C. R. Acad. Sci. Paris, t. , Série I
Théorie des Nombres/*Number Theory*

Utilisation des modules de Drinfeld en cryptologie

Roland GILLARD, Franck LEPREVOST, Alexei PANCHISHKIN, Xavier-François ROBLOT

Institut Fourier, Université de Grenoble I
UMR 5582 CNRS-UJF 38402 Saint-Martin-d'Hères (France)
E-mail : gillard@ujf-grenoble.fr, Franck.Leprevost@ujf-grenoble.fr,
panchish@ujf-grenoble.fr

Institut Girard Desargues, Université Claude Bernard (Lyon I)
UMR 5028 CNRS-UCB 69622 Villeurbanne (France)
E-mail : roblot@euler.univ-lyon1.fr

Résumé . Nous présentons dans cette note un nouveau et efficace cryptosystème à clef publique basé sur les modules de Drinfeld. Les détails apparaitront ailleurs.

Using Drinfeld modules in cryptology

Abstract . *We present in this note a new and efficient public-key cryptosystem based on Drinfeld modules. The details will appear elsewhere.*

Introduction . — La sécurité des cryptosystèmes à clef publique est basée sur des problèmes mathématiques difficiles à résoudre en pratique. Citons par exemple les cryptosystèmes RSA et de Rabin, dont la sécurité est fondée sur le problème de la factorisation des entiers, celui de El Gamal ou le protocole d'échange de clefs de Diffie-Hellman basés sur le problème du logarithme discret dans le groupe multiplicatif d'un corps fini, ou dans le groupe des points rationnels d'une courbe elliptique définie sur un corps fini. Ces protocoles ont été longuement étudiés et sont standardisés (par exemple, voir [6] et les références incluses). Néanmoins, il est utile de disposer d'autres cryptosystèmes, dont la sécurité est basée sur d'autres problèmes, et qui, le cas échéant, peuvent être inclus dans de nouveaux standards. Dans [5], Scanlon montre que des analogues naturels dans le cadre des modules de Drinfeld (voir [1] et [2], ainsi que [4] pour ces objets) des cryptosystèmes évoqués plus haut sont particulièrement faibles. Nous esquissons ici une approche différente, qui contourne les faiblesses soulevées par Scanlon, en construisant une fonction sens unique à trappe (FSUT). Les détails apparaitront dans un autre article (voir [3]), et une demande de brevet a été déposée sur ce nouveau cryptosystème.

Fonctions sens unique à trappe et applications à la cryptographie . — Une fonction ψ bijective d'un ensemble $\mathcal{E}$ dans un ensemble $\mathcal{F}$ est dite une *fonction sens unique* (FSU) si la seule donnée de ψ et de $y \in \mathcal{F}$ ne permet pas de résoudre en pratique le *problème d'inversion* consistant à trouver l'unique $x \in \mathcal{E}$ tel que $y = \psi(x)$. La fonction ψ est dite *fonction sens unique à trappe* (FSUT) si c'est une FSU telle que la connaissance d'une clé secrète $\mathbf{K}$ permette de résoudre le problème d'inversion. Une application cryptographique est la suivante, sous une forme très simplifiée, où *Alice* et *Bob* sont deux protagonistes cherchant à échanger une information confidentielle sur un canal ouvert. *Alice* publie la FSUT ψ de $\mathcal{E}$ (messages en clair) dans $\mathcal{F}$ (messages cryptés), et garde secrète la clé $\mathbf{K}$. Pour transmettre un message $x \in \mathcal{E}$, *Bob* calcule et envoie $y = \psi(x)$ à *Alice*. Pour décoder le message y, il faut calculer $x = \psi^{-1}(y)$. Seule *Alice*, qui possède $\mathbf{K}$, est à même de résoudre efficacement ce problème.

Présentation théorique du protocole. — Soit p un nombre premier. On note $\mathcal{A} = \mathbf{F}_p[T]$ l'anneau des polynômes à une variable et à coefficients dans le corps $\mathbf{F}_p$. Puis, on pose $\mathcal{A}\{\tau\}$ l'anneau des polynômes à coefficients dans A et en la variable τ avec les règles de l'addition usuelle et pour la multiplication, la règle de commutation $\tau^k \times a = a^{p^k} \times \tau^k$ pour $a \in \mathcal{A}$ et $k \geq 1$.

Nous définissons un module de Drinfeld comme un morphisme de $\mathbf{F}_p$-algèbre φ de $\mathcal{A}$ dans $\mathcal{A}\{\tau\}$ tel que $\varphi(T)$ est un polynôme (en τ) de degré ≥ 1 et dont le terme constant est T. Soit $a = \sum_{i=0}^{m} a_i T^i$ un élément de $\mathcal{A}$, on a donc $\varphi(a) = \varphi\big(\sum_{i=0}^{m} a_i T^i\big) = \sum_{i=0}^{m} a_i \varphi(T)^i$. Ainsi, le module de Drinfeld φ est complètement défini une fois que $\varphi(T)$ a été choisi. L'exemple le plus simple de module de Drinfeld est le module de Carlitz défini par $\varphi(T) = \tau + T$. Chaque élément de $\mathcal{A}\{\tau\}$ défini une application de $\mathcal{A}$ dans lui-même en identifiant chaque élément $a \in \mathcal{A}$ avec l'application $z \mapsto az$ et τ avec l'application de Frobenius $z \mapsto z^p$. La multiplication dans $\mathcal{A}\{\tau\}$ correspond alors à la composition des fonctions. Ainsi, on obtient : $\sum_{i=0}^{m} a_i \tau^i : z \mapsto \sum_{i=0}^{m} a_i z^{p^i}$. Pour $a \in \mathcal{A}$, on note φ_a l'application de $\mathcal{A}$ dans lui-même définie par $\varphi(a) \in \mathcal{A}\{\tau\}$. L'application φ_1 est l'identité sur $\mathcal{A}$.

Soient $d > 1$ un entier et $f(T) \in \mathcal{A}$ un polynôme irréductible de degré d. On pose $\mathcal{B} = \mathcal{A}/(f(T))$. Ainsi $\mathcal{B}$ est isomorphe à $\mathbf{F}_{p^d}$. Pour $a \in \mathcal{A}$, on note $\overline{a}$ la classe de a dans $\mathcal{B}$. L'anneau fini $\mathcal{B}$ a donc une structure naturelle de $\mathcal{A}$-module. Soit $a \in \mathcal{A}$, on note $\overline{\varphi_a}$ l'application de $\mathcal{B}$ dans $\mathcal{B}$ défini par : $\overline{\varphi_a} : \overline{b} \mapsto \overline{\varphi_a(b)}$. Ainsi, le module de Drinfeld φ permet de munir $\mathcal{B}$ d'une autre structure de $\mathcal{A}$-module donnée par : $a \times_\varphi \overline{b} = \overline{\varphi_a}(\overline{b})$.

On note $\mathcal{B}_\varphi$ l'ensemble $\mathcal{B}$ muni de la structure de $\mathcal{A}$-module défini par φ. Puisque $\mathcal{B}_\varphi$ est un $\mathcal{A}$-module de cardinal p^d, il existe un polynôme unitaire $f_\varphi \in \mathcal{A}$ de degré d tel que $\mathcal{B}_\varphi \simeq \mathcal{A}/(f_\varphi)$ comme $\mathcal{A}$-module. En général, f_φ n'est pas irréductible. Une remarque fondamentale pour notre propos est que deux éléments a_1 et a_2 de $\mathcal{A}$ donnent la même application $\overline{\varphi_{a_1}} = \overline{\varphi_{a_2}}$ si et seulement si $a_1 \equiv a_2 \pmod{f_\varphi}$. Ainsi, l'application $\overline{\varphi_a}$ est bijective si et seulement a est premier avec f_φ et, dans ce cas, l'application inverse est $\overline{\varphi_{a'}}$ où $a' \in \mathcal{A}$ est tel que $aa' \equiv 1 \pmod{f_\varphi}$.

Nous pouvons à présent expliquer comment construire une FSUT de $\mathcal{B}$ dans lui-même en utilisant les modules de Drinfeld. Les aspects algorithmiques de cette construction seront succintement détaillés dans la section suivante. On prend $\mathcal{E} = \mathcal{F} = \mathcal{B}$. La clé secrète $\mathbf{K}$ est composée de deux éléments c_1 et c_2 de $\mathcal{A}$, tous les deux premiers avec f_φ, et d'une bijection σ de $\mathcal{B}$ dans lui-même (voir la section suivante pour des exemples de choix de σ). La fonction ψ est alors définie par :

$$\psi(z) = \left(\overline{\varphi_{c_1}} \circ \sigma \circ \overline{\varphi_{c_2}}\right)(z)$$

donnée comme un polynôme de $\mathcal{B}[z]$. Cette fonction est bijective et sa réciproque est la fonction

$$\psi^{-1}(z) = \left(\overline{\varphi_{c_2'}} \circ \sigma^{-1} \circ \overline{\varphi_{c_1'}}\right)(z)$$

avec c'_1 et c'_2 deux éléments de $\mathcal{A}$ tels que $c_1\,c'_1 \equiv 1 \pmod{f_\varphi}$, et $c_2\,c'_2 \equiv 1 \pmod{f_\varphi}$. Cette fonction inverse ne peut être déterminée en pratique que si la clé secrète (c_1, c_2, σ) est connue.

Présentation pratique du protocole. — Supposons choisis les différents paramètres p, φ, d et f (les choix de ceux-ci seront discutés dans la section suivante), nous montrons comment trouver c_1, c_2 et σ, puis calculer les fonctions ψ et ψ^{-1} et s'en servir pour le chiffrement et le déchiffrement.

Pour calculer dans $\mathcal{B}$, nous représentons chaque classe par l'unique polynôme de degré $< d$ contenu dans la classe. Soit $a = \sum_{i=0}^{m} a_i T^i \in \mathcal{A}$ et $\beta = \sum_{j=0}^{d-1} \overline{b_j T^j} \in \mathcal{B}$. Alors :

$$\overline{\varphi_a}(\beta) = \sum_{i=0}^{m} \overline{a_i \varphi_{T^i}}(\beta) = \sum_{i=0}^{m} \sum_{j=0}^{d-1} \overline{a_i b_j}\, \overline{\varphi_{T^i}}(\overline{T^j}).$$

Donc il suffit de savoir calculer $\overline{\varphi_{T^i}}(\overline{T^j})$ pour en déduire $\overline{\varphi_a}(\beta)$ par cette formule. De surcroît, puisque $\overline{\varphi_a} = \overline{\varphi_{a'}}$ si $a \equiv a' \pmod{f_\varphi}$, si on prend pour a' le reste de la division euclidienne de a par f_φ, on voit qu'il est suffisant de considérer uniquement les exposants i tels que $0 \leq i \leq d-1$. On écrit $\varphi(T) = \sum_{k=0}^{d} \phi_k \tau^k$, avec $\phi_k \in \mathcal{A}$ et, par définition, $\phi_0 = T$. Alors, on a, pour tout $j \geq 0$: $\overline{\varphi_1}(\overline{T^j}) = \overline{T^j}$ et $\overline{\varphi_T}(\overline{T^j}) = \sum_{k=0}^{d} \overline{\phi_k \times T^{jp^k}}$, puis la formule de récurrence $\overline{\varphi_{T^{i+1}}}(\overline{T^j}) = \overline{\varphi_T}\Big(\overline{\varphi_{T^i}}(\overline{T^j})\Big)$ $(i \geq 1)$ pour le calcul des autres valeurs.

Le calcul de f_φ s'obtient comme suit. En tant que $\mathbf{F}_p$-espace vectoriel, $\mathcal{B}$ est de dimension d admettant la famille $\{\overline{1}, \overline{T}, \ldots, \overline{T}^{d-1}\}$ pour base. L'application $\overline{\varphi_T}$ est en fait un endomorphisme de cette espace vectoriel dont on calcule facilement la matrice, puis le polynôme caractéristique $C(T)$. Le calcul montre que $f_\varphi = C(T)$.

Il convient à présent de choisir la clé secrète, *i-e.* les deux éléments c_1 et c_2 de $\mathcal{A}$, de degré $< d$ et premiers avec f_φ, et la bijection σ. Pour c_1 et c_2, la meilleure méthode est de choisir au hasard un couple de deux polynômes distincts premiers avec f_φ. Des exprimentations montrent aussi qu'il vaut mieux choisir c_1 et c_2 sans coefficients nuls. On calcule aussi les deux polynômes c'_1 et c'_2 de degré $< d$ et tels que $c_1\,c'_1 \equiv 1 \pmod{f_\varphi}$ et $c_2\,c'_2 \equiv 1 \pmod{f_\varphi}$; ces deux polynômes serviront pour le décryptage. Pour la bijection σ, il faut une fonction simple et rapide à calculer, donnée sous la forme d'un polynôme. Soit $e < p^d - 1$ un entier non divisible par p et premier avec $p^d - 1$, on prend $\sigma : z \mapsto z^e + \delta$, où δ est un élément pris au hasard dans $\mathcal{B}$. Si on note f un entier tel que $ef \equiv 1 \pmod{p^d - 1}$, alors on a $\sigma^{-1} : z \mapsto (z - \delta)^f$.

A présent, on calcule la FSUT $\psi(z) = \left(\overline{\varphi_{c_1}} \circ \sigma \circ \overline{\varphi_{c_2}}\right)(z)$ sous la forme d'un polynôme. Pour cela, on écrit à nouveau : $a = \sum_{i=0}^{d-1} a_i T^i$, avec $a_i \in \mathcal{A}$, et donc $\overline{\varphi_a}(z) = \sum_{i=0}^{d-1} \overline{a_i}\, \overline{\varphi_{T^i}}(z)$, et les $\overline{\varphi_{T^i}}(z)$ sont reliés par la relation de récurrence déjà mentionnée ci-dessus. Pour utiliser cette formule, on écrit $\varphi(T) = \sum_{k=0}^{d} \phi_k \tau^k$, avec $\phi_i \in \mathcal{A}$, et, par définition, $\phi_0 = T$. Alors, pour tout polynôme $P(z) = \sum_{i=0}^{m} \pi_i z^i \in \mathcal{B}[z]$, on a

$$\overline{\varphi_T}(P(z)) = \sum_{k=0}^{d} \sum_{i=0}^{m} \overline{\phi_k} \pi_i^{p^k} z^{ip^k}.$$

Grâce à cette formule, on obtient une expression pour $\psi(z)$ donné comme un polynôme en z et à coefficients dans $\mathcal{B}$. Il est à noter que dans ce calcul, on doit remplacer les termes de la forme γz^m avec $m \geq p^d$ par γz^r, où r est le reste de la division euclidienne de m par $p^d - 1$, puisque $\beta^m = \beta^r$ pour tout $\beta \in \mathcal{B}$. On s'assure ainsi que tous les termes de $\psi(z)$ restent de degré $< p^d$.

Le cardinal de $\mathcal{B}$ est p^d, donc la FSUT ψ permet de coder des messages donnés sous la forme d'un nombre M vérifiant $0 \leq M < p^d$. En effet, étant donné un tel nombre M, on le transforme en un élément de $\mathcal{B}$ en utilisant l'écriture de M en base p : il existe des entiers uniques $m_0, \ldots, m_{d-1}$ vérifiant $0 \leq m_i < p$ et tels que $M = m_0 + m_1\, p + m_2\, p^2 + \cdots + m_{d-1}\, p^{d-1}$, ce qui permet d'associer de manière unique à l'entier M, l'élément

$$\mu = \overline{m_0 + m_1 T + \cdots + m_{d-1} T^{d-1}}$$

de $\mathcal{B}$. On calcule alors

$$\chi = \psi(\mu) = \overline{k_0 + k_1 T + \cdots + k_{d-1} T^{d-1}}$$

et le message codé C est alors donné, comme nombre entre 0 et p^d, par le procédé inverse : $C = k_0 + k_1\, p + \cdots + k_{d-1}\, p^{d-1}$. Inversement, pour décrypter le message C, on construit l'élément χ de $\mathcal{B}$ correspondant, on applique ψ^{-1} pour en déduire μ et finalement M. Il est à noter cependant que, plutôt que calculer $\psi^{-1}(z)$ sous forme d'un polynôme pour effectuer le calcul de $\psi^{-1}(\mu)$, il est plus efficace de faire ce calcul en utilisant l'expression : $\psi^{-1}(\mu) = \overline{\varphi_{c_2'}}\big(\sigma^{-1}\big(\overline{\varphi_{c_1'}}(\mu)\big)\big)$.

Choix des paramètres. — Plusieurs attaques du cryptosystème sont envisageables (calcul de cycle, factorisation, énumération, etc). La considération de ces attaques montre que l'on doit prendre p très grand par rapport à d. Cependant, par souci d'efficacité algorithmique et d'architecture d'ordinateur, il est souhaitable que $p < 2^{32}$. Par ailleurs, la taille du polynôme $\psi(z)$ augmente avec l'exposant e, donc e doit être petit. D'un autre côté, e doit être premier avec $p^d - 1$, donc e est impair. Des choix raisonnables sont $e = 5$ ou $e = 7$. Finalement, afin de pouvoir utiliser ce protocole pour crypter des clés AES, il faut que $p^d > 2^{160} \simeq 10^{18}$. En résumé, des restrictions souhaitables sur les paramètres sont les suivantes : p est un grand nombre premier plus petit que 2^{32} ; d est un entier tel que $p^d \geq 2^{160}$; f est un polynôme de $\mathbf{F}_p[T]$ irréductible de degré d ; e est petit et premier avec $p^d - 1$; φ est le module de Carlitz ; c_1 et c_2 sont deux polynômes de $\mathbf{F}_p[T]$ de degré $d - 1$, premiers avec f_φ, et sans coefficients nuls ; δ est un élément non nul de $\mathcal{B}$. Nous avons implémenté des exemples, et il en découle que le stockage des données de cryptage nécessite environ 26000 caractères. Sur un Pentium 700MHz, tournant sous Linux 2.4, une implémentation de ce protocole en `C++` avec la librairie NTL [7] donne un temps de codage de l'ordre de un centième de seconde. Le temps de décodage est négligeable.

Références bibliographiques

[1] **V.G.Drinfeld**, **1976**. Elliptic modules, *Math.USSR-Sbornik* 23, 561-592

[2] **V.G.Drinfeld**, **1977**. Elliptic modules, *Math.USSR-Sbornik* 31, 159-170

[3] **R. Gillard, F. Leprévost, A. Panchishkin, X.-F. Roblot**, **2002**. A new public-key cryptosystem based on Drinfeld modules, *(En préparation)*

[4] **A.A.Panchishkin**, **1993-94**. Algorithmes rapides pour factorisation des nombres et des polynômes, tests de primalité, courbes elliptiques et modules de Drinfeld, *Séminaire de Théorie des nombres (Caen)*, 1-10

[5] **T. Scanlon**, **2001**. Public key cryptosystems based on Drinfeld modules are insecure, *Journal of Cryptology* 14, 225-230

[6] **IEEE Standards**, **1999**. Group P1363: Standard Specification for Public-Key Cryptography, *page internet :* `http://grouper.ieee.org/groups/1363`

[7] **V. Shoup**, **2002**. NTL: A Library for doing Number Theory, *page internet :* `http://shoup.net/ntl/`