - If $m_1 \neq m_2$, then $h(m_1) \neq h(m_2)$.
  - Given only $h(m)$, it is impossible to find m.

  Ex : SHA-256, SHA-384 SHA-512, MD5, MD6, RIPEMD ...

- Nonces, denoted $n_a, n_b, ..., n_1, n_2, ...$ are randomly generated large values. **Informal security properties:**
  - $n_1 \neq n_2$ for any independent generated nonces.
  - Given only partial information about n, it is impossible to find entire n.

Propriétés associées à la crypto

- Confidentialité :

$$\{m\}_{K_{AB}},\ \{m\}_{K_A}$$

- Intégrité :

$$\{m\}_{K_{AB}},\ \{m\}_{K_A^{-1}} \text{ ou } m, \{hash(m)\}_{K_A^{-1}}$$

- Authentification d'un message :

$$\{m\}_{K_A^{-1}} \text{ ou } m, \{hash(m)\}_{K_A^{-1}}$$

- Authentification de la source de l'envoi ?

Applications

- Communications protégées (SSH, GSM . . .)
- Authentification d'agents (login, GSM, . . .)
- signature de contrats électroniques, vente aux enchères
- paiment bancaire par carte à puces
- paiment bancaire en ligne (transaction électronique)
- paiment bancaire hors ligne (monnaie électronique)
- vote électronique
- chaînes de télévision payantes

Définitions

Secret : un protocole assure le secret d'une donnée s si un intrus ne peut pas déduire s lorsque ce protocole est joué

Authentification d'un message : un protocole permet à un agent A d'authentifier un message m si A peut connaître de façon sûre l'émetteur de m.

Authentification d'entité : un protocole permet à un agent A d'authentifier un agent B si à la fin d'une session réussie du protocole, A a la garantie qu'il a bien réalisé le protocole avec B.

Fraîcheur : pendant une session de protocole, une donnée est fraîche si on peut garantir qu'elle a été émise spécifiquement pour cette session par un des acteurs.

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Premier essai :

1. $A \rightarrow B : \{s\}_{K_B}$

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Premier essai :

1. $A \rightarrow B : \{s\}_{K_B}$

De qui vient s ?

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Premier essai :

1. $A \rightarrow B : \{s\}_{K_B}$

De qui vient s ?

Second essai :

1. $A \rightarrow B : \{A, s\}_{K_B}$

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Premier essai :

1. $A \rightarrow B : \{s\}_{K_B}$

De qui vient s ?

Second essai :

1. $A \rightarrow B : \{A, s\}_{K_B}$

Peut être forgé par I se faisant passer pour A. B croit partager un secret avec A mais le partage avec I

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Troisième essai : authentification par signature

1. $A \rightarrow B : \{A, s, \{s\}_{K_A^{-1}}\}_{K_B}$

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Troisième essai : authentification par signature

1. $A \rightarrow B : \{A, s, \{s\}_{K_A^{-1}}\}_{K_B}$

$\Rightarrow$ On peut garantir :

- s secret
- s émis par A

Problème du rejeu :

1. $A \rightarrow B : \{A, s, \{s\}_{K_A^{-1}}\}_{K_B}$
2. $I(A) \rightarrow B : \{A, s, \{s\}_{K_A^{-1}}\}_{K_B}$

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Quatrième essai :

1. $B \rightarrow A : \{B, A, N_B\}_{K_A}$
2. $A \rightarrow B : \{B, A, N_B, s\}_{K_B}$

Principes de base

Envoi d'un secret s de A à B (chiffrement asymétrique)

Quatrième essai :

1. $B \rightarrow A : \{B, A, N_B\}_{K_A}$
2. $A \rightarrow B : \{B, A, N_B, s\}_{K_B}$

$\Rightarrow$ On peut garantir :

- s secret
- B authentifie s comme étant émis par A
- N_B frais et donc $\{B, A, N_B, s\}_{K_B}$ frais
- B authentifie A pour la session

Exemple de faille et capacité de l'intrus

Needham-Schroeder (1978) : un protocole d'authentification

1. $A \rightarrow B : \{N_A, A\}_{K_B}$
2. $B \rightarrow A : \{N_A, N_B\}_{K_A}$
3. $A \rightarrow B : \{N_B\}_{K_B}$

Propriétés ?

Exemple de faille et capacité de l'intrus

Needham-Schroeder (1978) : un protocole d'authentification

1. $A \rightarrow B : \{N_A, A\}_{K_B}$
2. $B \rightarrow A : \{N_A, N_B\}_{K_A}$
3. $A \rightarrow B : \{N_B\}_{K_B}$

Propriétés ?

- N_A et N_B secret partagé entre$A$ et B
- A authentifie B pour la session au pas 2 (témoin N_A)
- B authentifie A pour la session au pas 3 (témoin N_B)

Faille

Protocole prouvé correct (preuves manuelles). Faille trouvée par Lowe en 1995 par model-checking.

$\Rightarrow$ Une session entre A et I et une session entre I se faisant passer pour A et B.

1	$A \rightarrow I : \{N_A, A\}_{K_I}$	pas 1 session 1
2	$I \rightarrow B : \{N_A, A\}_{K_B}$	pas 1 session 2
3	$B \rightarrow A : \{N_A, N_B\}_{K_A}$	pas 2 session 2
4	$A \rightarrow I : \{N_B\}_{K_I}$	pas 3 session 1
5	$I \rightarrow B : \{N_B\}_{K_B}$	pas 3 session 2

- B croit avoir joué la session avec A et partager N_B avec A.
- I connait le secret N_B

Schéma tableau

Correction

Needham-Schroeder (1978) : un protocole d'authentification

1. $A \rightarrow B : \{N_A, A\}_{K_B}$
2. $B \rightarrow A : \{N_A, N_B, B\}_{K_A}$
3. $A \rightarrow B : \{N_B\}_{K_B}$

tableau : Attaque bloquée