TP6 : Analyse formelle de protocoles (introduction)

L'objectif de ce TP est de se familiariser avec le logiciel AVISPA disponible à l'adresse :

`http://www.avispa-project.org/`

1. Installez le logiciel.
2. Lisez les exemples de modélisation dans `avispa-1.1/testsuite/hlpsl`. Commencez par un modèle "simple" comme `EKE.hlpsl`.
3. Exécutez les exemples.
4. Modifiez les conditions vérifiées, le modèle, ...
5. Prenez un protocole considéré dans le cours (par exemple Woo-Lam) et modélisez-le dans l'outil.
6. Pouvez-vous retrouver l'attaque décrite dans le cours ? Y-a-t-il d'autres attaques possibles ?

TP7 : Analyse formelle de protocoles (problèmes)

L'objectif de ce travail est d'analyser différentes variantes d'un protocole dans le modèle de Dolev-Yao. On se place donc dans le cadre où les fonctions cryptographiques sont 'parfaitement sûres' et l'adversaire a un contrôle total du réseaux.

Le protocole doit permettre à deux participants A et B d'établir une nouvelle *information secrète* (une clé de session ou un nonce). Pour atteindre ce but A et B peuvent se servir d'un tiers de confiance, c'est-à-dire, d'un serveur S.

A la fin du protocole, les propriétés suivantes devraient être satisfaites (en supposant que A et B sont deux participants honnêtes) :

1. Seulement A et B (et éventuellement le serveur S) connaissent l'information secrète.
2. A (resp. B) est sûr que l'information secrète est partagée avec B (resp. A).
3. L'information secrète est nouvelle, c'est-à-dire elle ne provient pas d'une session précédente.

Le protocole doit assurer ces propriétés même s'il est répété plusieurs fois, éventuellement en parallèle, avec des rôles échangés et avec des participants malhonnêtes.

On présente un certain nombre de protocoles qui sont censés 'résoudre' le problème. Vous pouvez analyser ces protocoles avec 'papier et crayon' et à l'aide du logiciel AVISPA.

Pour chaque protocole vous aurez à :

1. Modéliser chaque rôle du protocole dans le langage HLPSL d'AVISPA.
2. Spécifier les propriétés souhaitées à l'aide d'annotations des rôles.
3. Vérifier si ces propriétés sont satisfaites en modulant la taille des paramètres (nombre de participants, nombre de sessions,...).
4. Donner une brève explication des attaques possibles.

Protocole 1

Hypothèse Chaque participant A partage avec le serveur S une clé K_{AS}.

Objectif Deux participants A, B veulent partager une nouvelle clé K avec l'aide de S.

Déroulement standard du protocole

1. A envoie à S : $\langle A, B \rangle$.
2. S détermine K_{AS} et K_{BS}, génère une nouvelle clé K, et envoie à A : $\langle \{K\}_{K_{AS}}, \{K\}_{K_{BS}} \rangle$.
3. A détermine K et envoie à B : $\langle S, A, \{K\}_{K_{BS}} \rangle$.
4. B détermine K et envoie à A un message secret chiffré avec K.

Protocole 2

Même hypothèse et objectif que dans le protocole 1 mais le premier pas est modifié de la façon suivante :

1. A envoie à S : $\langle A, \{B\}_{K_{AS}} \rangle$.

Protocole 3

Même hypothèse et objectif que dans le protocole 1.

Déroulement standard du protocole

1. A envoie à S $\langle A, B\rangle$.
2. S détermine K_{AS}, K_{BS}, génère une clé K, et envoie à A $\langle \{B, K\}_{K_{AS}}, \{A, K\}_{K_{BS}}\rangle$.
3. A reçoit le message $\langle \{B, K\}_{K_{AS}}, \{A, K\}_{K_{BS}}\rangle$, obtient B, K et envoie à B $\langle S, \{A, K\}_{K_{BS}}\rangle$.
4. B reçoit $\langle S, \{A, K\}_{K_{BS}}\rangle$, dérive A, K et envoie à A un message secret chiffré avec K.

Protocole 4

Même hypothèse et objectif que dans le protocole 1.

Déroulement standard du protocole

1. A génère un nonce N_A et envoie à S : $\langle A, B, N_A\rangle$.
2. S génère un clé K et envoie à A : $\{N_A, K, B, \{K, A\}_{K_{BS}}\}_{K_{AS}}$.
3. A déchiffre, vérifie le nonce N_A et l'identité B et envoie à B : $\langle S, \{K, A\}_{K_{BS}}\rangle$.
4. B déchiffre, vérifie l'identité A, génère un nonce N_B et envoie à A : $\{m, N_B\}_K$, où m est un message secret.
5. A réplique en envoyant à B : $\{m', N_B, N_B\}_K$, où m' est un autre message secret.

Protocole 5 (optionnel)

Hypothèse Dans ce protocole on utilise la cryptographie publique. Chaque participant A a une clé publique K_A et une clé privée K_A^{-1}. De même le serveur S a une clé publique K_S et une clé privée K_S^{-1}.

Objectif Deux participants A, B veulent partager un nouveau secret N_B avec l'aide de S.

Déroulement standard du protocole

1. A envoie à S : $\langle A, B\rangle$.
2. S envoie à A : $\{K_B, B\}_{K_S^{-1}}$.
3. A vérifie la signature, génère un nonce N_A et envoie à B : $\{N_A, A\}_{K_B}$.
4. B déchiffre, vérifie l'identité de A et envoie à S : $\langle B, A\rangle$.
5. S envoie à B : $\{K_A, A\}_{K_S^{-1}}$.
6. B vérifie la signature, génère un nonce N_B et envoie à A : $\{N_A, N_B\}_{K_A}$.
7. A déchiffre et envoie à B : $\{N_B\}_{K_B}$.