

Dos esquemas de compartición de secretos

H. Tapia-Recillas y N. Gutiérrez-Herrera

Dpto. de Matemáticas
UAM-I,

Dpto. Computación

Cinvestav

Abril, 2016



C O N T E N I D O

- 1 **Introducción**
- 2 **Esquema basado en $\mathbb{F}_q[x]$**
- 3 **Ejemplo**
- 4 **Esquema basado en polinomios de Dickson**
- 5 **Ejemplo**

Introducción

Los esquemas de compartición de secretos (ECSs) fueron introducidos en forma independiente por Shamir y Blakly (1970). Desde entonces han sido objeto de estudio y tienen una gran variedad de aplicaciones en seguridad informática.

Los ECSs que aparecen en la literatura están basados en diversas estructuras matemáticas entre los que se pueden mencionar anillos (enteros racionales, polinomios, euclidianos, enteros modulares), campos (finitos); y conceptos como el algoritmo de Euclides, interpolación de polinomios, códigos lineales, diversos tipos de funciones booleanas (casi-bent, casi-perfectamente no-lineales (APN), entre otros.

Introduction (cont.)

En términos generales un ECSs se puede describir de la siguiente manera:

- Se desea distribuir un secreto S entre un conjunto finito de participantes $\mathcal{P} = \{P_1, \dots, P_n\}$.
- Cada participante P_i (o un subconjunto de participantes) recibe de una Autoridad (\mathcal{A}) una porción s_i (share) del secreto S .
- El secreto se puede recuperar a partir de las porciones de los miembros de subconjuntos autorizados del conjunto de participantes.
- El secreto no se puede recuperar de subconjuntos no autorizados.

Introduction (cont.)

En general se pide que el conjunto de participantes sea **monótono**:

Si Γ_1 y Γ_2 son conjuntos autorizados entonces $\Gamma_1 \cup \Gamma_2$ también es un subconjunto autorizado para recuperar el secreto.

Esquema basado en $\mathbb{F}_q[x]$

A continuación recordaremos algunos conceptos y resultados que servirán para describir el ECSs.

- 1 \mathbb{F}_q : campo finito con $q = p^r$ elementos.
- 2 $A = \mathbb{F}_q[x]$: anillo de polinomios en x con coeficientes en \mathbb{F}_q .
- 3 Para $f \in A$, $\phi_q(f) = |\{g(x) \in A : (f, g) = 1\}|$: función Phi de Euler en A .

Esta función tiene varias aplicaciones incluyendo el estudio de polinomios linearizados y bases normales en campos finitos.

Observaciones:

- 1 Si $f \in A$ es irreducible de grado e : $\phi_q(f) = |\mathbb{F}_{q^e}^*| = q^e - 1$.
- 2 Si $f, g \in A$ irreducibles de grados r, s :

$$\phi_q(fg) = (q^r - 1)(q^s - 1)$$

El esquema

Paso inicial:

La autoridad \mathcal{A} hace lo siguiente:

- 1 Elige aleatoriamente $P_1, P_2 \in A$ distintos e irreducibles de grados r, s , $a = P_1 P_2$ y determina $\phi_q(a)$. Se puede suponer que $r = s$.
- 2 Elige aleatoriamente $f \in A$ de grado grande tal que $\gcd(\phi_q(f), \phi_q(a)) = 1$ y $(f(x), a(x)) = 1$.
- 3 Hace público $(f(x), a(x), q)$ y guarda en secreto $(P_1, P_2, \phi_q(a))$.

Cada participante hace lo siguiente:

- 1 Elige aleatoriamente $s_i \in [2, q^{r+s}]$ y los guarda en secreto.
- 2 Determina $R_i(x) = f(x)^{s_i} \bmod a(x)$.
- 3 Envía $R_i(x)$ a la autoridad \mathcal{A} .



Dividiendo el secreto:

La autoridad \mathcal{A} verifica que los varios $R_i(x)$ sean distintos, de otra manera solicita a los participantes elegir otro s_i . Con esta información hace lo siguiente:

- 1 Elige aleatoriamente $e \in [1, \phi_q(a)]$ tal que $\gcd(e, \phi_q(a)) = 1$ y determina $d = e^{-1} \bmod \phi_q(a)$.
- 2 Para $i = 1, 2, \dots, n$ determina $Q_i(x) = R_i(x)^d \bmod a(x)$.
- 3 Calcula $R_0 = f(x)^d \bmod a(x)$.
- 4 Hace público $(R_0(x), e)$.

Dividiendo el secreto (cont.)

El secreto a ser compartido es $s(x) = P_1(x)b(x)$, múltiplo de $P_1(x)$, de grado menor que el grado de $a(x)$.

Sean $\Gamma_1, \Gamma_2, \dots, \Gamma_t$, los subconjuntos de participantes autorizados para recuperar el secreto, no necesariamente de la misma cardinalidad ni ajenos. Sea $\Gamma_j = \{P_i : i \in A_j\}$ uno de tales subconjuntos. La autoridad \mathcal{A} determina:

$$b_j(x) = \left(s(x) \prod_{i \in A_j} Q_i(x) \right) \bmod a(x),$$

y hace público la pareja $(j, b_j(x))$, $1 \leq j \leq t$.

Obsérvese que parte del secreto se asigna a los subconjuntos de participantes autorizados y no a los participantes, como se hace otros ECSs.



Recuperando el secreto

Si el subconjunto Γ_j autorizado desea recuperar el secreto, cada elemento $P_i \in \Gamma_j$, i.e., con $i \in A_j$, hace lo siguiente:

- 1 Determina $T_i(x) = R_0^{-s_i}(x) \bmod a(x)$.
- 2 Envía $T_i(x)$ y $R_i(x)$ a la autoridad \mathcal{A} .

Recuperando el secreto (cont.)

La autoridad hace lo siguiente:

- 1 Determina $T_i^e(x) \bmod a(x)$, para cada $i \in A_j$.
- 2 Usando el hecho que $ed \equiv 1 \bmod \phi_q(a)$ y el Teorema de Fermat sobre $\mathbb{F}_q[x]$, verifica que $T_i^e(x) R_i(x) \equiv 1 \bmod a(x)$ para toda $i \in A_j$.

Si esta relación no se cumple para alguna i , la autoridad sabe que el participante P_i no envió el $R_i(x)$ correcto y detiene el proceso.

- 3 Si la información es correcta evalúa:

$$w(x) = b_j(x) \prod_{i \in A_j} T_i(x) \bmod a(x).$$

Un cálculo directo muestra que $w(x) = s(x)$, el secreto.



Observaciones

- 1 Se tiene suficientes polinomios irreducibles sobre un campo finito para evitar un ataque a “fuerza bruta”, gracias al siguiente resultado:

Theorem

El número de polinomios irreducibles mónicos de grado n sobre un campo finito \mathbb{F}_q es:

$$\frac{1}{n} \sum_{d|n} \mu\left(\frac{n}{d}\right) q^d,$$

donde μ es la función de Möbius, y la suma se toma sobre todos los divisores de n .

- 2 Ideas tipo RSA sobre $\mathbb{F}_q[x]$ dan a este sistema cierta robustez.



Ejemplo

Sea $q = 41$, $P_1(x) = 1 + 4x + x^2$, $P_2(x) = 27 + x^2$. Entonces:

- 1 $a(x) = 27 + 26x + 28x^2 + 4x^3 + x^4$,
- 2 $\phi_q(a) = 2,822,400$.

Sea $f(x) = 2 + 3x^2 + x^4$, primo relativo con $a(x)$.

El agente \mathcal{D} publica la terna:

$$(f(x), a(x), q) = (2 + 3x^2 + x^4, 27 + 26x + 28x^2 + 4x^3 + x^4, 41).$$

Ejemplo (cont.)

Sea $\Gamma_5 = \{P_2, P_5, P_7, P_8\}$ un conjunto de participantes autorizado para recuperar el secreto.

Cada uno de los miembros de este conjunto elige aleatoriamente $s_i \in [2, (41)^4]$, digamos:

$s_2 = 294350$, $s_5 = 434597$, $s_7 = 1398147$, $s_8 = 1656901$,
determina:

$$R_2(x) = 120 + 10x + 8x^2 + 11x^3, \quad R_5(x) = 7 + 4x + 16x^2 + 29x^3,$$

$$R_7(x) = 21 + 9x + 24x^2 + 14x^3, \quad R_8(x) = 25 + 26x + 25x^2 + 4x^3,$$

y lo envía al agente \mathcal{D} , guardando su s_i .

Ejemplo (cont.)

El agente \mathcal{D} elige aleatoriamente $e = 80767 \in [2, \phi_q(\alpha)]$ tal que $(e, \phi_q(\alpha)) = 1$ y determina $d = 1483903 \in [2, \phi_q(\alpha)]$, el inverso de e módulo $\phi_q(\alpha)$.

Determina:

$$R_0(x) = f(x)^e = 39 + 34x + 39x^2 + 21x^3,$$

$$Q_2(x) = 21 + 31x + 33x^2 + 30x^3, \quad Q_5(x) = 24 + 2x + 8x^2 + 35x^3,$$

$$Q_7(x) = 10 + 16x + 29x^2 + 34x^3 \text{ y } Q_8(x) = 2 + 7x + 2x^2 + 20x^3,$$

(todo módulo $\alpha(x)$).

Publica la pareja $(R_0(x), e)$.

Dividiendo el secreto

Supóngase que el secreto es:

$$s(x) = (2 + 5x^2 + 17x^3)P_1(x) \bmod \alpha(x) = 31 + 34x + 40x^2 + 33x^3 \in \mathbb{F}_{41}[x].$$

El agente toma $A_5 = \{2, 5, 7, 8\}$ y evalúa:

$$b_5(x) = s(x) \prod_{i \in A_5} Q_i(x) = 24 + 25x + 27x^2 + 11x^3,$$

(todas las operaciones módulo $a(x)$) y envía

$(5, 24 + 25x + 27x^2 + 11x^3)$ al grupo Γ_5 .

Recuperando el secreto

Si los miembros del conjunto autorizado Γ_5 desean recuperar el secreto, cada uno de ellos usa su valor secreto s_i , determina:

$$T_2(x) = 7 + 26x + 6x^2 + 4x^3, \quad T_5(x) = 4 + 14x + 4x^2 + 40x^3,$$

$$T_7(x) = 39 + 5x + 27x^2 + 26x^3, \quad T_8(x) = 7 + 4x + 16x^2 + 29x^3.$$

y envían la pareja $(T_i(x), R_i(x))$, $i \in \{2, 5, 7, 8\}$, a la entidad recuperadora.

Recuperando el secreto (cont.)

La entidad recuperadora determina $T_i^e(x) \bmod \alpha(x)$:

$$T_2^e(x) = 34 + 15x + 35x^2 + 37x^3, \quad T_5^e(x) = 2 + 7x + 2x^2 + 20x^3,$$

$$T_7^e(x) = 1 + 18x + 7x^2 + 28x^3, \quad T_8^e(x) = 35 + 20x + 39x^2 + 22x^3.$$

y verifica que para cada $i \in \{2, 5, 7, 8\}$, la igualdad $T_i^e(x) R_i(x) \equiv 1 \bmod a(x)$ es válida.

Si para alguna i esta relación no es cierta, el miembro P_i del conjunto no usó su valor s_i correcto, la entidad recuperadora no tiene la información correcta y detiene el proceso.

Recuperando el secreto (cont.)

Si la autoridad \mathcal{A} (entidad recuperadora) tiene la información correcta, toma $(5, b_5(x)) = (5, 24 + 25x + 27x^2 + 11x^3)$ correspondiente al conjunto Γ_5 y recupera el secreto evaluando:

$$w(x) = b_5(x) \prod_{i \in A_5} T_i(x) = 31 + 34x + 40x^2 + 33x^3,$$

(todos los cálculos módulo $a(x)$).

Esquema basado en polinomios de Dickson

A continuación recordaremos algunos conceptos y resultados sobre polinomios de Dickson que servirán para describir el esquema.

Los polinomios de Dickson tienen aplicación en diversas áreas como permutaciones, esquemas de cifrado, sumas exponenciales, curvas hiperélicas, entre otras.

Un polinomio de Dickson (de primera clase) de grado k sobre un anillo conmutativo R es:

$$D_k(x) = \sum_{i=0}^{\lfloor k/2 \rfloor} \frac{k}{k-i} \binom{k-i}{i} x^{k-2i}$$

Se define $D_0(x) = 2$.

Esos polinomios son cerrados bajo composición:

$$D_k(D_\ell(x)) = D_{k\ell}(x).$$



Propiedades de los polinomios de Dickson.

- 1 Sea $n = p_1 p_2$, primos distintos. Para cada entero k se tiene que $g(x; k) = D_k(x) \bmod n$ es una permutación sí y sólo si $\gcd(k, v(n)) = 1$ se satisface, donde $v(n) = \text{lcm}[p_1^2 - 1, p_2^2 - 1]$.
- 2 Si h es un entero positivo tal que $kh \equiv 1 \pmod{v(n)}$ entonces $D_{kh}(x)$ es igual a x , el polinomio identidad módulo n .
- 3 Para determinar el valor de $D_k(b)$ para $b \in R$, se resuelve la ecuación cuadrática $u^2 - bu + 1 = 0$ en el anillo $R[u] / (u^2 - bu + 1)$. Si u_0 es una solución $u_0^k = a_1 u_0 + a_0$ donde $a_1, a_0 \in R$ se tiene que $D_k(b) = a_1 b + 2a_0$.

El esquema

Sea $\{P_0, P_1, \dots, P_m\}$ el conjunto de participantes donde P_0 es el Agente \mathcal{A} (o Autoridad Certificadora (AC)).

Fase inicial

La Autoridad \mathcal{A} hace lo siguiente:

- 1 Elige aleatoriamente dos primos grandes distintos p_1, p_2 y toma $n = p_1 p_2$.
- 2 Elige un primo $q > n$.
- 3 Elige aleatoriamente un entero $\beta \in [2, n]$.
- 4 Hace público $(D_k(x), n, \beta, q)$ y guarda (p_1, p_2) .

Cada participante P_i , $1 \leq i \leq m$, hace lo siguiente:

- 1 Elige aleatoriamente un entero $s_i \in [2, m]$, y lo guarda.
- 2 Determina $R_i = D_{s_i}(\beta) \bmod n$.
- 3 Envía R_i a la Autoridad.

Dividiendo el secreto

Se toma el secreto s como un elemento de \mathbb{F}_q . El agente hace lo siguiente:

- 1 Determina $v(n) = \text{lcm}[p_1^2 - 1, p_2^2 - 1]$.
- 2 Aleatoriamente elige un entero $e < v(n)$ primo relativo con $v(n)$ y determina su inverso h módulo $v(n)$.
- 3 Determina $Q_i = D_e(R_i) \bmod n$, $1 \leq i \leq n$, y $R_0 = D_e(\beta) \bmod n$.
- 4 Aleatoriamente elige $b \in \mathbb{F}_q$ y considera el polinomio $f(x) = (s + bx) \in \mathbb{F}_q[x]$.

Dividiendo el secreto (cont.)

- 1 Hace público $f(1) = (s + b) \bmod q$.
- 2 Etiqueta los conjuntos de participantes que son permitido a recuperar el secreto como: $\Gamma_1, \Gamma_2, \dots, \Gamma_t$ where $\Gamma_1 = \{P_0\} = \{\mathcal{A}\}$.
- 3 Para cada conjunto autorizado $\Gamma_j = \{P_i : i \in A_j\}$ donde A_j es un subconjunto de $\{1, 2, \dots, m\}$, $j \geq 2$, determina $H_j = [f(j)]_2 \oplus [\oplus_{i \in A_j} [Q_i]_2]$, donde $[z]_2$ es la representación binaria del entero z y “ \oplus ” es la operación XOR.
- 4 Hace público $(R_0, h, (j, H_j))$, $2 \leq j \leq t$.

Recuperando el secreto

Supóngase que los participantes del conjunto Γ_j desean recuperar el secreto. Cada participante $P_i \in \Gamma_j$ obtiene (n, R_0) , que es público, y:

- 1 Determina $D_{s_i}(R_0) \bmod n$.
- 2 Envía R_i y $D_{s_i}(R_0) \bmod n$ a la entidad recuperadora del secreto.

La entidad recuperadora hace lo siguiente:

- 1 Determina $f(1)$ y verifica que $R_i = D_h(D_{s_i}(R_0)) \bmod n$ se satisface, de otra manera informa al participante P_i que su información no es correcta. Si toda la inf. es correcta, continua el proceso, de otra manera lo para.
- 2 Evalúa $[f(j)]_2 = [H_j]_2 \oplus [i \in A_l][Q_i]_2 = [H_j]_2 \oplus [\oplus_{i \in A_l}[D_{s_i}(R_0)]]_2$
y $b = [f(j) - f(1)](j - 1)^{-1} \bmod q$.
- 3 Recupera el secreto: $s = [f(1) - b] \bmod q$.

recuperando el secreto (cont.)

El secreto se recupera porque:

$$\begin{aligned}[f(j)]_2 &= [H_j]_2 \oplus \left[\bigoplus_{i \in A_j} [Q_i]_2 \right] = [H_j]_2 \oplus \left[\bigoplus_{i \in A_j} [D_e(R_i)]_2 \right] \\ &= [H_j]_2 \oplus \left[\bigoplus_{i \in A_j} [D_e(D_{s_i}(\beta))]_2 \right] \\ &= [H_j]_2 \oplus \left[\bigoplus_{i \in A_j} [D_{s_i}(D_e(\beta))]_2 \right] \\ &= [H_j]_2 \oplus \left[\bigoplus_{i \in A_j} [D_{s_i}(R_0)]_2 \right].\end{aligned}$$

Dividiendo el secreto

El la autoridad hace lo siguiente:

- 1 Elige $p_1 = 257$, $p_2 = 593$ y evalua $n = p_1 p_2 = 152401$.
- 2 Elige aleatoriamente $\beta = 76092 \in [1, n]$ y $q = 153359$.
- 3 Publica la terna $(n, \beta, q) = (152401, 76092, 153359)$.

Supóngase que hay 8 participantes, incluyendo el agente y que los subconjuntos autorizados para recuperar el secreto son:

$$\{P_0\}, \{P_1, P_2, P_3\}, \{P_2, P_5, P_7\}, \{P_3, P_6\}, \{P_4, P_7\}.$$

Ejemplo (cont.)

Se vera como el conjunto $\Gamma_3 = \{P_2, P_5, P_7\}$ recupera el secreto $s = 146781 \in \mathbb{F}_q$.

Los participantes P_j , $1 \leq j \leq 7$, hacen lo siguiente:

- 1 Aleatoriamente eligen enteros y guardan su número:

$$(s_1, \dots, s_7) = (101476, 711141, 494, 103315, 44479, 50707, 17636), s_i \in [2, n];$$

- 2 Determinan

$(R_1, \dots, R_7) = (66854, 41771, 26203, 23882, 39815, 31592, 53233)$, $R_i = D_{s_j}(\beta)$, y cada participante lo envia a la autoridad. Por ejemplo, para obtener 26203, el participante P_3 evalúa el polinomio $D_{494}(x) \bmod n$ en $x = 76092$. Los otros valores se obtienen en forma similar.

Ejemplo (cont.)

Para dividir el secreto el agente hace lo siguiente:

- 1 Determina $v(152401) = 241933824$.
- 2 Elige aleatoriamente $e = 105232901$ tal que $\gcd(e, v(n)) = 1$ y determina $h = e^{-1} = 5278925$ módulo $v(n)$.
- 3 Determina $R_0 = D_e(\beta) = 48858$, $Q_2 = D_e(41771) = 65139$, $Q_5 = D_e(39815) = 33639$, $Q_7 = D_e(53233) = 24807$ (todo módulo n) y los restantes valores Q_j .
- 4 Elige aleatoriamente $b = 83957 \in \mathbb{F}_q$ y encuentra $f(1) = (s + b) \bmod q = 77379$.
- 5 Para el subconjunto, $\{P_2, P_5, P_7\}$ determina:

$$H_3 = [f(3)]_2 \oplus [Q_2]_2 \oplus [Q_5]_2 \oplus [Q_7]_2 = [67339]_2.$$

- 6 Hace público $(R_0, h, f(1), (j, H_j)) = \{(48858, 5278925), 77379, (j, H_j), 2 \leq j \leq 5\}$.

Recuperando el secreto

Si el conjunto $\{P_2, P_5, P_7\}$ decide recuperar el secreto, cada miembro del conjunto hace lo siguiente:

- 1 Encuentra los valores públicos $n = 152401$ and $R_0 = 48858$.
- 2 Determina $D_{141494}(R_0) \bmod n = 107047$, $D_{44479}(R_0) \bmod n = 65139$, $D_{17636}(R_0) \bmod n = 80961$.
- 3 Envía $(26203, 107047)$, $(39815, 65139)$, $(53233, 80961)$ a la autoridad (entidad recuperadora).

Recuperando el secreto (cont.)

La autoridad (entidad recuperadora) recibe los valores $f(1) = 77379$, $q = 153359$, $h = 5278925$, $(3, [67339]_2)$ y recupera el secreto de la sig. manera:

- 1 Verifica que $26203 = D_h(107047)$, $39815 = D_h(65139)$, $53233 = D_h(80961)$. Si alguno de esas relaciones no es válida comunica a los participantes P_2, P_5, P_7 y detiene el proceso, de otra manera continua.
- 2 Determina $[f(3)]_2 = H_3 \oplus [107047]_2 \oplus [65139]_2 \oplus [80961]_2 = [91934]_2$.
- 3 Determina $(3 - 1)^{-1} \bmod q = 76680$ y $b = [91934 - 77379(76680) \bmod q = 83957$.
- 4 Recupera el secreto: $s = [77379 - 83957] \bmod q = 146781$.

Referencias

- 1 A. Beimel, Secret-sharing schemes: A survey. In Yeow Meng Chee, Zhenbo Guo, San Ling, Fengjing Shao, Yuansheng Tang, Huaxiong Wang, and Chaoping Xing, editors, *Coding and Cryptology - Third International Workshop, IWCC 2011, Qingdao, China, May 30-June 3, 2011. Proceedings*, volume 6639 of *Lecture Notes in Computer Science*, pages 11- 46. Springer, 2011.
- 2 G.R. Blakley. Safeguarding cryptographic keys. *Procc. of the National Computer Conference*, pp.313 - 317 (1979).
- 3 C. Blundo, A. De Santis, L. Gargano and U. Vaccaro. On the information rate of secret sharing schemes. *Theoretical Computer Sciences* 154, (1996), 283 - 306.
- 4 X. Dong, L. Hou and Y. Zhang. A Secret Sharing Scheme over the Ring of Eisenstein Integers. *J. of Computational Information Systems* 9: (2013), 4869 - 4876.

Referencias

- 1 T. Galibus and G. Matveev. Generalized Mignotte's sequences over polynomial rings. *Electr. Notes Theor. Comput. Sci.*,186 (2007) 43-48.
- 2 C. Carlet, C. Ding and J. Yuan, Linear Codes From Perfect Nonlinear Mappings and Their Secret Sharing Schemes, *IEEE Trans. on Inf. Theory*, vol. 51, no. 6, June 2005, pp. 2089-2102.
- 3 C. Ding and J. Yuan, Covering and secret sharing with linear codes. *Discrete Mathematics and Theoretical Computer Science*, Lecture Notes in Computer Science (2003), 2731: 11 - 25.
- 4 S. Iftene. General Secret sharing Based on the Chinese Remainder Theorem with Applications in E-voting. *Electr. Notes in Theor. Comput. Sci.*,186 (2007) 67-84.
- 5 M. Ito, A. Saito, and T. Nishizeki. Secret sharing scheme realizing general access structure. In *Proceedings of GLOBECOM87*, pages 99- 102, 1987.

GRACIAS !!



