

CESAM : Les Courbes Elliptiques pour la Sécurité des Appareils Mobiles

LIENS-ENS TANC-INRIA

26 avril 2005

<http://www.di.ens.fr/users/pointche/Projets/CESAM/>

Résumé

Les appareils et supports mobiles ont des capacités réduites, et pourtant, les opérations qu'ils effectuent ou les données qu'ils transportent sont souvent très sensibles. On pense bien sûr aux multiples cartes à puce qui gèrent nos transactions financières, ou stockent des données médicales, identifient et protègent nos communications téléphoniques, les assistants personnels numériques (ou PDA) et clés USB, mais on parle aussi de plus en plus de *sensor networks*, où des capteurs transmettent des données diverses à un serveur. La sécurité est rarement l'objectif premier de ces appareils, mais elle est cependant cruciale. Malheureusement, avec la technologie répandue actuellement, une bonne sécurité est coûteuse en ressources (stockage, calculs, alimentation, etc.). En effet, les principaux systèmes cryptographiques existants reposent sur RSA, ou tout du moins le calcul sur de très grands nombres. La cryptographie sur les courbes elliptiques a été vue comme une solution à ces problèmes de ressources, mais leur mise en œuvre pratique semble poser problème. Nous proposons donc de combiner les compétences d'une équipe spécialisée en protocoles cryptographiques et une autre spécialisée en la cryptographie avec les courbes elliptiques pour faire avancer une "cryptographie efficace" sur courbes elliptiques. Les courbes elliptiques nous permettent d'ailleurs d'espérer une "efficacité" à facettes multiples : temps de calcul, temps de communication et/ou stockage (en raison de données à transmettre et/ou stocker de petite taille), tout cela conduisant à une consommation d'énergie minimale.

1 Participants

Ce projet est coordonné par David Pointcheval (CR1 – CNRS – LIENS), et est effectué par les deux équipes

- GRECC – LIENS – Laboratoire d'informatique de l'École normale supérieure
 - Michel Abdalla, post-doctorant – LIENS
(financement CNRS d'août 2003 à juillet 2004, puis ENS depuis août 2004)
 - Dario Catalano, CR2 - CNRS – LIENS
 - Pierre-Alain Fouque, MdC – ENS – LIENS
 - Sébastien Kunz-Jacques, thésard – LIENS
(en thèse depuis octobre 2003, Ingénieur Corps Télécom – DCSSI)
 - Duong Hieu Phan, thésard – LIENS
(en thèse depuis octobre 2002, soutenance prévue en septembre 2005 – allocation de recherche du MENRT)
- Cryptologie – Projet TANC – Théorie algorithmique des nombres pour la cryptologie, INRIA Futurs
 - Régis Dupont, thésard – LIX
(en thèse depuis septembre 2003, financement INRIA)

- Andreas Enge, CR2 – INRIA
- Pierrick Gaudry, CR2 – CNRS – LIX
- Nicolas Gürel, ATER Marne-la-Vallée – LIX
- Thomas Houtmann, thésard – LIX

(en thèse depuis octobre 2004, financement DGA)

en collaboration avec Olivier Chevassut, du département des systèmes distribués, au Lawrence Berkeley National Laboratory (Département d’Energie aux États-Unis).

2 Rappel des objectifs fixés

Notre sujet est la cryptographie pour les environnements contraints, avec comme support l’utilisation des courbes elliptiques. Cependant, ces courbes elliptiques peuvent être soit utilisées comme un groupe classique (avec des objets de petite taille, qui ainsi apportent une efficacité supplémentaire) ou comme un groupe spécifique avec une structure particulière qui peut être exploitée par les protocoles cryptographiques.

La cryptographie pour les environnements contraints peut donc être étudiée dans deux directions indépendantes : proposer des protocoles cryptographiques adaptés aux environnement contraints (avec ou sans courbe elliptique), puis voir les apports spécifiques des courbes elliptiques à la cryptographie.

2.1 La cryptographie pour les environnement contraints

Deux thématiques ont été privilégiées : la cryptographie basée sur l’identité et la cryptographie à base de mots de passe. La première ayant connu un nouvel essor très récemment, à cause de nouvelles propriétés de certaines courbes elliptiques, mais certains protocoles peuvent s’en passer.

2.1.1 La cryptographie basée sur l’identité

La cryptographie à clé publique traditionnelle nécessite la mise en place d’une infrastructure (PKI) pour garantir l’authenticité des clés publiques. Déjà dans les systèmes qui abondent de ressources, la mise en œuvre d’une infrastructure de certification de clés est assez lourde. Dans les environnements à fortes contraintes auxquels nous nous intéressons, elle peut s’avérer impossible à mettre en place. Les cryptosystèmes fondés sur l’identité, dans lesquels le lien entre les identités des interlocuteurs et leurs clés publiques est immédiat, se révèlent donc d’un intérêt particulier dans les appareils à faibles ressources.

2.1.2 La cryptographie à base de mots de passe

La mise en accord de clé avec authentification mutuelle (*authenticated key exchange*) est certainement la primitive la plus importante. Dans bien des cas, au final, cette authentification des correspondants repose sur le fait que les interlocuteurs partagent un secret de petite taille (quelques dizaines de bits – mot de passe [IEEE]). Ce cas de figure semble le plus adapté aux environnements à fortes contraintes (notamment pour le *e-commerce* et le *m-commerce*), et cependant, il n’est pas encore très développé.

2.2 Les courbes elliptiques

2.2.1 Le logarithme discret elliptique

Les courbes elliptiques sont des courbes très particulières au sens où les points de la courbe forment un groupe : il est possible de définir une loi d’addition qui à deux points de la courbe

associe leur somme. Lorsque l'on choisit comme corps de base un corps fini, le groupe obtenu est un groupe fini, et l'on peut envisager de l'utiliser en cryptographie. Cette possibilité a été proposée il y a près de vingt ans [M-86, K-87], et malgré les tentatives de bon nombre de chercheurs renommés, aucun algorithme n'a été trouvé qui résolve efficacement le problème du logarithme discret elliptique, si ce n'est dans des cas bien particuliers faciles à détecter. Grâce à la difficulté de ce problème, les courbes elliptiques offrent la possibilité de concevoir des cryptosystèmes asymétriques d'une compacité et d'une efficacité inégalées. Pour donner une idée, utiliser un groupe elliptique de taille environ 2^{160} confère une sécurité équivalente à celle dans un groupe classique de taille 2^{1024} . Cela signifie que les calculs à faire lors des opérations cryptographiques se font sur des entiers bien plus petits, et que les tailles des paramètres cryptographiques (clés, signatures, etc.) sont réduites d'autant.

2.2.2 Protocoles elliptiques existants : analyse des normes

Les organismes de normalisation s'intéressent aux schémas cryptographiques sur les courbes elliptiques [Ca-00, Cb-00]. Pour garantir la sécurité, une preuve de sécurité est désormais nécessaire [P-02]. Cependant, si certains schémas admettent une preuve lorsqu'ils sont décrits génériquement (groupe cyclique, point sur la courbe, etc.), il n'en est pas toujours de même en fonction du codage et de la courbe utilisés. Des exemples récents ont montré des différences entre les résultats prouvés dans un cadre général et la sécurité effective du schéma concret [SPMS-02, S-01]. Néanmoins, l'efficacité de ces schémas semble satisfaisante.

2.2.3 Optimisations des protocoles

Un de nos objectifs majeurs est l'exploitation des spécificités des courbes elliptiques pour améliorer considérablement certains protocoles cryptographiques. Notamment, dans un groupe classique, le nombre de bits pour représenter un élément est bien plus grand que ce que l'on peut espérer pour une sécurité donnée (en raison notamment des algorithmes sous-exponentiels). Afin (entre autre) d'éviter de gaspiller la bande passante, on utilise une fonction de hachage afin de concentrer l'entropie dans moins de bits. Dans le cas des courbes elliptiques, la représentation des éléments est particulièrement compacte, voire quasi-optimale puisqu'un élément d'une courbe avec 2^{160} points peut être représenté avec 161 bits (contre plus de 1000 bits lors de l'utilisation de corps finis). Nous espérons donc pouvoir éviter certaines fonctions de hachage dans les protocoles, sans compromettre la preuve de sécurité, ni sacrifier la bande passante. En pratique, cela peut entraîner un gain en terme de temps de calcul, et du point de vue théorique cela permettrait peut-être de se passer de la désormais classique hypothèse de l'oracle aléatoire [BR-94], ou du moins d'en dépendre un peu moins.

2.2.4 Génération de paramètres

En plus de l'adaptation des protocoles, une difficulté lors de l'utilisation de courbes elliptiques à la place des groupes multiplicatifs des corps finis est qu'il faut trouver des courbes dont le cardinal est premier afin de ne pas compromettre la sécurité. Deux approches sont connues : soit l'on fabrique simultanément la courbe et son nombre de points en utilisant la théorie de la multiplication complexe, soit l'on prend des courbes au hasard dont on calcule le nombre de points jusqu'à en trouver une qui convient.

Par ailleurs de nouveaux besoins sont apparus avec les nouveaux protocoles. Les premiers cryptosystèmes se fondant sur l'identité utilisaient des courbes elliptiques supersingulières, dans lesquelles les couplages sous-jacents sont effectivement calculables. En revanche, ces courbes possèdent des structures algébriques très particulières, qui font douter de leur sécurité. Des

courbes aux paramètres raisonnablement élevés sont de plus obtenues seulement en caractéristique 3, qui se prête très peu à des implantations efficaces.

References

- [BR-94] M. Bellare and P. Rogaway. Optimal Asymmetric Encryption – How to Encrypt with RSA. In *Eurocrypt '94*, LNCS 950, pages 92–111. Springer-Verlag, Berlin, 1995.
- [Ca-00] Certicom Corp. SEC 1: Elliptic curve cryptography. Technical report, Standards for Efficient Cryptography Group, September 2000. Available at http://www.secg.org/collateral/sec1_final.pdf.
- [Cb-00] Certicom Corp. SEC 2: Recommended elliptic curve domain parameters. Technical report, Standards for Efficient Cryptography Group, September 2000. Available at http://www.secg.org/collateral/sec2_final.pdf.
- [IEEE] IEEE Standard 1363.2 Study Group. Password-Based Public-Key Cryptography. Available from <http://grouper.ieee.org/groups/1363/passwdPK>.
- [K-87] N. Koblitz. Elliptic curve cryptosystems. *Math. Comp.*, 48(177):203–209, 1987.
- [MV-93] A. J. Menezes and S. A. Vanstone. Elliptic Curve Cryptosystems and their Implementation. *Journal of Cryptology*, 6:209–224, 1993.
- [M-86] V. Miller. Use of elliptic curves in cryptography. In H. Williams, editor, *Advances in Cryptology — CRYPTO '85*, LNCS 218, pages 417–426, Berlin, 1986. Springer-Verlag.
- [P-02] D. Pointcheval. *Le chiffrement asymétrique et la sécurité prouvée*. PhD thesis, université de Paris VII, mai 2002. Thèse d’habilitation.
- [S-01] V. Shoup. A Proposal for an ISO Standard for Public-Key Encryption, Décembre 2001. ISO/IEC JTC 1/SC27.
- [SPMS-02] J. Stern, D. Pointcheval, J. Malone-Lee, and N. Smart. Flaws in Applying Proof Methodologies to Signature Schemes. In *Crypto '02*, LNCS 2442, pages 93–110. Springer-Verlag, Berlin, 2002.

3 Résultats obtenus

Comme espéré, plusieurs directions ont déjà été très fructueuses. Notamment, la collaboration de nos deux équipes a permis de concrétiser l’intuition évoquée dans le projet initial quant à l’utilisation de la spécificité des courbes elliptiques dans certains protocoles cryptographiques : pour résoudre le problème de l’extraction d’entropie, et pour améliorer l’efficacité de la diffusion de contenus chiffrés avec traçage de pirates.

Nous avons également de nombreux résultats au sujet de la cryptographie à base de mots de passe, dont certains ont déjà été publiés (et notamment ceux en cours de soumission lors du précédent rapport), de nouvelles soumissions ont aussi vu le jour, ainsi que publication. Puis de nouvelles pistes, et nouveaux modèles de sécurité, sont en cours d’étude.

Au sujet de la génération de courbes elliptiques, de nouvelles classes de courbes sont à proscrire en raison de leur faiblesse.

Enfin, au sujet des normes, nous étudions la sécurité de certains protocoles de mise en accord de clé sur courbes elliptiques, puis nous participons à l'introduction de mise en accord de clé à base de mots de passe dans OpenSSL, avec la rédaction d'une RFC.

3.1 Échange de clés avec authentification à base de mots de passe

3.1.1 Cas à deux participants

Le principal protocole à base de mots de passe est EKE (pour *Encrypted Key Exchange*) qui est un simple Diffie-Hellman dont les messages sont chiffrés à l'aide du mot de passe : Alice envoie le chiffrement de g^x , tandis que Bob envoie le chiffrement de g^y , et la clé de session est dérivée de g^{xy} . Ce chiffrement avec le mot de passe garantit l'authentification des participants. En effet, intuitivement, seul un message correctement chiffré (avec le bon mot de passe) sera reconstitué par l'interlocuteur qui connaît ce même mot de passe. Dans le cas contraire, ce dernier obtient un message aléatoire, qui conduit donc à une clé aléatoire.

Cependant, si le chiffrement n'est pas bien effectué, ce protocole est faible face à une attaque par dictionnaire : si on chiffre la chaîne de bits X qui code g^x avec un schéma de chiffrement classique, lorsque l'attaquant intercepte le chiffré X' , il lui suffit de tenter le déchiffrement avec tous les mots de passe possibles, très peu conduiront à un élément g^x (d'ordre q dans \mathbb{Z}_p^* , avec p sur 1024 bits et q sur 160 bits). Avec probabilité écrasante, un seul mot de passe conviendra : le protocole est cassé.

Il faut donc utiliser un chiffrement dont l'ensemble des messages clairs est le groupe G engendré par g . C'est le résultat que nous avons prouvé formellement dans [BCP-03]. Cependant, ce résultat nécessite un modèle fort, dit du "chiffrement idéal". Non seulement nous avons besoin d'un chiffrement sur G , mais pour chaque clé de chiffrement, nous faisons l'hypothèse que la fonction de chiffrement est une permutation parfaitement aléatoire sur G (puis toutes les permutations sont indépendantes les unes des autres).

Une proposition de chiffrement, supposée conduire à un schéma sûr, et qui est de plus très efficace, est $E_{pw}(m) = m \times H(pw)$ où la fonction H retourne des éléments aléatoires dans G . Cependant, même si cette construction était conjecturée sûre, la sécurité effective était un problème ouvert. Nous l'avons formellement confirmée dans [BCP-04]. Nous avons récemment étendu cette preuve à un modèle plus fort : la "forward-secrecy" [ACP-05]. Si le mot de passe est corrompu, les clés échangées par le passé restent confidentielles.

Dans les deux articles ci-dessus [BCP-03, BCP-04], nous avons même montré qu'il était suffisant de chiffrer l'un des deux messages uniquement (et non les deux), à condition d'apporter ultérieurement la confirmation que les participants connaissaient bien la clé finale : être en mesure de calculer la clé prouve aussi bien la capacité à chiffrer qu'à déchiffrer avec le mot de passe.

La technique utilisée dans le dernier article a l'avantage d'être générique, et ne nécessite pas l'utilisation du problème Diffie-Hellman. Le calcul d'exponentielles est tout de même coûteux en pratique, même sur courbes elliptiques. Nous l'avons donc généralisée [CPP-04], et une application au problème de la résiduosit  quadratique est sans aucun doute le protocole de mise en accord de clé à base de mots de passe le plus efficace : un des participants n'a que quelques carrés modulaires à effectuer.

De plus, nous avons constaté que la technique de chiffrement $E_{pw}(m) = m \times H(pw)$ pouvait encore être améliorée, en se passant de l'oracle aléatoire : $E_{pw}(m) = m \times U^{pw}$, où U est un élément du groupe dont le logarithme discret est inconnu [AP-05a].

3.1.2 Serveur d'authentification

Nous avons également considéré le cas où les interlocuteurs ne partagent pas de mot de passe, mais ont chacun un mot de passe commun avec un serveur d'authentification. Dans un tel scénario, de nouvelles attaques sont à considérer : le serveur ne doit servir qu'à garantir l'authentification, mais ne doit pas ensuite pouvoir prendre connaissance des communications échangées entre les participants ; Bob devient un attaquant privilégié pour apprendre le mot de passe qu'Alice possède en commun avec le serveur. Nous avons tout d'abord précisé le modèle de sécurité, avec un exemple de protocole admettant une preuve formelle [AFP-05]. Puis nous avons présenté un schéma très efficace, sous de nouvelles hypothèses calculatoires (variantes du Diffie-Hellman) [AP-05b].

3.2 Échange de clés entre appareils de faible puissance

Le problème de la mise en accord de clé peut être remplacé par une distribution de clé, ceci diminue considérablement le coût de calcul du côté des clients, tandis que tous les calculs sont reportés sur le serveur. Nous avons donc proposé un schéma de distribution de clé au sein d'un groupe, où seul le serveur a de nombreux calculs à effectuer, mais les clients n'ont presque rien à calculer [BCEP-03, BCEP-04].

3.3 Extraction d'entropie

Tout en restant dans la mise en accord de clé, nous sommes parvenus à exploiter la spécificité des courbes elliptiques pour améliorer des protocoles. En effet, la plupart des protocoles de mise en accord de clés aboutissent sur un élément commun g^{xy} , qui est donc indistinguable d'un élément aléatoire g^z dans le groupe (sous l'hypothèse du Diffie-Hellman décisionnel). Or, par la suite, ce secret commun sera utilisé comme clé de chiffrement symétrique : une chaîne de bits parfaitement aléatoire est nécessaire. Cependant, la chaîne de bits qui représente g^z est loin d'être aléatoire : dans le cas d'un sous-groupe de \mathbb{Z}_p^* , avec p sur 1024 bits, d'ordre q sur 160 bits, c'est indéniable ; dans le cas d'une courbe elliptique sur \mathbb{Z}_p^* , avec p sur 160 bits, d'ordre q sur 160 bits, environ une chaîne sur deux apparaît, et il est facile de savoir lesquelles. Ainsi, sans outils supplémentaires (tel que le "left-over hash lemma", mais qui nécessite des aléas certifiés) il n'est pas possible d'extraire, d'un élément aléatoire dans le groupe G , une chaîne de bits uniformément distribuée (ou proche de la distribution uniforme). En pratique, la seule méthode est alors d'utiliser le modèle de l'oracle aléatoire.

Dans IKE (Internet Key Exchange), normalisé par l'IETF, une famille de fonctions pseudo-aléatoires est utilisée pour faire office d'extracteur d'entropie, ce qui n'est pas correct en général. Nous avons mis en évidence un certain nombre de problèmes théoriques à ce sujet dans IKE. Puis nous avons proposé une solution originale : un extracteur déterministe (TAU) qui utilise une spécificité des courbes elliptiques, et notamment le twist quadratique [CFGP-05]. La génération des courbes spécifiques pour cet usage a également été étudiée.

Une autre piste pour l'extraction d'entropie a été explorée. Si on considère une courbe elliptique définie sur une extension de degré 2 d'un corps premier, alors des bornes adéquates peuvent être prouvées. Le résultat est le suivant [G-05] : à partir des coordonnées d'un point qui contient $2 \log p$ bits d'entropie cachés au sein de $2 \log p + 1$ bits, on peut extraire $\log p$ bits. Ainsi, si l'on a besoin d'une clef de 80 bits, en travaillant avec une courbe d'ordre premier définie sur $GF(p^2)$ avec p un premier proche de 2^{80} , on a la sécurité souhaitée. La preuve de ce résultat utilise une analyse de la géométrie des courbes qui sont incluses dans la restriction de Weil de la courbe elliptique de départ. Une fois la géométrie maîtrisée, les bornes de Weil permettent de conclure. Actuellement, nous sommes en train de chercher à obtenir un résultat similaire pour

les courbes définies sur un corps premier. Pour cela, l’approche géométrique ne fonctionne plus, et il faudra vraisemblablement utiliser des résultats sur les sommes de caractères.

3.4 Génération de paramètres

Nous avons également étudié la sécurité de courbes qui en général ne sont pas dans les standards, mais qui présenteraient de nombreux avantages. En effet, pour une utilisation des courbes elliptiques en environnement contraint, il est extrêmement tentant de choisir un corps de base “intermédiaire”, c’est-à-dire une extension de degré moyen d’un corps premier lui-même de taille moyenne, adaptée à l’architecture visée. Fabriquer de bonnes courbes pour ces corps est un problème peu étudié, car les paramètres se situent à la limite de validité de divers algorithmes. Avant de s’y atteler, il est prudent d’étudier la sécurité. En effet, une attaque par descente de Weil peut s’appliquer à certaines de ces situations, affaiblissant alors le cryptosystème. Dans le preprint [G-04], nous avons modifié l’attaque par descente de Weil, de manière à la rendre applicable à toutes les courbes elliptiques définies sur un corps de la forme $GF(p^n)$ avec n petit et $n \geq 3$. Ainsi, pour $n = 3$, on obtient un algorithme de complexité $O(p^{1.42})$, alors que la meilleure attaque connue auparavant (attaque générique) donnait $O(p^{1.5})$. Pour $n = 4$, la complexité est de $O(p^{1.55})$ au lieu de $O(p^2)$ auparavant. Pour $n \geq 5$, on obtient aussi des améliorations de complexité, mais la constante impliquée est telle que le résultat n’a pas d’implication pratique pour le moment.

Un nouvel algorithme permettant d’améliorer les attaques sur les courbes hyperelliptiques de petit genre a été développé [GTT-04]. Cette approche s’applique en fait très facilement au contexte de notre algorithme précédent pour attaquer les courbes elliptiques sur des extensions de petit degré. Ainsi en combinant les résultats de ces deux preprints, on obtient un algorithme de complexité $O(p^{1.33})$ pour les log discrets dans les courbes elliptiques sur $GF(p^3)$, et $O(p^{1.5})$ pour les courbes sur $GF(p^4)$.

Ces résultats confirment une impression qui règne depuis quelques temps dans la communauté, et qui est reflétée dans les standards : les courbes définies sur un corps premier ou sur une extension première de $GF(2)$ sont préférables. Toutefois, n’ayant pas d’argument frappant, on aurait pu être tenté d’intégrer les courbes sur les corps intermédiaires dans des produits où les contraintes sont énormes. Notre travail fournit un argument tangible pour affirmer que ce n’est pas du tout raisonnable, car ce que l’on gagne en efficacité du calcul, on risque de le perdre en sécurité.

Nous travaillons sur l’amélioration des deux méthodes connues pour fabriquer efficacement des courbes utilisables en cryptographie. D’une part nous avons montré que l’algorithme SEA de comptage de points s’adapte bien pour fabriquer efficacement des courbes convenables pour l’extracteur déterministe TAU. En effet, dans cette optique il est nécessaire d’avoir simultanément la primalité de l’ordre de la courbe et de sa twist quadratique. D’autre part la complexité théorique de la fabrication de courbes à multiplication complexe a été améliorée, ce qui se traduit aussi par des temps de calcul bien meilleurs que précédemment. Nous insistons sur le fait que la méthode de la multiplication complexe est la seule voie connue pour construire des courbes elliptiques non supersingulières qui permettent d’avoir des couplages calculables. Améliorer cette méthode revêt donc une importance capitale dans le contexte de la cryptographie basée sur l’identité, très à la mode ces derniers temps.

3.5 Diffusion chiffrée, et traçage de pirates

La diffusion chiffrée permet à un émetteur de transmettre un contenu numérique de façon à ce que seuls les abonnés puissent déchiffrer l’information. Une solution simple mais insuffisante consiste à fournir la clé (unique) de déchiffrement à tout abonné. Cependant, un traître pourrait

donner/vendre cette clé pour produire des décodeurs pirates. Ce traître ne pourra pas être repéré, puisque tout le monde a la même clé. Une solution naturelle serait alors de transmettre l'information chiffrée pour chaque abonné, chacun possédant une clé distincte. Mais ceci induit une communication linéaire en le nombre d'abonnés. Une solution intermédiaire est possible, mais elle ne peut pas exclure, de façon inconditionnelle, la production de décodeurs pirates à partir de la coalition d'un certain nombre d'individus, et donc la combinaison de plusieurs secrets.

Des solutions permettent de garantir, de façon calculatoire, l'impossibilité d'une telle coalition, sous peine de détection d'au moins un des traîtres, à condition que cette coalition ne soit pas trop nombreuse. Cependant, la taille du chiffré reste importante. Dans [CPP-05], nous avons exploité les courbes elliptiques et les couplages pour réduire cette taille. Nous avons du même coup introduit une nouvelle fonctionnalité : la traçabilité publique. En effet, malgré la possibilité de retrouver le traître à partir d'un décodeur pirate, cette étape est coûteuse, et nécessitait jusqu'à présent des informations secrètes (permettant de produire des décodeurs pirates anonymes). Nous avons rendu publique la phase la plus coûteuse, permettant ainsi de la distribuer.

3.6 Étude des normes

Nous avons commencé l'étude des normes utilisant les courbes elliptiques, et notamment les protocoles de mise en accord de clé, tels que ECMQV et MTI/A0, MTI/C0. Aucune preuve de sécurité n'existe pour le moment, nous tentons de déterminer si des hypothèses algorithmiques raisonnables permettraient de garantir la sécurité.

References

- [ACP-05] M. Abdalla, O. Chevassut, and D. Pointcheval. One-time Verifier-based Encrypted Key Exchange. In *PKC '05*, LNCS 3386, pages 47–64. Springer-Verlag, Berlin, 2005.
- [AFP-05] M. Abdalla, P. A. Fouque, and D. Pointcheval. Password-Based Authenticated Key Exchange in the Three-Party Setting. In *PKC '05*, LNCS 3386, pages 65–84. Springer-Verlag, Berlin, 2005.
- [AP-05b] M. Abdalla and D. Pointcheval. Interactive Diffie-Hellman Assumptions with Applications to Password-based Authentication. In *Financial Cryptography '05*, LNCS. Springer-Verlag, Berlin, 2005.
- [AP-05a] M. Abdalla and D. Pointcheval. Simple Password-Based Encrypted Key Exchange Protocols. In *CT – RSA '05*, LNCS, pages 191–208. Springer-Verlag, Berlin, 2005.
- [BCEP-03] E. Bresson, O. Chevassut, A. Essiari, and D. Pointcheval. Mutual Authentication and Group Key Agreement for Low-Power Mobile Devices. In *5th IFIP-TC6 International Conference on Mobile and Wireless Communications Networks*, pages 59–62. World Scientific Publishing, 2003.
- [BCEP-04] E. Bresson, O. Chevassut, A. Essiari, and D. Pointcheval. Mutual Authentication and Group Key Agreement for Low-Power Mobile Devices. In *Journal of Computer Communications*, Elsevier, 2004.
- [BCP-03] E. Bresson, O. Chevassut, and D. Pointcheval. Security Proofs for an Efficient Password-Based Key Exchange. In *10th ACM Conference on Computer and Communications Security*, pages 241–250. ACM Press, 2003.

- [BCP-04] E. Bresson, O. Chevassut, and D. Pointcheval. New Security Results on Encrypted Key Exchange. In *PKC '04*, LNCS 2947, pages 145–158. Springer-Verlag, Berlin, 2004.
- [CPP-04] D. Catalano, D. Pointcheval, and T. Pornin. IPAKE: Isomorphisms for Password-based Authenticated Key Exchange. In *Crypto '04*. Springer-Verlag, Berlin, 2004. Soumis.
- [CPP-05] H. Chabanne, D. H. Phan, and D. Pointcheval. Public Traceability in Traitor Tracing Schemes. In *Eurocrypt '05*, LNCS 3494, pages 542–558. Springer-Verlag, Berlin, 2005.
- [CFGP-05] O. Chevassut, P. A. Fouque, P. Gaudry, and D. Pointcheval. Key Derivation and Randomness Extraction. In *Crypto '05*. Springer-Verlag, Berlin, 2005. Soumis. Cryptology ePrint Archive, Report 2005/061.
- [G-04] P. Gaudry. Index calculus for abelian varieties and the elliptic curve discrete logarithm problem. Cryptology ePrint Archive, Report 2004/073.
- [GTT-04] P. Gaudry, N. Thériault and E. Thomé. A double large prime variation for small genus hyperelliptic index calculus. Cryptology ePrint Archive, Report 2004/153.
- [G-05] N. Gürel. Extracting bits from coordinates of a point of an elliptic curve. Available from <http://www.di.ens.fr/users/pointche/Projets/CESAM/>.