UNIVERSITATEA BUCUREŞTI
FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ
ŞCOALA DOCTORALĂ DE INFORMATICĂ

Group protocols in classical and lattice-based cryptography (Protocoale de grup în criptografia clasică şi criptografia bazată pe latici)

TEZĂ DE DOCTORAT
REZUMAT

Coordonator:
Prof. Dr. Adrian ATANASIU

Doctorand:
Adela MIHĂIŢĂ (GEORGESCU)

Bucureşti

Martie 2013

Cuprins

Capitolul 1

Introducere

Titlul acestei teze sugerează două abordări diferite ale critpografiei cu cheie publică pe care le facem în această lucrare relativ la protocoalele de grup. In criptografia cu cheie publică, securitatea primitivelor criptografice se bazează pe ipoteza că anumite probleme de teoria numerelor nu pot fi rezolvate în timp polinomial, cele mai folosite fiind problema factorizării numerelor mari şi problema logaritmului discret. Această abordare a criptografiei cu chei publică o vom numi în continuare *criptografia classica*, spre deosebire de mai noua şi deosebit de promitătoarea abordare bazată pe latici, cunoscută sub numele de *criptografia bazată pe latici*.

Pentru a evita orice posibilă confuzie, subliniem faptul că folosim laticile ca obiecte geometrice care constă în aranjamente regulate de puncte în spatiul euclidian, asemănator spaţiilor vectoriale. Există şi un alt înţeles complet diferit asociat aceluiaşi termen *latici* ce semnifică mulţimi parţial ordonate. In teza de faţă, lucrăm doar cu primul tip de latici, numite uneori în literatura şi *latici punctuale*.

Criptografia cu cheie publică necesită o sursă de probleme dificile computaţional care să asigure baza securităţii primitivelor sale criptografice. Mult timp, teoria numerelor a fost considerată o astfel de sursă şi totul a fost bine până când Shor a propus un algoritm [Sho97] care să arate că dificultatea acestor probleme poate fi depăşită cu ajutorul calculatoarelor cuantice suficient de mari. Această descoperire a aruncat o umbră de îngrijorare cu privire la necesitatea găsirii unei alternative rezistente cuantic pentru aceste probleme de teoria numerelor.

Criptografia laticeală îşi are bazele în lucrarea de pionierat a lui Ajtai [Ajt96] care în 1996 a stabilit o conexiune între complexitatea în cazul cel mai defavorabil şi cea în cazul mediu a problemelor de latici. Rezultatul e important

pentru că în criptografie, securitatea celor mai multe primitive se bazează pe dificultate în cazul mediu. De pildă, securitatea unui sistem de criptare care se bazează pe problema factorizării este compromisă atunci când un adversar este capabil să factorizeze anumite numere, dar nu toate numerele. Problemele de latici acoperă exact această arie a problemelor a căror fiecare instanţă este dificil de rezolvat. Atacarea cu succes a unei construcţii criptografice implică existenţa unui algoritm care să rezolve fiecare instanţa a unei probleme de latici care îi stă la bază.

Conexiunea între cazul mediu şi cazul cel mai defavorabil este una dintre caracteristicile cele mai reprezentative şi distinctive ale criptografiei bazate pe latici, dar nu este singura. Laticile beneficiază şi de simplitate, eficienţă şi implementări paralelizabile pentru că implică doar operaţii liniare (înmultiri şi adunări de vectori şi matrici) pe numere relativ mici (mult mai mici decât numerele foarte mari necesare pentru dificultatea problemelor de teoria numerelor).

O alta caracteristică fundamentală a criptografiei bazate pe latici este rezistenţa în faţa calculatoarelor cuantice. Deşi nu sunt încă o realitate practică, calculatoarele cuantice reprezintă o ameninţare serioasă pentru criptografia cu cheie publică de astăzi bazată pe probleme de teoria numerelor. Toţi algoritmii pentru problemele de latici cunoscuţi până acum rulează în timp exponetial (pentru versiunile exacte) sau ating factori de aproximare foarte mari (pentru valorile aproximative). Insă nici algoritmii cuantici pentru problemele de latici nu sunt semnificativ mai eficienţi decât cei clasici. Aşa încât, se poate conjectura ideea următoare: critpografia bazată pe latici reprezintă o alternativă plauzibilă pentru criptografia post-cuantică.

Ar părea că laticile reprezintă o mină de aur pentru criptografie: poate cel mai important rezultat de până acum este construcţia primei scheme de criptare complet homomorfă în 2009 [Gen09] bazată pe latici ideale. Acest rezultat era considerat imposibil de obţinut pâna atunci. Deşi schema este ineficientă, Gentry a aratat că este posibilă şi de atunci au aparut progrese semnificative în direcţia eficientizării. O altă descoperire foarte recentă este prima aplicaţie multiliniară derivată tot din latici ideale [GGH12]. Până acum, în criptografie existau doar perechile biliniare care stau la baza multor construcţii importante din criptografie, dar multiliniaritatea nu a putut fi obţinută în criptografie prin nici o alta metodă în afara laticilor.

Cu atât de multe beneficii oferite de latici, este de înteles explozia de prim-

itive criptografice bazate pe latici la care asistăm în ultima vreme: scheme de criptare, semnături digitale, scheme de criptare bazate pe identităţi, decriptare cu prag, criptare funcţională şi bazată pe atribute etc. Insă în ciuda acestei abundanţe de primitive, incă mai sunt multe progrese de facut în această zonă. Contribuţia tezei de faţă se înscrie în randul acelora menite să umple un mic gol din criptografia bazată pe latici: protocoale de stabilire a cheilor (atât de transfer cât şi de stabilire de comun acord) bazate pe latici.

In teza de faţă dăm o serie de rezultate în aria protocoalelor de grup începând cu criptografia clasică şi continuând apoi cu criptografia bazată pe latici. Incepem, în **Capitolul 2** prin a da noţiuni preliminare despre latici incluzând probleme computaţionale dificile atribuite laticilor, dar şi problema invăţării cu erori (LWE) indirect asociată, pe care se vor baza majoritatea rezultatelor personale din această teză. Prezentăm aici şi câteva primitive criptografice bazate pe latici de care vom face uz în construcţiile noastre. O versiune preliminară a acestui capitol a fost publicată în lucrarea [Mih10].

Contribuţiile proprii din această teză sunt prezentate în următoarele două capitole. In **Capitolul 3** introducem două protocoale de stabilire de chei diferite cu scopul de a rezolva o anumită problemă practică. Primul protocol se bazează pe recriptare proxy iar al doilea impune o condiţie suplimentară, anonimitatea membrilor unii faţă de ceilalţi. Ambele protocoale sunt construite în criptografia clasică având ca bază de securitate problema logaritmului discret. Conţinutul acestui capitol a fost publicat în două lucrări [AM11] şi [AG12].

In **Capitolul 4** construim protocoale criptografice cu ajutorul laticilor. Propunem mai întâi o schemă de partajare a secretelor simplă dar verificabilă, publicată în lucrarea [Geo11]. Descriem apoi un protocol de transfer de chei (pentru aceeaşi problemă practică precum precedentele protocoale) bazat pe latici şi dăm rezultate de securitate pentru el. Acest protocol a fost publicat în lucrarea [Geo12]. A treia parte importantă a capitolului propune o versiune echivalentă bazată pe latici pentru un protocol foarte cunoscut şi eficient de stabilire de comun acord de chei din criptografia clasică. Identificăm aici dificultăţi care apar la conversia protocolului în latici, cu precădere la adaptarea demonstraţiei de securitate. Rezolvăm aceste probleme şi dăm rezultate de securitate consistente. Articolul care conţine aceste rezultate este încă în lucru [GS13]. Ca o ultimă parte a acestui capitol, dăm o altă construcţie echivalentă în latici pentru o schea de criptare broadcast anonimă clasică din care ne-am inspirat pentru a obţine anonimitatea în protocoalele noastre. Acest ultim rezultat a fost publicat

in [Geo13].

In final, in **Capitolul 5** dăm câteva concluzii privind rezultatele din teză şi identificăm unele probleme deschise în rezultatele noastre, dar si în literatura de specialitate pe care le sugerăm ca activitate viitoare de cercetare.

Capitolul 2

Latici

Vectorii şi matricile vor fi notate cu litere ingroşate, de exemplu $\mathbf{v}$ e un vector iar $\mathbf{M}$ e o matrice. Vom nota cu $\langle x, y \rangle$ produsul scalar a doi vectori $\mathbf{x}$ şi $\mathbf{y}$ şi cu $||\mathbf{v}||$ norma euclidiană a vectorului $\mathbf{v}$. Vom folosi frecvent notaţia $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ pentru a desemna matrici $\mathbf{A}$ cu n linii, m coloane şi elemente din $\mathbb{Z}_q$. Dacă Ψ este o probabilitate de distribuţie, vom folosi notaţia $x \leftarrow \Psi$ care semnifică faptul că un element x este extras conform cu distribuţia Ψ. Vom defini mulţimea $[n] = \{1, 2, ..., n\}$.

2.1 Preliminarii

Definitie 2.1. *Fie* $\mathbf{B} = \{\mathbf{b_1}, ...\mathbf{b_k}\} \in \mathbb{R}^{n \times k}$ *vectori liniar independenţi din* $\mathbb{R}^n$. *Laticea generată de* $\mathbf{B}$ *este mulţimea tuturor combinaţiilor liniare întregi ale vectorilor din* $\mathbf{B}$

$$\mathcal{L}(\mathbf{B}) = \left\{ \sum_{i=1}^{n} x_i \cdot \mathbf{b_i} \ : \ x_i \in \mathbb{Z} \right\}.$$

Cel mai simplu exemplu de latice este $\mathbb{Z}^n$, mulţimea tuturor vectorilor n-dimensionali cu elemente întregi, generată de matricea identitate I_n (în acest caz, $n = k$). Criptografia lucrează cu latici întregi, i.e. submulţimi infinite ale lui $\mathbb{Z}^n$.

Matricea $\mathbf{B}$ având vectorii $\mathbf{b_1}, ..., \mathbf{b_k}$ drept coloane formează o bază a laticii. Numărul de astfel de vectori din bază defineşte dimensiunea laticii dim $\mathcal{L}(\mathbf{B})$. Orice latice admite mai multe baze, dintre care unele (cu vectori scurţi şi aproape ortogonali) sunt mai bune decât altele (cu vectori lungi) (a se vedea figura 2.1). In criptografie ne interesează bazele cu vectori scurţi.

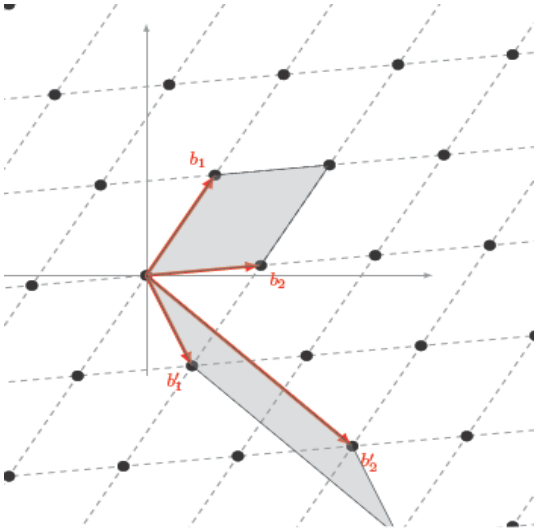

Figure 2.1: *O latice 2-dimensională cu două baze diferite; b_1 şi b_2 determină o bază scurtă şi aproape ortogonală*

Laticile sunt caracterizate de diverşi parametri însă deocamdată ne vom îndrepta atenţia asupra unora care nu depind de o bază aleasă care generează laticea, anume cei mai scurţi n vectori din latice.

Definitie 2.2. *Distanţa minimă a unei latici $\mathcal{L}(\mathbf{B})$ notată λ_1, este distanţa minimă între oricare două puncte distincte din latice şi este egală cu lungimea celui mai scurt vector nenul din latice.*

$\lambda_1(\mathcal{L}(\mathbf{B})) = min\{||x-y||: x \neq y \in \mathcal{L}(\mathbf{B})\} = min\{||x||: x \in \mathcal{L}(\mathbf{B})\backslash\{0\}\}.$

Definitie 2.3. *Al i-lea* minim succesiv $\lambda_i(\mathcal{L}(\mathbf{B}))$ *al unei latici este cel mai mic r pozitiv asa încât sfera de rază r centrată în origine conţine cel puţin i vectori liniar independenţi.*

In tot ceea ce vom studia în continuare, vom fi intersaţi de trei tipuri de latici. Fixăm mai întâi următorii parametri $k \leq n \leq q$ unde k este principalul parametru de securitate. De obicei n este un multiplu mic al lui k (de exemplu $n = O(k)$) şi q este un numar prim mic cu $O(\log k)$ biţi. Remarcăm că q este mult mai mic decât numerele prime folosite în criptografia clasică bazată pe probleme din teoria numerelor.

Vom folosi o notaţie puţin diferită pentru a diferenţia aceste latici de cele obişnuite. Fiind dată o matrice $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, pentru q, m, n întregi, primul tip de latici conţine vectori ortogonali pe liniile matricii $\mathbf{A}$

$$\Lambda^{\perp}(\mathbf{A}) = \{\mathbf{z} \in \mathbb{Z}^m \ : \ \mathbf{Az} = \mathbf{0} \bmod q\}.$$

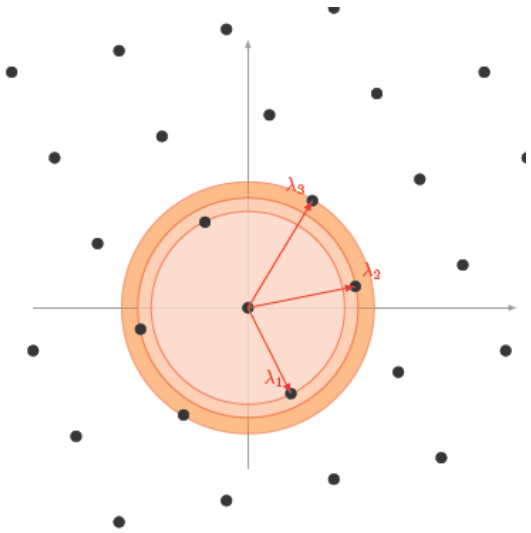

Figure 2.2: *Latice 2-dimensionala cu primele doua minime succesive*

Al doilea tip de latici este generat de liniile matricii $\mathbf{A}$

$$\Lambda(\mathbf{A}) = \{\mathbf{z} \in \mathbb{Z}^m \ : \ \exists\, \mathbf{s} \in \mathbb{Z}_q^n \text{ aşa încât } \mathbf{z} = \mathbf{A}^\top \mathbf{s} \bmod q\}.$$

Pentru orice $\mathbf{u} \in \mathbb{Z}_q^n$ aşa încât $\mathbf{Ax} = \mathbf{u} \bmod q$ admite o soutie întreagă, considerăm laticea

$$\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}) = \{\mathbf{z} \in \mathbb{Z}^m \ : \ \mathbf{Az} = \mathbf{u} \bmod q\} = \Lambda^{\perp}(\mathbf{A}) + \mathbf{x}.$$

In criptografia bazată pe latici, vom folosi vectori de eroare (sau perturbare, zgomot) aleşi conform distribuţiei normale (gaussiene) D_α cu următoarea funcţie centrată în $\mathbf{c}$ de parametru r:

$$\forall \mathbf{x} \in \mathbb{R}^n, \rho_{r,c}(\mathbf{x}) = exp(-\pi||\mathbf{x} - \mathbf{c}||^2/r^2)$$

Funcţia se poate extinde la mulţimi (deci şi latici) astfel:

$$\rho_{r,c}(\mathrm{A}) = \sum_{\mathbf{x} \in A} \rho_{r,c}(\mathbf{x})$$

pentru orice mulţime numărabilă A.

Definitie 2.4. *(Distribuţia normală discretă)*

Pentru o latice n-dimensională $\Lambda = \mathcal{L}(\mathbf{B})$, distribuţia normală discretă Λ se defineşte astfel

$$\forall \mathbf{x} \in \Lambda, D_{\Lambda,r,c}(x) = \frac{\rho_{r,\mathbf{c}}(\mathbf{x})}{\rho_{r,\mathbf{c}}(\Lambda)}.$$

In situaţia în care coloanele lui $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ generează $\mathbb{Z}_q^n$, fixăm $\mathbf{u} \in \mathbb{Z}_q^n$ şi fie $\mathbf{t} \in \mathbb{Z}^m$ o soluţie arbitrară a ecuaţiei $\mathbf{At} = \mathbf{u} \bmod q$. Distribuţia normală peste $\Lambda_y^{\perp}(\mathbf{A})$ reprezentând distribuţia condiţionată $D_{\mathbb{Z}^m,r}$ fiind dat $\mathbf{Az} = \mathbf{y} \bmod q$ are formula

$$\forall \mathbf{x} \in \Lambda, D_{\Lambda_{\bar{y}}^{\perp}(\mathbf{A}),r}(x) = \frac{\rho_r(\mathbf{x})}{\rho_r(\mathbf{t} + \Lambda^{\perp}(\mathbf{A}))}$$

2.2 Probleme de latici

Calculul parametrilor pe care i-am prezentat în secţiunea precedentă pentru latici de dimensiuni mari generează probleme computaţional dificile care sunt fundamentul criptografiei bazate pe latici. Principala problemă dificilă asociată laticilor este problema celui mai scurt vector *Shortest Vector Problem (SVP)*.

Definitie 2.5. *(Shortest Vector Problem - SVP)*

Fiind dată o latice $\mathcal{L}$ generată de baza $\mathbf{B}$, problema cere să se găsească un cel mai scurt vector nenul $\mathbf{x}$ in $\mathcal{L}(\mathbf{B})$ (i.e. $0 \neq \mathbf{x}$ aşa încat $||\mathbf{x}|| = \lambda_1(\mathcal{L}(\mathbf{B}))$).

In criptografie se foloseşte versiunea aproximativă a acestei probleme SVP_γ care cere să se găsească un vector nenul $||x|| \in \mathcal{L}(\mathbf{B})$ a cărui normă este de cel mult γ ori mai mare decât a celui mai scurt vector. γ se numeşte factor de aproximare.

Pentru dimensiuni mari ale laticii, problema devine dificilă. Pentru o soluţie exactă sau o aproximare până la un factor polinomial, cel mai bun algoritm necesită timp exponenţial.

2.2.1 LWE - Problema invăţării cu erori

Problema invăţării cu erori (LWE - Learning With Errors) a fost introdusa în 2005 [Reg05] devenind extrem de populară într-un timp foarte scurt. Este foarte celebră pentru faptul că este considerată la fel de dificilă ca şi problemele bazate pe latici în cazul cel mai defavorabil deşi nu este direct relaţionată laticilor.

Problema cere găsirea unui vector secret $\mathbf{s} \in \mathbb{Z}_q^n$ fiind dată o serie de ecuaţii liniare aleatoare în $\mathbf{s}$ "aproximative" în sensul că sunt corecte până la o eroare aditivă de ± 1 (au fost perturbate cu o mică cantitate de zgomot). Formal:

Definitie 2.6. *(Problema LWE)*

Fixăm parametrii: dimensiunea $n \geq 1$, un modul $q \geq 2$ şi o probabilitate de distribuţie a erorii $\chi \in \mathbb{Z}_q$. Fie $A_{s,\chi}$ peste $\mathbb{Z}_q^n \times \mathbb{Z}_q$ distribuţia de probabilitate obţinută

alegând un vector $\boldsymbol{a}$ *uniform aleator, alegând* $e \in \mathbb{Z}_q$ *conform distributiei* χ *şi returnând perechi de forma* $(\mathbf{a}, \mathbf{a} * \mathbf{s} + e)$ *unde adunările se efectuează în* $\mathbb{Z}_q$*. Se spune că un algoritm rezolvă problema LWE cu modulul* q *şi probabilitatea de distribuţie* χ *dacă pentru orice* $\mathbf{s} \in \mathbb{Z}_q^n$*, fiind dat un număr arbitrar de eşantioane independente din* $A_{s,\chi}$ *întoarce* s *(cu o probabilitate foarte mare).*

Alegerea **parametrilor** se face astfel: χ este distribuţia normală cu deviaţia standard αq cu α de obicei $1/\text{poly}(n)$, modulul q este polinomial în n iar numărul de ecuaţii este de cele mai multe ori nesemnificativ.

Varianta decizională a problemei cere să se distingă între eşantioane (perechi) LWE $(\mathbf{a}, b)$ unde $\mathbf{a}$ este ales uniform din $\mathbb{Z}_q$ iar $b = \mathbf{as} + e$ şi perechi uniforme unde ambii $\mathbf{a}$ şi b sunt aleşi uniform peste $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

Problema LWE este considerată dificilă din punct de vedere computaţional, la fel de dificilă precum aproximarea problemelor de latici în cazul cel mai defavorabil. Regev [Reg05] demonstrează acest rezultat reducând problema LWE la o versiune a problemei SVP asociată laticilor. Pe de altă parte, deocamdată nu se cunosc nici algoritmi cuantici mai buni decât cei clasici pentru problemele de latici, ceea ce întăreşte rezultatul lui Regev privind dificultatea problemei LWE.

Un rezultat important pentru construcţiile noastre viitoare este următorul: schimbarea distribuţiei $A_{s,\chi}$ prin alegerea secretului $\mathbf{s}$ din aceeaşi distribuţie χ ca şi zgomotul (în locul distribtiei uniforme cum se întâmplă în varianta originală a problemei) nu schimbă cu nimic dificultatea problemei LWE. Spunem că în această situaţie distribuţia LWE are forma normală Hermite (HNF - Hermite Normal Form).

Lema 2.1. *(De la LWE către HNF-LWE) [ACPS09]*

Fie $q = p^e$ *cu* p *prim. Există o transformare* T *în timp polinomial care pentru un* $\mathbf{s} \in \mathbb{Z}_q^n$ *şi o distribuţie a erorii* χ *duce* $A_{\mathbf{s},\chi}$ *în* $A_{\overline{\mathbf{x}},\chi}$*, unde* $\overline{\mathbf{x}} \leftarrow \chi^n$ *si* $U(\mathbb{Z}_q^n \times \mathbb{Z}_q)$ *în el însuşi.*

2.2.2 Problema invăţării cu erori într-un inel - Ring-LWE

După cum am văzut, problema LWE beneficiază de proprietăţi de securitate puternice însă eficienţa este scăzută; aceasta se întamplă pentru că matricea $\mathbf{A}$ (care are o dimensiune mare) este folosită drept cheie publică. Pentru a înlătura acest dezavantaj, cercetătorii au propus o versiune mai compactă a problemei LWE definita într-un inel de polinoame astfel: fie $f(x) = x^n + 1 \in \mathbb{Z}[x]$ unde parametrul de securitate n este o putere a lui 2. Fie $q = 1 \bmod 2n$ un număr

prim suficient de mare. Notăm cu $R_q = \mathbb{Z}_q/\langle f(x)\rangle$ inelul de polinoame întregi modulo $f(x)$ si q. Inelul are q^n elemente care pot fi reprezentate ca polinoame de grad mai mic sau egal cu n cu coeficienţi din $\mathbb{Z}_q$. Fie χ distribuţia erorii peste R_q având elemente "mici" (i.e polinoame cu coeficienţi numere mici).

Definitie 2.7. *(Ring-LWE) [BGV12], [Reg10]*

Fie $s \leftarrow R_q$ un element uniform. Ipoteza Ring-LWE spune că perechi de forma $(a_i, b_i = a_i \cdot s + e_i) \in R_q \times R_q$, unde a_i este uniform aleator în R_q iar e_i este ales cu distribuţia χ, sunt dificil de diferentiat, din punct de vedere computaţional, de perechi uniforme peste $(R_q \times R_q)$.

Lyubashevski et al [LPR10] arată că problema Ring-LWE este dificilă prin reducere la problema SVP.

2.2.3 Problema solutiei întregi scurte - SIS

In problema SIS (Small Integer Solution) se dă o secvenţă de vectori $\mathbf{a_1}, ..., \mathbf{a_n}$ aleşi uniform din $\mathbb{Z}_q^n$ şi se cere să se găsească o combinatie a lor cu coeficienţi mici care se însumează la zero.

Definitie 2.8. *($SIS_{q,m,\beta}$)*

Fie q un întreg, o matrice $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ si un numar real β, problema SIS cere găsirea unui vector întreg $\mathbf{z} \in \mathbb{Z}^m \setminus \{\mathbf{0}\}$ aşa încât $\mathbf{Az} = \mathbf{0}$ şi $||z|| \leq \beta$.

Problema SIS este fel de dificil de rezolvat în cazul mediu precum alte probleme standard de latici (SVP) în cazul cel mai defavorabil.

2.3 Constructii criptografice bazate pe latici

In continuare prezentăm câteva primitive criptografice bazate pe latici pe care le vom folosi în construcţiile noastre din capitolele următoare.

2.3.1 Sistem de criptare cu cheie publică

Prezentăm mai jos un sistem de criptare cu cheie publică introdus în [GPV08] ca dualul sistemului de criptare cu cheie publică a lui Regev [Reg05].

Parametrii sistemului sunt $r > \omega(\sqrt{log\ m})$ pentru m pozitiv întreg care descrie distribuţia normală $D_{\mathbb{Z}^m,r}$ din care sunt alese cheile secrete. Matricea $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ este publică şi descrie funcţia $f_{\mathbf{A}}(e) = \mathbf{Ae}$ mod q. Toate operaţiile se

efectueaza în $\mathbb{Z}_q$. Descriem versiunea extinsă a sistemului care permite criptarea unui mesaj de lungime k=poly(n) biţi.

KeyGen Se aleg k vectori $\mathbf{e_i} \leftarrow D_{\mathbb{Z}^m,r}$, $1 \leq i \leq k$ şi se consideră matricea $\mathbf{E} \in \mathbb{Z}_q^{k \times m}$ ale cărei coloane sunt vectorii $\mathbf{e_i}$, ca fiind cheia secretă. Cheia publică $\mathbf{U} \in \mathbb{Z}_q^{n \times k}$ constă din k vectori $\{\mathbf{u_1}, ..., \mathbf{u_k}\}$ aşa încât $\mathbf{u_i} = f_{\mathbf{A}}(e_i) = \mathbf{Ae_i} \bmod q$.

Criptare(pk **= U, M)** Pentru criptarea mesajului M, se alege uniform aleator $\mathbf{s} \in \mathbb{Z}_q^n$, şi se calculează $\mathbf{p} = \mathbf{A^T s} + \mathbf{x_1} \in \mathbb{Z}_q^m$ unde $\mathbf{x_1} \leftarrow \chi^m$. De asemenea, se calculează $\mathbf{c} = \mathbf{U^T s} + \mathbf{x_2} + M \cdot \lfloor q/2 \rfloor \in \mathbb{Z}_q^k$ unde $\mathbf{x_2} \leftarrow \chi^k$. Rezultatul criptării îl constituie perechea (**p,c**).

Decriptare(E, (p,c)) Interpretează **c** ca $[c_1, ..., c_k] \in \mathbb{Z}_q^k$. Pentru toţi $1 \leq j \leq k$ calculează $\bar{b_j} = c_j - \mathbf{e_j^T p} \in \mathbb{Z}_q$. Calculează biţii mesajului M după cum urmează: $b_j = 0$ dacă $\bar{b_j}$ este mai aproape de zero decât de $\lfloor q/2 \rfloor$; altfel $b_j = 1$. Mesajul decriptat este $M = [b_1, ..., b_k]$.

Securitatea sistemului de criptare se bazează pe dificultatea problemei LWE.

2.3.2 Schimb de chei tip Diffie-Hellman bazat pe latici

Să presupunem că A şi B doresc să stabilească o cheie comună secretă având la dispoziţie un canal de comunicare nesigur. Descriem un protocol în criptografia bazată pe latici care îndeplineşte acest scop.

Schimb de chei Diffie-Hellman bazat pe latici

Setări Alege m, n şi q numere întregi pozitive şi o matrice aleatoare $M \in \mathbb{Z}_q^{n \times m}$ publică.

Execuţie

- A alege un vector aleator $\mathbf{x} \in \mathbb{Z}_q^n$ şi un vector scurt "eroare" $\mathbf{e} \in \mathbb{Z}_q^m$ şi îi trimite lui B mesajul

$$A \longrightarrow B : \tilde{\mathbf{x}} = \mathbf{M^T x} + \mathbf{e} \in \mathbb{Z}_\mathbf{q}^\mathbf{m}$$

- B alege un vector aleator scurt $\mathbf{y} \in \mathbb{Z}_q^m$ şi îi răspunde lui A cu mesajul

$$B \longrightarrow A : \tilde{\mathbf{y}} = \mathbf{My} \in \mathbb{Z}_{\mathbf{q}}^{\mathbf{n}}$$

La finalul protocolului, ambii A şi B pot calcula o cheie comună pe baza valorilor generate de ei şi a celor primite.

A calculează $\mathbf{x^T} \cdot \tilde{\mathbf{y}} = \mathbf{x^T My}$ iar B calculează $\tilde{\mathbf{x}} \cdot \mathbf{y} = \mathbf{x^T My} + \mathbf{ey}$ unde $\mathbf{ey}$ este "mic".

Aplicând funcţia $round(\cdot)$, amândoi vor calcula aceeaşi cheie

$$K = round(\mathbf{x} \cdot \tilde{\mathbf{y}}) = round(\tilde{\mathbf{x}} \cdot \mathbf{y})$$

Funcţia $round(\cdot)$ are o variantă de bază folosită în [Reg05] pentru decriptare:

$$round(x) = \begin{cases} 1, & x \in [0, \lfloor q/2 \rfloor] \\ 0, & \text{altfel} \end{cases}$$

In această teză vom folosi o variantă extinsă a funcţiei care rotunjeşte la intervale mai mici $round(x) = a$ if $x \in [a \cdot q/I, (a+1) \cdot q/I]$ unde I este numărul total de intervale.

Securitatea acestui schimb de chei se bazează pe dificultatea problemelor LWE şi SIS.

Capitolul 3

Protocoale de grup în criptografia clasică

In acest capitol introducem următoarea problema practică propunând totodata două protocoale pentru rezolvarea ei.

Problemă

Există o bază de date cu experţi. Pentru evaluarea unui proiect, este ales un manager din baza de date. El îşi alege o echipă de experţi cu următoarele condiţii:

- *Doar membrii echipei stu că au fost aleşi; nimeni din afara echipei nu cunoaşte acest lucru.*
- *Membrii echipei trebuie să stabilească de comun acord o cheie de comunicare pe care numai ei o cunosc.*

Am dezvoltat mai întâi un protocol în care managerul îşi alege o echipă ai cărei membri aleg împreuna o cheie secretă de comunicare. O versiune preliminară a acestui protocol a fost publicată în [AM11]. Protocolul foloseşte o schemă de re-criptare proxy dar nu este foarte eficient datorită numarului mare de mesaje transmise. In al doilea protocol impunem o condiţie suplimentară: anonimitatea membrilor echipei unii faţă de ceilalţi şi sporim eficienţa protocolului renuntând la schema de re-criptare proxy.

3.1 Protocol de stabilire de comun acord de chei folosind o schemă de re-criptare proxy

Prima versiune a protocolului nostru se bazează pe problema vectorului rucsac şi pe o schemă de re-criptare proxy. Am sugerat folosirea unui sistem de criptare rucsac bazat pe problema logaritmului discret peste curbe eliptice [SLC05] cu o mică modificare propusă în [BMH10]. In acest caz, trapa secretă este un vector super-crescator (cheia privată) care permite rezolvarea liniară a problemei. Pentru mai multe detalii despre această construcţie, indicăm lucrarea [SLC05].

Datele iniţiale ale problemei sunt următoarele:

- Bazei de date $\mathcal{B} = \{P_1, ..., P_n\}$ îi asociem un vector rucsac $A = (a_1, ..., a_n)$ şi un numar mare prim $p > max_{1 \leq i \leq n}\{a_i\}$, ambele publice.
- Problema rucsacului este computaţional dificilă pentru oricine din afara bazei de date. Membrii bazei de date pot rezolva problema în timp liniar dacă deţin o trapă secretă.
- Fiecare P_i are o cheie publică e_i pentru criptare, o cheie secretă pentru decriptare d_i şi un algoritm de semnatură (sig_i, ver_i).

3.1.1 Schema de re-criptare proxy

Pentru construcţia protocolului folosim o schemă de re-criptare proxy bazată pe identităţi. In continuare amintim această noţiune.

O schemă de recriptare proxy [BBS98] îi permite unui proxy să transforme un text criptat destinat iniţial lui Alice în acelaşi text criptat destinat lui Bob. Pentru conversie, proxy-ul necesită o cheie de recriptare emisă de Alice care însă nu îi dezvaluie nimic despre textul clar. O astfel de schemă bazata pe identităţi va folosi identitătile drept chei publice.

Notăm o schemă de re-criptare proxy bazată pe identităţi cu IBPRE (Identity-Based Proxy Re-Encryption) şi o definim mai jos.

Setup(k, nr) primeşte la intrare un parametru k şi o valoare nr care indică numărul maxim de recriptări consecutive permise de schemă. Intoarce parametri publici destinaţi utilizatorilor şi o cheie master secretă msk.

KeyGen(id, msk) primeşte la intrare o identitate şi o cheie secretă master şi întoarce o cheie de decriptare sk_{id}.

Enc(pk, id, m) primeşte la intrare parametri publici pk, o identitate id şi un text clar m şi întoarce criptarea lui m sub acea identitate.

RkGen(sk_{id_1}, id_1, id_2) primeşte la intrare o cheie secretă sk_{id_1} şi identităţile id_1, id_2, şi întoarce *o cheie de re-criptare* $rk_{id_1 \to id_2}$.

ReEnc$(c_{id_1}, rk_{id_1 \to id_2})$ primeşte un text criptat c_{id_1} sub indentitatea id_1 şi o cheie de re-criptare $rk_{id_1 \to id_2}$ şi întoarce textul recriptat c_{id_2}.

Dec(c_{id}, sk_{id}) decriptează textul c_{id} folosind cheia secretă sk_{id} şi întoarce mesajul m sau un simbol de eroare $\perp$.

3.1.2 Protocolul nostru bazat pe o schemă IBPRE

PREKAG protocol

1. M alege o echipă $T_m = \{P_{i_1}, ..., P_{i_k}\}$, $I = \{i_1, ..., i_k\}$ şi un vector rucsac $A = (a_1, ..., a_n)$ pe care îl face public.
2. M calculează şi face publică suma $S = \sum_{i \in I} a_i \bmod p$. Această instanţă a problemei rucsac este dificil de rezolvat pentru toţi experţii.
3. M trimite fiecărui P_j $(j \in I)$ un nonce criptat sub identitatea lui împreună cu trapa secretă criptată sub identitatea lui M (deocamdată nici unul dintre P_j nu poate decripta).
4. Fiecare P_j decriptează nonce-ul şi îl trimite înapoi lui M numai dacă acceptă invitaţia de a face parte din echipă; remarcăm faptul că P_j nu poate încă decripta trapa secretă.

   (a) Dacă toţi P_j răspund pozitiv, M calculează (cu algoritmul ReEnc din schema IBPRE) şi face public un vector de chei de re-criptare $Rk = (rk_{id_{i_1}}, ..., rk_{id_{i_k}})$, unde fiecare rk_{id_j} îi corespunde lui P_j, $\forall j \in I$. Apoi fiecare P_j foloseşte cheia publicată pentru el care îi permite să recripteze trapa secretă primită sub propria identitate după care poate decripta uşor.

   (b) Ar putea exista experţi care refuză invitaţia; în aceasţa situaţie, M alege alţi experţi şi repetă paşii 3 şi 4. Dacă ei acceptă, M publică

cheile de re-criptare pentru fiecare membru la fel ca în situaţia precendentă. Desigur că inainte de acest pas, M reactulizează valoarea lui S conform cu noii membri aleşi.

5. Fiecare P_j se află acum în posesia trapei secrete, şi deci poate rezolva problema rucsacului şi afla componenţa echipei; apoi generează un număr aleator α_j şi trimite mesajul $\{a_j, \alpha_j, sig_j(\alpha_j)\}_{e_i}$ fiecărui P_i, unde $i \in I, i \neq j$.

6. Fiecare membru P_j al echipei urmează paşii

   (a) Decriptează cele $k-1$ mesaje primite

   (b) Verifică dacă $\sum_{j \in I} a_j = S \bmod p$;

   (c) Verifică dacă $ver_i(\alpha_i, sig_i(\alpha_i)) = True$, $\forall i \in I - \{j\}$;

   (d) Dacă ambele condiţii sunt satisfacute, atunci calculează cheia secretă comună astfel

$$K = \sum_{j \in I} \alpha_j (\bmod p)$$

7. Ultimul pas verifică dacă toţi membri deţin aceeaşi cheie secretă:

   (a) Fiecare P_j, $j \in I$ generează aleator β_j şi trimite lui M mesajul $\{\{\beta_j\}_{e_j}, a_j, sig_j(a_j)\}_K$;

   (b) M trimite înapoi lui P_i mesajul $\{\beta_j, a_j - 1, sig_M(a_j - 1)\}_K$.

3.2 Protocol de transfer de chei autentificat

In această secţiune descriem o nouă soluţie pentru problema anterioară, impunand condiţia de anonimitate: numai managerul cunoaşte componenţa echipei, membrii echipei nu se cunosc între ei. Spre deosebire de cel anterior, acesta este un protocol de transfer de chei în care managerul alege o cheie pe care o transmite în mod sigur echipei sale. Protocolul beneficiază de un număr redus de mesaje în comparaţie cu protocolul propus în secţiunea precedentă.

Protocolul se desfaşoară astfel (menţionăm ca toti participanţii au acces la o funcţie hash $\mathcal{H} : \{0,1\}^* \longrightarrow \{0,1\}^k$).

GKTA protocol

Stabilirea echipei

1. Pentru fiecare expert $P_i \in \mathcal{B}$, M alege două nonce-uri: α_i si a_i (ambele având k biţi) pe care le trimite semnate cu cheia secretă. Acest pas reprezintă invitaţia adresată tuturor experţilor de a face parte din baza de date.

$$M \longrightarrow P_i : \{id_M, \mathcal{E}_{e_i}(\{\alpha_i\}), g^{a_i}\}_{sig_M}$$

pentru fiecare $1 \leq i \leq n$

2. Dacă P_i acceptă, atunci generează b_i şi trimite răspunsul lui M (semnat cu cheia secretă)

$$P_i \longrightarrow M : \{id_i, \mathcal{D}_{d_i}(\alpha_i), g^{a_i}, g^{b_i}\}_{sig_i}$$

Dintre expetii care au acceptat, M verifică dacă $\mathcal{E}_{e_i}(\mathcal{D}_{d_i}(\alpha_i)) = \alpha_i$ şi selectează echipa $T_M = \{P_{i_1}, \cdots, P_{i_k}\}$, $M \in T_M$. Fie $I = \{i_1, \cdots, i_k\}$. M calculează pentru fiecare membru al echipei valoarea $(g^{b_i})^{a_i} = g^{b_i a_i}$.

Transferul cheii de sesiune

3. M calculează cheia de sesiune pentru echipa aleasă

$$K = \sum_{j \in I} \alpha_j \pmod p$$

generază două nonce-uri n_i si r_i (de câte k biţi) pentru fiecare expert P_i care a răspuns la pasul 2 dar nu a fost ales în echipă şi face publice următoarele valori:

$$\{kp_i = (K - \alpha_i) + g^{a_i b_i}, \mathcal{H}(g^{a_i b_i})\}_{sig_M}$$

$\forall i \in I$, pentru membrii aleşi în echipă

şi

$$\{kp_i = n_i + g^{a_i b_i}, r_i\}_{sig_M}$$

$\forall i \notin I$ pentru cei care nu au fost aleşi

4. Fiecare P_i care a răspuns la pasul 2 procedează astfel:

(a) verifică validitatea semnăturii mesajului primit

(b) dacă semnatura este validă, el calculează valoarea $(g^{a_i})^{b_i} = g^{a_i b_i}$ şi $\mathcal{H}(g^{a_i b_i})$

(c) verifică dacă a doua componentă a mesajului este egală cu $\mathcal{H}(g^{a_i b_i})$; in caz pozitiv, el a fost ales în echipă şi poate recupera cheia secretă din prima componentă a mesajului; altfel, nu a fost ales şi nu poate calcula cheia secretă.

Teorema 3.1. *Protocolul de mai sus este sigur dacă ipoteza Diffie-Hellman clasică este adevarată iar schema de semnatură este sigură.*

Demonstraţia de securitate este realizată în cadrul modelului propus de Bellare şi Rogaway [BR94] pentru protocoale de stabilire de comun acord de chei.

Capitolul 4

Protocoale de grup în criptografia bazata pe latici

In acest capitol prezentăm câteva construcţii criptografice pe care le-am dezvoltat în criptografia bazată pe latici: prima este o schemă de partajare a secretului bazată pe problema LWE publicată în [Geo11], urmează apoi o variantă bazata pe latici a protocolului de transfer de chei publicată în [Geo12]; continuăm cu un protocol de schimb de chei bazat pe latici [GS13] şi o schemă de broadcast encryption anonimă bazată pe latici [Geo13].

4.1 O schemă de partajare a secretelor bazată pe problema LWE

O schemă de partajare a secretelor permite ca un secret să fie partajat de o mulţime de participanţi aşa încât numai o submulţime autorizată (sau chiar întreaga mulţime) poate reconstitui secretul. O schemă (t, n) cu pragul $(t \leq n)$ necesită prezenţa a cel putin t părţi (până la n) pentru refacerea secretului în timp ce orice submulţime de $t-1$ sau mai puţini participanţi nu poate recupera secretul.

In această secţiune propunem o schemă de partajare cu pragul $t = n$ verificabilă (participanţii pot verifica faptul că parţile primite sunt valide). Construcţia este foarte simplă şi foloseşte o schemă de commitment [KTX08] pe care o prezentăm succint mai jos.

4.1.1 Schema de commitment bazată pe latici

O schemă de commitment transformă o valoare secretă s într-o pereche (a, b) unde a este valoarea de commitment care ascunde valoarea secretă s iar b este cheia care permite descoperirea lui s aşa încât: a nu dezvăluie nici o informaţie despre s în timp ce singura modalitate de a descoperi secretul este folosind a şi b. Prezentăm în continuare o schemă de commitment bazată pe latici [KTX08] care are două proprietăti importante: (1) un adversar nu poate face diferenţa între două valori de commitment pentru două şiruri de biţi diferite şi (2) un adversar nu poate crea o nouă pereche de commitment $(\bar{a}, \bar{b})$ pentru acelaşi secret s care să înlocuiască perechea originală.

O funcţie de commitment are la intrare un şir de m biti $\mathbf{x}$ constând dintr-un şir aleator $t = (t_1, ..., t_{m/2}) \in \mathbb{Z}_2^{m/2}$ concatenat cu secretul care se doreşte a fi ascuns $s = (s_1, ..., s_{m/2}) \in \mathbb{Z}_2^{m/2}$ i.e. $\mathbf{x} = (t; s)$.

Setup(n) Având la intrare parametrul n, algoritmul întoarce o matrice $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$.

Commit$(\mathbf{A}, msg = (t; s))$ Având la intrare matricea $\mathbf{A}$ şi secretul s, algoritmul alege $t \in \mathbb{Z}_2^{m/2}$, calculează $c \leftarrow \texttt{Commit}(\mathbf{A}, msg = (t; s)) = A \cdot msg \bmod q$ şi întoarce c si t.

Verif$(\mathbf{A}, c, s, t)$ Algoritmul verifică dacă $s, t \in \mathbb{Z}_2^{m/2}$ şi $\mathsf{Commit}(\mathbf{A}, (t; s)) = c$ şi întoarce 1 dacă totul este verificat sau 0 în caz contrar.

Securitatea acestei scheme se bazează pe dificultatea problemei SIS [KTX08].

4.1.2 Schema noastră de partajare a secretelor

Faza de iniţializare

Notăm participanţii la schemă cu P_i, $1 \leq i \leq n$ şi dealer-ul cu $\mathcal{D}$; el fixează un modul q, un întreg m şi o distribuţie a erorii $\chi \in \mathbb{Z}_q$, toate având aceleaşi dimensiuni precum cele sugerate în [Reg10] pentru problema LWE. $\mathbf{s} \in \mathbb{Z}_q^m$ este secretul care va fi partajat.

Distribuirea părţilor

Dealer-ul procedează astfel:

1. Pentru fiecare $1 \leq i \leq n-1$

   - alege un vector uniform aleator $\mathbf{a}_i$ din $\mathbb{Z}_q^m$, $e_i \in \mathbb{Z}_q$ conform cu distribuţia erorii χ şi execută $\mathsf{Setup}(n)$ pentru a genera matricea publică $\mathbf{M} \in \mathbb{Z}_q^{n \times m}$.
   - calculează perechea $S_i = (\mathbf{a}_i, b_i) = (\mathbf{a}_i, \mathbf{a}_i\mathbf{s} + e_i)$ (adunarea se efectuează în $\mathbb{Z}_q$) reprezentând partea corespunzatoare lui P_i.

2. Pentru ultimul participant, alege $\mathbf{a}_n$ uniform aleator din $\mathbb{Z}_q^m$ iar eroarea aditivă o calculează ca fiind: $e_n = (-e_1) + (-e_2) + \cdots + (-e_{n-1})$. Deci ultima parte este perechea $S_n = (\mathbf{a}_n, \mathbf{a}_n\mathbf{s} - \sum_{i=1}^{n-1} e_i)$.

3. Aplică algoritmul Commit care generează $t_i \in \mathbb{Z}_2^{m/2}$ pentru fiecare $i \in \{1, .., n\}$, calculează şi trimite valorile $c_i \leftarrow \mathsf{Commit}(\mathbf{M}, (t_i; \mathbf{a_i s}))$ şi apoi generează $\bar{t}_i \in \mathbb{Z}_2^{m/2}$ pentru fiecare $i \in \{1, .., n\}$, calculează şi trimite valorile $\bar{c}_i \leftarrow \mathsf{Commit}(\mathbf{A}, (\bar{t}_i; \mathbf{a_i s} + e_i))$

4. trimite fiecărui participant partea S_i împreună cu şirul $\bar{t}_i$.

Verificarea părţilor şi recuperarea secretului

După primirea părţii corespunzătoare, P_i calculează valoarea $\mathsf{Commit}(\mathbf{M}, (\bar{t}_i; b_i)) = \mathbf{M}(t_i; \mathbf{a_i s} + e_i)$ şi o compară cu valoarea primită $\bar{c}_i$; dacă valorile comparate sunt egale, verifică dacă $\sum_{i=1}^{n} c_i = \sum_{i=1}^{n} \bar{c}_i$. Dacă este adevarat, acceptă partea sa de secret ca fiind validă.

Pentru recuperarea secretului, fiecare participant contribuie cu partea sa. Prin însumarea tuturor parţilor,

$$S_1 + \cdots + S_n = \left(\sum_{i=1}^{n} \mathbf{a_i}, \sum_{i=1}^{n} \mathbf{a_i s} \right)$$

eroarea aditivă se anulează şi cum fiecare participant P_i cunoaşte valoarea lui $\mathbf{a_i}$, secretul poate fi calculat imediat $\mathbf{s}$. Odată secretul recuperat nu mai rămane decât să se calculeze $\mathsf{Commit}(\mathbf{M}, (t_i; \mathbf{a_i s}))$ pentru fiecare a_i şi să se compare cu c_i pentru a fi siguri că secretul calculat este valid.

Se poate verifica foarte uşor că orice grup de $n-1$ sau mai puţini utilizatori nu pot calcula secretul decât dacă pot rezolva problema dificilă LWE.

4.2 Un protocol de transfer de chei bazat pe latici

Prezentăm în cele ce urmează o nouă soluţie pentru problema managerului care doreşte să işi formeze o echipă cu care să comunice secret. Propunem un protocol bazat pe latici, care păstreaza proprietatea de anonimitate, obţinuta printr-o metodă diferită, inspirată din [LPQ12].

Stabilirea echipei

1. M schimbă cu fiecare $P_i \in \mathcal{B}$ o cheie tip Diffie-Hellman bazată pe latici :

   (a) M generează aleator $\mathbf{a_i} \in \mathbb{Z}_q^n$, un vector scurt $\mathbf{e_i} \in \mathbb{Z}_q^n$ şi o pereche de chei pentru semnătura digitală $(SK_M, VK_M) \leftarrow \mathcal{G}(\lambda)$ şi trimite următorul mesaj lui P_i

   $$M \longrightarrow P_i : \{\tilde{\mathbf{a}} = \mathbf{M^T} \cdot \mathbf{a_i} + \mathbf{e_i}, \sigma, VK_M\}$$

   pentru fiecare $1 \leq i \leq n$.

   unde $\sigma = \mathcal{S}(SK_M, \tilde{\mathbf{a}})$. Acest pas reprezintă cererea de participare adresată tuturor experţilor din baza de date.

   (b) P_i verifică validitatea semnăturii şi dacă acceptă, generează aleator un vector scurt în $\mathbf{b_i} \in \mathbb{Z}_q^m$, $\mathbf{v_i} \in \mathbb{Z}_q^{n \times t}$ public, perechea de chei pentru semnătura $(SK_i, VK_i) \leftarrow \mathcal{G}(\lambda)$ şi generează matricea $\mathbf{A_i} \in \mathbb{Z}_q^{n \times m}$ (cheie publică) împreună cu o bază scurtă T_i pentru $\Lambda^{\perp}(\mathbf{A_i})$ şi îi trimite înapoi lui M mesajul

   $$P_i \longrightarrow M : \{\tilde{\mathbf{b}} = \mathbf{M} \cdot \mathbf{b_i}, \sigma, VK_i\}$$

   unde $\sigma = \mathcal{S}(SK_i, \tilde{\mathbf{b}})$.

   La sfârşitul acestui pas, M împarte cu fiecare P_i o cheie secretă $k_i = round(\mathbf{a_i} \cdot \tilde{\mathbf{b}}) = round(\tilde{\mathbf{a}} \cdot \mathbf{b_i})$

2. M alege o echipă de experţi $T = \{P_{i_1}, \ldots, P_{i_k}\}$, $M \in T$ dintre cei care au acceptat. Fie $I = \{i_1, \ldots, i_k\}$.

   De acum încolo, ne vom referi la orice literă care are indexul i_j cu indexul j. Pentru simplitate, vom folosi P_j în loc de P_{i_j} s.a.m.d.

Transferul cheii de sesiune

M calculează cheia de sesiune

$$K = \sum_{j \in I} k_j \pmod q$$

şi o publică după cum urmează

- M calculează pentru fiecare membru ales P_j perechea $C_j = (l_j, c_j) = \mathsf{PKE.Enc}(\mathbf{v_j}, K)$ şi valoarea $H_j = \mathcal{H}(k_j)$
- M alege permutarea aleatoare $\pi : \{1, ..., k\} \to \{1, ..., k\}$ şi publică textul criptat ca fiind

$$C = \{VK_M, (H_{\pi(1)}, C_{\pi(1)}), ..., (H_{\pi(k)}, C_{\pi(k)}), \sigma\}$$

unde $\sigma = \mathcal{S}(SK_M, C)$.

3. Fiecare P_i: verifică dacă semnătura mesajului este validă; dacă verificarea eşuează, atunci el renunţă la protocol; altfel, calculează $H = \mathcal{H}(k_i)$ şi dacă $H \neq H_j$ pentru fiecare $j \in \{1, ..., k\}$ înseamnă că el nu a fost ales în echipă.

   Altfel, pentru fiecare $1 \leq d \leq t$ el generează $\mathbf{e_{j_d}} = \mathsf{SamplePre}(\mathbf{A_j}, T_j, \mathbf{v_d}, r(k+1))$, notează cu $\mathbf{e} = [\mathbf{e_{j_1}}, ..., \mathbf{e_{j_d}}]$ şi recuperează mesajul M = $\mathsf{PKE.Dec}(\mathbf{e}, C_j)$ reprezentând cheia secretă partajată cu restul echipei.

Corectitudinea schemei este asigurată de sistemul de criptare dual sistemului Regev din [GPV08] şi de proprietăţile trapelor secrete pe care le-am prezentat în capitolul 2.

4.3 Protocol de stabilire de comun acord de chei bazat pe latici

In această secţiune prezentăm un protocol pe care l-am obţinut transformând protocolul de stabilire de chei propus de Burmester si Desmedt [BD95] în criptografia clasica în acelaşi protocol în criptografia bazată pe latici. Principala noastra contribuţie a fost să identificăm şi să rezolvăm câteva aspecte ne-triviale care apar la conversie în demonstraţia de securitate.

4.3.1 Protocol de stabilire de chei Burmester şi Desmedt

Burmester şi Desmedt [BD95] prezintă primul protocol practic şi sigur de stabilire de chei. Ei au extins protocolul Diffie-Hellman la n părţi obţinând cel mai eficient protocol cu privire la numărul de runde.

Cadrul de desfăşurare a protocolului : există o reţea de n utilizatori $U_1, ..., U_n$ care doresc să stabilească o cheie secretă comună. Presupunem un grup şi un generator $g \in \mathbb{G}$ publici utilizatorilor. Considerăm utilizatorii aranjaţi în cerc aşa încât după U_n urmează U_1.

Pasul 1 Fiecare utilizator U_i alege aleator $r_i \in \mathbb{Z}_q$ şi transmite $z_i = g^{r_i}$.

Pasul 2 Fiecare utilizator U_i transmite $X_i = (z_{i+1}/z_{i-1})^{r_i}$

Calculul cheii Fiecare U_i calculează cheia de sesiune astfel

$$sk_i = z_{i-1}^{n \cdot r_i} \cdot X_i^{n-1} \cdot X_{i+1}^{n-2} \cdots X_{i+n-2}.$$

Este uşor de verificat faptul că toţi utilizatorii calculează aceeaşi cheie $g^{r_1 r_2 + r_2 r_3 + ... + r_n r_1}$.

Securitatea protocolului se bazează pe dificultatea problemei logaritmului discret.

4.3.2 Protocol de stabilire de chei bazat pe latici

Prezentăm aici conversia protocolului precedent în latici pe care îl dezvoltăm în cadrul dat de problema Ring-LWE. Reamintim că R_q este inelul de polinoame întregi $R_q = \mathbb{Z}_q^n / \langle x^n + 1 \rangle$ unde n este o putere a lui 2 şi q este un modul prim aşa încât $q = 1 \bmod 2n$. χ este distribuţia zgomotului definită pe elemente mici din R_q. Fie m un element public din R_q.

Incepem cu o versiune preliminară în care am întâmpinat dificultăţi la conversie şi pe care am modificat-o rezultând o a doua versiune protocolului pe care o vom prezenta puţin mai târziu.

LKAG1- un protocol de stabilire de chei bazat pe latici (prima versiune)

Pasul 1. Fiecare U_i, $i = 1, ..., n$ alege aleator a_i si b_i din distribuţia χ şi transmite tuturor

$$z_i = m \cdot a_i + b_i$$

Pasul 2. Fiecare U_i, $i = 1, ..., n$ alege aleator $\tilde{b}_i$ din distribuţia χ şi transmite tuturor

$$X_i = (z_{i+1} - z_{i-1}) \cdot a_i + \tilde{b}_i$$

unde indicii sunt consideraţi în ciclu.

Calculul cheii Fiecare U_i, $i = 1, ..., n$ calculează cheia de sesiune

$$K_i = \text{round}(n(z_{i-1} \cdot a_i + b_i) + (n-1)X_i + (n-2)X_{i+1} + ... + X_{i-2})$$

Atragem atenţia asupra faptului că în pasul 3 fiecare utilizator calculează cheia $K = m(a_1a_2 + a_2a_3 + ... + a_na_1)$ + zgomot.

Funcţia *round* folosită aici este cea prezentată în capitolul 2.

Securitatea acestui protocol se bazează pe dificultatea problemei DDH-RLWE pe care o definim în continuare.

DDH-RLWE. Fiind dat un tuplu $(g, y_1 = g \cdot x + e_x, y_2 = g \cdot y + e_y, \Gamma)$ unde g este ales uniform aleator din R_q, x, y, e_x, e_y sunt alese cu distribuţia χ, se cere să se determine cu probabilitate semnificativă dacă tuplul anterior în care $\Gamma = y_1 \cdot y + e_3$ cu e_3 ales independent din distribuţia χ este acelaşi cu tuplul în care Γ este ales uniform aleator şi independent din R_q.

O condiţie suficientă pentru această problemă este prezentată in continuare.

Lema 4.1. *Dificultatea problemei DDH-RLWE se bazează pe dificultatea problemei HNF-RLWE.*

Teorema 4.1. *Protocolul LKAG1 este sigur împotriva atacurilor pasive în ipoteza în care problema DDH-RLWE este dificilă.*

Demonstraţia acestui rezultat am organizat-o ca pe o secvenţă de jocuri între adversar şi o entitate care emite provoari. In fiecare joc, considerăm distribuţia transcriptului şi a cheii de sesiune generate pe baza valorilor z_i dar şi a altora pe care noi le-am definit. Observăm însă că în relatia de mai jos există scurgere de informaţie.

- $\sum_i X_i = \sum_i (b_{i+1} - b_{i-1}) \cdot a_i + \tilde{b}_i =$ "scurt".

Elementele a_i, b_i şi $\tilde{b_i}$ au valori mici şi sunt secrete iar valorile X_i sunt publice, de aici rezultând scurgerea de informaţii. Pentru a evita această problemă, am încercat să folosim valori mult mai mari pentru $\tilde{b_i}$ decât suma celorlalte valori b_i şi a_i aşa încât să le mascăm cu valoarea lui $\tilde{b_i}$. Insă această soluţie nu este cea mai bună pentru că în acest fel creşte şi factorul de aproximare al problemei de latici aferente, problema devenind mai vulnerabilă în faţa unui atac bazat pe reducerea bazei laticii.

Prezentăm, aşadar, o soluţie mai bună care evită ineficienţa datorată soluţiei anterioare. Observăm faptul că protocolul rămâne corect indiferent de valoarea exactă a sumei $\sum_{i=1}^{n} X_i$ depinzând doar de dimensiunea ei (e important ca suma să fie mică). Prin urmare, impunem condiţia ca $\sum_{i=1}^{n} X_i = 0$, ceea ce putem obţine astfel: ultimul utilizator U_i nu mai trimite nimic la ultimul pas ci calculează valoarea $X_n = -\sum_{i=1}^{n-1} X_i$. Protocolul modificat este:

LKAG2- un protocol de stabilire de chei bazat pe latici (a doua versiune)

Pasul 1. Fiecare U_i, $i = 1, ..., n$ alege aleator a_i şi b_i din distribuţia χ şi transmite tuturor

$$z_i = m \cdot a_i + b_i$$

Pasul 2. Fiecare U_i, $i = 1, ..., n-1$ alege aleator $\tilde{b_i}$ din distribuţia χ şi transmite tuturor

$$X_i = (z_{i+1} - z_{i-1}) \cdot a_i + \tilde{b_i}$$

unde indicii sunt consideraţi in ciclu.

La acest pas, U_n doar calculează $X_n = -\sum_{i=1}^{n-1} X_i$.

Calculul cheii Fiecare U_i, $i = 1, ..., n$ calculează cheia de sesiune

$$K_i = \text{round}(n(z_{i-1} \cdot a_i + b_i) + (n-1)X_i + (n-2)X_{i+1} + ... + X_{i-2})$$

Ipoteza pe care se bazează securitatea protocolului modificat este acum alta.

Teorema 4.2. *Protocolul LKAG2 este sigur împotriva atacurilor pasive în ipoteza în care problema Ring-LWE este dificilă.*

Linia demonstraţiei securităţii protocolului rămâne aceeaşi precum la protocolul precedent.

4.3.3 Protocol de stabilire de chei autentificat bazat pe latici

In subsecţiunea precedentă, am prezentat un protocol sigur împotriva atacurilor pasive. Pentru securitate împotriva atacurilor active, propunem metoda definită în [KY03] care transformă un protocol sigur împotriva atacurilor pasive într-unul sigur impotriva atacurilor active. Pornind de la protocolul nostru, adăugam câteva mesaje semnate care asigură autentificarea (şi deci rezistenţa în faţa atacurilor active). Vom folosi o schemă de semnătura digitală $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$

LAKAG - un protocol de stabilire de chei autentificat bazat pe latici

- Fiecare U_i, $i = 1, ..., n$ execută $\mathsf{Gen}(k)$ pentru a genera chei de verificare/semnare (pk_i, sk_i).
- Fiecare U_i, $i = 1, ..., n$ alege o valoare aleatoare $r_i \in \{0,1\}^k$ şi transmite $U_i|0|r_i$ (mesajul are numărul de secvenţă "0").
- Fiecare instanţă Π^j_U setează o valoare $\mathsf{nonces}^j_U = U_1|r_1|...|U_n|r_n$.
- Fiecare instanţă Π^i_U alege aleator a_i and b_i cu distribuţia χ, calculează

$$z_i = m \cdot a_i + b_i$$

şi $\sigma = \mathsf{Sign}_{sk_i}(j|z_i|\mathsf{nonces}_{U_i})$ şi trimite $U|j|z_i|\sigma$ (unde j este numărul de secvenţă corespunzător).

- Fiecare instanţă Π^i_U, la primirea mesajului $V|j|m|\sigma$, verifică:
  1. $V \in pid^i_U \setminus \{U\}$;
  2. j este următorul număr de secvenţă aşteptat al mesajelor din V;
  3. semnătura este validă $\mathsf{Vrfy}_{pk_V}(j|m|\mathsf{nonces}^i_U, \sigma) = 1$.

  Dacă numai una dintre aceste condiţii nu este adevarată, Π^i_U renunţă, altfel alege $\tilde{b}_i$ cu distribuţia χ şi trimite

$$X_i = (z_{i+1} - z_{i-1}) \cdot a_i + \tilde{b}_i$$

La acest pas, fiecare instanţă a lui U_n calculează $X_n = -\sum_{i=1}^{n-1} X_i$.

- Fiecare U_i, $i = 1, ..., n$ calculează cheia comună

$$K_i = \text{round}(n(z_{i-1} \cdot a_i + b_i) + (n-1)X_i + (n-2)X_{i+1} + ... + X_{i-2})$$

4.4 Schemă de criptare broadcast anonimă

In această secţiune prezentăm o nouă construcţie criptografică bazată pe latici: o schemă de criptare broadcast anonimă obtinuţă prin conversia schemei de criptare propusă în [LPQ12] din cadrul clasic în cadrul criptografiei bazate pe latici. Folosim două primitive criptografice de bază: sisteme hint bazate pe etichete şi sisteme de criptare sigure împotriva atacurilor cu text criptat ales.

O versiune preliminară [Geo13] a conţinutului acestei secţiuni a fost acceptată pentru prezentare şi publicare la conferinţa AsiaARES 2013.

Schema de criptare broadcast anonimă aduce în plus, fată de precedentele scheme de criptare broadcast, conceptul de anonimitate: utilizatorii implicaţi în schemă nu se cunosc între ei. Libert, Paterson si Quaglia [LPQ12] propun o aplicaţie practică imediată: în sistemele de Pay-TV, utilizatorii care plătesc pentru aceleaşi programe nu trebuie să se cunoască între ei.

Prezentăm mai întai definiţia unei scheme de criptare broadcast

Definitie 4.1. *O schemă de criptare broadcast cu parametrul de securitate λ şi mulţimea utilizatorilor $U = \{1, ..., n\}$ constă din algoritmii*

`Setup`(λ, n) *are la intrare parametrul λ şi numărul de utilizatori şi întoarce o cheie publică master MPK şi o cheie secretă master MSK.*

`KeyGen`(MPK, MSK, i) *are la intrare MPK, MSK şi $i \in U$ şi întoarce cheia secretă sk_i corespunzătoare utilizatorului i.*

`Enc`(MPK, m, S) *are la intrare MPK şi un mesaj m care se doreşte a fi transmis unei submulţimi a utilizatorilor $S \subseteq U$ şi întoarce textul criptat c.*

`Dec`(MPK, sk_i, c) *are la intrare MPK, o cheie secretă sk_i şi un text criptat c şi întoarce mesajul m sau un simbol de eşec.*

Un sistem hint anonim bazat pe etichete (TAHS) [LPQ12] este un fel de sistem de criptare sub o etichetă t şi o cheie publică pk. Ieşirea este o pereche (U, H) unde H este un hint. Dacă se folosesc două chei publice diferite, ar trebui să fie dificil computaţional a spune dacă perechile rezultate sunt diferite. Sistemul constă din următorii algoritmi:

`KeyGen`(λ) având la intrare parametrul de securitate λ, întoarce o pereche de chei (sk, pk).

`Hint`(t, pk, r) are la intrare o cheie publică pk şi o etichetă t; întoarce o pereche (U, H) constând dintr-o valoare U şi un hint H. Se cere ca U să depindă numai de un r aleator iar nu de pk.

`Invert`(sk, t, U) are la intrare o valoare U, o etichetă t şi o cheie secretă sk. Intoarce un hint H sau un simbol de eroare.

Corectitudinea cere ca pentru orice pereche $(sk, pk) \leftarrow$ `KeyGen`(λ) şi orice r aleator, dacă $(U, H) \leftarrow$ `Hint`(t, pk, r), atunci `Invert`$(sk, t, U) = H$.

Pentru a arăta că putem construi această primitivă în criptografia bazată pe latici, dăm un exemplu de sistem hint anonim bazat pe problema DDH-RLWE pe care am introdus-o în secţiunea precedentă. Acesta este echivalentul sistemului hint introdus in [LPQ12] şi bazat pe problema decizională Diffie-Hellman clasică

Lucrăm din nou în cadrul inelului de polinoame întregi R_q descris în capitolul 2, secţiunea 2.2.2 $R_q = \mathbb{Z}_q^n / \langle x^n + 1 \rangle$ cu n o putere a lui 2 iar q un modul prim aşa încât $q = 1 \bmod 2n$. Reamintim că distribuţia χ este concentrată pe polinoame din R_q cu coeficienţi mici; s este un element fixat din R_q.

Atragem atenţia asupra faptului că, spre deosebire de sistemul hint original bazat pe etichete, algoritmul nostru `Hint` întoarce o valoare H_1 uşor diferită de valoarea H_2 calculată în urma algoritmului `Invert` (printr-o valoare mică din χ aşa cum se arată mai jos) şi numai deţinătorul cheii secrete sk poate calcula o valoare H din H_1 si H_2. Subliniem faptul că valoarea finală H este aceeaşi pentru fiecare utilizare a schemei.

`KeyGen`(λ) are la intrare valorile aleatoare $x_1, x_2, y_1, y_2, e_1, e_2, \tilde{e}_1, \tilde{e}_2 \leftarrow \chi$ şi calculează $X_i = s \cdot x_i + e_i$ şi $Y_i = s \cdot y_i + \tilde{e}_i$. Cheia publică este $pk = (X_1, X_2, Y_1, Y_2)$ iar cheia secretă este $sk = (x_1, x_2, y_1, y_2)$.

`Hint`(t, pk, r) alege e, e_x, e_y cu distribuţia χ şi calculează (U, H_1) astfel

$$U = s \cdot r + e; \quad H_1 = (V, W) = ((X_1 \cdot t + e_x + X_2)r, (Y_1 \cdot t + e_y + Y_2)r)$$

$\texttt{Invert}(sk, t, U)$ transformă sk in (x_1, x_2, y_1, y_2), calculează

$$H_2 = (V, W) = (U(t \cdot x_1 + x_2), U(t \cdot y_1 + y_2))$$

şi verifică dacă diferenţa $H_2 - H_1$ este mică (i.e. are distributia χ). Dacă acest lucru este adevărat, întoarce

$round(H_2) = (round(U(t \cdot x_1 + x_2)), round(U(t \cdot y_1 + y_2))) = round(H_1) = H$

Considerăm următorul model de securitate care asigură anonimitatea unei sistem hint bazat pe etichete.

Definitie 4.2. *[LPQ12]*

Un sistem hint bazat pe etichete este anonim dacă nu există nici un adversar în timp polinomial care să aibă avantaj ne-neglijabil în următorul joc:

1. *Adversarul $\mathcal{A}$ alege o etichetă $\tilde{t}$ pe care o trimite entităţii care lansează provocarea.*
2. *Entitatea care provoacă generează două perechi $(sk_0, pk_0), (sk_1, pk_1) \leftarrow \texttt{KeyGen}(\lambda)$ şi dă adversarului pk_0, pk_1.*
3. *Următoarea etapă se repetă de un număr polinomial de ori: $\mathcal{A}$ invocă un oracol de verificare pe tripletul (U, H, t) aşa încât $t \neq \tilde{t}$. In replică, entitatea care provoacă întoarce biţii $d_0, d_1 \in \{0, 1\}$ unde $d_0 = 1$ dacă şi numai dacă $H = \texttt{Invert}(sk_0, t, U)$ şi $d_1 = 1$ dacă şi numai dacă $H = \texttt{Invert}(sk_1, t, U)$.*
4. *In etapa de provocare, entitatea care provoacă alege aleator un bit $b \leftarrow \{0, 1\}$ şi $\tilde{r} \leftarrow R_q$ aleator şi întoarce $(\tilde{U}, \tilde{H}) = \texttt{Hint}(\tilde{t}, pk_b, \tilde{r})$.*
5. *Lui $\mathcal{A}$ îi este permis să adreseze în continuare cereri dar care să nu implice eticheta $\tilde{t}$.*
6. *$\mathcal{A}$ întoarce bit-ul $b'\{0, 1\}$ şi castigă jocul dacă $b' = b$.*

Lema 4.2. *Sistemul hint bazat pe etichete pe care l-am definit mai sus este anonim dacă ipoteza DDH-RLWE este adevarată în inelul R_q.*

Capitolul 5

Concluzii şi cercetări viitoare

Contribuţiile pe care le-am adus în această teză reprezintă o tranziţie de la criptografia clasică spre criptografia bazată pe latici. Rezultatele proprii pot fi gasite în articolele [Mih10], [AM11], [AG12], [Geo11], [Geo12], [Geo13], [GS13].

Vom face o scurtă trecere în revistă a rezultatelor originale din această teză.

In Capitolul 2 am propus o problemă practică şi am introdus două protocoale de stabilire de chei drept soluţie:

- primul protocol bazat pe o schemă de recriptare proxy rezolvă problema însă este ineficient datorită numărului mare de mesaje necesare pentru a stabili echipa; de asemenea, schema de recriptare proxy reprezintă o sursă de ineficienţă, şi arătăm, în următoarea versiune a protocolului, că aceasta poate fi evitată.

- al doilea protocol este mai eficient: am exclus schema de recriptare proxy şi am redus numărul de mesaje schimbate între experti; am introdus de asemenea o cerinţă suplimentară practică: anonimitatea membrilor echipei unii faţă de ceilalţi. Considerăm că acest protocol ar putea fi eficientizat şi mai mult prin găsirea altei metode de a obţine anonimitatea (decât schimbul de chei Diffie-Hellman executat de manager cu fiecare membru).

In Capitolul 3, am introdus mai multe construcţii criptografice bazate pe latici:

- am propus o schemă de partajare a secretelor unanimă, a cărei securitate rezidă în dificultatea problemei LWE şi a problemei SIS; schema oferă pentru participanţi şi posibilitatea de a verifica dacă părţile primite sunt valide. Credem că schema ar putea fi îmbunătăţită schimbând pragul de la $k = n$ la $k < n$.

- am propus o construcţie echivalentă pentru protocolul de transfer de chei anonim în criptografia bazată pe latici, obtinând anonimitatea prin altă metodă, mai eficientă, decât cea propusă iniţial. Din nou, o idee de cercetare ulterioară ar consta în găsirea unei metode de păstra proprietatea de anonimitate fără a mai folosi schimburi de chei Difie-Hellman.

- am dezvoltat un protocol de stabilire de chei bazat pe latici, pornind de la unul foarte eficient din criptografia clasică; protocolul beneficiază de un număr mic de runde (numai două) şi a fost dezvoltat în contextul dat de problema Ring-LWE, care oferă implementare compactă şi eficientă. Am identificat şi am rezolvat problemele aparute la conversie în ceea ce priveşte demonstraţia de securitate. De asemenea, am introdus problema DDH-RLWE, corespondenta problemei decizionale Diffie-Hellman în criptografia bazată pe latici; am demonstrat dificultatea acestei probleme bazându-ne pe dificultatea problemei LWE.

- am transferat o schemă anonimă de criptare broadcast din criptografia clasică [LPQ12] în criptografia bazata pe latici. Principala contribuţie a fost translatarea in latici a schemei de hint bazată pe tag-uri prin care se obţine anonimitatea. Am lucrat, ca şi la protocolul anterior, în cadrul dat de problema Ring-LWE, obţinând o variantă eficientizată a schemei. Aşa cum se afirmă în lucrarea originală, mai sunt multe alte proprietăţi ce trebuiesc luate în considerare pentru ca schemă să devină cu adevărat practică.

Domeniul criptografiei bazat pe latici este încă unul tânăr dar care evoluează rapid. Mai este foarte mult de lucru până când vom avea o colecţie completă de primitive criptografice sigure bazate pe dificultatea problemelor laticeale. De asemenea, aceste probleme aferente laticilor sunt într-un stadiu de continuă îmbunătăţire şi eficientizare.

După cum am subliniat mai sus, contribuţiile noastre lasă loc şi probleme deschise pentru studii viitoare. O problemă comună a primelor trei protocoale propuse este obţinerea anonimităţii fără a folosi schimburi de chei Diffie-Hellman cu fiecare utilizator. O alta problemă importantă rămasă deschisă este dezvoltarea unei scheme de partajare a secretelor cu pragul $k < n$. Considerăm că, în acest scop, am putea folosi o metoda similară cu cea din [BD10] unde este descrisă o schemă de decriptare cu prag bazată pe latici (aceasta implică existenţa unui grup de membri care numai împreună pot decripta secretul, iar nu individual aşa cum se întâmplă la schemele de criptare clasice).

In viitor, am dori să ne îndreptăm atenţia şi asupra construcţiilor din criptografia clasică bazate pe perechi biliniare. Acestea, la rândul lor, fiind corelate cu problema logaritmului discret, sunt uşor de atacat în criptografia cuantică. Conversia acestor tipuri de construcţii în criptografia bazată pe latici este mai dificilă, pentru că deocamdată nu există perechi biliniare bazate pe latici (deşi foarte recent a fost propus un candidat valid pentru aplicaţii multiliniare derivate din latici ideale [GGH12]). Există câteva construcţii în latici echivalente ca funcţionalitate celor din criptografia bazată pe perechi biliniare: scheme de criptare bazate pe identitate, scheme de criptare fuzzy bazate pe identitate, scheme de criptare bazate pe atribute. Desigur că mai sunt multe alte primitive criptografice bazate pe perechi biliniare care aşteaptă să fie translatate în latici. La o astfel de construcţie lucrăm şi noi în prezent: încercăm să adaptăm la criptografia bazată pe latici un protocol asimetric de stabilire de chei [WMS+09]. In acest tip de protocol, participanţii stabilesc de comun acord o cheie de criptare. Acesteia îi corespund diferite chei de decriptare, fiecare dintre ele putând fi calculate numai de către un membru al grupului.

Bibliografie

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. *Advances in Cryptology-CRYPTO 2009*, pages 595–618, 2009. 9

[AG12] Adrian Atanasiu and Adela Georgescu. A secure authenticated group key transfer protocol. In *Infusing Research and Knowledge in South- East Europe - 7th South East European Doctoral Student Conference*, pages 625–634. South-East European Research Center, 2012. 3, 31

[Ajt96] Miklos Ajtai. Generating hard instances of lattice problems. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 99–108. ACM, 1996. 1

[AM11] Adrian Atanasiu and Adela Mihaita. A key agreement protocol based on identity-based proxy re-encryption. In *Proceedings of the 2011 International Conference on Security and Management SAM 2011*, pages 738–742. CSREA Press, 2011. 3, 13, 31

[BBS98] Matt Blaze, Gerrit Bleumer, and Martin Strauss. Divertible protocols and atomic proxy cryptography. *Advances in CryptologyEUROCRYPT8*, pages 127–144, 1998. 14

[BD95] Mike Burmester and Yvo Desmedt. A secure and efficient conference key distribution system. In *Advances in CryptologyEUROCRYPT4*, pages 275–286. Springer, 1995. 23, 24

[BD10] Rikke Bendlin and Ivan Damgård. Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems. *Theory of Cryptography*, pages 201–218, 2010. 32

[BGV12] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. Fully homomorphic encryption without bootstrapping. *Innovations in Theoretical Computer Science, ITCS*, pages 309–325, 2012. 10

[BMH10] Jingguo Bi, Xianmeng Meng, and Lidong Han. Cryptanalysis of two knapsack public-key cryptosystems, 2010. 14

[BR94] Mihir Bellare and Philip Rogaway. Entity authentication and key distribution. In *Advances in CryptologyCRYPTO93*, pages 232–249. Springer, 1994. 18

[Gen09] Craig Gentry. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, 2009. `crypto.stanford.edu/craig`. 2

[Geo11] Adela Georgescu. A lwe-based secret sharing scheme. *IJCA Special Issue on Network Security and Cryptography*, NSC(3):27–29, December 2011. Published by Foundation of Computer Science, New York, USA. 3, 19, 31

[Geo12] Adela Georgescu. An lwe-based key transfer protocol with anonymity. *Tatra Mountains Mathematical Publications*, 53(1):119–135, 2012. 3, 19, 31

[Geo13] Adela Georgescu. Anonymous lattice-based broadcast encryption. In *Proceedings of the First Conference ICT-EurAsia 2013*, pages 30–40. Springer, 2013. 4, 19, 28, 31

[GGH12] Sanjam Garg, Craig Gentry, and Shai Halevi. Candidate multilinear maps from ideal lattices and applications. Technical report, Cryptology ePrint Archive, Report 2012/610, 2012. http://eprint. iacr. org, 2012. 2, 33

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 197–206. ACM, 2008. 10, 23

[GS13] Adela Georgescu and Ron Steinfeld. Lattice-based key agreement protocols. *In preparation*, 2013. 3, 19, 31

[KTX08] Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. *Advances in Cryptology-ASIACRYPT 2008*, pages 372–389, 2008. 19, 20

[KY03] Jonathan Katz and Moti Yung. Scalable protocols for authenticated group key exchange. *Advances in Cryptology-CRYPTO 2003*, pages 110–125, 2003. 27

[LPQ12] Benot Libert, Kenneth G. Paterson, and Elizabeth A. Quaglia. Anonymous broadcast encryption: adaptive security and efficient constructions in the standard model. *Public Key Cryptography–PKC 2012*, pages 206–224, 2012. 22, 28, 29, 30, 32

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. *Advances in Cryptology–EUROCRYPT 2010*, pages 1–23, 2010. 10

[Mih10] Adela Mihaita. Lattice problems in cryptography. *JITCS - Journal of Information Technology and Communication Security*, 37(1):19–28, 2010. 3, 31

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 84–93. ACM, 2005. 8, 9, 10, 12

[Reg10] Oded Regev. The learning with errors problem. *Invited survey in CCC*, 2010. 10, 20

[Sho97] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997. 1

[SLC05] Pin-Chang Su, Erl-Huei Lu, and Henry Ker-Chang Chang. A knapsack public-key cryptosystem based on elliptic curve discrete logarithm. *Applied mathematics and computation*, 168(1):40–46, 2005. 14

[WMS+09] Qianhong Wu, Yi Mu, Willy Susillo, Bo Qin, and Josep Domingo-Ferrer. Asymmetric group key agreement. *Advances in Cryptology-EUROCRYPT 2009*, pages 153–170, 2009. 33