
La méthode TAU pour l'échange de clef authentifié

O. Chevassut, P.-A. Fouque, P. Gaudry, D. Pointcheval

ACI Sécurité – Cesam

Les courbes elliptiques pour la sécurité des appareils mobiles

ENS – Ecole polytechnique

CNRS – INRIA

Lawrence Berkeley National Lab

Rappel : objectifs de Cesam

- En cryptographie asymétrique, on s'appuie sur des problèmes algorithmiques réputés difficiles : factorisation, décodage de codes aléatoires, **logarithme discret**, ...
- Les protocoles cryptographiques qui s'appuient sur LD sont conçus et analysés pour un groupe quelconque.
- En pratique, les **courbes elliptiques** fournissent de très bons groupes. En particulier pour les milieux contraints (calculs, bande passante du réseau).

Objectif : Revisiter les protocoles classiques en remplaçant les groupes génériques par des courbes elliptiques. On peut parfois **simplifier** le protocole, sans casser les preuves de sécurité.

Autres objectifs de Cesam

- Étudier et améliorer les protocoles à base de **mots de passe**.
 - Très adaptés au monde réel : secret très court (un cerveau, c'est nul !)
 - Sécurité : résistance aux attaques passives. De plus, chaque attaque active ne permet de tester qu'un seul mot de passe.
- Rendre plus asymétrique les protocoles : les calculs sont fait essentiellement par le serveur ; la charge du client (mobile) est allégée.

Échange de clef Diffie–Hellman

$G = \langle g \rangle$ groupe cyclique d'ordre n .

Alice	réseau	Bob
choisit $x_A \in [1, n]$ calcule $K_A = g^{x_A}$	$\xrightarrow{K_A}$	choisit $x_B \in [1, n]$ calcule $K_B = g^{x_B}$
calcule $K_{AB} = K_B^{x_A}$	$\xleftarrow{K_B}$	calcule $K_{AB} = K_A^{x_B}$

$$K_{AB} = g^{x_A x_B}$$

La clef K_{AB} est utilisée comme **clef de session** dans un algorithme de chiffrement symétrique.

Échange de clef authentifié

- Lutte contre les attaques «**Man in the middle**» : on suppose qu'Alice et Bob disposent de clefs publiques dans une PKI. Alors on peut authentifier les échanges (essentiellement en signant les flux de données).
- Authentification mutuelle (optionnel) : on rajoute deux passes pour être sûr que l'échange de clef s'est bien passé.

Exemple : protocole SIGMA dans IPSEC (Internet Key Exchange).

Problème de la dérivation de clef

Dérivation de clef : Déduire la clef de session K_s (chaîne de bits) à partir de l'élément du groupe K_{AB} .

- En cryptographie symétrique, tous les bits de K_s doivent être vraiment aléatoires !
- K_{AB} est un élément aléatoire du groupe, représenté par une chaîne de bits (pas tous aléatoires...)
- Comment dériver une clef de session K_s à partir de K ?

Exemple de difficulté

Si G est le groupe multiplicatif de \mathbb{F}_p pour un p premier.

L'hypothèse Decisional Diffie Hellman (DDH) dit que (g^x, g^y, g^{xy}) et (g^x, g^y, g^z) sont indistingables.

Mais (g^x, g^y, g^{xy}) et (g^x, g^y, R) (pour un chaîne de bits aléatoires R) ne sont pas indistingables :

Parfois le symbole de Legendre peut suffir pour distinguer.
De plus, R ne correspond pas forcément à la représentation d'un élément de G .

- Fonction de hachage classique
On calcule K_s comme image de K_{AB} par une fonction de hachage cryptographiquement sûre. Cela signifie que l'on aura une preuve seulement sous l'hypothèse de l'**oracle aléatoire**.
De plus, cela rajoute une opération.
- **Left-over-Hash Lemma**
On calcule K_s à partir de K_{AB} à l'aide d'une fonction de hachage universelle et de bits aléatoires.
Solution élégante et générique, mais on extrait peu d'entropie : si on a en entrée une entropie de s bits (240), on extrait $k = s - 2e$ bits (80) avec un biais 2^{-e} (2^{-80})

- Cas particulier des **courbes elliptiques** : on espère mieux !
 - On connaît les propriétés de l'encodage des éléments du groupe.
 - On utilise une courbe et sa **Twist Quadratique** pour obtenir une extraction de bits d'entropie très efficace.
⇒ méthode TAU

Courbes elliptiques et twists quadratiques – 1

Soit \mathbb{F}_p le corps fini à p éléments (p fait 200 bits).

Def Une **courbe elliptique** sur \mathbb{F}_p est

$$\mathbb{E}_{a,b} = \{(x, y) \in \mathbb{F}_p \times \mathbb{F}_p : y^2 = x^3 + ax + b\} \cup \{\infty_{\mathbb{E}}\}$$

Bien connu : on peut « additionner » deux points de $\mathbb{E}_{a,b}$ (construction géométrique). **Loi de groupe** sur $\mathbb{E}_{a,b}$.

Soit c un non-résidu quadratique de \mathbb{F}_p .

Def La **twist quadratique** de $\mathbb{E}_{a,b}$ est

$$\tilde{\mathbb{E}}_{a,b} = \{(x, y) \in \mathbb{F}_p \times \mathbb{F}_p : cy^2 = x^3 + ax + b\} \cup \{\infty_{\tilde{\mathbb{E}}}\}.$$

Courbes elliptiques et twists quadratiques – 2

Remarques :

- La courbe twistée est elle-même une courbe elliptique sur \mathbb{F}_p (changement de variables : $x' = cx$ et $y' = c^2y$).
- Il y a donc aussi une loi de groupe sur $\tilde{\mathbb{E}}_{a,b}$.
- Soit x un élément de \mathbb{F}_p . Alors
 - ou bien x est l'abscisse d'un point de $\mathbb{E}_{a,b}$,
 - ou bien x est l'abscisse d'un point de $\tilde{\mathbb{E}}_{a,b}$.

En effet, si $x^3 + ax + b$ est un carré, alors on a un point sur $\mathbb{E}_{a,b}$, sinon $c(x^3 + ax + b)$ est un carré, et on a un point sur $\tilde{\mathbb{E}}_{a,b}$.

Protocole d'échange de clef TAU

TAU : Twist-AUGmented key exchange protocol.

- **En deux mots.** On fait tourner deux protocoles Diffie–Hellman en parallèle : l'un sur $\mathbb{E}_{a,b}$, l'autre sur $\tilde{\mathbb{E}}_{a,b}$.
- Alice envoie sa contribution sur $\mathbb{E}_{a,b}$ et sur $\tilde{\mathbb{E}}_{a,b}$.
- Bob en choisit une au hasard, y ajoute sa contribution et en déduit K_{AB} . Il renvoie à Alice sa contribution ainsi qu'une contribution factice sur l'autre courbe.
- Alice reconnaît la bonne et déduit K_{AB} .

Propriété fondamentale : l'abscisse de K_{AB} est un élément complètement aléatoire de \mathbb{F}_p . Si $p \approx 2^\ell$, le stockage traditionnel de l'abscisse de K_{AB} donne directement ℓ bits (presque) aléatoires (sans fonction de hachage).

- Possibilité de **prouver** la sécurité sans utiliser l'hypothèse de l'oracle aléatoire.
La sécurité repose seulement sur la difficulté du Decisional Diffie Hellman.
- Plus efficace que d'utiliser le Left-over hash lemma.

Rem : Dans les protocoles habituels (SIGMA), la preuve est incomplète, ou repose sur des hypothèses non-standards.

Thm (Hasse) Il existe un entier $t_{\mathbb{E}}$ tel que

$$\#\mathbb{E} = p + 1 - t_{\mathbb{E}}$$

$$\#\tilde{\mathbb{E}} = p + 1 + t_{\mathbb{E}}$$

$$|t_{\mathbb{E}}| \leq 2\sqrt{p}.$$

Deux possibilités pour calculer exactement $t_{\mathbb{E}}$:

- Calcul direct par l'algorithme de Schoof-Elkies-Atkin (SEA).
- Théorie de la «[multiplication complexe](#)» qui permet de construire simultanément \mathbb{E} et $t_{\mathbb{E}}$. (courbes particulières)

Construction de paramètres

Théorème Chinois :

Pour que le problème DDH soit difficile sur \mathbb{E} et $\tilde{\mathbb{E}}$, il faut que leur nombre de points soit premier.

On tire des courbes au hasard, et on compte les points jusqu'à ce que ces deux nombres soient premiers (probabilité $\approx \log(p)^{-2}$).

Amélioration : Dans l'algorithme SEA, on connaît très tôt la valeur de $t_{\mathbb{E}}$ modulo 2, 3, 5, 7, ...

Construire une bonne courbe prend environ 3 heures sur un PC.

- La méthode TAU permet de réparer des trous dans les preuves de sécurité de SIGMA et autres ;
 - La preuve ne fait pas intervenir l'hypothèse de l'oracle aléatoire ;
 - Plus efficace que la méthode générique «Left-over hash lemma».
- ⇒ Réétudier le protocole en instanciant le groupe permet d'améliorer l'efficacité et la sécurité.