Rapport de stage de DEA

Une comparaison des preuves de sécurité

Duong Hieu Phan

Laboratoire d'accueil : Laboratoire d'informatique de l'École normale supérieure
Directeur de stage : David Pointcheval

Année 2001–2002

Remerciements

Je tiens tout d'abord à exprimer ma profonde reconnaissance à David Pointcheval pour m'avoir encadré pendant ce stage, et surtout pour m'avoir fait découvrir et aimer la recherche dans le domaine de la cryptographie. Je tiens beaucoup à remercier Jacques Stern pour m'avoir donné de précieux conseils et pour m'avoir aidé à choisir le domaine de mon stage. Je remercie beaucoup Phong Nguyen pour m'avoir aidé à m'adapter au milieu de travail. Je tiens à remercier Emmanuel Bresson et Dario Catalano avec qui j'ai partagé un même bureau et des discussions intéressantes. Finalement, je remercie tous les membres du Département d'informatique qui m'ont très agréablement accueilli.

Table des matières

Introduction

Dans le monde actuel, la demande d'échange d'informations est énorme, la sécurité de ces informations joue un rôle primordial pour plusieurs activités non seulement militaires mais aussi économiques, sociales, ... Avec le développement très rapide des outils calculatoires, plusieurs protocoles modernes de chiffrement, d'authentification, et de signature sont proposés pour garantir les exigences principales des informations échangées : la confidentialité, l'authenticité et l'intégrité. Mais pourtant, la puissance de calcul importante donne aussi aux attaquants des moyens efficaces pour casser ces systèmes. Alors, pour estimer la sécurité effective d'un protocole, il faut considérer les moyens des attaquants, ou bien les ressources dont ils disposent, et des notions de sécurité, ou bien les niveaux de sécurité que l'on veut garantir.

Pour analyser la sécurité des protocoles, depuis une quinzaine d'années, on développe principalement deux approches bien différentes : l'une repose sur les méthodes formelles, notamment la logique BAN proposée par Burrows, Abadi, Needham [6] que l'on considère dans ce rapport, par déduction logique avec des règles de ré-écriture ; l'autre repose sur des méthodes par réduction, ainsi la sécurité est démontrée dans le contexte de la théorie de la complexité. Ces deux méthodes reposent sur des hypothèses. Pour la logique BAN, quand on analyse un protocole d'authentification, on doit faire l'hypothèse que le schéma de chiffrement utilisé dans ce protocole est idéalement sûr. Pour la méthode par réduction, quand on analyse la sécurité d'un schéma de chiffrement par exemple, on doit utiliser une hypothèse de difficulté de problèmes mathématiques comme le problème du logarithme discret, le problème Difie-Hellman Calculatoire ou bien le problème Difie-Hellman Décisionnel,... et les preuves sont faites par une réduction d'une instance du problème difficile à une attaque du système.

Dans ce rapport, nous allons essayer de considérer ces méthodes de preuve de sécurité, comparer les efficacités sur certains protocoles, et nous allons notamment développer une extension de la logique BAN qui formalise certaines notions calculatoires avec l'objectif de construire un pont entre ces deux approches.

Dans le premier chapitre, nous analysons l'efficacité des deux méthodes en considérant un protocole particulier - le protocole de mise en accord de clé Otway-Rees. A la fin de ce chapitre, nous proposons une extension de ce protocole pour garantir l'authentification mutuelle.

Ensuite, au chapitre 2, nous introduisons une extension de la logique BAN pour donner une méthode formelle de preuve de sécurité qui se rapproche des preuves par réduction. Nous essaierons de prouver, par cette nouvelle méthode formelle, la sécurité (avec les notions de sécurité utilisées dans les preuves par réduction) de la clé échangée lors des

protocoles Otway-Rees et Otway-Ress étendu. Enfin, nous ferons apparaître des différences entre les trois méthodes en considérant une modification du protocole Otway-Rees.

Enfin, nous montrons, dans le chapitre 3, les capacités de la logique BAN étendue en examinant des protocoles de chiffrement asymétrique. Les résultats bien connus de sécurité du schéma de chiffrement ElGamal et d'une construction générique [1, 14] sont montrés à nouveau par cette méthode formelle.

Chapitre 1

Deux méthodes d'analyse d'un protocole d'authentification

1.1 Analyse par la logique BAN

1.1.1 Introduction

La logique BAN [6] a été proposée en 1989 comme une méthode formelle pour analyser des protocoles d'authentification. Dans un protocole d'authentification, on souhaite qu'un participant connaisse avec certitude sa partenaire, c'est pourquoi la logique BAN se concentre sur la question de croyance.

Comme pour décrire un système formel, on présente d'abord les notations, les postulats, puis les règles de déduction de la logique BAN.

Notations de base. Le formalisme de la logique BAN est composé d'un ensemble d'objets et d'un ensemble de règles de déduction. Les auteurs distinguent plusieurs sortes d'objets : participant légitime du protocole, clé de chiffrement, formule. Dans les notations ci-dessous, A, B, S désignent des participants spécifiques ; K_{ab}, K_{as}, K_{bs} désignent des clés spécifiques ; N_a, N_b, N_c désignent des aléas spécifiques. Les symboles P, Q, R réprésentent les participants ; X, Y les aléas ; K les clés

$P \models X$: P *croit* X, P fonctionne en supposant que X est vrai.

$P \triangleleft X$: P *voit* X, quelqu'un a envoyé un message contenant X à P et P peut lire et répéter X (peut-être après une étape de déchiffrement.)

$P \succ X$: P *a dit* X, P a envoyé un message contenant X à un moment quelconque. On ne sait pas si ce message est envoyé au cours du protocole courant mais on sait que P croit X au moment où il a envoyé X.

$P \Rightarrow X$: P a *juridiction* sur X. Le participant P a autorité sur X. Cette construction est utilisée quand on exige qu'un participant puisse faire des énoncés crédibles. Par exemple, la clé de chiffrement doit être générée de manière très convenable et dans certains protocoles, un serveur est considéré capable d'assurer cette propriété.

$\#(X)$: Une formule X est *fraîche*, c'est-à-dire que X n'a pas encore été envoyée avant la session courante du protocole.

$P \overset{K}{\leftrightarrow} Q$: K est une bonne clé partagée entre P et Q, c'est-à-dire exceptés P et Q ou un participant crédible par P et Q, personne ne connaît K.

$\{X\}_K$: Le message X est chiffré sous la clé K.

Postulats logiques. Pour le raisonnement, les auteurs ont donné les postulats de la logique BAN. On présente ici quelques règles importantes que l'on va utiliser dans l'analyse des protocoles.

— R1 : *message meaning règle* :

$$\frac{P \models P \overset{K}{\leftrightarrow} Q, P \triangleleft \{X\}_K}{P \models Q \succ X}$$

Cette règle signifie que si P croit en la clé K partagée avec Q et s'il voit le message X chiffré sous cette clé K, alors P croit que Q a dit X.

— R2 : *Nonce-verification règle* :

$$\frac{P \models \#(X), P \models Q \succ X}{P \models Q \models X}$$

Cette règle signifie que si P croit que X a été envoyé récemment (dans la session courante) et que Q a dit X, alors P croit que Q croit X.

— R3 : *juridiction règle* :

$$\frac{P \models Q \Rightarrow X, P \models Q \models X}{P \models X}$$

— R4 : *freshness règle* :

$$\frac{P \models \#(X)}{P \models \#(X, Y)}$$

Cette règle signifie qu'un message est fraîs si il contient une partie fraîche..

Protocole idéalisé. Pour analyser un protocole selon la logique BAN, il nous faut transformer le protocole réel en protocole formel que l'on appelle protocole idéalisé qui contient des formules déduites des messages échangés. Remarquons que le protocole réel est informel, tandis que le protocole idéalisé est formel. Cette phase de transformation n'est donc pas réalisée par une méthode formelle avec les règles explicites. La formule associée à chaque message ne peut être déterminée en considérant uniquement le message et ses composantes. Il est nécessaire d'avoir une vision globale du protocole et plus précisément des utilisations et interprétations de chaque message par les différents participants pour pouvoir transformer le plus correctement le protocole ordinaire en sa version idéalisée. Par exemple, si dans le protocole réel on a une étape :

$$A \to B \ : \{A, K_{ab}\}_{K_{bs}},$$

ceci dit à B, qui connaît la clé K_{bs}, que K_{ab} est une clé partagée avec A. Cette étape peut être idéalisée en :

$$A \to B \ : \{A \overset{K_{ab}}{\leftrightarrow} B\}_{K_{bs}}.$$

Hypothèses initiales. En plus des formules décrivant les échanges de messages, il faut donner les hypothèses et les conditions relatives à l'état initial et les croyances des différents participants au début d'une session d'exécution du protocole. Comme dans la phase de l'idéalisation, on ne peut pas réaliser cette étape de manière exacte mais par la vue globale du protocole, on essaie de donner les hypothèses les plus convenables.

Formaliser le but de l'authentification. L'authentification est souvent la première phase d'une communication protégée par une clé partagée. Alors, il faut au moins que les

participants croient en la qualité de la clé partagée :

$$\begin{aligned} A &\models A \overset{K_{ab}}{\leftrightarrow} B \\ B &\models A \overset{K_{ab}}{\leftrightarrow} B. \end{aligned}$$

Mais de la part d'un participant, A par exemple, la conclusion $A \models A \overset{K_{ab}}{\leftrightarrow} B$ n'est pas suffisante : il sait que K est une bonne clé partagée avec B (seuls A et B peuvent la connaître mais pas les autres) mais il ne sait pas si B a reçu K ou pas. Par conséquent, il est possible que A accepte K mais pas B. C'est pourquoi, dans certains protocoles, on essaie d'obtenir plus :

$$\begin{aligned} A \models B \models A \overset{K_{ab}}{\leftrightarrow} B \\ B \models A \models A \overset{K_{ab}}{\leftrightarrow} B. \end{aligned}$$

Concrètement, quand A termine et accepte la clé échangée, i.e $A \models B \models A \overset{K_{ab}}{\leftrightarrow} B$ et $A \models A \overset{K_{ab}}{\leftrightarrow} B$, il faut que B accepte la clé échangée, i.e $B \models A \overset{K_{ab}}{\leftrightarrow} B$ et réciproquement quand B termine et accepte la clé échangé, il faut que A accepte la clé échangée. Cette notion de sécurité est appelée authentification mutuelle.

Les étapes pour analyser un protocole

1 : Dériver le protocole idéalisé à partir du protocole original.

2 : Donner les hypothèses de l'état initial.

3 : Attacher les formules logiques à chaque étape du protocole, ces formules décrivent l'état du système après chaque étape.

4 : Déduire les assertions de croyance (exécutées par chaque participant) en utilisant les postulats logiques.

En réalité, les étapes 3 et 4 sont souvent réalisées en même temps.

1.1.2 Remarque

La logique BAN fait des hypothèses idéales sur la sécurité : $A \overset{K_{ab}}{\leftrightarrow} B$ signifie que seuls A et B connaissent la clé K_{ab}, et un tiers, même tout puissant, ne peut exploiter aucune information sur un chiffré ou sur K_{ab}.

1.2 Analyse par réduction

D'abord, on remarque qu'un attaquant tout puissant, ou bien un attaquant non-limité dans le temps, peut effectuer une recherche exhaustive pour balayer tous les cas possibles et il peut casser le protocole. Donc, pour analyser la sécurité des protocoles, il nous faut donner des hypothèses algorithmiques et donner des notions de sécurité.

Avec une preuve par réduction, sous des hypothèses algorithmiques précises, on montre la sécurité des protocoles. Pour cela, on considère un attaquant qui peut casser le protocole, puis on utilise cet attaquant pour construire une attaque contre une des hypothèses algorithmiques.

1.2.1 Notions de sécurité du chiffrement symétrique

D'abord, on précise des notations très souvent utilisées ci-après :

Coins : l'ensemble des séquences infinies
$\mathcal{M}$: $\mathsf{Espace-Message}$: espace des messages
$\mathcal{K}$: $\mathsf{Espace-Cle}$: espace des clés
$\mathcal{C}$: $\mathsf{Espace-Chiffrement} = \{0,1\}^\star$

Pour évaluer la sécurité effective des protocoles d'authentification, on formalise d'abord les notions de sécurité d'un schéma de chiffrement symétrique qui est utilisé dans des protocoles d'authentification. Il faut aussi préciser les informations accessibles à l'attaquant, les *moyens* dont il peut disposer.

Définition 1 *Un schéma de chiffrement symétrique $\mathcal{S} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ est défini par 3 algorithmes :*

$\mathcal{G}$: $\mathsf{Coins} \to \mathcal{K}$
$\mathcal{E}$: $\mathcal{K} \times \mathcal{M} \times \mathsf{Coins} \to \mathcal{C}$
$\mathcal{D}$: $\mathcal{K} \times \mathcal{C} \to \mathcal{M} \bigcup \{\perp\}$

Un schéma de chiffrement symétrique $\mathcal{S} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ est dit *déterministe* si $\mathcal{E}$ est déterministe, c'est-à-dire :

$$\mathcal{E} : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$$

On appelle $\mathcal{E}$ algorithme de chiffrement, $\mathcal{D}$ algorithme de déchiffrement et $\mathcal{G}$ algorithme de génération de clés. Il faut que $\mathcal{D}(\mathcal{E}(K, m, r)) = m$ pour tout $K \in \mathcal{K}, m \in \mathcal{M}, r \in \mathsf{Coins}$. Dans les schémas de chiffrement, $\mathcal{D}(K, c) = \perp$ est utilisé dans le cas où c n'est le chiffré d'aucun message m sous la clé K.

Définition 2 *Un schéma de chiffrement symétrique $\mathcal{S} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ est dit $\epsilon-$ sémantiquement sûr face à un adversaire $\mathcal{A}$ si :*

$$\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind}}(\mathcal{A}) \stackrel{\mathsf{def}}{=} \left| 2 \times \Pr_{b,r} \left[\begin{array}{c} m_0, m_1 \leftarrow \mathcal{A}_1, K \leftarrow \mathcal{K}, \\ c = \mathcal{E}(K, m_b, r), b' = \mathcal{A}_2(m_0, m_1, c) : b' = b \end{array} \right] - 1 \right|$$

est inférieur à ϵ.
Un schéma de chiffrement symétrique est dit $(t, \epsilon)-$ sémantiquement sûr si :

$$\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind}}(t) \stackrel{\mathsf{def}}{=} \underset{|\mathcal{A}| \leq t}{\mathrm{Max}}(\mathsf{Adv}_{\mathcal{S}}^{ind}(\mathcal{A})) \leq \epsilon.$$

Les moyens d'un attaquant contre le chiffrement symétrique

Contrairement au chiffrement asymétrique (que l'on présente après) où un attaquant peut chiffrer tout message de son choix, dans le contexte symétrique, l'accès à l'algorithme de chiffrement (avec la restriction de ne pas l'utiliser sur m_0 et m_1) donne à l'attaquant des informations puisqu'il ne connaît pas la clé sécrète. Cette attaque est appelée attaque à *clairs choisis* (ou *chosen-plaintext attack* - CPA)

Dans certains cas, l'attaquant peut avoir accès à l'algorithme de déchiffrement (avec la restriction de ne pas l'utiliser sur le challenge). Cette attaque est appelée attaque à *chiffrés choisis* - CCA. On note le CPA/CCA l'attaque qui donne accès à la fois à l'algorithme de chiffrement et à l'algorithme de déchiffrement.

1.2.2 Mise en accord de clé

Une notion de sécurité qu'on doit examiner est celle de la clé échangée dans un tel protocole. Pour cela, on suppose d'abord que l'attaquant contrôle le réseau et tous les messages échangés. Il peut de plus interagir avec les participants en leur posant une $\mathsf{Test}-\mathsf{query}$: quand il pose une $\mathsf{Test}-\mathsf{query}$ à un des deux participants (A, par exemple) qui a accepté le protocole, on choisit β aléatoire, si $\beta = 0$ on retourne K_0—la clé obtenue dans le protocole, si $b = 1$ on retourne K_1—une chaîne aléatoire. On note $\mathrm{Prot}(K_0)$ ($\mathrm{Prot}(K_1)$ respectivement) le protocole dans lequel la clé échangée reçoit la valeur K_0 (K_1 respectivement).

Définition 3 *La clé échangée dans un protocole d'authentification est dite ϵ - sémantiquement sûre face à un adversaire $\mathcal{A}$ si :*

$$\mathsf{Adv}_P^{\mathsf{ind}}(\mathcal{A}) \stackrel{\mathsf{def}}{=} \left| 2 \times \Pr_{b,r}\left[\begin{array}{c} K_0, K_1 \leftarrow \mathcal{K}, \beta \leftarrow \{0,1\} \\ K_\beta \leftarrow \mathsf{Test}-\mathsf{query}, \beta' = \mathcal{A}(\mathrm{Prot}(K_0), K_\beta) : \beta' = \beta \end{array} \right] - 1 \right|$$

est inférieur à ϵ.
La clé échangée dans un protocole d'authentification est dite $(t,\epsilon)-$sémantiquement sûre si :

$$\mathsf{Adv}_P^{\mathsf{ind}}(t) \stackrel{\mathsf{def}}{=} \max_{|\mathcal{A}| \leq t}(\mathsf{Adv}_P^{\mathsf{ind}}(\mathcal{A})) \leq \epsilon.$$

Maintenant, on donne la définition de l'authentification mutuelle, une notion de sécurité également importante pour un protocole d'authentification. Le but est qu'après chaque session, le participant qui accepte la clé échangée veut s'assurer que sa partenaire l'accepte aussi. Puisqu'il est impossible d'obtenir à la fois que A accepte implique B accepte et que B accepte implique A accepte, on change un peu le but. Intuitivement, un participant *accepte* lorsque toutes les étapes sont consistantes et lui permettent d'extraire la clé partagée. Un participant *termine* le protocole si il n'y a plus de message à échanger [5]. Les définitions exactes de ces deux notions sont introduites pour chaque protocole.

Définition 4 *Un protocole d'échange de clé garantit l'authentification mutuelle si aucun attaquant ne peut pas contrefaire le rôle d'un participant. Plus précisément, lorsqu'un participant termine en acceptant, sa partenaire a nécessairement accepté.*

1.3 Analyse du protocole d'Otway-Rees

1.3.1 Protocole d'Otway-Rees

Otway et Rees ont proposé un protocole d'échange de clé partagée en 1987 [13]. Ce protocole implique deux participants et un serveur. Il est intéressant en raison du petit nombre de messages échangés. On présente le protocole ci-dessous, A et B sont deux participants, S est le serveur. K_{as} est la clé secrète que A partage avec S, K_{bs} est la clé

secrète que B partage avec S. N_a, N_b et M sont des aléas générés par A et B. Le serveur S génère K_{ab} qui deviendra la clé de session partagée entre A et B.

$$
\begin{array}{lrcl}
\textit{Message 1} & A \rightarrow B & : & M, A, B, \{N_a, M, A, B\}_{K_{as}} \\
\textit{Message 2} & B \rightarrow S & : & M, A, B, \{N_a, M, A, B\}_{K_{as}}, \{N_b, M, A, B\}_{K_{bs}} \\
\textit{Message 3} & S \rightarrow B & : & M, \{N_a, K_{ab}\}_{K_{as}}, \{N_b, K_{ab}\}_{K_{bs}} \\
\textit{Message 4} & B \rightarrow A & : & M, \{N_a, K_{ab}\}_{K_{as}}
\end{array}
$$

1.3.2 Analyse par la logique BAN

On exécute les étapes comme présenté précédemment pour analyser le protocole Otway-Rees :

1 : Dériver le protocole idéalisé à partir du protocole original

$$
\begin{array}{lrcl}
\textit{Message 1} & A \rightarrow B & : & \{N_a, N_c\}_{K_{as}} \\
\textit{Message 2} & B \rightarrow S & : & \{N_a, N_c\}_{K_{as}}, \{N_b, N_c\}_{K_{bs}} \\
\textit{Message 3} & S \rightarrow B & : & \{N_a, (A \overset{K_{ab}}{\leftrightarrow} B), B \succ N_c\}_{K_{as}}, \\
 & & & \{N_b, (A \overset{K_{ab}}{\leftrightarrow} B), A \succ N_c\}_{K_{bs}} \\
\textit{Message 4} & B \rightarrow A & : & \{N_a, (A \overset{K_{ab}}{\leftrightarrow} B), B \succ N_c\}_{K_{as}}
\end{array}
$$

Dans les protocoles idéalisés, les auteurs [6] omettent tous les messages clairs échangés car ils n'apparaissent pas dans les règles de déduction et ils les remplacent souvent par une forme chiffrée. Dans ce protocole, N_c est l'aléa commun entre A et B et il correspond à M, A, B dans le protocole réel. Pour faire un lien entre deux participants, ils ajoutent les énoncés $A \succ N_c$ et $B \succ N_c$ dans les messages 3 et 4.

2 : Donner les hypothèses de l'état initial.

$$
\begin{array}{rr}
A \models A \overset{K_{as}}{\leftrightarrow} S & B \models B \overset{K_{bs}}{\leftrightarrow} S \\
S \models A \overset{K_{as}}{\leftrightarrow} S & S \models B \overset{K_{bs}}{\leftrightarrow} S \\
S \models A \overset{K_{ab}}{\leftrightarrow} B & \\
A \models (S \Rightarrow A \overset{K_{ab}}{\leftrightarrow} B) & B \models (S \Rightarrow A \overset{K_{ab}}{\leftrightarrow} B) \\
A \models (S \Rightarrow B \succ X) & B \models (S \Rightarrow A \succ X) \\
A \models \#(N_a) & B \models \#(N_b) \\
A \models \#(N_c) &
\end{array}
$$

3, 4 : Attacher les formules logiques à chaque étape du protocole et déduire les assertions de croyance :

— Du message 1 : $B \triangleleft \{N_a, N_c\}_{K_{as}}$
B peut voir le message mais il ne peut pas le comprendre. Il génère un message de même forme, puis il l'envoie à S avec le message de A.

— Du message 2 : $S \triangleleft \{N_a, N_c\}_{K_{as}}$ et $\{N_b, N_c\}_{K_{bs}}$
Par l'hypothèse : $S \models A \overset{K_{as}}{\leftrightarrow} S$ et $S \models B \overset{K_{bs}}{\leftrightarrow} S$, en utilisant la règle R1, on déduit :

$$S \models A \succ (N_a, N_c)$$

$$S \models B \succ (N_b, N_c)$$

S ne sait pas si ces messages sont rejoués ou pas mais il peut les utiliser pour répondre à A et B.

— Du message 3 : $B \triangleleft \{N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \succ N_c\}_{K_{bs}}$
Par l'hypothèse : $B \models B \overset{K_{bs}}{\leftrightarrow} S$, en utilisant la règle R1, on déduit :

$$B \models S \succ \{N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \succ N_c\}.$$

De l'hypothèse $B \models \#(N_b)$, d'après la règle R4 : $B \models \#(\{N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \succ N_c\})$. En utilisant la règle R2, on déduit :

$$B \models S \models \{N_b, A \overset{K_{ab}}{\leftrightarrow} B, A \succ N_c\}.$$

Une fois de plus, par les hypothèses $B \models (S \Rightarrow A \overset{K_{ab}}{\leftrightarrow} B)$ et $B \models (S \Rightarrow A \succ X)$, en utilisant la règle R3, on déduit :

$$B \models A \overset{K_{ab}}{\leftrightarrow} B, B \models A \succ N_c.$$

— Du message 4, on déduit de la même façon :

$$A \models A \overset{K_{ab}}{\leftrightarrow} B, A \models B \succ N_c.$$

Mais, par l'hypothèse $A \models \#(N_c)$, en utilisant la règle R2, de $A \models B \succ N_c$, on déduit :

$$A \models B \models N_c.$$

Alors, on obtient les conclusions suivantes :

$$A \models A \overset{K_{ab}}{\leftrightarrow} B \qquad A \models B \models N_c$$
$$B \models A \overset{K_{ab}}{\leftrightarrow} B \qquad B \models A \succ N_c$$

Avec cette conclusion, A et B savent que K_{ab} est une bonne clé partagée mais l'un ne sait pas si l'autre connaît, ou non cette clé. A est dans une meilleure position que B, il sait que B a émis un aléa, donc B a participé à cette session, tandis que B sait seulement que A a utilisé un aléa mais il ne sait pas si cet aléa est rejoué ou pas.

1.3.3 Analyse par réduction

Preuve de la sécurité sémantique de la clé

Hypothèse de la sécurité du chiffrement. Dans les protocoles d'authentification considérés, on fait l'hypothèse que le schéma de chiffrement satisfait les conditions suivantes :

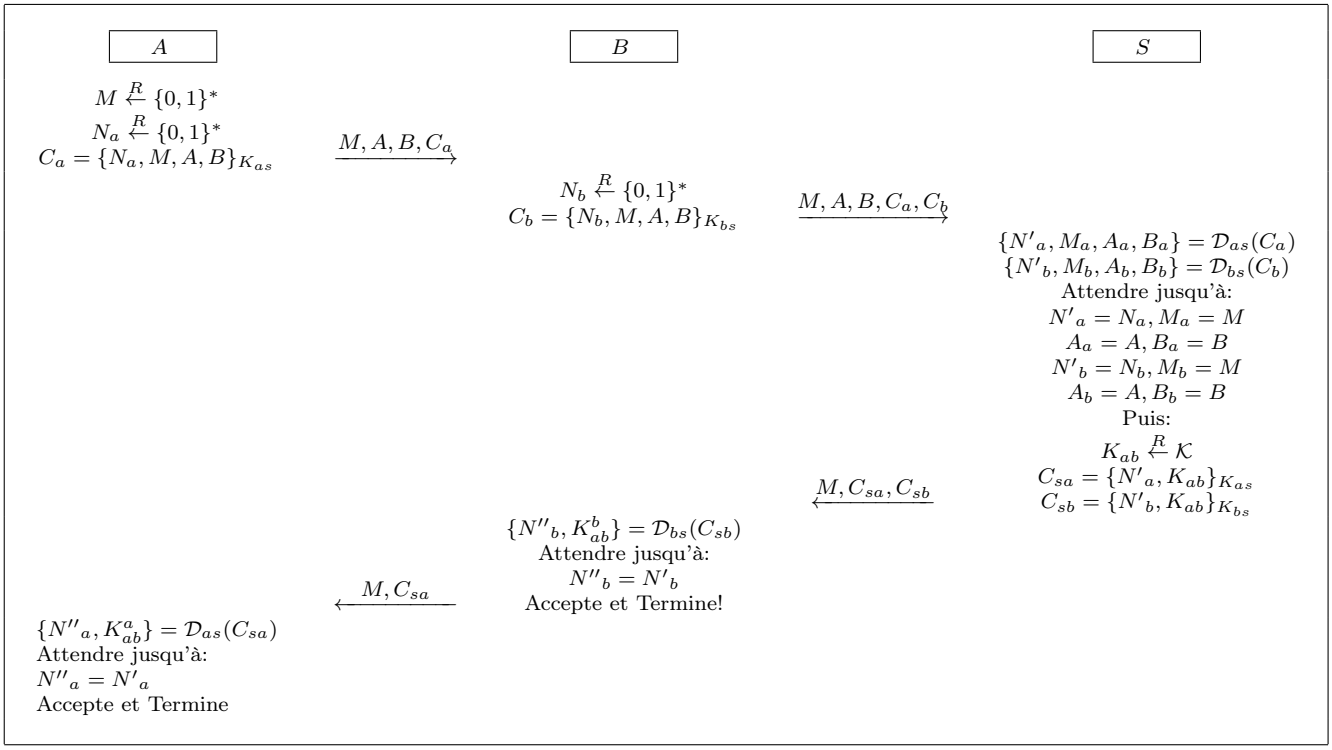

FIGURE 1.1 – L'action des participants dans le Protocole d'Otway-Rees

P1 : l'algorithme de chiffrement est déterministe : on supprime l'aspect probabiliste dans la définition de l'algorithme de chiffrement, donc :

$$\mathcal{E} : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$$

P2 : l'algorithme de chiffrement est une permutation, ce qui revient à dire que $\mathcal{M}$ et $\mathcal{C}$ sont identiques (qu'on note $\mathcal{M}$) et, pour chaque clé K, l'algorithme de chiffrement $\mathcal{E}$ sous la clé K est une permutation des messages de même longueur de $\mathcal{M}$ à $\mathcal{M}$. On note ci-dessous k la longueur des messages clairs et des messages chiffrés.

P3 : le schéma de chiffrement est sémantiquement sûr selon des attaques à clairs choisis et chiffrés choisis ($\mathsf{IND} - \mathsf{CPA/CCA}$).

P4 : l'algorithme de génération de clés génère des clés de manière uniformément distribuée.

Hypothèse du temps. Le temps pour accéder à un oracle (obtenir une réponse de l'oracle) ainsi que le temps pour vérifier si un élément est dans une liste, sont une même constante (que l'on considère comme une unité).

Théorème 1 *La clé échangée dans le protocole Otway-Rees est sémantiquement sûre sous l'hypothèse que le schéma de chiffrement utilisé dans protocole soit sématiquement sûr selon des attaques à clairs et chiffrés choisis :*

$$\mathsf{Adv}_P^{\mathsf{ind}}(\mathcal{A}) \leq 2N \cdot \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$$

Preuve.

Game$_0$: Le protocole est réalisé avec les clés K^i_{ab} ($i = \overline{1, N}$) choisies aléatoirement par S pour N sessions éventuellement simultanées. On note $\text{Prot}(K^i_{ab})$ le protocole correspondant à la session i. L'attaquant "casse" la clé de la session t en posant la $\mathsf{Test} - \mathsf{query}$ à un oracle (soit A, soit B) et il reçoit K_β (β est choisi aléatoirement, $K_1 = K^t_{ab}$ et $K_0 = K'_{ab}$ choisi aléatoirement). L'attaquant doit déterminer si K_β est K^t_{ab} ou pas, il retourne son choix β'. Avec probabilité $(\epsilon{+}1)/2$, $\beta' = \beta$ (plus clairement, $\beta' = (K_\beta = K^t_{ab})$). On note cet événement S_0 ainsi que S_i dans les jeux Game_i ci-dessous : $\Pr[S_0] = (\epsilon + 1)/2$.

Game$_1$: On choisit aléatoirement i et on suppose que l'attaquant veut casser la clé K^i_{ab}(que l'on note K_{ab}) du protocole $\text{Prot}(K^i_{ab})$ (que l'on note par clarté $\text{Prot}(K_{ab})$). On stoppe les exécutions où la session i n'est pas testée et on retourne β' aléatoire. La probabilité que cette session soit justement celle choisie par l'attaquant est $1/N$. Alors :

$$\begin{aligned}
\Pr[S_1] &= \Pr[\beta = \beta'] \\
&= \Pr[\beta = \beta' \wedge i \neq t] + \Pr[\beta = \beta' \wedge i = t] \\
&= \Pr[i \neq t].\Pr[\beta = \beta' | i \neq t] + \Pr[i = t].\Pr[\beta = \beta' | i = t] \\
&= \frac{N-1}{N} \cdot \frac{1}{2} + \frac{1}{N} \cdot \frac{\epsilon + 1}{2} \\
&= \frac{1}{2} + \frac{\epsilon}{2N}.
\end{aligned}$$

Game$_2$: On modifie encore un peu ce jeu réel. Les clés K_{ab} et K'_{ab} ainsi que β sont choisis aléatoirement dès le début du jeu : K_{ab}, $K'_{ab} \overset{R}{\leftarrow} \mathcal{K}$, $\beta \overset{R}{\leftarrow}\{0,1\}$: $\Pr[S_2] = \Pr[S_1]$.

Game$_3$: On remplace les chiffrements et déchiffrements avec la clé K_{as} et la clé K_{bs} par les couples d'oracles $(\mathcal{E}_{as}, \mathcal{D}_{as})$ et $(\mathcal{E}_{bs}, \mathcal{D}_{bs})$ respectivement. En utilisant ces oracles, l'attaquant retourne son choix β' : $\Pr[S_3] = \Pr[S_2]$.

Game$_4$: On choisit aléatoirement une valeur b : $b \overset{R}{\leftarrow} \{0, 1\}$.

— Si $b = 0$, on simule $\text{Prot}(K_{ab})$, l'attaquant retourne β' : $\Pr[S_4] = \Pr[\beta' = \beta]$, ou bien $\Pr[S_4] = \Pr[b \bigoplus \beta' = \beta]$.

— si $b = 1$, on simule $\text{Prot}(K'_{ab})$, l'attaquant retourne β' : $\Pr[S_4] = \Pr[\beta' = \neg(\beta)]$ (puisque $K_0 = K'_{ab}$ et $\Pr[S_4] = \Pr[\beta' = (K_\beta = K'_{ab})]$), ou bien $\Pr[S_4] = \Pr[b \bigoplus \beta' = \beta]$.

On combine ces deux cas et on définit $b' = b \bigoplus \beta'$. On note S'_4 l'événement $b' = \beta$, alors $\Pr[S'_4] = \Pr[S_4]$.

Game$_5$: comme ci-dessus, mais dans ce jeu, b sera le bit choisi par le challengeur pour évaluer la sécurité sémantique du schéma de chiffrement $\mathcal{S}$. D'où, la simulation qui utilise β mais plus b. Dans le jeu précédent, on voit que $\Pr[b' = \beta] = \Pr[b = \beta \bigoplus \beta']$. Alors, on retourne maintenant $b'' = \beta \bigoplus \beta'$. On note cet événement S''_5 ainsi que S''_i dans les jeux ci-dessous. On a $\Pr[S''_5] = \Pr[b'' = b] = \Pr[S'_5] = \Pr[S'_4]$.

Game$_6$: On ne modifie pas mais on récrit le jeu précédent. K_{ab}, $K'_{ab} \overset{R}{\leftarrow} \mathcal{K}$, $b \overset{R}{\leftarrow}\{0,1\}$. On note $m_0 = \{N_a, K_{ab}\}, m'_0 = \{N_b, K_{ab}\}$ et $m_1 = \{N_a, K'_{ab}\}, m'_1 = \{N_b, K'_{ab}\}$. On reçoit dans les messages 3 et 4 le chiffrement d'un des deux couples (m_0, m'_0) et (m_1, m'_1) : $c = (\mathcal{E}_{as}(m_b), \mathcal{E}_{bs}(m'_b))$ et on évalue b'', $\Pr[S''_6] = \Pr[S''_5]$.

Game$_7$: On remplace les deux couples (m_0, m'_0) et (m_1, m'_1) par (m_1, m'_0) et (m_1, m'_1). La différence est l'avantage de distinguer $(\mathcal{E}_{as}(m_0), \mathcal{E}_{bs}(m'_0))$ et $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_0))$. Pour ce dernier, on remplace les oracles $\mathcal{E}_{bs}$ et $\mathcal{D}_{bs}$ par les algorithmes de chiffrement et déchiffrement avec la clé K_{bs}. Ça ne change en rien le jeu, et l'avantage de distinguer $(\mathcal{E}_{as}(m_0), \mathcal{E}_{bs}(m'_0))$ et $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_0))$, en utilisant les oracles $\mathcal{E}_{as}$, $\mathcal{E}_{bs}$, $\mathcal{D}_{as}$, $\mathcal{D}_{bs}$ est égal à l'avantage de distinguer $\mathcal{E}_{as}(m_0)$ et $\mathcal{E}_{as}(m_1)$ en utilisant les oracles $\mathcal{E}_{as}$ et $\mathcal{D}_{as}$. $|\Pr[S''_7] - \Pr[S''_6]| \leq \mathsf{Adv}_{\mathcal{E}}^{\mathsf{ind-cpa/cca}}(t)$.

Game$_8$: On remplace les deux couples (m_1, m'_0) et (m_1, m'_1) par (m_1, m'_1) et (m_1, m'_1). La différence est l'avantage de distinguer $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_0))$ et $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_1))$. Pour ce dernier, on remplace les oracles $\mathcal{E}_{as}$ et $\mathcal{D}_{as}$ par les algorithmes de chiffrement et déchiffrement avec la clé K_{as}. Ça ne change en rien le jeu, et l'avantage de distinguer $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_0))$ et $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_1))$, en utilisant les oracles $\mathcal{E}_{as}$, $\mathcal{E}_{bs}$, $\mathcal{D}_{as}$, $\mathcal{D}_{bs}$ est égal à l'avantage de distinguer $\mathcal{E}_{bs}(m_0)$ et $\mathcal{E}_{bs}(m_1)$ en utilisant les oracles $\mathcal{E}_{bs}$ et $\mathcal{D}_{bs}$: $|\Pr[S''_8] - \Pr[S''_7]| \leq \mathsf{Adv}_{\mathcal{E}}^{\mathsf{ind-cpa/cca}}(t)$.
Cependant, dans ce dernier jeu Game$_9$, on doit distinguer $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_1))$ et $(\mathcal{E}_{as}(m_1), \mathcal{E}_{bs}(m'_1))$ qui sont identiques, donc b est indépendant de la vue du simulateur, ainsi $\Pr[S''_9] = \frac{1}{2}$.
L'inégalité triangulaire nous donne :

$$\begin{aligned} \frac{\epsilon}{2N} = \frac{\epsilon/N + 1}{2} - \frac{1}{2} &= |\Pr[S_1] - \Pr[S''_9]| \\ &\leq 2\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t). \end{aligned}$$

□

Sous l'hypothèse de la sécurité sémantique du schéma de chiffrement, on déduit la sécurité sémantique de la clé K_{ab}.

Analyse d'authentification mutuelle

— L'attaquant peut prendre un ancien message 1 de A et le renvoyer à B, puis il retient le message 4 que B envoie à A. Bien que l'attaquant ne connaisse pas la clé commune K_{ab}, A ne la connaît pas non plus puisqu'il ne participe pas au protocole. Alors, B accepte la clé K_{ab} mais pas A. Ceci montre que l'attaquant peut casser l'authentification $A - B$.

— L'attaquant suit le protocole jusqu'au le message 3, puis il empêche le message 3 que S envoie à B. Il ne l'envoie pas à B mais il joue le rôle de B pour envoyer le message 4, qui est explicite dans le message 3 à A. Alors A accepte la clé K_{ab} sans connaître que B ne reçoit pas K_{ab}. Ceci montre que l'attaquant peut casser l'authentification $B - A$.

1.4 Analyse du protocole d'Otway-Rees étendu

1.4.1 Protocole d'Otway-Rees étendu

Dans cette partie, on propose une extension du protocole d'Otway-Rees et on montre que ce protocole garantit de plus l'authentification mutuelle. Pour cela, on ajoute une étape de confirmation de connaissance de la clé.

$$
\begin{array}{lrcl}
\textit{Message 1} & A \to B & : & M, A, B, \{N_a, M, A, B\}_{K_{as}} \\
\textit{Message 2} & B \to S & : & M, A, B, \{N_a, M, A, B\}_{K_{as}}, \{N_b, M, A, B\}_{K_{bs}} \\
\textit{Message 3} & S \to B & : & M, \{N_a, N_b, K_{ab}\}_{K_{as}}, \{N_a, N_b, K_{ab}\}_{K_{bs}} \\
\textit{Message 4} & B \to A & : & M, \{N_a, N_b, K_{ab}\}_{K_{as}}, \{N_a\}_{K_{ab}} \\
\textit{Message 5} & A \to B & : & \{N_b\}_{K_{ab}}
\end{array}
$$

1.4.2 Analyse par la logique BAN

1 : Dériver le protocole idéalisé à partir du protocole original.

$$
\begin{array}{lrcl}
\textit{Message 1} & A \to B & : & \{N_a, N_c\}_{K_{as}} \\
\textit{Message 2} & B \to S & : & \{N_a, N_c\}_{K_{as}}, \{N_b, N_c\}_{K_{bs}} \\
\textit{Message 3} & S \to B & : & \{N_a, N_b, (A \overset{K_{ab}}{\leftrightarrow} B), B \succ N_c\}_{K_{as}}, \\
 & & & \{N_a, N_b, (A \overset{K_{ab}}{\leftrightarrow} B), A \succ N_c\}_{K_{bs}}, \\
\textit{Message 4} & B \to A & : & \{N_a, N_b, (A \overset{K_{ab}}{\leftrightarrow} B), B \succ N_c\}_{K_{as}}, \\
 & & & \{N_a, (A \overset{K_{ab}}{\leftrightarrow} B)\}_{K_{ab}} \\
\textit{Message 5} & A \to B & : & \{N_b, (A \overset{K_{ab}}{\leftrightarrow} B)\}_{K_{ab}}
\end{array}
$$

Dans le message 4 du protocole initial, B envoie $\{N_a\}_{K_{ab}}$ à A pour montrer qu'il croit que la clé partagée est bonne, donc l'énoncé $(A \overset{K_{ab}}{\leftrightarrow} B)$ est attaché au message 4 du protocole idéalisé de façon naturelle. De même, on attache $(A \overset{K_{ab}}{\leftrightarrow} B)$ au message 5 du protocole idéalisé.

2 : Les hypothèses sont celles du protocole original.

3, 4 : Attacher les formules logiques à chaque étape du protocole et déduire les assertions de croyance :
Avec la même analyse que pour le protocole original, on obtient :

$$
\begin{array}{ll}
A \models A \overset{K_{ab}}{\leftrightarrow} B & A \models B \models N_c \\
B \models A \overset{K_{ab}}{\leftrightarrow} B & B \models A \succ N_c.
\end{array}
$$

De plus, quand A reçoit $\{N_a, (A \overset{K_{ab}}{\leftrightarrow} B)\}_{K_{ab}}$ dans le message 4 :

$$A \triangleleft \{N_a, (A \overset{K_{ab}}{\leftrightarrow} B)\}_{K_{ab}}.$$

Comme on a démontré $A \models A \overset{K_{ab}}{\leftrightarrow} B$, en utilisant la règle R1, on déduit :

$$A \models B \succ \{N_a, A \overset{K_{ab}}{\leftrightarrow} B\}.$$

Par l'hypothèse $A \models \#(N_a)$, on a $A \models \#(\{N_a, A \overset{K_{ab}}{\leftrightarrow} B\})$. En utilisant la règle R3, on déduit :

$$A \models B \models A \overset{K_{ab}}{\leftrightarrow} B.$$

On analyse le message 5 de la même manière. On obtient alors :

$$A \models B \models A \overset{K_{ab}}{\leftrightarrow} B$$

$$B \models A \models A \overset{K_{ab}}{\leftrightarrow} B.$$

Cette conclusion est beaucoup plus intéressante que celle du protocole original puisque l'un sait que l'autre croit en la clé partagée.

1.4.3 Analyse par réduction

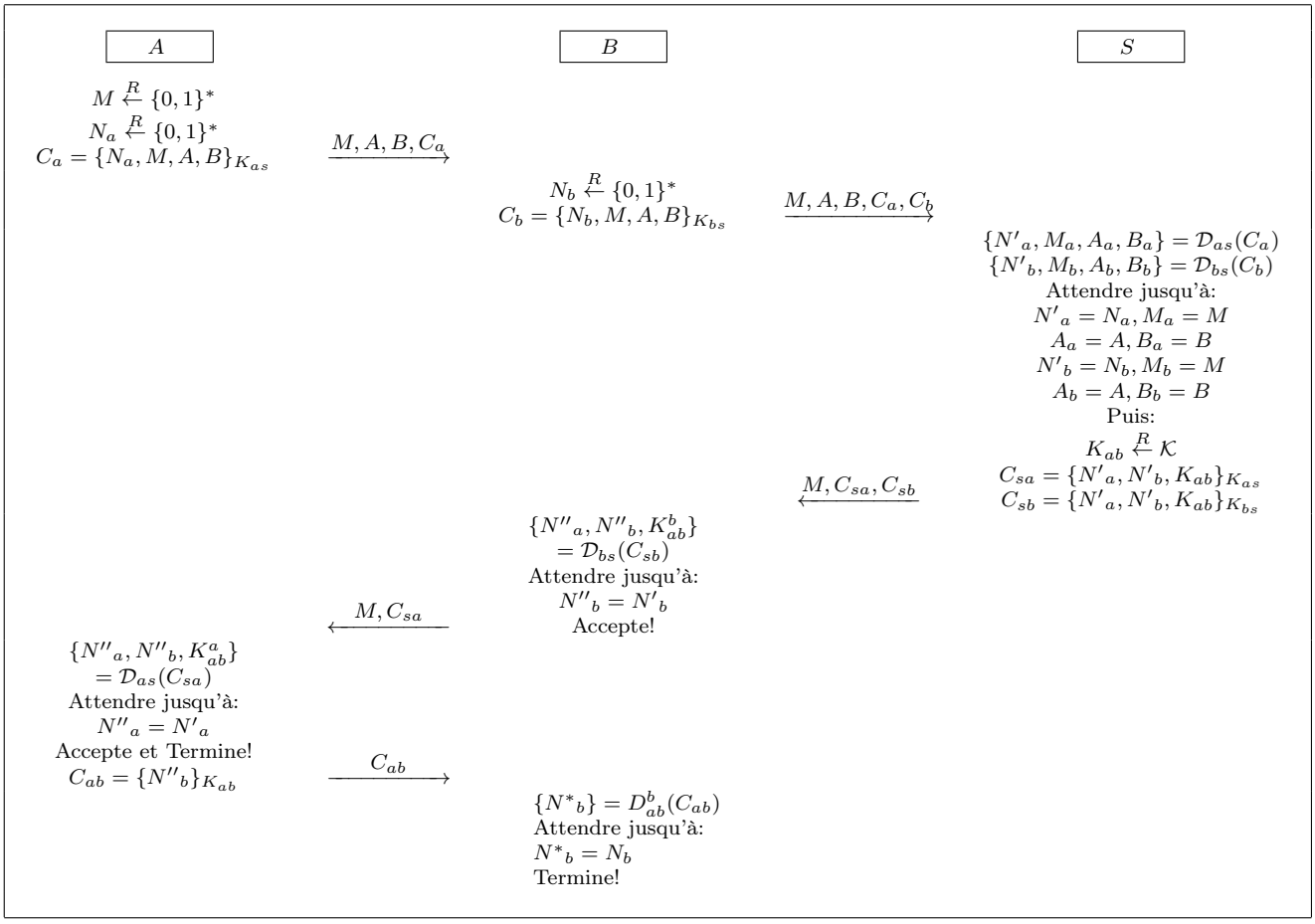

FIGURE 1.2 – L'action des participants dans le Protocole Otway-Rees étendu

Preuve de la sécurité sémantique de la clé

Théorème 2 *La clé échangée dans le protocole d'extension d'Otway-Rees est sémantiquement sûre sous l'hypothèse que le schéma de chiffrement utilisé dans protocole soit*

sémantiquement sûr selon des attaques à clairs et chiffrés choisis :

$$\mathsf{Adv}_P^{\mathsf{ind}}(t) \leq 12N \cdot \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$$

Preuve.

Game$_0$: Le protocole est réalisé avec les clés K_{ab}^i ($i = \overline{1, N}$) choisies aléatoirement par S pour N sessions éventuellement simultanées. On note $\mathrm{Prot}(K_{ab}^i)$ le protocole correspondant à la session i. L'attaquant casse la clé d'un session t en posant la $\mathsf{Test-query}$ à un oracle (soit A soit B) et il reçoit K_β (β est choisi aléatoirement, $K_0 = K_{ab}^t$ et $K_1 = K'_{ab}$ choisie aléatoirement). L'attaquant retourne son choix β'. Avec probabilité $(\epsilon{+}1)/2$, $\beta = \beta'$. On note cet événement S_0 ainsi que S_i dans les jeux Game$_i$ ci-dessous : $\Pr[S_0] = (\epsilon + 1)/2$.

Game$_1$: On choisit aléatoirement i et on suppose que l'attaquant veut casser la clé K_{ab}^i (que l'on note K_{ab}) du protocole $\mathrm{Prot}(K_{ab}^i)$ (que l'on note par clarté $\mathrm{Prot}(K_{ab})$). On stoppe les exécutions où la $i$ème session n'est pas testée et on retourne β' aléatoire. La probabilité que cette session est justement celle choisie par l'attaquant est $1/N$. Comme on a montré dans le jeu Game$_1$ dans la preuve du théorème 1, on a :

$$\Pr[S_1] = \Pr[\beta = \beta'] = \frac{1}{2} + \frac{\epsilon}{2N}.$$

Game$_2$: On remplace les chiffrements et déchiffrements avec la clé K_{as} et la clé K_{bs} par les couples d'oracles $(\mathcal{E}_{as}, \mathcal{D}_{as})$ et $(\mathcal{E}_{bs}, \mathcal{D}_{bs})$ respectivement. En utilisant ces oracles, l'attaquant retourne son choix β' : $\Pr[S_2] = \Pr[S_1]$.

Game$_3$: Dans ce jeu, on remplace N_a, N_b dans les messages chiffrés par K_{as}, K_{bs} par N'_a, N'_b aléatoires.

Lemme 1 $|\Pr[S_3] - \Pr[S_2]| \leq 4\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$.

Preuve.

Game$_0^+$: On note $m_0 = \{N_a, M, A, B\}$, $m_1 = \{N_b, M, A, B\}$, $m_2 = m_3 = \{N_a, N_b, K_{ab}\}$ et $m'_0 = \{N'_a, M, A, B\}$, $m'_1 = \{N'_b, M, A, B\}$, $m'_2 = m'_3 = \{N'_a, N'_b, K_{ab}\}$. On joue ce jeu comme le jeu Game$_3$, c'est-à-dire on joue avec les messages (m_0, m_1, m_2, m_3). $\Pr[S_0^+] = \Pr[S_3]$.

Game$_1^+$: On remplace le tuple (m_0, m_1, m_2, m_3) par (m'_0, m_1, m_2, m_3). La différence est l'avantage de distinguer $((\mathcal{E}_{as}(m_0), \mathcal{E}_{bs}(m_1)), (\mathcal{E}_{as}(m_2), \mathcal{E}_{bs}(m_3)))$ et $((\mathcal{E}_{as}(m'_0), \mathcal{E}_{bs}(m_1)), (\mathcal{E}_{as}(m_2), \mathcal{E}_{bs}(m_3)))$. De même que dans le jeu Game$_8$ dans la preuve du théorème 1, on a : $|\Pr[S_1^+] - \Pr[S_0^+]| \leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(\mathrm{t})$.

Game$_2^+$: On remplace le tuples (m'_0, m_1, m_2, m_3) par (m'_0, m'_1, m_2, m_3). La différence est l'avantage de distinguer $((\mathcal{E}_{as}(m'_0), \mathcal{E}_{bs}(m_1)), (\mathcal{E}_{as}(m_2), \mathcal{E}_{bs}(m_3)))$ et $((\mathcal{E}_{as}(m'_0), \mathcal{E}_{bs}(m'_1)), (\mathcal{E}_{as}(m_2), \mathcal{E}_{bs}(m_3)))$. De même que dans le jeu Game$_8$ dans la preuve du théorème 1, on a : $|\Pr[S_2^+] - \Pr[S_1^+]| \leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$.

Game$_3^+$: On remplace le tuples (m'_0, m'_1, m_2, m_3) par (m'_0, m'_1, m'_2, m_3). La différence est l'avantage de distinguer $((\mathcal{E}_{as}(m'_0), \mathcal{E}_{bs}(m'_1)), (\mathcal{E}_{as}(m_2), \mathcal{E}_{bs}(m_3)))$ et $((\mathcal{E}_{as}(m'_0), \mathcal{E}_{bs}(m'_1)), (\mathcal{E}_{as}(m'_2), \mathcal{E}_{bs}(m_3)))$. De même que dans le jeu Game$_8$ dans la preuve du théorème 1, on a : $|\Pr[S_3^+] - \Pr[S_2^+]| \leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$.

Game$_4^+$: On remplace deux tuples (m'_0, m'_1, m'_2, m_3) par (m'_0, m'_1, m'_2, m'_3). La différence est l'avantage de distinguer $((\mathcal{E}_{as}(m'_0), \mathcal{E}_{bs}(m'_1)), (\mathcal{E}_{as}(m'_2), \mathcal{E}_{bs}(m_3))$ et $((\mathcal{E}_{as}(m'_0),$

$\mathcal{E}_{bs}(m'_1)$), ($\mathcal{E}_{as}(m'_2)$, $\mathcal{E}_{bs}(m'_3)$. De même que dans le jeu Game_8 dans la preuve du théorème 1, on a : $|\Pr[S_8^+] - \Pr[S_7^+]| \leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$.

On remarque que le jeu Game_4^+ est le même que le jeu Game_2, on en déduit :

$$|\Pr[S_3] - \Pr[S_2]| = |\Pr[S_4^+] - \Pr[S_0^+]| \leq 4\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t).$$

□

Game_4 : On remplace les messages $\{N_a\}_{K_{ab}}$ et $\{N_b\}_{K_{ab}}$ dans les messages 4 et 5 par c_a, c_b choisis aléatoirement. De l'hypothèse que le chiffrement $\mathcal{E}$ est une permutation, N_a est implicitement déduit de c_a, et N_b est implicitement déduit de c_b. Puisque N_a, N_b sont indépendants des autres messages, ce remplacement ne change rien : $\Pr[S_4] = \Pr[S_3]$.

Game_5 : On modifie un peu le jeu précédent. Les clés K_{ab} et K'_{ab}, et β sont choisis aléatoirement au début du jeu : K_{ab}, $K'_{ab} \xleftarrow{R} \mathcal{K}$, $\beta \xleftarrow{R} \{0,1\}$: $\Pr[S_5] = \Pr[S_4]$.

Game_6 : On choisit aléatoirement une valeur b, $b \xleftarrow{R} \{0,1\}$. Si $b = 0$, on simule $\text{Prot}(K_{ab})$, si $b = 1$, on simule $\text{Prot}(K'_{ab})$. A la fin, l'attaquant retourne β' et on définit $b' = b \bigoplus \beta'$. On note S'_6 ainsi que S'_i dans les jeux ci-dessous l'événement $b' = \beta$, alors $\Pr[S'_6] = \Pr[b' = \beta = b \bigoplus \beta'] = \Pr[S_5]$.

Game_7 : comme ci-dessus, mais dans ce jeu, b sera le bit choisi par le challengeur pour évaluer la sécurité sémantique du schéma de chiffrement $\mathcal{E}$. D'où, la simulation qui utilise β mais plus b. Dans le jeu précédent, on voit que $\Pr[b' = \beta] = \Pr[b = \beta \bigoplus \beta']$. Alors, on retourne maintenant $b'' = \beta \bigoplus \beta'$. On note cet événement S''_7 ainsi que S''_i dans les jeux ci-dessous, on a $\Pr[S''_7] = \Pr[b'' = b] = \Pr[S'_6]$.

Game_8 : On ne modifie pas mais on récrit le jeu précédent. K_{ab}, $K'_{ab} \xleftarrow{R} \mathcal{K}$, $b \xleftarrow{R} \{0,1\}$. On note $m_0 = \{N'_a, N'_b, K_{ab}\}_{K_{as}}$, et $m_1 = \{N'_a, N'_b, K'_{ab}\}$. On reçoit dans les messages 3 et 4 le chiffrement d'un des deux couples (m_0, m_0) et (m_1, m_1) : $c = (\mathcal{E}_{as}(m_b), \mathcal{E}_{bs}(m_b))$ et on évalue b'', $\Pr[S''_8] = \Pr[S''_7]$.

Game_9 : On remplace les deux couples (m_0, m_0) et (m_1, m_1) par (m_1, m_0) et (m_1, m_1). La différence est l'avantage de distinguer ($\mathcal{E}_{as}(m_0)$, $\mathcal{E}_{bs}(m_0)$) et ($\mathcal{E}_{as}(m_1)$, $\mathcal{E}_{bs}(m_0)$). De même que dans le jeu Game_7 dans la preuve du théorème 1, on a : $|\Pr[\text{S}''_9] - \Pr[\text{S}''_8]| \leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$.

Game_{11} : On remplace les deux couples (m_1, m_0) et (m_1, m_1) par (m_1, m_1) et (m_1, m_1). La différence est l'avantage de distinguer ($\mathcal{E}_{as}(m_1)$, $\mathcal{E}_{bs}(m'_0)$) et ($\mathcal{E}_{as}(m_1)$, $\mathcal{E}_{bs}(m'_1)$). De même que dans le jeu Game_8 du théorème 1, on a : $|\Pr[\text{S}''_{10}] - \Pr[\text{S}''_9]| \leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t)$.

On remarque que dans ce dernier jeu Game_{10}, on doit distinguer ($\mathcal{E}_{as}(m_1)$, $\mathcal{E}_{bs}(m_1)$) et ($\mathcal{E}_{as}(m_1)$, $\mathcal{E}_{bs}(m_1)$) qui sont identiques, donc b est indépendant de la vue de l'attaquant, ainsi $\Pr[\text{S}''_{10}] = \frac{1}{2}$.

L'inégalité triangulaire nous donne :

$$\begin{aligned} \frac{\epsilon}{2N} &= \frac{\epsilon/N + 1}{2} - \frac{1}{2} \\ &= |\Pr[\text{S}_1] - \Pr[\text{S}''_{11}]| \\ &\leq 6\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa/cca}}(t). \end{aligned}$$

□

Sous l'hypothèse de la sécurité sémantique du schéma de chiffrement, on déduit la sécurité sémantique de la clé K_{ab}. pour l'authentification, une preuve plus délicat serait necessaire.

1.4.4 Une comparaison des deux méthodes d'analyse

La logique BAN nous donne une méthode, pour analyser des protocoles, que l'on peut utiliser pour construire des preuves automatiques. Un peu plus précisément, à partir des formules qui décrivent l'état initial du système, on peut construire des algorithmes qui choisissent les règles de déduction pour déduire des nouvelles formules jusqu'à ce que les conclusions soient obtenues. Une autre avantage de cette méthode induite de la déduction est trouver des redondances ou détecter des bogues dans le protocole. Si dans le processus de déduction, pour obtenir les conclusions, il n'est pas necéssaire d'utiliser toutes les informations données dans le protocole, alors on peut trouver des redondances dans le protocole original (c'est le cas, où les auteurs [6] montrent que N_a dans le protocole Otway-Rees ci-dessous peut être éliminée). Dans un autre cas, si on ne peut pas déduire de nouvelles formules (même si on a appliqué toutes les règles sur toutes les formules que l'on dispose) et qu'une des conclusions n'est pas encore obtenue, on peut penser que le protocole original a des bogues et on doit ajouter des messages au protocole ou bien des nouvelles hypothèses (c'est le cas, où les auteurs [6] ont montré qu'un attaquant peut rejouer dans le protocole Needham-Schroeder et ils ont réglé ce problème en faisant une hypothèse sur la fraicheur du message envoyé par le serveur).

Par contre, il y a des critiques sur cette méthode d'analyse. Boyd et Wenbo Mao ont montré [11, 10] que la phase d'idéalisation et la phase qui précise les hypothèses initiales ne reflètent pas tout à fait le protocole original, donc, il est possible qu'il y ait des protocoles démontrés sûrs par la logique BAN mais pas vraiment sûrs dans la réalité (ils ont considéré une version modifiée du protocole Otway-Rees).

La différence principale entre la méthode d'analyse par la logique BAN et celle par réduction est que les hypothèses dans la logique BAN sont idéales, c'est-à-dire, quand on fait l'hypothèse qu'un élément M (un message, une clé, ...) est sûr, un attaquant, même tout puissant, ne peut exploiter aucune information sur M. Par contre, l'analyse par réduction aborde les aspects plus effectifs. Cette méthode précise d'une part les buts que l'attaquant veut atteindre et que l'on veut éviter et d'autre part elle explicite les informations, les moyens dont l'attaquant dispose. En conséquence, l'analyse par réduction aborde des aspect plus réalistes que celle par logique BAN. Et on va voir dans le chapitre suivant, quand l'attaquant utilise des propriétés arithmétiques pour casser un système, l'analyse par la logique BAN peut ne pas les détecter.

Chapitre 2

Une extension de la logique BAN

Dans cette partie, on introduit une extension de la logique BAN pour s'approcher des notions de sécurité classiques en cryptographie. Pour cela, on ajoute à la logique les variables, les prédicats et notamment, la valeur des propositions.

2.1 Les notions

Dans les notations ci-dessous, $\mathcal{A}, \mathcal{B}$ désignent des participants spécifiques ; K_{ab}, K_{as}, K_{bs} désignent des clés spécifiques ; N_a, N_b, N_c désignent des aléas spécifiques ; f, g les fonctions spécifiques. Les symboles P, Q, R représentent les participants ; X, Y les aléas ; K les clés. Les symboles $\mathbf{P}$, $\mathbf{Q}$ représentent les prédicats ; F, G les formules. On utilise aussi le symbole $\mathbf{U}, \mathbf{V}$ pour représenter une instance quelconque qui peut être un ensemble de prédicats, de fonctions, etc.

- $P \xrightarrow{?} \mathcal{E}(m)$: le prédicat qui signifie que P interroge l'oracle $\mathcal{E}$ sur m.
- $P \rightarrow c$: P sait calculer la valeur c où c peut être une constance, ou la valeur d'une fonction. Exemple : $P \rightarrow \mathcal{E}_K(m)$: un nouveau prédicat qui signifie que P peut donner une valeur qui est égale à la valeur chiffrée du message m.
- $\mathsf{Var}(P)$: l'ensemble des variables apparaissant dans P.
  Exemple : $\mathsf{Var}(\mathcal{E}_{as}(m_1, m_2)) = \{K_{as}, m_1, m_2\}$.
- $\mathsf{Perm}(f)$: prédicat qui signifie que f est une permutation de son domaine.
- $t(\mathcal{A})$: une fonction qui limite le temps que $\mathcal{A}$ peut utiliser.
- $< \mathbf{P}, v >$: on peut rendre le prédicat $\mathbf{P}$ vrai avec la probabilité au moins v en préservant les restrictions sur les distributions des variables de $\mathbf{P}$.
  Pour le prédicat $P \rightarrow c$, on définit : $< P \rightarrow c, v > \stackrel{\mathsf{def}}{=} (\mathcal{A} \rightarrow c') \wedge < c' = c, v >$
  Si $\mathbf{P}$ s´écrit sous la forme $\mathbf{P} = \mathbf{Q} : \mathcal{A} \rightarrow c$, on définit : $< \mathbf{P}, v > \equiv \mathbf{Q} :< \mathcal{A} \rightarrow c, v >$.

- $\mathsf{val}(\mathbf{P})$: la probabilité maximale avec laquelle $\mathbf{P}$ est vrai. Elle est définie par :

$$\mathsf{val}(\mathbf{P}) = \sup_v \{v / < \mathbf{P}, v >\}$$

- $\mathbf{U}[x/x']$: on remplace la variable x dans $\mathbf{U}$ par une autre variable x' (x peut être un prédicat, une fonction, ..). Si x n'apparaît pas dans $\mathbf{U}$, on convient que $\mathbf{U}[x/x'] = \mathbf{U}$.
- $\mathbf{U}[x|d]$ où x est une variable dans $\mathbf{U}$ et d est le tuple des autres variables dans $\mathbf{U}$. Cette notation est utilisée quand on ne veut considérer que la variable x. Si x n'apparaît pas dans $\mathbf{U}$, on convient que $\mathbf{U}[x|d] = \mathbf{U}[d]$.

2.2 Les définitions

On utilise les notions ci-dessus pour définir les notions de confidentialité et d'authenticité comme l'avantage pour distinguer deux messages chiffrés, soit la notion de sécurité sémantique d'un schéma de chiffrement ainsi que la notion de sécurité sémantique de la clé d'un protocole d'échange de clé, et la probabilité de générer un nouveau couple clair-chiffré.

— $\mathsf{Nv}(\mathcal{A}, f, T)$ où $\mathcal{A}$ est un participant et f est une fonction : ce prédicat signifie que $\mathcal{A}$ peut générer une nouvelle valeur de f pendant le temps T en utilisant l'oracle f. Formellement :

$$\mathsf{Nv}(\mathcal{A}, f, T) = \mathcal{A} \triangleleft f \wedge (\exists m : \neg(\mathcal{A} \xrightarrow{?} f(m)) :< \mathcal{A} \rightarrow f(m) > \wedge(t(\mathcal{A}) \leq T).$$

Exemple : $\mathsf{Nv}(\mathcal{A}, \mathcal{E}_K, T) = \mathcal{A} \triangleleft \mathcal{E}_K \wedge (\exists m : \neg(\mathcal{A} \xrightarrow{?} \mathcal{E}_K(m)) :< \mathcal{A} \rightarrow \mathcal{E}_K(m) > \wedge(t(\mathcal{A}) \leq T)$.
Ce prédicat signifie qu'un attaquant $\mathcal{A}$ en utilisant l'oracle de chiffrement $\mathcal{E}_K$ peut générer un nouveau couple clair-chiffré de l'algorithme de chiffrement $\mathcal{E}$ sous la clé K.
A partir de $\mathsf{Nv}(\mathcal{A}, f, T)$, on définit aussi :

$$\mathsf{Nv}(f, T) = \exists \mathcal{A} : \mathsf{Nv}(\mathcal{A}, f, T).$$

Ce prédicat signifie qu'une nouvelle valeur de f peut être calculée par un participant qui a le droit d'utiliser l'oracle f en temps T.

— $\mathsf{Ind}(\mathcal{A}, f, m_0, m_1, T)$: ce prédicat signifie que l'attaquant $\mathcal{A}$ en utilisant l'oracle f peut distinguer la valeur de f sur les valeurs m_0, m_1 pendant le temps T :

$$\begin{aligned} \mathsf{Ind}(\mathcal{A}, f, m_0, m_1, T) \stackrel{\mathsf{def}}{=} \quad & (\beta \xleftarrow{R} \{0,1\}) \wedge (\mathcal{A}(\triangleleft f, f(m_\beta), m_0, m_1)) \\ \wedge \quad & \neg(\mathcal{A} \xrightarrow{?} f(m_0) \vee \mathcal{A} \xrightarrow{?} f(m_1)) \\ : \quad & < \mathcal{A} \rightarrow \beta > \wedge(t(\mathcal{A}) \leq T). \end{aligned}$$

— $\mathsf{Ind}(\mathcal{A}, f, f^{-1}, m_0, m_1, T)$: ce prédicat signifie que l'attaquant $\mathcal{A}$ en accédant aux oracles f et f^{-1} peut distinguer la valeur de f sur les valeurs m_0, m_1 pendant le temps T :

$$\begin{aligned} \mathsf{Ind}(\mathcal{A}, f, f^{-1}, m_0, m_1, T) \stackrel{\mathsf{def}}{=} \quad & (\beta \xleftarrow{R} \{0,1\}) \wedge (\mathcal{A} \triangleleft (f, f^{-1}, c = f(m_\beta), m_0, m_1)) \\ \wedge \quad & \neg(\mathcal{A} \xrightarrow{?} f(m_0) \vee \mathcal{A} \xrightarrow{?} f(m_1) \vee \mathcal{A} \xrightarrow{?} f^{-1}(c)) \\ : \quad & < \mathcal{A} \rightarrow \beta > \wedge(t(\mathcal{A}) \leq T). \end{aligned}$$

et on définit aussi :

$$\begin{aligned} \mathsf{Ind}(\mathcal{A}, f, T) &\stackrel{\mathsf{def}}{=} \exists m_0, m_1 \in \mathcal{M} : \mathsf{Ind}(\mathcal{A}, f, m_0, m_1, T) \\ \mathsf{Ind}(\mathcal{A}, f, f^{-1}, T) &\stackrel{\mathsf{def}}{=} \exists m_0, m_1 \in \mathcal{M} : \mathsf{Ind}(\mathcal{A}, f, f^{-1}, m_0, m_1, T) \\ \mathsf{Ind}(f, T) &\stackrel{\mathsf{def}}{=} \exists \mathcal{A} : \mathsf{Ind}(\mathcal{A}, f, T) \\ \mathsf{Ind}(f, f^{-1}, T) &\stackrel{\mathsf{def}}{=} \exists \mathcal{A} : \mathsf{Ind}(\mathcal{A}, f, f^{-1}, T) \end{aligned}$$

Le prédicat $\mathsf{Ind}(\mathcal{A}, f, T)$ ($\mathsf{Ind}(\mathcal{A}, f, f^{-1}, T)$ respectivement) signifie que l'attaquant $\mathcal{A}$ en utilisant l'oracle f (les oracles f et f^{-1} respectivement) peut distinguer deux valeurs quelconques de f pendant le temps T.
Le prédicat $\mathsf{Ind}(f, T)$ ($\mathsf{Ind}(f, f^{-1}, T)$ respectivement) signifie qu'il existe un attaquant qui peut distinguer deux valeurs quelconques de f en utilisant l'oracle f (les oracle f et f^{-1} respectivement) pendant le temps T.

Exemples :

$$\begin{aligned}\mathsf{Ind}(\mathcal{A}, \mathcal{E}_K, T) = \quad & \exists m_0, m_1 \in \mathcal{M} : \beta \xleftarrow{R} \{0,1\} \wedge (\mathcal{A} \triangleleft (\mathcal{E}_K, \mathcal{E}_K(m_\beta), m_0, m_1)) \\ \wedge \quad & \neg(\mathcal{A} \xrightarrow{?} \mathcal{E}_K(m_0) \vee \mathcal{A} \xrightarrow{?} \mathcal{E}_K(m_1)) \\ : \quad & < \mathcal{A} \to \beta > \wedge (t(\mathcal{A}) \leq T).\end{aligned}$$

Ce prédicat signifie qu'un attaquant à clairs choisis $\mathcal{A}$ peut casser la sécurité sémantique de l'algorithme de chiffrement $\mathcal{E}$ sous la clé K.
Pour les schémas de chiffrement, on définit aussi le prédicat $\mathsf{Ind}(\mathcal{A}, \mathcal{E}, T)$:

$$\mathsf{Ind}(\mathcal{A}, \mathcal{E}, T) \stackrel{\mathsf{def}}{=} K \xleftarrow{R} \mathcal{K} \wedge \mathsf{Ind}(\mathcal{A}, \mathcal{E}_K, T)$$

$$\begin{aligned}\mathsf{Ind}(\mathcal{A}, \mathcal{E}_K, \mathcal{D}_K, T) = \quad & \exists m_0, m_1 \in \mathcal{M} : (\beta \xleftarrow{R} \{0,1\}) \\ \wedge \quad & (\mathcal{A} \triangleleft (\mathcal{E}_K, \mathcal{D}_K, c = \mathcal{E}_K(m_\beta), m_0, m_1)) \\ \wedge \quad & \neg(\mathcal{A} \xrightarrow{?} \mathcal{E}_K(m_0) \vee \mathcal{A} \xrightarrow{?} \mathcal{E}_K(m_1) \vee \mathcal{A} \xrightarrow{?} \mathcal{D}_K(c)) \\ : \quad & < \mathcal{A} \to \beta > \wedge (t(\mathcal{A}) \leq T).\end{aligned}$$

Ce prédicat signifie qu'un attaquant à clairs et chiffrés choisis $\mathcal{A}$ peut casser la sécurité sémantique de l'algorithme de chiffrement $\mathcal{E}$ sous la clé K.
On définit aussi le prédicat $\mathsf{Ind}(\mathcal{A}, \mathcal{E}, \mathcal{D}, T)$:

$$\mathsf{Ind}(\mathcal{A}, \mathcal{E}, \mathcal{D}, T) \stackrel{\mathsf{def}}{=} K \xleftarrow{R} \mathcal{K} \wedge \mathsf{Ind}(\mathcal{A}, \mathcal{E}_K, \mathcal{D}_K, T)$$

— $\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, m_0, m_1, T)$: ce prédicat caractérise l'avantage d'un participant $\mathcal{A}$ pour distinguer deux valeurs de la fonction f sur deux valeurs m_0, m_1 pendant le temps T en accédant à l'oracle f. Formellement :

$$\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, m_0, m_1, T) \stackrel{\mathsf{def}}{=} v \xleftarrow{R} [0,1] \wedge (\mathsf{val}(\mathsf{Ind}(\mathcal{A}, f, m_0, m_1, T)) \geq (1+v)/2)$$

— $\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, f^{-1}, m_0, m_1, T)$: ce prédicat caractérise l'avantage d'un participant $\mathcal{A}$ pour distinguer deux valeurs de la fonction f sur deux valeurs m_0, m_1 pendant le temps T en accédant aux oracles f et f^{-1}. Formellement :

$$\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, f^{-1}, m_0, m_1, T) \stackrel{\mathsf{def}}{=} v \xleftarrow{R} [0,1] \wedge (\mathsf{val}(\mathsf{Ind}(\mathcal{A}, f, f^{-1}, m_0, m_1, T)) \geq (1+v)/2)$$

Remarque :
La probabilité de la vérité de $\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, m_0, m_1, T)$ est la probabilité que l'on obtienne une valeur v satisfaisant $\mathsf{val}(\mathsf{Ind}(\mathcal{A}, f, m_0, m_1)) \geq (1+v)/2$, on en déduit :

$$\mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, m_0, m_1, T)) = 2 \cdot \mathsf{val}(\mathsf{Ind}(\mathcal{A}, f, m_0, m_1, T)) - 1.$$

De même :

$$\mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, f^{-1}, m_0, m_1, T)) = 2 \cdot \mathsf{val}(\mathsf{Ind}(\mathcal{A}, f, f^{-1}, m_0, m_1, T)) - 1.$$

— $\mathsf{Unif}(f, x)$: La distribution de la valeur de la fonction f sur la variable x est uniforme, formellement :

$$\mathsf{Unif}(f, x) = \forall d, \forall y_1, y_2 \in \mathsf{Im}(f) : |\{x/f[x|d] = y_1\}| = |\{x/f[x|d] = y_2\}|$$

— $\mathsf{Unif}(f)$: La distribution de la valeur de la fonction f est uniforme pour toute variable dans f.

2.3 Les postulats

Dans ces règles ci-dessous, on utilise la notation $\delta(t)$ pour indiquer la différence du temps limite du participant de la conclusion par rapport celui du participant de l'hypothèse.

· **R1 :** $\dfrac{\forall \mathbf{U}_0 \neq \mathbf{U}_1 : (\beta \xleftarrow{R} \{0,1\}) \wedge \mathcal{A} \triangleleft (\mathbf{U}_0, \mathbf{U}_\beta)}{< \mathcal{A} \rightarrow \beta, 1 > \wedge (t(\mathcal{A}) = 0)}$

Cette règle est évidente, si $\mathcal{A}$ voit $\mathbf{U}_0$, il peut distinguer $\mathbf{U}_0$ des autres instances.

· **R2 :** $\dfrac{\beta \xleftarrow{R} \{0,1\}}{\forall \mathcal{A} :< \mathcal{A} \rightarrow \beta, 1/2 > \wedge (t(\mathcal{A}) = 0)}$

Cette règle est évidente, on peut toujours avoir une chance sur 2 pour deviner β.

· **R3 :** $\dfrac{m_0, m_1 \xleftarrow{R} \mathcal{M}}{\forall \mathcal{A} : \mathsf{val}(\mathcal{A}, (m_0 = m_1)) = 1/|\mathcal{M}| \wedge (t(\mathcal{A}) = 0)}$

Cette règle est évidente, si m_0, m_1 sont choisis aléatoirement, la probabilité qu'ils sont égaux est $1/|\mathcal{M}|$.

· **R4 :** $\dfrac{\begin{array}{c} Cond(X) :< \mathcal{A} \rightarrow \mathbf{P}(u), v_1 > \wedge (t(\mathcal{A}) \leq T_1) \\ \neg Cond(X) :< \mathcal{A} \rightarrow \mathbf{P}(u), v_2 > \wedge (t(\mathcal{A}) \leq T_2 \\ p = \mathsf{val}(Cond(X)) \end{array}}{< \mathcal{A} \rightarrow \mathbf{P}(u), p \cdot v_1 + (1-p) \cdot v_2 > \wedge (t(\mathcal{A}) \leq \max(T_1, T_2))}$

où $Cond(X)$ est un prédicat qui exprime une condition sur l'ensemble X des variables considérées, $\max(T_1, T_2)$ est la valeur la plus grande parmi T_1, T_2.

· **R5 :** $\dfrac{(\mathcal{A} \triangleleft (\mathbf{U}, \mathbf{V}) :< \mathcal{A} \rightarrow f(.), v >) \wedge (\mathsf{Var}(\mathbf{V}) \cap (\mathsf{Var}(\mathbf{U}) \cup \mathsf{Var}(f(.)) = \emptyset)}{(\mathcal{A} \triangleleft \mathbf{U} :< \mathcal{A} \rightarrow f(.), v >)}$, avec $\delta(t) = 0$.

Cette règle est évidente : si $\mathbf{V}$ est indépendant de la cible et des variables restantes, alors $\mathcal{A}$ peut jeter $\mathbf{V}$.

· **R6 :** $\dfrac{(\mathsf{val}(\mathsf{Ind}(\mathcal{A}, f, T)) = 1/2) \wedge (\exists X_1 : (\mathcal{A} \triangleleft f(X_1) :< \mathcal{A} \rightarrow x, v >))}{(\forall X_2 : \mathcal{A} \triangleleft f(X_2)) :< \mathcal{A} \rightarrow x, v >}$, avec $\delta(t) = 0$.

Quand on ne peut pas distinguer la fonction f sur deux valeurs, on peut remplacer un élément dans le domaine de f par un autre.

· **R7 :** $\dfrac{m \xleftarrow{R} \mathcal{M} \wedge \mathsf{Unif}(f, m)}{u \xleftarrow{R} \mathsf{Im}(f), \mathsf{Prot}[f/u]}$, avec $\delta(t) = 0$

Si f est une fonction qui est uniformément distribuée pour $m \in \mathcal{M}$, et si m est généré de manière aléatoire, on peut remplacer la fonction f partout dans le protocole où apparait la variable m par une valeur aléatoire u.

Exemple : Si on veut remplacer $\mathcal{E}_K(m)$ par u et dans le protocole, s'il y a un message contenant m, par exemple $g(y, m)$, on doit remplacer $g(y, m)$ par $g(y, D_K(u))$.

· **R8 :** $\dfrac{(w \xleftarrow{R} \mathcal{W}) \wedge \mathsf{Unif}(f,w) \wedge (w \in \mathsf{Var}(f))(\mathcal{A} \triangleleft U) \wedge (w \notin (\mathsf{Var}(U))}{\mathsf{val}(\mathcal{A}, f) = 1/|f(\mathcal{W})| \wedge (t(\mathcal{A}) = 0)}$

La valeur d'une fonction f est une valeur aléatoire si la fonction f contient une variable aléatoire indépendante et f est uniformément distribuée pour cette variable.

· **R9 :** $\dfrac{\mathcal{A} \triangleleft f}{\forall x \in \mathsf{Dom}(f) : (\mathcal{A} \to f(x)) \wedge (t(\mathcal{A}) = T_f)}$

où $\mathsf{Dom}(f)$ est le domaine de la fontion f et T_f est le temps pour calculer la fonction f. Cette règle est évidente mais elle donne la capacité de calcul à un participant.

2.4 Les propositions

Pour illustrer la déduction dans cette logique BAN étendue, on présente et prouve la proposition suivante par réduction et puis par la logique BAN étendue.

Proposition 1 *La probabilité de produire un nouveau couple clair-chiffré dans un schéma de chiffrement symétrique déterministe est négligeable si l'avantage contre la sécurité sémantique du protocole selon des attaques à clairs choisis est négligeable, plus précisément :*

$$\mathsf{Succ}_{\mathcal{S}}^{\mathsf{ef-cma}}(t) = \Pr[c = \mathcal{E}_k(m) | \mathcal{A}^{\mathcal{E}} \to (m,c)] \leq 2\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa}}(t) + 1/2^k$$

Preuve.

Game$_0$: on utilise $\mathcal{E}_K$, la clé K est choisie aléatoirement. L'attaquant à clairs choisis produit un nouveau couple (m,c). On note S_0 l'événement c $= \mathcal{E}_K(m)$ ainsi que S_i dans les jeux Game$_i$ ci-dessous : $\Pr[S_0] = \epsilon$.

Game$_1$: On considère le nouveau couple (m,c) retourné par l'attaquant. On définit $m_0 = m$ et on choisit aléatoirement m_1. Après avoir reçu le chiffré $c' = \mathcal{E}_K(m_b)$ où b est choisi aléatoirement. Si $c = c'$, on retourne $b' = 0$, sinon on retourne b' aléatoirement. On note S_1' l'événement $b' = b$.
On a :

$$\begin{aligned}
2\Pr[b' = b] - 1 &= \Pr[b' = 0 | b = 0] - \Pr[b' = 0 | b = 1] \\
&= \Pr[c' = c | b = 0] + 1/2(\Pr[c \neq c' | b = 0]) \\
&\quad - \Pr[c' = c | b = 1] - 1/2(\Pr[c' \neq c | b = 1]) \\
&= \Pr[c' = c | b = 0] + 1/2(1 - \Pr[c = c' | b = 0]) \\
&\quad - \Pr[c' = c | b = 1] - 1/2(1 - \Pr[c' = c | b = 1]) \\
&= 1/2\Pr[c' = c | b = 0] - 1/2\Pr[c' = c | b = 1] \\
&= \frac{\epsilon}{2} - \frac{1}{2}(1/2^k).
\end{aligned}$$

(on a utilisé l'hypothèse du déterminisme du chiffrement pour évaluer $\Pr[c' = c | b = 0] = \epsilon$). Alors,

$$\Pr[S_2] = \Pr[b' = b] = \frac{1}{2} + \frac{\epsilon}{4} - \frac{1}{4}(1/2^k).$$

On remarque que le jeu Game$_2$ nous donne un avantage pour casser la sécurité sémantique de chiffrement. Alors : $\frac{\epsilon}{2} - \frac{1}{2}(1/2^k) \leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa}}(t)$, soit $\epsilon \leq 2\mathsf{Adv}_{\mathcal{S}}^{\mathsf{ind-cpa}}(t) + 1/2^k$

□

Proposition 2 $\dfrac{< \mathcal{A} \to \mathsf{Nv}(\mathcal{E}_K, T), v >}{< \mathcal{A} \to \mathsf{Ind}(\mathcal{E}_K, T), v_\beta >}$

où

$$v_\beta = \frac{1}{2} + \frac{v}{2}(1 - \frac{1}{|\mathcal{M}|}).$$

Preuve. On récrit $< \mathcal{A} \to \mathsf{Nv}(\mathcal{E}_K, T), v >$:

$$\mathcal{A} \triangleleft \mathcal{E}_K \wedge (\exists m^* : \neg(\mathcal{A} \stackrel{?}{\to} \mathcal{E}_K(m^*)) :< \mathcal{A} \to c > \wedge < c = \mathcal{E}_K(m^*), v > \wedge(t(\mathcal{A}) \leq T).$$

On considère deux cas :

1. Soit $m \neq m^*$: On prend $\beta \stackrel{R}{\leftarrow} \{0,1\}$ et on suppose que $\mathcal{A} \triangleleft \mathcal{E}_K(m_\beta)$

   $$(\mathcal{A} \triangleleft (m_0 = m^*, m_1 = m, c, \mathcal{E}_K(m_\beta))) \wedge (\mathcal{E}_K(m_0) \neq \mathcal{E}_K(m_\beta))$$

   On a :
   — $c = \mathcal{E}_K(m^*)$. D'après le postulat R1 en remarquant que $\mathcal{E}_K$ est déterministe :

   $$(\mathcal{A} \to \beta') \wedge < \beta' = \beta, 1 > \wedge(t(\mathcal{A}) \leq T).$$

   — $c \neq \mathcal{E}_K(m^*)$ D'après le postulat R2 :

   $$(\mathcal{A} \to \beta') \wedge < \beta' = \beta, 1/2 > \wedge(t(\mathcal{A}) = 0).$$

   Puisque $< c = \mathcal{E}_K(m^*), v >$. D'après le postulat R4, on a :

   $$(\mathcal{A} \to \beta') \wedge < \beta' = \beta, v \cdot 1 + (1 - v) \cdot 1/2 = (1 + v)/2 > \wedge(t(\mathcal{A}) \leq T).$$

   En résumé, dans ce cas :

   $$(\mathcal{A} \to \beta') \wedge < \beta' = \beta, v \cdot 1 + (1 - v) \cdot 1/2 = (1 + v)/2 > .$$

2. Soit $m = m^*$. D'après le postulat R2 :

   $$m = m^* : (\mathcal{A} \to \beta') \wedge < \beta' = \beta, 1/2 > \wedge(t(\mathcal{A}) = 0).$$

Alors, quand on prend $m \stackrel{R}{\leftarrow} \mathcal{M}$, qui entraîne : $< m = m^*, 1/|\mathcal{M}| >$, on combine les deux cas, d'après le postulat R4 :

$$(\mathcal{A} \to \beta') \wedge < \beta' = \beta, v_\beta = (1 - 1/|\mathcal{M}|) \cdot (1 + v)/2 + (1/|\mathcal{M}|) \cdot 1/2 > \wedge(t(\mathcal{A}) \leq T).$$

Enfin :

$$\frac{\mathcal{A} \triangleleft \mathcal{E}_K \wedge (\exists m_0 = m^*, m_1 \stackrel{R}{\leftarrow} \mathcal{M}) \wedge (\beta \stackrel{R}{\leftarrow} \{0,1\}) \wedge (\mathcal{A} \triangleleft \mathcal{E}_K(m_\beta)) \wedge (t(\mathcal{A}) \leq T)}{(\mathcal{A} \to \beta') \wedge < \beta' = \beta, v_\beta = 1/2 + v/2 - (v/2|\mathcal{M}|) > \wedge(t(\mathcal{A}) \leq T)}$$

ou bien $< \mathcal{A} \to \mathsf{Ind}(\mathcal{E}_K, T), v_\beta >$. □

Maitenant, on introduit une proposition qui est très utilisée par la suite.

Proposition 3 *Si on a :*

$$v_1 = \mathsf{val}(\mathcal{A} \triangleleft (f(X_1)) : (\mathcal{A} \to x) \wedge (t(\mathcal{A}) \leq T_1)).$$
$$v_2 = \mathsf{val}(\mathcal{A} \triangleleft (f(X_2)) : (\mathcal{A} \to x) \wedge (t(\mathcal{A}) \leq T_2)).$$

alors :

$$|v_1 - v_2| \leq \mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T'))$$

où $T' = \max(T_1, T_2)$

Preuve. Supposons que l'on ait :

$$\mathcal{A} \triangleleft (f(X_1)) :< \mathcal{A} \to x, v_1 > \wedge (t(\mathcal{A}) \leq T_1).$$

— $\neg(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2))$, ou bien $\mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T)) = 0$.
D'après le postulat R6, on a :

$$\mathcal{A} \triangleleft (f(X_2)) :< \mathcal{A} \to x, v_1 > \wedge (t(\mathcal{A}) \leq T_1).$$

— $(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T))$, ou bien $\mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T)) = 1$.
On a toujours :

$$\mathcal{A} \triangleleft (f(X_2)) :< \mathcal{A} \to x, u \geq 0 > \wedge (t(\mathcal{A}) \leq T_1)$$

Alors, d'après le postulat R4 :

$$\mathcal{A} \triangleleft (f(X_2)) :< \mathcal{A} \to x, v_2' > \wedge (t(\mathcal{A}) \leq T_1),$$

où $v_2' = v_1(1 - \mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T))) + u\mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T))$.
Puisque $u \geq 0$, on déduit :

$$\begin{array}{rcl} v_2' & \geq & (v_1(1 - \mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T_1))) \\ & \geq & v_1 - \mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T_1))). \end{array}$$

D'autre part : $v_2 \geq v_2'$, alors : $v_1 - v_2 \leq v_1 - v_2' \leq \mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T_1))$.
On montre de même : $v_2 - v_1 \leq \mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, f, X_1, X_2, T_2))$.
D'où le résultat. □

2.5 Analyse de la sécurité de la clé du protocole d'Otway-Rees

On définit $\mathsf{Ind\_key}(\mathsf{Prot}, T)$ le prédicat qui signifie la probabilité de distinguer deux interprétations du protocole d´échange de clé (que l'on note Prot avec deux clés différentes).

Définition 5

$$\begin{array}{rcll} \mathsf{Ind\_key}(\mathcal{A}, \mathsf{Prot}, T) & \stackrel{\mathsf{def}}{=} & & (K_0, K_1 \stackrel{R}{\leftarrow} \mathcal{K}) \wedge \beta \stackrel{R}{\leftarrow} \{0,1\} \\ & & \wedge & \mathcal{A} \triangleleft (\mathsf{Prot}[K_{ab}/K_0], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\ & & \wedge & (b = (K_\beta = K_0)) : \mathcal{A} \to b \\ \mathsf{Ind\_key}(\mathsf{Prot}, T) & \stackrel{\mathsf{def}}{=} & \exists \mathcal{A} & : \mathsf{Ind\_key}(\mathcal{A}, \mathsf{Prot}, T) \end{array}$$

Théorème 3 *La clé échangée dans le protocole Otway-Rees* ProtOR *est sémantiquement sûre sous l'hypothèse que le schéma de chiffrement* $\mathcal{S}$ *utilisé dans le protocole soit sémantiquement sûr selon des attaques à clairs et chiffrés choisis, plus précisément :*

$$\mathsf{val}(\mathsf{Ind\_key}(\mathsf{ProtOR}), T) \leq 1/2 + \mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T))$$

Preuve. On récrit le protocole d'Otway-Ree.

$$\begin{array}{lrcl}
\textit{Message 1} & A \to B & : & M, A, B, \{N_a, M, A, B\}_{K_{as}} \\
\textit{Message 2} & B \to S & : & M, A, B, \{N_a, M, A, B\}_{K_{as}}, \{N_b, M, A, B\}_{K_{bs}} \\
\textit{Message 3} & S \to B & : & M, \{N_a, K_{ab}\}_{K_{as}}, \{N_b, K_{ab}\}_{K_{bs}} \\
\textit{Message 4} & B \to A & : & M, \{N_a, K_{ab}\}_{K_{as}}
\end{array}$$

On note $v = \mathsf{val}(\mathsf{Ind\_key}(\mathsf{ProtOR}, T))$.
On considère $\mathcal{A}$ tel que : $\mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtOR}, T)) = v$.
On a :

$$\begin{array}{rcl}
(K_0, K_1 \xleftarrow{R} \mathcal{K}) \wedge \beta \xleftarrow{R} \{0,1\} & \wedge & \mathcal{A} \triangleleft (\mathsf{ProtOR}[K_{ab}/K_0], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\
& \wedge & (b = (K_\beta = K_0)) : (\mathcal{A} \to b') \wedge < b' = b, v > \wedge (t(\mathcal{A}) \leq T)
\end{array}$$

Puisque les rôles de K_0 et K_1 sont identiques, on a aussi :

$$\begin{array}{rcl}
(K_0, K_1 \xleftarrow{R} \mathcal{K}) \wedge \beta \xleftarrow{R} \{0,1\} & \wedge & \mathcal{A} \triangleleft (\mathsf{ProtOR}[K_{ab}/K_1], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\
& \wedge & (b = (K_\beta = K_1)) : (\mathcal{A} \to b') \wedge < b' = b, v > \wedge (t(\mathcal{A}) \leq T)
\end{array}$$

On considère une variable w dont la valeur est soit 0 soit 1.

— $w = 0$: on peut remplacer K_0 par K_w et on a :

$$\begin{array}{rcl}
(w = 0) & \wedge & \mathcal{A} \triangleleft (\mathsf{ProtOR}[K_{ab}/K_0], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\
& \wedge & (b = (K_\beta = K_w)) : (\mathcal{A} \to b') \wedge < b' = b, v > \wedge (t(\mathcal{A}) \leq T)
\end{array}$$

ou bien :

$$\begin{array}{rcl}
(w = 0) & \wedge & \mathcal{A} \triangleleft (\mathcal{E}_{as}[K_0], \mathcal{E}_{bs}[K_0], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\
& \wedge & (b = (K_\beta = K_w)) : (\mathcal{A} \to b') \wedge < b' = b, v > \wedge (t(\mathcal{A}) \leq T)
\end{array}$$

D'après la proposition 3, on déduit :

$$\begin{array}{rcl}
(w = 0) & \wedge & \mathcal{A} \triangleleft (\mathcal{E}_{as}[K_0/K_1], \mathcal{E}_{bs}[K_0], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\
& \wedge & (b = (K_\beta = K_w)) : (\mathcal{A} \to b') \wedge < b' = b, v_1 > \wedge (t(\mathcal{A}) \leq T)
\end{array}$$

avec

$$|v_1 - v| \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}_{as}, \mathcal{D}_{as}, T) \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T) \tag{2.1}$$

D'après la proposition **??**, on déduit :

$$\begin{array}{rcl}
(w = 0) & \wedge & \mathcal{A} \triangleleft (\mathcal{E}_{as}[K_0/K_1], \mathcal{E}_{bs}[K_0/K_1], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\
& \wedge & (b = (K_\beta = K_w)) : (\mathcal{A} \to b') \wedge < b' = b, v_2 > \wedge (t(\mathcal{A}) \leq T)
\end{array}$$

avec

$$|v_2 - v_1| \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{A}, \mathcal{E}_{bs}, \mathcal{D}_{as}, T) \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T) \tag{2.2}$$

— $w = 1$ On peut remplacer K_1 par K_w et on a :

$$\begin{array}{rcl}
(w = 1) & \wedge & \mathcal{A} \triangleleft (\mathcal{E}_{as}[K_1], \mathcal{E}_{bs}[K_1]), \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta)) \\
& \wedge & (b = (K_\beta = K_w)) : (\mathcal{A} \to b') \wedge < b' = b, v > \wedge (t(\mathcal{A}) \leq T)
\end{array}$$

Quand on prend $w \stackrel{R}{\leftarrow} \{0,1\}$, ou bien : $< w = 0, 1/2 >$, d'après le postulat R4 :

$$\begin{aligned} \mathcal{A} &\triangleleft (\mathcal{E}_{as}[K_1], \mathcal{E}_{bs}[K_1], \mathcal{D}_{as}, \mathcal{D}_{bs}, \mathcal{E}_{as}, \mathcal{E}_{bs}, K_\beta) \\ &\wedge (b = (K_\beta = K_w)) : (\mathcal{A} \rightarrow b') \wedge < b' = b, (v + v_2)/2 > \wedge (t(\mathcal{A}) \leq T) \end{aligned}$$

ou bien :

$$\begin{aligned} w \stackrel{R}{\leftarrow} \{0,1\} \quad \wedge \quad & \mathcal{A} \triangleleft (\mathcal{E}_{as}[K_1], \mathcal{E}_{bs}[K_1]), \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta)) \\ : \quad & (\mathcal{A} \rightarrow b') \wedge < b' = (K_\beta = K_w), (v + v_2)/2 > \wedge (t(\mathcal{A}) \leq T) \end{aligned}$$

On voit que w est choisie aléatoirement et elle est indépendante de ce que $\mathcal{A}$ voit.
D'après le postulat R8, en remarquant que $|\mathsf{Im}(b' = (K_\beta = K_w))| = 2$, on déduit : $\mathsf{val}(b' = (K_\beta = K_w)) = 1/2$, donc $\mathcal{A} \rightarrow < b', u \leq 1/2 >$, et alors

$$(v + v_2)/2 \leq 1/2 \tag{2.3}$$

Des équations (2.1) et (2.2) on a :

$$|v_2 - v| \leq 2 \cdot \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T).$$

ce qui entraîne : $v - 2 \cdot \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T) \leq v_2$. Alors : $(v + v_2)/2 \geq v - \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T)$
De (2.3), on a :

$$v \leq 1/2 + \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T).$$

□

2.6 Analyse de la sécurité de la clé du protocole d'Otway-Rees étendu

Théorème 4 *La clé échangée dans le protocole d'Otway-Rees étendu* ProtExOR *est sémantiquement sûre sous l'hypothèse que le schéma de chiffrement $\mathcal{S}$ utilisé dans protocole soit déteministe et sémantiquement sûr selon des attaques à clairs et chiffrés choisis, plus précisément :*

$$\mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}), T) =\leq 1/2 + 5\mathsf{val}(\mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T))$$

Preuve. On récrit le protocole d'extension d'Otway-Ree.

$$\begin{aligned} &\textit{Message 1} \quad A \rightarrow B \;:\; M, A, B, \{N_a, M, A, B\}_{K_{as}} \\ &\textit{Message 2} \quad B \rightarrow S \;:\; M, A, B, \{N_a, M, A, B\}_{K_{as}}, \{N_b, M, A, B\}_{K_{bs}} \\ &\textit{Message 3} \quad S \rightarrow B \;:\; M, \{N_a, N_b, K_{ab}\}_{K_{as}}, \{N_a, N_b, K_{ab}\}_{K_{bs}} \\ &\textit{Message 4} \quad B \rightarrow A \;:\; M, \{N_a, N_b, K_{ab}\}_{K_{as}}, \{N_a\}_{K_{ab}} \\ &\textit{Message 5} \quad A \rightarrow B \;:\; \{N_b\}_{K_{ab}} \end{aligned}$$

On note $v = \mathsf{val}(\mathsf{Ind\_key}(\mathsf{ProtExOR}, T))$.
On considère $\mathcal{A}$ tel que : $\mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}, T)) = v$.
On prend N'_a, N'_b aléatoirement et on note :

$$\begin{aligned} m_0 = \{N_a, M, A, B\}, m_1 = \{N_b, M, A, B\}, m_2 = m_3 = \{N_a, N_b, K_{ab}\}, \\ m'_0 = \{N'_a, M, A, B\}, m'_1 = \{N'_b, M, A, B\}, m'_2 = m'_3 = \{N'_a, N'_b, K_{ab}\} \end{aligned}$$

On définit $v_1 = \mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}_1 = \mathsf{ProtExOR}[m_0/m'_0], T))$
D'après la proposition 2, on a : $|v_1 - v| \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}_{as}, T) \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, T)$. De même :

— $v_2 = \mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}_2 = \mathsf{ProtExOR}_1[m_1/m_1'], T))$
avec $|v_2 - v_1| \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T)$

— $v_3 = \mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}_3 = \mathsf{ProtExOR}_2[\mathcal{E}_{as}(m_2/m_2')], T))$
avec $|v_3 - v_2| \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T)$

— $v_4 = \mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}_4 = \mathsf{ProtExOR}_3[\mathcal{E}_{bs}(m_3/m_3')], T))$
avec $|v_4 - v_3| \leq \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T)$

Alors :

$$|v_4 - v| \leq 4\mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T) \tag{2.4}$$

Maintenant, on prend $c_a, c_b \xleftarrow{R} \mathcal{M}$. Puisque $\mathcal{E}_K$ est une permutation pour toute clé K,et donc $\mathsf{Unif}(\mathcal{E}, K_{ab})$, d'après le postulat R7 :

$$\mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}_5 = \mathsf{ProtExOR}_4[\mathcal{E}_{ab}(N_a)/c_a], T) = v_4.$$

De même :

$$\mathsf{val}(\mathsf{Ind\_key}(\mathcal{A}, \mathsf{ProtExOR}_6 = \mathsf{ProtExOR}_5[\mathcal{E}_{ab}(N_b)/c_b], T) = v_4.$$

Maintenant, on peut écrire :

$$\begin{array}{lll} (K_0, K_1 \xleftarrow{R} \mathcal{K}) \wedge \beta \xleftarrow{R} \{0,1\} & \wedge & \mathcal{A} \triangleleft (\mathsf{ProtExOR}_6[K_{ab}/K_0], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\ & \wedge & (b = (K_\beta = K_0)) : (\mathcal{A} \to b') \wedge < b' = b, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

par :

$$\begin{array}{lll} (w = 0) & \wedge & \mathcal{A} \triangleleft (\mathcal{E}_{as}[K_0], \mathcal{E}_{bs}[K_0], \mathcal{E}_{as}, \mathcal{E}_{bs}, \mathcal{D}_{as}, \mathcal{D}_{bs}, K_\beta) \\ & \wedge & (b = (K_\beta = K_w)) : (\mathcal{A} \to b') \wedge < b' = b, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

puisque K_{ab} n'apparaît que dans les messages chiffrés par $\mathcal{E}_{as}$ et $\mathcal{E}_{bs}$.
Alors, maintenant, on montre exactement comme dans le théorème 3 que :

$$v_4 \leq 1/2 + \mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T).$$

En utilisant (2.4), on a :

$$v \leq 1/2 + 5\mathsf{Adv}^{\mathsf{ind}}(\mathcal{E}, \mathcal{D}, T).$$

□

2.7 Une comparaison entre des méthodes d'analyse sur un protocole

Dans cette partie, on considère une modification du protocole d'Otway-Rees étendu et on montre que ce protocole garantit, selon le modèle, l'authentification mutuelle ou non. Pour cela, on modifie l'étape de confirmation de connaissance de la clé (messages 4 et 5).

2.7.1 Protocole d'Otway-Rees modifié

$$
\begin{array}{lrcl}
\textit{Message 1} & A \to B & : & M, A, B, \{N_a, M, A, B\}_{K_{as}} \\
\textit{Message 2} & B \to S & : & M, A, B, \{N_a, M, A, B\}_{K_{as}}, \{N_b, M, A, B\}_{K_{bs}} \\
\textit{Message 3} & S \to B & : & M, \{N_a, K_{ab}\}_{K_{as}}, \{N_a, N_b, K_{ab}\}_{K_{bs}} \\
\textit{Message 4} & B \to A & : & M, \{N_a, N_b, K_{ab}\}_{K_{as}}, \{N_a, N_b\}_{K_{ab}} \\
\textit{Message 5} & A \to B & : & \{N_b\}_{K_{ab}}
\end{array}
$$

2.7.2 Analyse par réduction

Proposition 4 *La clé échangée dans le protocole d'Otway-Rees modifié n'est pas sémantiquement sûre même sous l'hypothèse que le schéma de chiffrement utilisé dans protocole soit sémantiquement sûr selon des attaques à clairs et chiffrés choisis :*

Preuve. Le protocole est réalisé avec les clés K_{ab}^i ($i = \overline{1, N}$) choisies aléatoirement par S pour N sessions éventuellement simultanées. On note $\text{Prot}(K_{ab}^i)$ le protocole correspondant à la session i.
On montre que l'attaquant peut casser la clé d'une session t quelconque. Pour cela, l'attaquant pose la $\mathsf{Test-query}$ à un oracle (soit A, soit B) et il reçoit K_β (β choisi aléatoirement) où $K_0 = K_{ab}^t$ et $K_1 = K'_{ab}$ choisi aléatoirement) et on montre qu'il peut retourner β' qui est égal à β avec une probabilité $(\epsilon+1)/2$ où ϵ est non-négligible. En effet, en utilisant la clé K_β, l'attaquant :

- déchiffre le message $\{N_a, N_b\}_{K_{ab}}$ dans le message 4, il obtient m_4.
- déchiffre le message $\{N_b\}_{K_{ab}}$ dans le message 5, il obtient m_5.
- si m_5 est un suffixe de m_4, alors il croit que K_β est justement la clé K_{ab} et il retourne $\beta' = 0$, sinon, il retourne $\beta' = 1$.

On peut voir que la probabilité pour que l'attaquant retourne une fausse réponse est la probabilité, pour une clé K'_{ab} choisie aléatoirement, que le déchiffré par K'_{ab} de $\{N_b\}_{K_{ab}}$ soit un suffixe du déchiffré de $\{N_a, N_b\}_{K_{ab}}$. Il est évident que cette probabilité est négligeable, d'où le résultat. □

2.7.3 Analyse par la logique BAN

Proposition 5 *La clé échangée dans le protocole d'Otway-Rees modifié est sûre.*

Preuve.

- Avec le même argument que dans la preuve de la section 1.4.2, on déduit le protocole

idéalisé :

$$
\begin{array}{lll}
\textit{Message 1} & A \to B & : \{N_a, N_c\}_{K_{as}} \\
\textit{Message 2} & B \to S & : \{N_a, N_c\}_{K_{as}}, \{N_b, N_c\}_{K_{bs}} \\
\textit{Message 3} & S \to B & : \{N_a, N_b, (A \overset{K_{ab}}{\leftrightarrow} B), B \succ N_c\}_{K_{as}}, \\
 & & \quad \{N_a, (A \overset{K_{ab}}{\leftrightarrow} B), A \succ N_c\}_{K_{bs}}, \\
\textit{Message 4} & B \to A & : \{N_a, N_b, (A \overset{K_{ab}}{\leftrightarrow} B), B \succ N_c\}_{K_{as}}, \\
 & & \quad \{N_a, N_b, (A \overset{K_{ab}}{\leftrightarrow} B)\}_{K_{ab}} \\
\textit{Message 5} & A \to B & : \{N_b, (A \overset{K_{ab}}{\leftrightarrow} B)\}_{K_{ab}}
\end{array}
$$

— On obtient enfin par le même raisonnement que dans la preuve de la section 1.4.2 :

$$A \models B \models A \overset{K_{ab}}{\leftrightarrow} B$$

$$B \models A \models A \overset{K_{ab}}{\leftrightarrow} B$$

□

2.7.4 Analyse par la logique BAN étendue

Proposition 6 *La clé échangée dans le protocole d'Otway-Rees modifié n'est pas sémantiquement sûre même sous l'hypothèse que le schéma de chiffrement utilisé dans protocole soit sémantiquement sûr selon des attaques à clairs et chiffrés choisis.*

Preuve. Selon le même argument que dans la preuve du théorème 4, on peut remplacer N_a, N_b dans les chiffrements avec les clés K_{as} et K_{bs} pour transformer le protocole initial ProtORMod en le protocole $\mathsf{ProtORMod}_4$.
Puis, d'après le postulat R7, puisque N_b est choisi aléatoirement, on peut remplacer $\mathcal{E}_{ab}(N_b)$ par $c_b \overset{R}{\leftarrow} \mathcal{M}$. On doit remplacer aussi $\{N_a, N_b\}_{K_{ab}}$ par $\{N_a, \mathcal{D}_{ab}(c_b)\}_{K_{ab}}$.
Alors, maintenant, on ne peut pas appliquer le postulat R7 pour remplacer $\{N_a, \mathcal{D}_{ab}(c_b)\}_{K_{ab}}$ par $c_a \overset{R}{\leftarrow} \mathcal{M}$ puisque $\{N_a, \mathcal{D}_{ab}(c_b)\}$ n'est plus aléatoire comme N_a dans le protocole d'Otway-Rees étendu.
Alors, à partir d'ici, on ne peut plus raisonner et par conséquent, on ne peut pas conclure que le protocole garantit l'authentication mutuelle. □

2.7.5 Conclusion

De cette analyse, on peut voir une limite de l'analyse par la logique BAN : quand les attaques utilisent des propriétés arithmétiques, elles peuvent passer au travers de cette méthode. En revanche, comme on a doté la logique BAN étendue de certaines propriétés arithmétiques, cette méthode peut détecter des bogues du protocole, comme la méthode par réduction.

Chapitre 3

Analyse de la sécurité du chiffrement asymétrique

Dans cette partie, nous allons appliquer la logique BAN étendue que nous avons introduite dans le chapitre précédent pour prouver la sécurité de certains schémas de chiffrement asymétrique. Ces analyses nous donnent l'espoir que cette méthode d'analyse peut aborder plusieurs types de protocoles cryptographiques.

3.1 Notions de sécurité du chiffrement asymétrique

3.1.1 Chiffrement à clé publique

Un schéma de chiffrement à clé publique $\mathcal{S} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ est défini par les trois algorithmes suivants :

- $\mathcal{G}$:*algorithme de génération des clés.* En fonction d'un paramètre de sécurité k, l'algorithme $\mathcal{G}(1^k)$ retourne une paire de clés publique/privée associées $(\mathsf{pk}, \mathsf{sk})$. Cet algorithme $\mathcal{G}$ est probabiliste.
- $\mathcal{E}$: *algorithme de chiffrement.* Étant donné un message $m \in \mathcal{M}$ et une clé publique pk, $\mathcal{E}_{\mathsf{pk}}(m)$ produit un chiffré c de m. Cet algorithme peut être probabiliste. Dans ce cas, on utilisera la notation $\mathcal{E}_{\mathsf{pk}}(m, r)$, où r est l'aléa fourni à l'algorithme $\mathcal{E}$.
- $\mathcal{D}$: *algorithme de déchiffrement* Étant donné un chiffré c et la clé privée sk (associée à pk), $\mathcal{D}_{\mathsf{sk}}(c)$ retourne le message clair m correspondant, ou $\perp$ pour un chiffré non valide. Cet algorithme est nécessairement déterministe.

Pour évaluer la sécurité effective d'un schéma de chiffrement, il faut formaliser les notions que l'on souhaite garantir. Pour cela, on précise les *buts* qu'un attaquant peut vouloir atteindre et les *moyens* qu'il peut disposer.

3.1.2 Buts d'un attaquant

Plusieurs notions de sécurité sont introduits [9, 7, 3, 2] mais dans ce raport, on se concentre sur la notion de *sécurité sémantique* qui exprime que tous attaquants polynomiaux ne peuvent prédire un seul bit du message clair. On présente brievement quand même quelques autres notions de sécurité et leurs relations [4].

Définition 6 *Un schéma de chiffrement asymétrique $\mathcal{S} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ est dit $\epsilon-$ à sens unique (ou one-wayness - OW) face à un adversaire $\mathcal{A}$ si :*

$$\mathsf{Succ}^{\mathsf{ow}}_{\mathcal{S}}(\mathcal{A}) \stackrel{\mathsf{def}}{=} \Pr_{m,r}\left[(\mathsf{pk},\mathsf{sk}) \leftarrow \mathcal{G}(1^k) : \mathcal{A}(\mathsf{pk}, \mathcal{E}_{\mathsf{pk}}(m,r)) = m\right].$$

est inférieur à ϵ.

Définition 7 *Un schéma de chiffrement asymétrique $\mathcal{S} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ est dit $\epsilon-$ sémantiquement sûr (ou indistinguabilité des chiffrés -IND) [9] face à un adversaire $\mathcal{A}$ si :*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{ind}}_{\mathcal{S}}(\mathcal{A}) &\stackrel{\mathsf{def}}{=} \left| 2 \times \Pr_{b,r}\left[\begin{array}{c}(\mathsf{pk},\mathsf{sk}) \leftarrow \mathcal{G}(1^k), (m_0,m_1,s) \leftarrow A_1(\mathsf{pk}),\\ c = \mathcal{E}_{\mathsf{pk}}(m_b,r), b' = A_2(m_0,m_1,s,c) : b' = b\end{array}\right] - 1 \right| \\ &= \left| \Pr_r[b' = 1 \,|\, b = 1] - \Pr_r[b' = 1 \,|\, b = 0] \right|,\end{aligned}$$

est inférieur à ϵ.

Dans cette définition, l'attaquant $\mathcal{A}$ fonctionne en deux temps, (A_1, A_2) : d'abord, à la vue de la clé publique, l'algorithme A_1 choisit deux messages de même taille et puis l'algorithme A_2 devra distinguer les chiffrés sur le challenge c. La variable s permet seulement à A_1 de transmettre formellement de l'information à la deuxième étape A_2.

Définition 8 *Un schéma de chiffrement asymétrique $\mathcal{S} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ est dit $\epsilon-$ non - malléabilité - NM [7] face à un adversaire $\mathcal{A}$ si :*

$$\mathsf{Adv}^{\mathsf{nm}}_{\mathcal{S}}(\mathcal{A}) \stackrel{\mathsf{def}}{=} \left|\mathsf{Succ}^{M}_{\mathcal{S}}(\mathcal{A}) - \mathsf{Succ}^{\$}_{\mathcal{S}}(\mathcal{A})\right|$$

est inférieur à ϵ.

avec

$$\left.\begin{aligned}\mathsf{Succ}^{M}_{\mathcal{S}}(\mathcal{A}) &= \Pr\left[\begin{array}{c}y \notin \mathbf{y} \ \wedge \ \perp \notin \mathbf{x}\\ \wedge\ R(x,\mathbf{x})\end{array}\right]\\ \mathsf{Succ}^{\$}_{\mathcal{S}}(\mathcal{A}) &= \Pr\left[\begin{array}{c}y \notin \mathbf{y} \ \wedge \ \perp \notin \mathbf{x}\\ \wedge\ R(x^\star,\mathbf{x})\end{array}\right]\end{aligned}\right\} \begin{array}{l}\text{sur l'espace de probabilités défini par}\\ \\ (\mathsf{pk},\mathsf{sk}) \leftarrow \mathcal{G}(1^k), (M,s) \leftarrow A_1(\mathsf{pk}),\\ x, x^\star \leftarrow M, y = \mathcal{E}_{\mathsf{pk}}(x,r),\\ (R,\mathbf{y}) \leftarrow A_2(M,s,y), \mathbf{x} = \mathcal{D}_{\mathsf{sk}}(\mathbf{y}).\end{array}$$

Dans cette définition, on considère un attaquant $\mathcal{A} = (A_1, A_2)$ en deux étapes. Dans un premier temps, l'algorithme A_1, à la vue de la clé publique pk, retourne une distribution sur l'ensemble des messages, caractérisée par un algorithme d'échantillonnage M. Un tel algorithme ne doit retourner avec une probabilité non nulle que des messages de même taille ; Dans un deuxième temps, l'algorithme A_2 reçoit le chiffré y d'un message aléatoire x (selon la distribution M). Cet adversaire retourne une relation R et un vecteur $\mathbf{y}$ de chiffrés (tous différents de y). Il espère que $R(x, \mathbf{x})$ soit satisfaite, où $\mathbf{x}$ est le déchiffrement de $\mathbf{y}$.

Les moyens d'un attaquant

- *attaque à clairs choisis* (ou *chosen-plaintext attack* – CPA) : dans le contexte asymétrique, grâce à la clé publique, un attaquant peut chiffrer tout message de son choix, donc, ce type d'attaque est mis en œuvre de manière évidente.
- *attaque à chiffrés choisis non-adaptatives* (ou *chosen-plaintext attack* – CCA1)[12] : l'attaquant a accès à l'algorithme de déchiffrement mais cet accès n'est possible qu'avant la vue du challenge (dans la première étape de l'attaque),

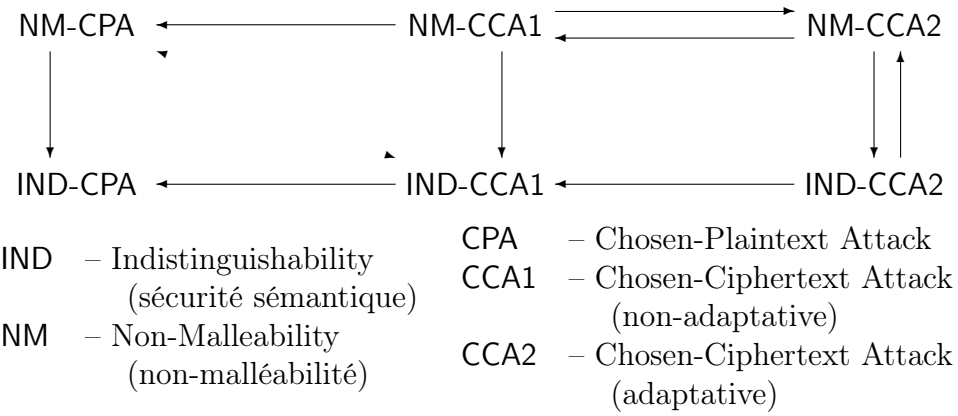

FIGURE 3.1 – Relations entre les notions de sécurité

— *attaque à chiffrés choisis adaptatives* (ou *chosen-plaintext attack* – CCA2)[15] : l'attaquant a accès à l'algorithme de déchiffrement et cet accès est illimité (avec la restriction naturelle de ne pas l'utiliser sur le challenge)

Dans les parties ci-dessous, on note $\mathsf{Succ}^{\mathsf{xxx}}(t(k))$ ou $\mathsf{Adv}^{\mathsf{xxx}}(t(k))$ la probabilité, respectivement succès ou avantage, maximale qu'un attaquant de type xxx (définissant l'objectif et les moyens, par exemple ind-cpa) en temps $t(k)$, où k est le paramètre de sécurité. Ce dernier sera par la suite omis mais sous-entendu. Si pour un système donné, ε est supérieure à $\mathsf{Succ}^{\mathsf{xxx}}(t)$ ou $\mathsf{Adv}^{\mathsf{xxx}}(t)$, alors on dit que ce système est (t, ε)–XXX-sûr.

3.1.3 Relations entre les notions de sécurité

Une étude plus complète des relations entre les différentes notions de sécurité a été menée dans [4]. Elle présente les preuves des implications présentées sur la figure 3.1.

Problèmes difficiles

Comme on a présenté, les preuves de sécurité sont produites sous l'hypothèses de la difficulté de certaines problèmes. Dans cette partie, on présente quelques problèmes que l'on utilisera dans la suite :

Considérons un groupe cyclique fini $\mathcal{G}$, d'ordre q, (on le considèrera premier par la suite), ainsi qu'un générateur g (*i.e.* $\mathcal{G} = \langle g \rangle$). Dans de tels groupes, on considère les problèmes suivants :

— le problème du **logarithme discret** (DL) : étant donné $y \in \mathcal{G}$, calculer $x \in \mathbb{Z}_q$ tel que $y = g^x$. On définit le succès d'un algorithme $\mathcal{A}$ par

$$\mathsf{Succ}^{\mathsf{dl}}(\mathcal{A}) = \Pr_{x \in \mathbb{Z}_q}[\mathcal{A}(g^x) = x].$$

— le problème **Diffie–Hellman Calculatoire** (CDH) : étant donné deux éléments dans le groupe $\mathcal{G}$, $A = g^a$ et $B = g^b$, calculer $C = g^{ab}$. On définit alors $C = \mathsf{DH}(A, B)$ ainsi que le succès d'un algorithme $\mathcal{A}$ par

$$\mathsf{Succ}^{\mathsf{cdh}}(\mathcal{A}) = \Pr_{a,b \in \mathbb{Z}_q}[\mathcal{A}(g^a, g^b) = g^{ab}].$$

Soient p un nombre premier de la forme $\kappa q + 1$, où q est un grand nombre premier également (typiquement p est sur 512 bits et q sur 160 bits) et

— Données communes : p, q et g

— Clé privée d'Alice : $x \in \mathbb{Z}_q$

— Clé publique d'Alice : $y = g^x \bmod p$

— Chiffrement : soit m un message à chiffrer pour Alice. Ce message m est vu comme un élément de $\langle g \rangle$. Bob choisit un élément $k \in \mathbb{Z}_q$ puis calcule

$$r = g^k \bmod p \text{ et } s = y^k \times m \bmod p.$$

Le chiffré de m est alors constitué de la paire (r, s).

— Déchiffrement : seule Alice est capable de retrouver m à partir du chiffré, grâce à sa connaissance de x. En effet,

$$y^k = g^{xk} = (g^k)^x = r^x \bmod p$$

Ainsi, $m = s/r^x \bmod p$.

FIGURE 3.2 – Chiffrement de El Gamal

— le problème **Diffie–Hellman Décisionnel** (DDH) : étant donné trois éléments dans le groupe $\mathcal{G}$, $A = g^a$, $B = g^b$ et $C = g^c$, décider si $C = \mathsf{DH}(A, B)$, ce qui est équivalent à décider si $c = ab \bmod q$. On définit l'avantage d'un distingueur $\mathcal{D}$ par

$$\mathsf{Adv}^{\mathsf{ddh}}(\mathcal{D}) = | \Pr_{a,b,c \in \mathbb{Z}_q}[1 \leftarrow \mathcal{D}(g^a, g^b, g^c)] - \Pr_{a,b \in \mathbb{Z}_q}[1 \leftarrow \mathcal{D}(g^a, g^b, g^{ab})] |.$$

Ces problèmes sont classés du plus difficile au plus facile. En effet, la résolution du logarithme discret permet de résoudre les problèmes Diffie–Hellman. De même, quand on peut calculer le DH, on peut le décider.

3.2 Schéma de chiffrement d'ElGamal

3.2.1 Description

En 1985, El Gamal a proposé un cryptosystème [8] basé sur le problème Diffie-Hellman. Une description est donnée figure 3.2. On présente d'abord les résultats de preuve de sécurité par une analyse par réduction [14], et puis on donne des preuves de ces résultats par l'analyse par la logique BAN étendue.

3.2.2 Analyse par réduction

Théorème 5 *L'inversion (*OW-CPA*) du chiffrement de El Gamal est équivalente au problème Diffie-Hellman Calculatoire :* $\mathsf{Succ}^{\mathsf{ow-cpa}}(t) \leq \mathsf{Succ}^{\mathsf{cdh}}(t)$.

Preuve. Considérons l'attaquant $\mathcal{A}$ contre OW-CPA, on utilisera cet attaquant comme sous-programme pour résoudre une instance aléatoire $(A = g^a, B = g^b)$ du problème Diffie-Hellman. Nous dénommons ce jeu réel Game_0.

Game_0 **:** on exécute l'algorithme de génération de clés qui retourne $y = g^x$ pour un x aléatoire dans $\mathbb{Z}_q$. Après avoir un challenge $(r = g^k, s = my^k)$ pour un message

$m \stackrel{R}{\leftarrow} \mathcal{G}$ aléatoire. l'attaquant retourne $m = s/y^k$ avec probabilité ε. On note cet événement S_0, ainsi que S_i dans les jeux Game_i ci-dessous : $\Pr[S_0] = \varepsilon$.

Game_1 **:** on modifie un peu le jeu réel, au lieu de prendre $y = g^x$ comme clé publique, on utilise $y = A$, ce qui revient à prendre $x = a$. La construction de s change un peu : on remplace $s = my^k$ par $s = mA^k$. Puisque l'instance (A, B) est choisie aléatoirement, les distributions de x et y sont identiques à celles du jeu précédent, alors : $\Pr[S_1] = \Pr[S_0]$.

Game_2 **:** on modifie désormais la construction du challenge. Puisque k est choisi aléatoirement, on peut remplacer $r = g^k$ par B qui revient à prendre $k = b$. La construction de s change un peu : on remplace $s = mA^k$ par $s = mA^b$. La distribution de r est identique à celle du jeu précédent : $\Pr[S_2] = \Pr[S_1]$.

Game_3 **:** enfin, puisque on choisit aléatoirement un message m : $m \stackrel{R}{\leftarrow} \mathcal{G}$, au lieu de définir $s = mA^b$, on choisit $s \stackrel{R}{\leftarrow} \mathcal{G}$ aléatoire. La structure de groupe fait que la distribution de s est uniforme dans les deux cas : $\Pr[S_3] = \Pr[S_2]$.

On peut récrire l'événement S_3 de la façon suivante :

$$\begin{aligned} \varepsilon &= \Pr[S_3] = \Pr[s \stackrel{R}{\leftarrow} \mathcal{G}, a, b \stackrel{R}{\leftarrow} \mathbb{Z}_q, y = g^a, r = g^b : \mathcal{A}(y, r, s) = s/y^b] \\ &= \Pr[A, B, s \stackrel{R}{\leftarrow} \mathcal{G}, y = A, r = B : \mathcal{A}(y, r, s) = s/\mathsf{DH}(A, B)]. \end{aligned}$$

La probabilité ε est donc bornée par la probabilité de résoudre le problème CDH, en le même temps que l'exécution de $\mathcal{A}$. □

Théorème 6 *La sécurité sémantique (*$\mathsf{IND-CPA}$*) du chiffrement de El Gamal est équivalente au problème Diffie-Hellman Décisionnel :* $\mathsf{Adv}^{\mathsf{ind-cpa}}(t) \leq 2 \times \mathsf{Adv}^{\mathsf{ddh}}(t)$.

Preuve. Comme ci-dessus, considérons l'attaquant $\mathcal{A}$ contre $\mathsf{IND\text{-}CPA}$, on utilisera cet attaquant comme sous-programme pour résoudre une instance aléatoire $(A = g^a, B = g^b)$ du problème Diffie-Hellman Décisionnel, avec C comme candidat. Nous dénommons ce jeu réel Game_0.

Game_0 **:** on exécute l'algorithme de génération de clés qui retourne $y = g^x$ pour un x aléatoire dans $\mathbb{Z}_q$. Sur y, A_1 retourne deux messages (m_0, m_1). Sur le chiffré $\gamma = (r, s)$ de m_δ, A_2 retourne son choix δ'. Avec probabilité $(\varepsilon+1)/2$, $\delta' = \delta$. On note cet événement S_0, ainsi que S_i dans les jeux Game_i ci-dessous : $\Pr[S_0] = (\varepsilon + 1)/2$.

Game_1 **:** comme ci-dessus, on modifie un peu le jeu réel, en utilisant $y = A$ puis $r = B$, ce qui revient à prendre $x = a$ et $k = b$. La construction de s reste inchangée : $s = m_\delta A^b$. Les distributions de x, y et r sont identiques, en raison de l'instance aléatoire (A, B) : $\Pr[S_1] = \Pr[S_0]$.

Game_2 **:** puis, au lieu de définir $s = m_\delta A^b$, on définit $s = m_\delta C$, pour $C = \mathsf{DH}(A, B)$. Alors, $\Pr[S_2] = \Pr[S_1]$.

Game_3 **:** maintenant, on remplace $C = \mathsf{DH}(A, B)$ par un candidat $C = g^c$ aléatoire. Puisque l'événement $\delta' = \delta$ est détectable, on peut définir le distingueur $\mathcal{D}$ qui exécute le même jeu que le Game_2, qui peut être effectivement le Game_2 ou le Game_3 selon que $C = \mathsf{DH}(A, B)$ ou non, ce que l'on ignore. Toujours est-il que le distingueur, à la fin du jeu, retourne 0 si $\delta' \neq \delta$, et 1 si $\delta' = \delta$:

$$\Pr[1 \leftarrow \mathcal{D} \,|\, C \stackrel{R}{\leftarrow} \mathcal{G}] = \Pr[S_3] \text{ et } \Pr[1 \leftarrow \mathcal{D} \,|\, C = \mathsf{DH}(A, B)] = \Pr[S_2].$$

Ainsi,

$$|\Pr[S_3] - \Pr[S_2]| \leq \mathsf{Adv}^{\mathsf{ddh}}(\mathcal{D}) \leq \mathsf{Adv}^{\mathsf{ddh}}(t).$$

Enfin, il est aisé de remarquer que $\Pr[S_3] = 1/2$. Appliquons alors l'inégalité triangulaire :

$$\frac{\varepsilon}{2} = \frac{1+\varepsilon}{2} - \frac{1}{2} = |\ \Pr[S_3] - \Pr[S_0]\ | \leq \mathsf{Adv}^{\mathsf{ddh}}(t),$$

d'où le résultat. □

3.2.3 Analyse par la logique BAN étendue

Pour $A, B \in \mathcal{G} : A = g^a, B = g^b$, on définit :

$$\begin{aligned} \mathsf{DH}(A,B) &= g^{ab} \\ \mathsf{DL}(A) &= a \end{aligned}$$

On définit les prédicats :

$$\begin{aligned}
\mathsf{CDH}(\mathcal{A},\mathcal{G},T) &= A, B \xleftarrow{R} \mathcal{G} \wedge (\mathcal{A} \triangleleft A, B) : \mathcal{A} \rightarrow \mathsf{DH}(A,B) \wedge (t(\mathcal{A}) \leq T) \\
\mathsf{DDH}(\mathcal{A},\mathcal{G},T) &= (A, B \xleftarrow{R} \mathcal{G}) \wedge (\delta \xleftarrow{R} \{0,1\}) \wedge (C_0 = \mathsf{DH}(A,B)) \wedge (C_1 \xleftarrow{R} \mathcal{G}) \\
&\quad \wedge \mathcal{A} \triangleleft (A, B, C_\delta) : (\mathcal{A} \rightarrow \delta) \wedge (t(\mathcal{A}) \leq T) \\
&= (A, B \xleftarrow{R} \mathcal{G}) \wedge (\delta \xleftarrow{R} \{0,1\}) \wedge (C_\delta = \mathsf{DH}(A,B)) \wedge (C_{1-\delta} \xleftarrow{R} \mathcal{G}) \\
&\quad \wedge (\mathcal{A} \triangleleft A, B, C_0, C_1) : (\mathcal{A} \rightarrow \delta) \wedge (t(\mathcal{A}) \leq T) \\
\mathsf{OW}-\mathsf{CPA}(\mathcal{A}, ProtElg, T) &= (x, k \xleftarrow{R} \mathbb{Z}_q) \wedge (m \xleftarrow{R} \mathcal{G}) \wedge (\mathcal{A} \triangleleft y = g^x, r = g^k, s = m \cdot y^k) \\
&\quad : (\mathcal{A} \rightarrow m) \wedge (t(\mathcal{A}) \leq T) \\
\mathsf{IND}-\mathsf{CPA}(\mathcal{A}, \mathsf{ProtElg}, T) &= (x, k \xleftarrow{R} \mathbb{Z}_q) \wedge (m_0, m_1 \xleftarrow{R} \mathcal{G}) \wedge (\delta \xleftarrow{R} \{0,1\}) \\
&\quad \wedge (\mathcal{A} \triangleleft m_0, m_1, y = g^x, r = g^k, s = m_\delta \cdot y^k) \\
&\quad : (\mathcal{A} \rightarrow \delta) \wedge (t(\mathcal{A}) \leq T)
\end{aligned}$$

Théorème 7 *Le chiffrement de El Gamal est à sens unique sous l'hypothèse que le problème Diffie-Hellman Calculatoire soit difficile, plus précisément :*

$$\mathsf{val}(\mathsf{OW}-\mathsf{CPA}(\mathsf{ProtElg}), T) \leq \mathsf{val}(\mathsf{CDH}(\mathcal{G}, T))$$

Preuve. On note $v = \mathsf{val}(\mathsf{OW}-\mathsf{CPA}(\mathsf{ProtElg}, T))$, et on considère $\mathcal{A}$ tel que : $\mathsf{val}(\mathsf{OW}-\mathsf{CPA}(\mathcal{A}, \mathsf{ProtElg}, T)) = v$.
Alors :

$$\begin{aligned}
&(x, k \xleftarrow{R} \mathbb{Z}_q) \wedge (m \xleftarrow{R} \mathcal{G}) \wedge (\mathcal{A} \triangleleft y = g^x, r = g^k, s = m \cdot y^k) \\
&: (\mathcal{A} \rightarrow m') \wedge < m' = m, v > \wedge (t(\mathcal{A}) \leq T))
\end{aligned}$$

— Avec $x \xleftarrow{R} \mathbb{Z}_q$, quand on considère y comme une fonction de x on a $\mathsf{Unif}(y = g^x, x)$, avec $\mathsf{Im}(g) = \mathcal{G}$. D'après le postulat R7 on peut remplacer $y = g^x$ par $A : A \xleftarrow{R} \mathcal{G}$:

$$\begin{aligned}
&(k \xleftarrow{R} \mathbb{Z}_q) \wedge (m, A \xleftarrow{R} \mathcal{G}) \wedge (\mathcal{A} \triangleleft (A = g^a, r = g^k, s = m \cdot A^k)) \\
: \quad &(\mathcal{A} \rightarrow m') \wedge < m' = m, v > \wedge (t(\mathcal{A}) \leq T)
\end{aligned}$$

— Avec $m \stackrel{R}{\leftarrow} \mathcal{G}$, quand on considère s comme une fonction de m, on a $\mathsf{Unif}(s = m \cdot y^k, m)$, avec $\mathsf{Im}(s) = \mathcal{G}$. D'après le postulat R7 on peut remplacer $s = m \cdot A^k$ par $D : D \stackrel{R}{\leftarrow} \mathcal{G}$ et m par D/A^k :

$$\begin{array}{rl} & (k \stackrel{R}{\leftarrow} \mathbb{Z}_q) \wedge (A, D \stackrel{R}{\leftarrow} \mathcal{G}) \wedge (\mathcal{A} \triangleleft (A = g^a, r = g^k, D)) \\ : & (\mathcal{A} \rightarrow m') \wedge < m' = D/A^k, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

— Avec $k \stackrel{R}{\leftarrow} \mathcal{G}$, quand on considère r comme une fonction de k, on a $\mathsf{Unif}(r = g^k, k)$, avec $\mathsf{Im}(r) = \mathcal{G}$. D'après le postulat R7 on peut remplacer $r = g^k$ par $B : B \stackrel{R}{\leftarrow} \mathcal{G}$ et k par $\mathsf{DL}(B)$:

$$\begin{array}{rl} & (A, B, D \stackrel{R}{\leftarrow} \mathcal{G}) \wedge (\mathcal{A} \triangleleft (A = g^a, B = g^b, D)) \\ : & (\mathcal{A} \rightarrow m') \wedge < m' = D/A^{\mathsf{DL}(B)}, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

On remarque que $A^{\mathsf{DL}(B)} = \mathsf{DH}(A, B)$. Alors :

$$\begin{array}{rl} & (A, B, D \stackrel{R}{\leftarrow} \mathcal{G}) \wedge (\mathcal{A} \triangleleft (A = g^a, B = g^b, D)) \\ : & (\mathcal{A} \rightarrow m') \wedge < m' = D/\mathsf{DH}(A, B), v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

ce qui entraîne :

$$\begin{array}{rll} \exists \mathcal{B} & : & (A, B, D \stackrel{R}{\leftarrow} \mathcal{G}) \wedge (\mathcal{B} \triangleleft (A = g^a, B = g^b, D)) \\ & & : (\mathcal{B} \rightarrow D^+ = D/m') \wedge < D^+ = \mathsf{DH}(A, B), v > \wedge (t(\mathcal{B}) \leq T) \end{array}$$

On en déduit :

$$v \leq \mathsf{val}(\mathsf{CDH}(\mathcal{G}, T))$$

□

Théorème 8 *Le chiffrement de El Gamal est sémantiquement sûr sous l'hypothèse que le problème Diffie-Hellman Décionnel est difficile, plus précisément :*

$$\mathsf{val}(\mathsf{IND} - \mathsf{CPA}(\mathsf{ProtElg}, T)) \leq \mathsf{val}(\mathsf{DDH}(\mathcal{A}, \mathcal{G}, T))$$

Preuve. On note $v = \mathsf{val}(\mathsf{IND} - \mathsf{CPA}(\mathsf{ProtElg}, T))$, et on considère $\mathcal{A}$ tel que : $\mathsf{val}(\mathsf{IND} - \mathsf{CPA}(\mathcal{A}, \mathsf{ProtElg}, T)) = v$.
Alors :

$$\begin{array}{rl} & (x, k \stackrel{R}{\leftarrow} \mathbb{Z}_q) \wedge (m_0, m_1 \stackrel{R}{\leftarrow} \mathcal{G}) \wedge \delta \stackrel{R}{\leftarrow} \{0, 1\} \\ \wedge & (\mathcal{A} \triangleleft (m_0, m_1, y = g^x, r = g^k, s = m_\delta \cdot y^k)) \\ : & < \mathcal{A} \rightarrow \delta, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

— Avec $x \stackrel{R}{\leftarrow} \mathbb{Z}_q$, quand on considère y comme une fonction de x, on a $\mathsf{Unif}(y = g^x, x)$, avec $\mathsf{Im}(y) = \mathcal{G}$. D'après le postulat R7 on peut remplacer $y = g^x$ par $A : A \stackrel{R}{\leftarrow} \mathcal{G}$:

$$\begin{array}{rl} & (k \stackrel{R}{\leftarrow} \mathbb{Z}_q) \wedge (m_0, m_1, A \stackrel{R}{\leftarrow} \mathcal{G}) \wedge (\delta \stackrel{R}{\leftarrow} \{0, 1\}) \\ \wedge & (\mathcal{A} \triangleleft (m_0, m_1, A = g^a, r = g^k, s = m_\delta \cdot A^k)) \\ : & < \mathcal{A} \rightarrow \delta, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

— Avec $k \xleftarrow{R} \mathcal{G}$, quand on considère r comme une fonction de k, on a $\mathsf{Unif}(r = g^k, k)$, avec $\mathsf{Im}(r) = \mathcal{G}$. D'après le postulat R7 on peut remplacer $r = g^k$ par $B : B \xleftarrow{R} \mathcal{G}$ et k par $\mathsf{DL}(B)$:

$$(m_0, m_1, A, B \xleftarrow{R} \mathcal{G}) \wedge \delta \xleftarrow{R} \{0,1\} \wedge (\mathcal{A} \triangleleft (m_0, m_1, A = g^a, B = g^b, s = m_\delta \cdot A^{\mathsf{DL}(B)}))$$
$$:< \mathcal{A} \to \delta, v > \wedge (t(\mathcal{A}) \leq T)$$

On remarque que $A^{\mathsf{DL}(B)} = \mathsf{DH}(A, B)$. Alors :

$$\begin{array}{rl} & m_0, m_1, A, B \xleftarrow{R} \mathcal{G} \wedge \delta \xleftarrow{R} \{0,1\} \\ \wedge & (\mathcal{A} \triangleleft (m_0, m_1, A = g^a, B = g^b, s = m_\delta \cdot \mathsf{DH}(A, B)) \\ : & < \mathcal{A} \to \delta, v > \wedge (t(\mathcal{A}) \leq T)) \end{array}$$

On assigne $C_0 = s/m_0$, $C_1 = s/m_1$. Alors : $C_\delta = \mathsf{DH}(A, B)$ et $C_{1-\delta} = \frac{m_\delta}{m_{1-\delta}} \cdot \mathsf{DH}(A, B)$.
On peut remplacer $(\mathcal{A} \triangleleft (m_0, m_1, s = m_\delta \cdot \mathsf{DH}(A, B)))$ par $(\mathcal{A} \triangleleft (C_0, C_1, s = m_\delta \cdot \mathsf{DH}(A, B)))$.
On a :

$$\begin{array}{rl} & m_0, m_1, A, B \xleftarrow{R} \mathcal{G} \wedge \delta \xleftarrow{R} \{0,1\} \wedge (C_\delta = \mathsf{DH}(A, B)) \wedge (C_{1-\delta} = \dfrac{m_\delta}{m_{1-\delta}} \cdot C_\delta) \\ \wedge & (\mathcal{A} \triangleleft (A = g^a, B = g^b, s = m_\delta \cdot C_\delta, C_\delta, C_{1-\delta})) \\ : & < \mathcal{A} \to \delta, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

Avec $C_{1-\delta} = \frac{m_\delta}{m_{1-\delta}} \cdot C_\delta$, quand on considère C_δ comme une fonction de $m_{1-\delta}$, on déduit $\mathsf{Unif}(\mathcal{C}_{1-\delta}, m_{1-\delta})$, avec $\mathsf{Im}(\mathcal{C}_{1-\delta}) = \mathcal{G}$. D'après le postulat R7, on peut remplacer $C_{1-\delta} = \frac{m_\delta}{m_{1-\delta}} \cdot C_\delta$ par $C_{1-\delta} \xleftarrow{R} \mathcal{G}$:

$$\begin{array}{rl} & m_\delta, A, B \xleftarrow{R} \mathcal{G} \wedge \delta \xleftarrow{R} \{0,1\} \wedge (C_\delta = \mathsf{DH}(A, B)) \wedge C_{1-\delta} \xleftarrow{R} \mathcal{G} \\ \wedge & (\mathcal{A} \triangleleft (A = g^a, B = g^b, s = m_\delta \cdot C_\delta, C_\delta, C_{1-\delta})) \\ : & < \mathcal{A} \to \delta, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

Avec $s = m_\delta \cdot C_\delta$, quand on considère s comme une fonction de m_δ, on déduit $\mathsf{Unif}(s, m_\delta)$,avec $\mathsf{Im}(\mathcal{C}_s) = \mathcal{G}$ D'après le postulat R7, on peut remplacer s par $u \xleftarrow{R} \mathcal{G}$

$$\begin{array}{rl} & A, B, u, C_{1-\delta} \xleftarrow{R} \mathcal{G} \wedge \delta \xleftarrow{R} \{0,1\} \wedge (C_\delta = \mathsf{DH}(A, B)) \\ \wedge & (\mathcal{A} \triangleleft (A = g^a, B = g^b, u, C_\delta, C_{1-\delta})) \\ : & < \mathcal{A} \to \delta, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

D'après le postulat R5, on peut éliminer u :

$$\begin{array}{rl} & A, B, C_{1-\delta} \xleftarrow{R} \mathcal{G} \wedge \delta \xleftarrow{R} \{0,1\} \wedge (C_\delta = \mathsf{DH}(A, B)) \\ \wedge & (\mathcal{A} \triangleleft (A = g^a, B = g^b, C_\delta, C_{1-\delta})) \\ : & < \mathcal{A} \to \delta, v > \wedge (t(\mathcal{A}) \leq T) \end{array}$$

On en déduit : $v \leq \mathsf{val}(\mathsf{DDH}(\mathcal{G}, T))$. □

3.3 Une construction générique

Bellare et Rogaway [1] ont proposé une construction générique permettant de construire un cryptosystème IND-CCA2 à partir de toute permutation à sens-unique à trappe. Cette construction est décrite comme suit :

- f : une permutation de l'espace E à sens-unique à trappe qui est considérée comme clé publique, et son inverse g comme clé privée (possible grâce à la trappe).
- G, H : deux oracles aléatoires à valeur dans $\{0,1\}^n$ et $\{0,1\}^{k_1}$ respectivement.
- *chiffrement* : Pour un message $m \in \{0,1\}^n$, on choisit $r \xleftarrow{R} E$, puis on calcule le chiffré de m par :
$$\mathcal{E}(m;r) = f(r) \,\|\, m \oplus G(r) \,\|\, H(m,r).$$
- *déchiffrement* : pour un chiffré $C = a \,\|\, b \,\|\, c$, on le déchiffre en deux étapes : tout d'abord, on retrouve $r = g(a)$, grâce à la trappe de f, puis $m = b \oplus G(r)$; ensuite, avant de retourner le message m, on vérifie la consistance du chiffré, à savoir si $c = H(m,r)$.

On présente maintenant les analyses de la sécurité de cette construction par deux méthodes : l'une par réduction et l'autre par la logique BAN étendue.

3.3.1 Analyse par réduction

Dans [1], les auteurs ont montré le résultat suivant :

Théorème 9 *Considérons un adversaire $\mathcal{A}$ contre cette construction, selon une attaque à chiffrés choisis adaptative. Supposons qu'après q_D questions à l'oracle de déchiffrement et q_G, q_H questions aux oracles G et H, $\mathcal{A}$ a un avantage ε en temps t, alors on peut inverser f avec succès $\varepsilon/2 - (q_H/2^n + 1/2^{k_1}) \cdot q_D$, en temps $t + (q_G + q_H)T_f$, où T_f désigne le temps d'une évaluation de f.*

Lemme 2 *Soient E, F et G des événements dans un espace de probabilités, alors*

$$\Pr[\mathsf{E} \wedge \neg\mathsf{G}] = \Pr[\mathsf{F} \wedge \neg\mathsf{G}] \implies |\Pr[\mathsf{E}] - \Pr[\mathsf{F}]| \leq \Pr[\mathsf{G}].$$

Preuve. La différence $|\Pr[\mathsf{E}] - \Pr[\mathsf{F}]|$ est égale à

$$\begin{aligned}&|\Pr[\mathsf{E} \wedge \neg\mathsf{G}] + \Pr[\mathsf{E} \wedge \mathsf{G}] - \Pr[\mathsf{F} \wedge \neg\mathsf{G}] - \Pr[\mathsf{F} \wedge \mathsf{G}]| = |\Pr[\mathsf{E} \wedge \mathsf{G}] - \Pr[\mathsf{F} \wedge \mathsf{G}]| \\ &= |\Pr[\mathsf{E} \mid \mathsf{G}] \cdot \Pr[\mathsf{G}] - \Pr[\mathsf{F} \mid \mathsf{G}] \cdot \Pr[\mathsf{G}]| \leq |\Pr[\mathsf{E} \mid \mathsf{G}] - \Pr[\mathsf{F} \mid \mathsf{G}]| \cdot \Pr[\mathsf{G}] \leq \Pr[\mathsf{G}].\end{aligned}$$

□

Preuve. Considérons l'attaquant $\mathcal{A} = (A_1, A_2)$ contre ce schéma. Dans les deux étapes, A_1 et A_2 ont accès à l'oracle de déchiffrement.

Game_0 **:** on exécute l'algorithme de génération de clés qui retourne une permutation f et son inverse g. On choisit aléaloirement $x \xleftarrow{R} E$ et $y = f(x)$. Après avoir vu la description de la fonction f, A_1 retourne deux messages m_0 et m_1. Après avoir reçu le chiffré $C = a \,\|\, b \,\|\, c$ du message m_δ, A_2 retourne un bit δ'. On note r l'unique élément tel que $C = \mathcal{E}(m_\delta, r)$. Avec probabilité $(\varepsilon + 1)/2$, $\delta' = \delta$. On note cet événement S_0, ainsi que S_i dans les jeux Game_i ci-dessous : $\Pr[S_0] = (1 + \varepsilon)/2$.

Game_1 **:** On simule les oracles G et H : pour toute nouvelle question à l'un des ces oracles, on répond par une chaîne aléatoire dans l'espace correspondant, puis on stocke les questions-réponses dans les listes Liste_G et Liste_H respectivement. Il s'agit de simulations parfaites, $\Pr[S_1] = \Pr[S_0]$.

Game_2 : Maintenant, on simule l'oracle de déchiffrement. À la question $C' = a' \,\|\, b' \,\|\, c'$, si $(b' \oplus G(r'), r')$ n'est pas dans Liste_H, on rejette le chiffré ; dans les autres cas, on continue à utiliser l'oracle de déchiffrement. On ne peut refuser un chiffré valide que si $(b' \oplus G(r'), r')$ n'a pas été demandé à H. Mais alors, $H(b' \oplus G(r'), r')$ retourne un élément parfaitement aléatoire qui est égal à c' avec probabilité $1/2^{k_1}$. D'après le lemme 2 (on applique q_D fois), on obtient : $|\Pr[S_2] - \Pr[S_1]| \leq q_D/2^{k_1}$.

Game_3 : On continue à simuler l'oracle de déchiffrement. À la question $C' = a' \,\|\, b' \,\|\, c'$, pour $a' = f(r')$, si r' n'est pas dans Liste_G, on rejette le chiffré ; On ne peut refuser un chiffré valide que si $(b' \oplus G(r'), r')$ a été demandé à H mais r' n'a pas été demandé à G (car si $(b' \oplus G(r'), r')$ n'a pas été demandé à H, le chiffré C' a été rejetté dans le jeu Game_2). Mais dans ce cas, $G(r')$ retourne un élément parfaitement aléatoire, par conséquent, $b' \oplus G(r')$ est aussi un élément parfaitement aléatoire qui apparaît dans les requêtes de H avec probabilité $q_H/2^n$. D'après le lemme 2 (on applique q_D fois), on obtient : $|\Pr[S_3] - \Pr[S_2]| \leq (q_H/2^n) \cdot q_D$.

Game_4 : on poursuit la simulation de l'oracle de déchiffrement, sur $C' = a' \,\|\, b' \,\|\, c'$, pour $a' = f(r')$. On sait que $r' \in \mathsf{Liste}_G$ et $(b' \oplus G(r'), r') \in \mathsf{Liste}_H$. On peut alors trouver ce r' (en testant si $f(r') = a'$ sur toutes les questions à G, grâce à la propriété de permutation de f), puis déchiffrer correctement : $\Pr[S_4] = \Pr[S_3]$.

Game_5 : dans ce jeu, on définit $a = y = f(x)$, $b = m_\delta \oplus g^+$ et $c = h^+$, où x, g^+ et h^+ sont aléatoires. De plus, à la question $G(x)$, on répond g^+, et à la question $H(m_\delta, x)$ on répond h^+. Il s'agit simplement de spécifier certaines valeurs de G et H, par des valeurs aléatoires, on ne modifie donc pas les distributions : $\Pr[S_5] = \Pr[S_4]$.

Game_6 : maintenant, on supprime les modifications locales de G et H. Les réponses aux question $G(x)$ et $H(m_\delta, x)$ sont indépendantes de x et m_δ. La seule différence apparaît si l'événement « x a été demandé à G ou H », nommé Ask. D'après le lemme 2 : $|\Pr[S_6] - \Pr[S_5]| \leq \Pr[\mathsf{Ask}]$. Cependant, dans ce dernier jeu, δ est indépendant de la vue de l'attaquant, ainsi $\Pr[S_6] = 1/2$.

À nouveau, l'inégalité triangulaire nous donne

$$\frac{\varepsilon}{2} = \frac{1+\varepsilon}{2} - \frac{1}{2} = |\Pr[S_0] - \Pr[S_6]| \leq (\frac{q_H}{2^n}) \cdot q_D + \frac{q_D}{2^{k_1}} + \Pr[\mathsf{Ask}].$$

Cependant, l'événement Ask permet d'inverser $f(x)$ en testant toutes les questions posées à G et H qui prend en temps $(q_G + q_H)T_f$, d'où le résultat. □

3.3.2 Analyse par la logique BAN étendue

Dans les parties ci-dessus, pour alléger les notations, les fonctions peuvent être considérées comme des prédicats de la manière suivantes :

$$f(x) \equiv (f(x) \neq 0)$$

$$\neg f(x) \equiv (f(x) = 0)$$

et les prédicats peuvent être considérés comme des fonctions de la manière suivante :

$$(\mathbf{P}(x) = 1) \equiv \mathbf{P}(x)$$

$$(\mathbf{P}(x) = 0) \equiv \neg\mathbf{P}(x)$$

On note :

$$\begin{aligned} \mathsf{Em} &= \{0,1\}^n \\ \mathsf{Ec} &= \{0,1\}^k \\ G &: E \to \mathsf{Em} \\ H &: \mathsf{Em} \times E \to \mathsf{Ec} \end{aligned}$$

Pour un chiffré C, on définit le prédicat $\mathsf{Valide}(C)$ pour signifier que C est un chiffré valide (qui correspond effectivement à un clair) :

$$\mathsf{Valide}(C) = \mathsf{Valide}(a||b||c) \stackrel{\mathsf{def}}{=} \exists r : (a = f(r)) \wedge (b = m \oplus G(r) \wedge (c = H(m,r)).$$

On définit les fonctions $\mathsf{List}_G, \mathsf{List}_H$:

$$\begin{aligned} \mathsf{List}_G : E &\to \mathsf{Em} \\ r &\mapsto (A \stackrel{?}{\to} G(r)) \cdot G(r), \end{aligned}$$

quand la valeur r est demandée à l'oracle G, $\mathsf{List}_G(r)$ retourne $G(r)$, sinon il retourne 0.

$$\begin{aligned} \mathsf{List}_H : \mathsf{Em} \times E &\to \mathsf{Ec} \\ (m,r) &\mapsto (A \stackrel{?}{\to} H(m,r)) \cdot H(m,r), \end{aligned}$$

quand le couple (m,r) est demandé à l'oracle G, $\mathsf{List}_H(m,r)$ retourne $H(m,r)$, sinon il retourne 0.

Remarque :

— $\forall r \in E : A \stackrel{?}{\to} G(r) = \mathsf{List}_G(r)$

— $\forall (m,r) \in \mathsf{Em} \times E : A \stackrel{?}{\to} H(m,r) = \mathsf{List}_H(m,r)$

Sous l'hypothèse que H et G sont parfaitement aléatoires, on dispose de propriétés évidentes :

$$\forall r \in E \;:\; \neg(\mathsf{List}_G(r)) : G(r) \stackrel{R}{\leftarrow} \mathsf{Em} \tag{3.1}$$

$$\forall (m,r) \in \mathsf{Em} \times E \;:\; \neg(\mathsf{List}_H(m,r)) : H(m,r) \stackrel{R}{\leftarrow} \mathsf{Ec} \tag{3.2}$$

— La propriété (3.1) signifie que si r n'a pas été demandé à l'oracle G (ou bien s'il n'est pas dans la List_G), la valeur $G(r)$ est choisie de manière parfaitement aléatoire.

— La propriété (3.2) signifie que si (m,r) n'a pas été demandé à l'oracle H (ou bien s'il n'est pas dans la List_H), la valeur $H(m,r)$ est choisie de manière parfaitement aléatoire.

On appelle $\mathsf{ProtCons}$ le schéma de chiffrement considéré.

On définit le prédicat $\mathsf{IND-CCA2}(\mathcal{A}, \mathsf{ProtCons}, T)$ pour signifier l'avantage contre la sécurité sémantique de cette construction selon une attaque à chiffrés choisis adaptive et le prédicat $\mathsf{OW}(A, \mathsf{ProtCons}, f, T)$ pour signifier l'avantage d'inverser la fonction f utilisée dans cette construction.

$$\begin{aligned} \mathsf{IND-CCA2}(\mathcal{A}, \mathsf{ProtCons}, T, m_0, m_1) &= r \stackrel{R}{\leftarrow} E \wedge (\mathcal{A} \triangleleft G, H, \mathcal{D}, f, m_0, m_1, C = \mathcal{E}(m_\delta, r)) \\ &\quad :< \mathcal{A} \to \delta, v > \; \wedge (t(\mathcal{A}) \leq T) \end{aligned}$$

$$\mathsf{OW}(\mathcal{A}, f, \mathsf{ProtCons}, T) = x \stackrel{R}{\leftarrow} E \wedge (\mathcal{A} \triangleleft G, H, \mathcal{D}, f, f(x)) : (\mathcal{A} \rightarrow x) \wedge (t(\mathcal{A}) \leq T)$$

et puis :

$$\begin{aligned}
\mathsf{IND}-\mathsf{CCA2}(\mathcal{A}, \mathsf{ProtCons}, T) &= \exists m_0, m_1 \in \mathcal{G} : \mathsf{IND}-\mathsf{CCA2}(\mathcal{A}, \mathsf{ProtCons}, T, m_0, m_1) \\
\mathsf{IND}-\mathsf{CCA2}(\mathsf{ProtCons}, T) &= \exists \mathcal{A} : \mathsf{IND}-\mathsf{CCA2}(\mathcal{A}, \mathsf{ProtCons}, T) \\
\mathsf{OW}(f, \mathsf{ProtCons}, T) &= \exists \mathcal{A} : \mathsf{OW}(\mathcal{A}, f, \mathsf{ProtCons}, T)
\end{aligned}$$

Théorème 10

$$\begin{aligned}
& \mathsf{val}(\mathsf{IND}-\mathsf{CCA2}(\mathsf{ProtCons}, T)) \\
\leq\ & 1/2 + \mathsf{val}(\mathsf{OW}(f, \mathsf{ProtCons}, T + (q_G + q_H) \cdot T_f)) + q_D(1/|\mathsf{Ec}| + q_H/|\mathsf{Em}|).
\end{aligned}$$

Preuve. On note $v = \mathsf{val}(\mathsf{IND}-\mathsf{CCA2}(\mathsf{ProtCons}, T))$
On considère $\mathcal{A}$ tel que : $v = \mathsf{val}(\mathsf{IND}-\mathsf{CCA2}(\mathcal{A}, \mathsf{ProtCons}, T))$ Alors :

$$\begin{aligned}
\exists m_0, m_1 \in \mathcal{G} \ &: \ (r \stackrel{R}{\leftarrow} E \wedge (\mathcal{A} \triangleleft G, H, \mathcal{D}, f, m_0, m_1, C = \mathcal{E}(m_\delta, r))) \\
&\quad :< \mathcal{A} \rightarrow \delta, v > \wedge (t(\mathcal{A}) \leq T)
\end{aligned}$$

ou bien :

$$\begin{aligned}
\exists m_0, m_1 \in \mathcal{G} \ &: \ (r \stackrel{R}{\leftarrow} E \wedge (\mathcal{A} \triangleleft G, H, \mathcal{D}, f, m_0, m_1, a = f(r), b = m \oplus G(r), c = H(m, r)) \\
&\quad :< \mathcal{A} \rightarrow \delta, v > \wedge (t(\mathcal{A}) \leq T))
\end{aligned}$$

Proposition 7

$$\mathsf{val}(\mathsf{Valide}(C = a||b||c) \wedge \neg(\mathsf{List}_H(b \oplus G(g(a)), g(a))) \leq 1/|\mathsf{Ec}|$$

Preuve. Supposons que : $\neg(\mathsf{List}_H(b \oplus G(g(a)), g(a)))$.
D'après (3.2), on a : $c' = H(b \oplus G(g(a)), g(a)) \stackrel{R}{\leftarrow} \mathsf{Ec}$.
D'après le postulat R3 : $\mathsf{val}(c' = c) = 1/|\mathsf{Ec}|$.

□

Proposition 8

$$\forall a : \mathsf{val}(\neg(\mathsf{List}_G(g(a))) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a))) \leq q_H/|\mathsf{Em}|$$

Preuve. Supposons que : $\neg(\mathsf{List}_G(g(a))$
D'après (3.1), on a : $c = G(g(a)) \stackrel{R}{\leftarrow} \mathsf{Em}$
alors, quand on considère $f(c) = b \oplus c_0$ comme une fonction de c, on a $\mathsf{Unif}(f, c)$, avec $\mathsf{Im}(f) = \mathsf{Em}$. D'après le postulat R7 on peut remplacer $b \oplus c$ par $c' : c' \stackrel{R}{\leftarrow} \mathsf{Em}$: D'après le postulat R3 : $\forall c \in \mathsf{Em} : \mathsf{val}(c' = c) = 1/|\mathsf{Em}|$
Par conséquent, si List_H contient q_H éléments : $\mathsf{val}(\mathsf{List}_H(c' = b \oplus G(g(a)), g(a))) \leq q_H/|\mathsf{Em}|$
D'où, le résultat.

□

Maintenant, on simule l'oracle de déchiffrement $\mathcal{D}$ par List_G et List_H.

Proposition 9 $v = \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^m) \wedge (\mathcal{A} \rightarrow x))$,
alors : $v' = \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^{m-1}, \mathsf{List}_G, \mathsf{List}_H) \wedge (\mathcal{A} \rightarrow x)) \wedge (\delta(t) = 0)$,*avec* $|v - v'| \leq \frac{1}{|\mathsf{Ec}|} + \frac{q_H}{|\mathsf{Em}|}$,
où x est une cible quelconque que $\mathcal{A}$ veut calculer et $\mathcal{D}^m$ signifie qu'on peut poser à m question à l'oracle de déchiffrement $\mathcal{D}$.

Preuve. $(\mathcal{A} \triangleleft \mathcal{D}^m) \wedge < \mathcal{A} \to x, v >$ signifie : $\exists C : \mathcal{A} \xrightarrow{?} \mathcal{D}(C) \wedge (\mathcal{A} \triangleleft \mathcal{D}^{m-1}) \wedge < \mathcal{A} \to x, v >$. On note $\mathcal{D}(C)$ la valeur que l'oracle $\mathcal{D}$ retourne sur le requête $C = a||b||c$, si C est invalide, on convient $\mathcal{D}(C) = 0$.
On considère deux cas :

1. $\mathsf{List}_G(g(a)) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a))$:
   Dans ce cas $\mathcal{D}(C) = \mathsf{List}_H(b \oplus \mathsf{List}_G(g(a)), g(a))$, alors :

   $$\mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^m) \wedge (\mathcal{A} \to x)) = \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^{m-1}, \mathsf{List}_G, \mathsf{List}_H) \wedge (\mathcal{A} \to x)) \wedge (\delta(t) = 0)$$

   Sous la condition considérée, on note :

   $$\begin{aligned} v_1 &= \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^m) \wedge (\mathcal{A} \to x)), \\ v_1' &= \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^{m-1}, \mathsf{List}_G, \mathsf{List}_H) \wedge (\mathcal{A} \to x)), \end{aligned}$$

   on a :

   $$v_1 = v_1'. \tag{3.3}$$

2. $\neg(\mathsf{List}_G(g(a)) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a)))$:
   Dans ce cas, on a toujours : $\mathsf{List}_H(b \oplus \mathsf{List}_G(g(a)), g(a)) = 0$ (ou bien la simulation rejette le chiffré C). Alors, on divise cette situation en deux cas :
   — $\neg(\mathsf{Valide}(C))$: alors $\mathcal{D}(C) = 0$ et par conséquent :

   $$\mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^m) \wedge (\mathcal{A} \to x)) = \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^{m-1}, \mathsf{List}_G, \mathsf{List}_H) \wedge (\mathcal{A} \to x)) \wedge (\delta(t) = 0)$$

   Sous la condition considérée, on note :

   $$\begin{aligned} v_{21} &= \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^m) \wedge (\mathcal{A} \to x)), \\ v_{21}' &= \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^{m-1}, \mathsf{List}_G, \mathsf{List}_H) \wedge (\mathcal{A} \to x)) \wedge, \end{aligned}$$

   on a :

   $$v_{21} = v_{21}'. \tag{3.4}$$

   — $(\mathsf{Valide}(C))$: On a alors : $\neg(\mathsf{List}_G(g(a)) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a))) \wedge (\mathsf{Valide}(C))$ ou bien :

   $$\begin{aligned} & (\neg(\mathsf{List}_H(b \oplus G(g(a)), g(a)) \wedge (\mathsf{Valide}(C))) \\ \vee \quad & (\mathsf{List}_H(b \oplus G(g(a)), g(a)) \wedge \neg(\mathsf{List}_G(g(a)) \wedge (\mathsf{Valide}(C))) \end{aligned}$$

   alors, utilisant les proposition (7) et (8), on a :

   $$\mathsf{val}(\mathsf{Valide}(C) \leq \frac{1}{|\mathsf{Ec}|} + \frac{q_H}{|\mathsf{Em}|} \tag{3.5}$$

   Dans ce cas, on a toujours :

   $$\begin{aligned} v_{22} &= \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^m) \wedge (\mathcal{A} \to x)) \leq 1 \\ v_{22}' &= \mathsf{val}((\mathcal{A} \triangleleft \mathcal{D}^{m-1}, \mathsf{List}_G, \mathsf{List}_H) \wedge (\mathcal{A} \to x)) \geq 0 \end{aligned}$$

   D'après le postulat R4 :

   $$\begin{aligned} v_2 &= v_{21} \cdot (1 - \mathsf{val}(\mathsf{Valide}(C))) + v_{22} \cdot (\mathsf{val}(\mathsf{Valide}(C)) \\ v_2' &= v_{21}' \cdot (1 - \mathsf{val}(\mathsf{Valide}(C))) + v_{22}' \cdot (\mathsf{val}(\mathsf{Valide}(C)) \end{aligned}$$

   Des équations (3.4) et (3.5), on a :

   $$v_2 - v_2' \leq \frac{1}{|\mathsf{Ec}|} + \frac{q_H}{|\mathsf{Em}|} \tag{3.6}$$

D'après le postulat R4 :

$$\begin{aligned} v &= v_1 \cdot (1 - \mathsf{val}(\mathsf{List}_G(g(a)) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a)))) \\ &\quad + v_2 \cdot (\mathsf{val}(\mathsf{List}_G(g(a)) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a)))) \\ v' &= v_1 \cdot (1 - \mathsf{val}(\mathsf{List}_G(g(a)) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a)))) \\ &\quad + v'_2 \cdot (\mathsf{val}(\mathsf{List}_G(g(a)) \wedge \mathsf{List}_H(b \oplus G(g(a)), g(a)))) \end{aligned}$$

En utilisant les équations (3.3) et (3.6), on déduit :

$$v - v' \leq v_2 - v'_2 \leq \frac{1}{|\mathsf{Ec}|} + \frac{q_H}{|\mathsf{Em}|},$$

d'où le résultat. □

On applique la proposition 9 q_D fois, on obtient :

$$\begin{aligned} \exists m_0, m_1 \in \mathcal{G} \ : \ & (r \xleftarrow{R} E \wedge (\mathcal{A} \triangleleft \mathsf{List}_G, , \mathsf{List}_H, f, m_0, m_1, a, b, c) \\ & :< \mathcal{A} \to \delta, v' > \wedge (t(\mathcal{A}) \leq T)), \end{aligned}$$

avec

$$v - v' \leq q_D \cdot (1/|\mathsf{Ec}| + q_H \cdot 1/|\mathsf{Em}|) \tag{3.7}$$

ou bien :

$$\begin{aligned} \exists m_0, m_1 \in \mathcal{G} \ : \ & a \xleftarrow{R} E \wedge (\mathcal{A} \triangleleft \mathsf{List}_G, \mathsf{List}_H, f, m_0, m_1, a, b = m_\delta \oplus G(g(a)), c = H(m, g(a))) \\ & :< \mathcal{A} \to \delta, v' > \wedge (t(\mathcal{A}) \leq T). \end{aligned}$$

— $\mathsf{List}_G(g(a)) \vee \mathsf{List}_H(m, g(a))$:
On a toujours :

$$< \mathcal{A} \to \delta, v_1 \leq 1 > \wedge (t(\mathcal{A}) = 0) \tag{3.8}$$

D'après le postulat R9, $\mathcal{A}$ peut calculer $g(a)$ comme suit :

$$\begin{aligned} \Big(\mathcal{A} \to (r = \sum_{x:\mathsf{List}_G(x) \vee \mathsf{List}_H(m,x)} ((f(x) = a)) \cdot x)\Big) \ & \wedge \ < r = g(a), 1 > \\ & \wedge \ (\delta(t) = (q_G + q_H) \cdot T_f) \end{aligned}$$

Alors, on déduit : $\mathsf{val}(\mathsf{List}_G(g(a))) \leq \mathsf{val}(\mathsf{OW}(f, T' = T + (q_G + q_H) \cdot T_f))$

— $\neg(\mathsf{List}_G(g(a))) \wedge \neg(\mathsf{List}_H(m, g(a)))$:
$g(a)$ n'a pas été demandé à G et $(m, g(a))$ n'a pas été non plus demandé à H.
De (3.1) et (3.2), on peut remplacer $\mathsf{List}_G(g(a))$ et $\mathsf{List}_H(m, g(a))$ par des aléas :

$$\begin{aligned} \exists m_0, m_1 \in \mathcal{G} \ : \ & (r \xleftarrow{R} E) \wedge (g^+ \xleftarrow{R} \mathsf{Em}) \wedge (h^+ \xleftarrow{R} \mathsf{Ec}) \\ & \wedge (\mathcal{A} \triangleleft \mathsf{List}_G, \mathsf{List}_H, f, m_0, m_1, a = f(r), b = m_\delta \oplus g^+, h^+) \\ & :< \mathcal{A} \to \delta, v_2 > \wedge (t(\mathcal{A}) \leq T)) \end{aligned}$$

Puisque $g^+ \xleftarrow{R} \mathsf{Em}$, quand on considère $b = m_\delta \oplus g^+$ comme une fonction de g^+, on a $\mathsf{Unif}(m_\delta \oplus g^+, g^+)$, avec $\mathsf{Im}(b) = \mathcal{G}$. D'après le postulat R7, on peut remplacer $m_\delta \oplus g^+$ par $u \xleftarrow{R} \mathsf{Em}$. Alors :

$$\begin{aligned} \exists m_0, m_1 \in \mathcal{G} \ : \ & (r \xleftarrow{R} E \wedge u \xleftarrow{R} \mathsf{Em} \wedge h^+ \xleftarrow{R} \mathsf{Ec} \\ & \wedge (\mathcal{A} \triangleleft \mathsf{List}_G, \mathsf{List}_H, f, m_0, m_1, a = f(r), b = u, h^+) \\ & :< \mathcal{A} \to \delta, v_2 > \wedge (t(\mathcal{A}) \leq T)) \end{aligned}$$

D'après le postulat R8, du fait que $\delta \notin \mathsf{Var}(U)$ (où U représente tout ce que $\mathcal{A}$ peut voir dans l'équation ci-dessus) avec $f = (\delta' = \delta)$, on déduit : $\mathsf{val}(\delta' = \delta) = 1/2$ qui entraîne, en utilisant le postulat R2 : $\mathsf{val}(\mathcal{A} \to \delta) = 1/2$
Alors :

$$v_2 = 1/2 \tag{3.9}$$

De (3.8), (3.9), d'après le postulat R4, on déduit :

$$< \mathcal{A} \to \delta, v' = v_1 \cdot \mathsf{val}(\mathsf{OW}(f)) + (1 - \mathsf{val}(\mathsf{OW}(f, T + (q_G + q_H) \cdot T_f))) \cdot (1/2) > \wedge (t(\mathcal{A}) \leq T))$$

Alors :

$$\begin{aligned} v' &\leq 1 \cdot \mathsf{val}(\mathsf{OW}(f)) + (1 - \mathsf{val}(\mathsf{OW}(f, T + (q_G + q_H) \cdot T_f))) \cdot (1/2) \\ &\leq 1/2 + \mathsf{val}(\mathsf{OW}(f, T + (q_G + q_H) \cdot T_f) \end{aligned}$$

En utilisant l'équation (3.7), on déduit :

$$\begin{aligned} & \mathsf{val}(\mathsf{IND} - \mathsf{CCA2}(\mathcal{A}, \mathsf{ProtCons}, T)) \\ \leq\ & 1/2 + \mathsf{val}(\mathsf{OW}(\mathcal{A}, f, \mathsf{ProtCons}, T + (q_G + q_H) \cdot T_f) + q_D \cdot (1/|\mathsf{Ec}| + q_H \cdot 1/|\mathsf{Em}|) \end{aligned}$$

D'où le résultat.

□

Avec l'hypothèse que f est une permutation à sens-unique à trappe, on déduit la sécurité sémantique de cette construction selon des attaques à chiffrés choisis adaptatives.

Conclusion

Dans ce rapport, nous avons considéré deux méthodes d'analyse d'un protocole : méthode formelle avec la logique BAN et méthode d'analyse par réduction. On a aussi proposé une extension de la logique BAN comme un pont entre ces deux méthodes : elle permet de mener des analyses formelles similaires à celles usuellement faites avec les méthodes issues de la théorie de la complexité. Dans le cadre du stage, nous avons montré que cette méthode peut prouver la sécurité des même protocoles que la méthode par réduction. Mais pour examiner vraiment l'efficacité de cette méthode, il faut analyser d'autres aspects comme les propriétés (consistance, complétude) de cette logique et, surtout, la capacité de cette méthode à produire des preuves automatiques.

La question des preuves automatiques se pose pour les méthodes formelles. Pour cela, il faut construire des algorithmes de déduction dont la complexité soit acceptable. Pour la logique BAN, on peut construire un algorithme qui, à chaque étape, essaie d'appliquer toutes les règles possibles jusqu'à trouver la conclusion. Le coût de cet algorithme peut devenir très élevé. Pour la logique BAN étendue, excepté la règle calculatoire R9, toutes les autres règles considèrent les variables dans le problème, donc, le nombre de possibilités d'application des règles à chaque pas est fini et on peut ainsi construire un algorithme exhautif de déduction. Et comme on le voit avec la preuve du schéma d'ElGamal, la hauteur de trace de déduction est bornée, donc le nombre des règles appliquées pour obtenir la conclusion est fini et alors, théoriquement, l'algorithme s'arrête. Quand on ajoute la règle R9, le nombre de règles appliquées peut devenir très grand puisque l'on applique cette règle non sur les variables mais sur des valeurs du domaine de la fonction f, et comme le domaine d'une fonction dans un schéma est normalement très grand, il se peut que l'on ne puisse pas construire un algorithme de déduction. Enfin, les méthodes formelles considérées contiennent la logique propositionnelle et on sait que pour la logique propositionelle, le problème de déduction est NP-complet. Donc, il est difficile de trouver des algorithmes qui utilisent des règles de la logique BAN ou la logique BAN étendue pour raisonner automatiquement et efficacement. Cependant, chaque protocole cryptographique a des propriétés particulières qui entraînent peut-être des restrictions sur les espaces de recherche, donc, peut-être est il possible de trouver les algorithmes efficaces pour certaines familles de protocoles. Dans les recherches qui vont suivre, nous espérons avoir des réponses exactes dans certains cas à cette question très intéressante et concrète.

Même si les preuves automatiques efficaces sont difficiles à construire, on peut espérer obtenir les techniques de preuves semi-automatiques ou bien, on peut espérer appliquer ces méthodes formelles pour la phase de vérification.

Bibliographie

[1] M. Bellare and Rogaway. Random Oracles Are Practical : a Paradigm for Designing Efficient Protocol. In *Proc. of the 1st CCS*, pages 62–73, New York, 1993. ACM Press.

[2] O. Baudron, D. Pointcheval, and J. Stern. Extended Notions of Security for Multicast Public Key Cryptosystem. In *Proc. of the 27th ICALP*, volume LNCS 1853, pages 499–511. Springer-Verlag, 2000.

[3] M. Bellare, A. Boldyreva, and S. Micali. Public-key Encryption in a Multi-user Setting Security Proofs and Improvement. In *Proc. of CRYPTO '98*, pages 259–274, Berlin, 2000. Springer-Verlag.

[4] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among Notions of Security for Public-key Encryption Schemes. In *Proc. of CRYPTO '98*, volume LNCS 1462, pages 26–45, Berlin, 1998.

[5] M. Bellare and P. Rogaway. Entity Authentication and Key Distribution. In D. Stinson, editor, *Advances in Cryptology - Crypto 93 Proceedings*, volume LNCS 773, pages 232–249. Springer-Verlag, 1994.

[6] M. Burrows, M. Abadi, and R. Needham. A Logic of Authentication. *Proc. Cambridge Phil. Soc.*, 60 :699–700, 1989.

[7] D. Dolev, C. Dwork, and M. Naor. Non-Malleable Cryptography. *SIAM Journal on Computing*, 2000.

[8] T. ElGamal. A Public-key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Transactions on Information Theory*, IT-31(4) :469–472, 1985.

[9] S. Goldwasser and S. Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28 :270–299, 1984.

[10] W. Mao and C. Boyd. On a Limitation of BAN Logic. In *Advances in Cryptology - Proceedings of EUROCRYPT 93*, pages 240–247. Springer-Verlag, 1993.

[11] W. Mao and C. Boyd. Towards a Formal Analysis of Security Protocols. In *Proceedings of the Computer Security Foundations Workshop VI*, pages 147–158. IEEE Computer Society Press, 1993.

[12] M. Naor and M. Yung. Public-key Cryptosystems Provably Secure against Chosen Ciphertext Attack. In *Proc. of the 22nd ACM STOC*, pages 427–437, NewYork, 1990. ACM Press.

[13] D. Otway and O. Rees. Efficient and Timely Mutual Authentication. *Operating Systems Review*, 21(1) :8–10, 1987.

[14] D. Pointcheval. *Le chiffrement asymétrique et la sécurité prouvée*. Thèse d'habilitation, université de Paris VII, juin 2002.

[15] C. Rackoff and D. R. Simon. Non-interactive Zero-knowledge Proof of Knowledge and Chosen Ciphertext Attack. In *Proc. of CRYPTO '91*, volume LNCS 576, pages 433–444, Berlin, 1992. Springer-Verlag.