Sisteme de partajare a secretelor

Prof. Dr. Adrian Atanasiu

Universitatea București

May 5, 2018

- Într-o bancă, seiful trebuie deschis zilnic.
  Banca are trei directori, dar nu încredințează combinația seifului nici unuia din ei.
  Ea dorește să dispună de un sistem de acces prin care orice asociere de doi directori să poată deschide seiful, dar acest lucru să fie imposibil pentru unul singur.

- Într-o bancă, seiful trebuie deschis zilnic.
  Banca are trei directori, dar nu încredințează combinația seifului nici unuia din ei.
  Ea dorește să dispună de un sistem de acces prin care orice asociere de doi directori să poată deschide seiful, dar acest lucru să fie imposibil pentru unul singur.
- În Rusia, accesul la arma nucleară se bazează pe un sistem similar *doi - din - trei*.
  Cele trei persoane sunt Președintele țării, Președintele Parlamentului și Ministrul Apărării.

Fiind dat un secret S, se cere împărțirea lui la n participanți ($n \geq 2$), astfel încât:

1. Cel puțin k din cei n participanți pot regăsi S prin combinarea informațiile lor;

Fiind dat un secret S, se cere împărțirea lui la n participanți ($n \geq 2$), astfel încât:

1. Cel puțin k din cei n participanți pot regăsi S prin combinarea informațiile lor;
2. Nici o asociere de mai puțin de k participanți nu poate recompune S.

Fiind dat un secret S, se cere împărțirea lui la n participanți ($n \geq 2$), astfel încât:

1. Cel puțin k din cei n participanți pot regăsi S prin combinarea informațiile lor;
2. Nici o asociere de mai puțin de k participanți nu poate recompune S.

Acest lucru se poate realiza prin *partajarea* secretului S în n componente ("shares") $S_1, S_2, \ldots, S_n$ și distribuirea câte unei componente fiecărui participant.

În funcție de "cantitatea" de informație secretă pe care o pot obține grupurile neautorizate, sistemele de partajare a secretelor se clasifică în

- Sisteme perfecte de partajare: componentele deținute de grupurile neautorizate nu oferă nici o informație (în sensul teoretic al termenului) despre secretul S;

În funcție de "cantitatea" de informație secretă pe care o pot obține grupurile neautorizate, sistemele de partajare a secretelor se clasifică în

- Sisteme perfecte de partajare: componentele deținute de grupurile neautorizate nu oferă nici o informație (în sensul teoretic al termenului) despre secretul S;
- Sisteme computațional - sigure de partajare: grupurile neautorizate au acces la o anumită cantitate de informație relativă la S, dar problema aflării secretului plecând de la această informație formează o problemă $\mathcal{NP}$ - completă.

Scheme de partajare majoritară

Definiție

Fie $k, n \in \mathbb{Z}$ $(2 \leq k \leq n)$. O schemă de partajare (n, k) - majoritară este o metodă de partajare a unui secret S între membrii unei mulțimi

$$\mathcal{P} = \{P_1, \ldots, P_n\}$$

de participanți, astfel încât orice asociere de k participanți să poată calcula S, lucru imposibil pentru asocieri de $k-1$ sau mai puțini participanți.

Scheme de partajare majoritară

Definiție

Fie $k, n \in \mathbb{Z}$ $(2 \leq k \leq n)$. O schemă de partajare (n, k) - majoritară este o metodă de partajare a unui secret S între membrii unei mulțimi

$$\mathcal{P} = \{P_1, \ldots, P_n\}$$

de participanți, astfel încât orice asociere de k participanți să poată calcula S, lucru imposibil pentru asocieri de $k-1$ sau mai puțini participanți.

Exemplele date la început sunt sisteme $(3, 2)$ - majoritare.

Definiție

Se numește structură de acces $\mathcal{A} \subseteq 2^{\mathcal{P}}$ mulțimea tuturor grupurilor "autorizate" care dispun de informația legală necesară construirii sistemului.
Celelalte grupuri sunt numite "grupuri neautorizate".

Definiție

Se numește structură de acces $\mathcal{A} \subseteq 2^{\mathcal{P}}$ mulțimea tuturor grupurilor "autorizate" care dispun de informația legală necesară construirii sistemului.
Celelalte grupuri sunt numite "grupuri neautorizate".

Fie $B \in \mathcal{A}$ și $B \subseteq C \subseteq \mathcal{P}$. Dacă mulțimea de participanți C caută să determine secretul S, ea va reuși lucrând numai cu participanții din B și ignorând participanții din $C \setminus B$.
Deci structura de acces $\mathcal{A}$ satisface condiția de monotonie:

Dacă $B \in \mathcal{A}$ și $B \subseteq C \subseteq \mathcal{P}$, atunci $C \in \mathcal{A}$.

Definiție

Se numește structură de acces $\mathcal{A} \subseteq 2^{\mathcal{P}}$ mulțimea tuturor grupurilor "autorizate" care dispun de informația legală necesară construirii sistemului.
Celelalte grupuri sunt numite "grupuri neautorizate".

Fie $B \in \mathcal{A}$ și $B \subseteq C \subseteq \mathcal{P}$. Dacă mulțimea de participanți C caută să determine secretul S, ea va reuși lucrând numai cu participanții din B și ignorând participanții din $C \setminus B$.
Deci structura de acces $\mathcal{A}$ satisface condiția de monotonie:

$$\text{Dacă } B \in \mathcal{A} \text{ și } B \subseteq C \subseteq \mathcal{P}, \text{ atunci } C \in \mathcal{A}.$$

Vom presupune că orice structură de acces este monotonă.

Definiție

Fie $2 \leq k \leq n$. *Structura*

$$\mathcal{A} = \{A \in 2^{\mathcal{P}} \mid card(A) \geq k\}$$

se numește structură de acces (n, k) - *majoritară.*

Deci o schemă de partajare (n, k) - majoritară este o structură monotonă de acces (n, k) - majoritară.

Definiție

Fie $2 \leq k \leq n$. Structura

$$\mathcal{A} = \{A \in 2^{\mathcal{P}} \mid card(A) \geq k\}$$

se numește structură de acces (n, k) - majoritară.

Deci o schemă de partajare (n, k) - majoritară este o structură monotonă de acces (n, k) - majoritară.
Un element $B \in \mathcal{A}$ este minimal dacă

$$(\forall A \in \mathcal{P})[A \subset B \implies A \notin \mathcal{A}]$$

$\mathcal{A}_{min}$ – mulțimea elementelor minimale autorizate din $\mathcal{A}$ caracterizează complet $\mathcal{A}$:

$$\mathcal{A} = \{C \subseteq \mathcal{P} \mid \exists B \in \mathcal{A}_{min},\ B \subseteq C\}.$$

Exemplu

Fie $\mathcal{P} = \{P_1, P_2, P_3, P_4\}$ și

$$\mathcal{A}_{min} = \{\{P_1, P_2, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3\}\}$$

Vom avea
$\mathcal{A} = \{\{P_1, P_2, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3\}, \{P_1, P_2, P_3\},$
$\{P_2, P_3, P_4\}, \{P_1, P_2, P_3, P_4\}\}.$

Invers, fiind dat $\mathcal{A}$, se vede imediat că $\mathcal{A}_{min}$ este mulțimea părților sale minimale.

Dacă o mulțime $B \subseteq \mathcal{P}$ este neautorizată, orice submulțime a sa va fi de asemenea neautorizată.
Deci se va lua ca bază de lucru mulțimea grupurilor maximale neautorizate.

Definiție

O mulțime $B \in 2^{\mathcal{P}} \setminus \mathcal{A}$ este maximal neautorizată dacă

$$(\forall C \in 2^{\mathcal{P}})\ [B \subset C \quad \Longrightarrow \quad C \in \mathcal{A}].$$

Dacă o mulțime $B \subseteq \mathcal{P}$ este neautorizată, orice submulțime a sa va fi de asemenea neautorizată.
Deci se va lua ca bază de lucru mulțimea grupurilor maximale neautorizate.

Definiție

O mulțime $B \in 2^{\mathcal{P}} \setminus \mathcal{A}$ este maximal neautorizată dacă

$$(\forall C \in 2^{\mathcal{P}})\ [B \subset C \quad \Longrightarrow \quad C \in \mathcal{A}].$$

Vom nota cu $\mathcal{NA}_{max}$ mulțimea mulțimilor maximal neautorizate.
Atunci o structură de acces neautorizată $\mathcal{NA} = 2^{\mathcal{P}} \setminus \mathcal{A}$ este

$$\mathcal{NA} = \left\{A \in 2^{\mathcal{P}} \mid (\exists B \in \mathcal{NA}_{max})\ (A \subseteq B)\right\}$$

Exemplu

Fie $n = 4$ și structura de acces

$\mathcal{A}= \{\{P_1, P_2\}, \{P_1, P_2, P_3\}, \{P_1, P_2, P_4\}, \{P_1, P_2, P_3, P_4\},$
$\{P_3, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3, P_4\}\}.$

Atunci:

$\mathcal{A}_{min} = \{\{P_1, P_2\}, \{P_3, P_4\}\},$

Exemplu

Fie $n = 4$ și structura de acces

$\mathcal{A}$= $\{\{P_1, P_2\}, \{P_1, P_2, P_3\}, \{P_1, P_2, P_4\}, \{P_1, P_2, P_3, P_4\},$
$\{P_3, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3, P_4\}\}$.

Atunci:

$\mathcal{A}_{min} = \{\{P_1, P_2\}, \{P_3, P_4\}\}$,

$$\mathcal{NA}_{max} = \{\{P_1, P_3\}, \{P_1, P_4\}, \{P_2, P_3\}, \{P_2, P_4\}\},$$

Exemplu

Fie $n = 4$ și structura de acces
$\mathcal{A}$= $\{\{P_1, P_2\}, \{P_1, P_2, P_3\}, \{P_1, P_2, P_4\}, \{P_1, P_2, P_3, P_4\},$
$\{P_3, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3, P_4\}\}$.
Atunci:
$\mathcal{A}_{min} = \{\{P_1, P_2\}, \{P_3, P_4\}\}$,

$$\mathcal{NA}_{max} = \{\{P_1, P_3\}, \{P_1, P_4\}, \{P_2, P_3\}, \{P_2, P_4\}\},$$

$\mathcal{NA} = \{\emptyset, \{P_1\}, \{P_2\}, \{P_3\}, \{P_4\}, \{P_1, P_3\}, \{P_1, P_4\},$
$\{P_2, P_3\}, \{P_2, P_4\}\}$.

Vom nota cu S_i componenta lui S, cunoscută de participantul $P_i \in \mathcal{P}$. Valoarea lui S este aleasă de un *arbitru* $D \notin \mathcal{P}$.

Vom nota cu S_i componenta lui S, cunoscută de participantul $P_i \in \mathcal{P}$.
Valoarea lui S este aleasă de un *arbitru* $D \notin \mathcal{P}$.
D va distribui – printr-un canal securizat – componentele $S_1, \ldots S_n$ (ale secretului) membrilor grupului $\mathcal{P}$, astfel încât nici un participant P_i să nu cunoască componentele celorlalți și nici să fie capabil ca din S_i să poată recompune secretul S.

Vom nota cu S_i componenta lui S, cunoscută de participantul $P_i \in \mathcal{P}$.
Valoarea lui S este aleasă de un *arbitru* $D \notin \mathcal{P}$.
D va distribui – printr-un canal securizat – componentele $S_1, \ldots S_n$ (ale secretului) membrilor grupului $\mathcal{P}$, astfel încât nici un participant P_i să nu cunoască componentele celorlalți și nici să fie capabil ca din S_i să poată recompune secretul S.

Ulterior, participanții unei submulțimi $B \subseteq \mathcal{P}$ pot pune în comun componentele cunoscute de ei (sau să le dea unei autorități în care au încredere) cu scopul de a determina S.
Ei trebuie să poată reuși în această tentativă dacă și numai dacă $card(B) \geq k$.

Vom nota cu S_i componenta lui S, cunoscută de participantul $P_i \in \mathcal{P}$.
Valoarea lui S este aleasă de un *arbitru* $D \notin \mathcal{P}$.
D va distribui – printr-un canal securizat – componentele $S_1, \ldots S_n$ (ale secretului) membrilor grupului $\mathcal{P}$, astfel încât nici un participant P_i să nu cunoască componentele celorlalți și nici să fie capabil ca din S_i să poată recompune secretul S.

Ulterior, participanții unei submulțimi $B \subseteq \mathcal{P}$ pot pune în comun componentele cunoscute de ei (sau să le dea unei autorități în care au încredere) cu scopul de a determina S.
Ei trebuie să poată reuși în această tentativă dacă și numai dacă $card(B) \geq k$.

Să notăm cu $\mathcal{K}$ spațiul tuturor secretelor posibile S și cu $\mathcal{S}$ spațiul componentelor (toate componentele S_i posibile ale unui secret S).

Schema lui Blakely

Fie q prim, k, n $(0 < k \leq n)$ și $\mathcal{K} = \mathbb{Z}_q^k$, $\mathcal{S} = \mathbb{Z}_q^{k+1}$.
Dacă $S = (a_1, a_2, \ldots, a_k)$ este un secret, atunci:

1. D alege $\alpha_{ij}, \beta_i \in \mathbb{Z}_q$ $(1 \leq i \leq n,\ 1 \leq j \leq k)$ astfel ca

$$\begin{pmatrix} \alpha_{11} & \alpha_{12} & \ldots & \alpha_{1k} \\ \alpha_{21} & \alpha_{22} & \ldots & \alpha_{2k} \\ & & \vdots & \\ \alpha_{n1} & \alpha_{n2} & \ldots & \alpha_{nk} \end{pmatrix} \cdot \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix}$$

și $\begin{pmatrix} \alpha_{11} & \alpha_{12} & \ldots & \alpha_{1k} \\ \alpha_{21} & \alpha_{22} & \ldots & \alpha_{2k} \\ & & \vdots & \\ \alpha_{n1} & \alpha_{n2} & \ldots & \alpha_{nk} \end{pmatrix}$ are rangul k (calcule modulo q).

2 Trimite fiecărui participant P_i componenta

$$S_i = (\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{ik}, \beta_i)$$

Schema este o structură de acces (n, k) - majoritară deoarece:

- Fiecare participant P_i va construi – din componenta sa – ecuația diofantică

$$\alpha_{i1}x_1 + \cdots + \alpha_{ik}x_k = \beta_i \pmod{q}$$

care are printre soluțiile sale și secretul $S = (a_1, \ldots, a_k)$.

Schema este o structură de acces (n, k) - majoritară deoarece:

- Fiecare participant P_i va construi – din componenta sa – ecuația diofantică

$$\alpha_{i1}x_1 + \cdots + \alpha_{ik}x_k = \beta_i \pmod q$$

  care are printre soluțiile sale și secretul $S = (a_1, \ldots, a_k)$.
- Orice grup de k parteneri va putea recompune secretul S rezolvând un sistem format din cele k ecuații liniare puse în comun.

Exemplu

Fie $q = 31$ și să o schemă $(3, 2)$ - majoritară, unde componentele participanților P_1, P_2, P_3 sunt:

$$S_1 = (4, 29, 8),\ S_2 = (2, 1, 8),\ S_3 = (3, 27, 1)$$

Niciun participant nu poate afla singur $S = (x, y) \in \mathbb{Z}_{31} \times \mathbb{Z}_{31}$.

Exemplu

Fie $q = 31$ și să o schemă $(3, 2)$ - majoritară, unde componentele participanților P_1, P_2, P_3 sunt:

$$S_1 = (4, 29, 8),\ S_2 = (2, 1, 8),\ S_3 = (3, 27, 1)$$

Niciun participant nu poate afla singur $S = (x, y) \in \mathbb{Z}_{31} \times \mathbb{Z}_{31}$.

Dacă se aliază însă P_1 cu P_3, ei au de rezolvat – în $\mathbb{Z}_{31}$ – sistemul

$$\begin{cases} 4x + 29y = 8 \\ 3x + 27y = 1 \end{cases}$$

cu soluția (unică) $S = (3, 2)$.

Exemplu

Fie $q = 31$ și să o schemă $(3, 2)$ - majoritară, unde componentele participanților P_1, P_2, P_3 sunt:

$$S_1 = (4, 29, 8),\ S_2 = (2, 1, 8),\ S_3 = (3, 27, 1)$$

Niciun participant nu poate afla singur $S = (x, y) \in \mathbb{Z}_{31} \times \mathbb{Z}_{31}$.

Dacă se aliază însă P_1 cu P_3, ei au de rezolvat – în $\mathbb{Z}_{31}$ – sistemul

$$\begin{cases} 4x + 29y = 8 \\ 3x + 27y = 1 \end{cases}$$

cu soluția (unică) $S = (3, 2)$.

Structura de acces este $(3, 2)$ - majoritară:

$$\mathcal{A} = \{\{P_1, P_2\}, \{P_1, P_3\}, \{P_2, P_3\}, \{P_1, P_2, P_3\}\}$$

Securitate

Schema lui Blakely nu constituie un sistem perfect: orice grup neautorizat știe că secretul se află undeva printre soluțiile ecuațiilor pe care le dețin membrii săi.
Deci grupul posedă o anumită cantitate de informație suplimentară, ceea ce reduce dimensiunea variantelor posibile.

Securitate

Schema lui Blakely nu constituie un sistem perfect: orice grup neautorizat știe că secretul se află undeva printre soluțiile ecuațiilor pe care le dețin membrii săi.
Deci grupul posedă o anumită cantitate de informație suplimentară, ceea ce reduce dimensiunea variantelor posibile.

O securitate perfectă se atinge atunci când secretul S este numai una din coordonatele $a \in \mathbb{Z}_q$ ale soluției $(a_1, \ldots, a_k)$.

Schema lui Shamir

Fie q ($q \geq n+1$) prim și $\mathcal{K} = \mathbb{Z}_q$, $\mathcal{S} = \mathbb{Z}_q$.
Fiecare participant P_i cunoaște un punct (x_i, y_i) de pe graficul unui polinom generat aleator, de grad cel mult $k-1$ și cu termen liber S.

1. D alege $x_1, \ldots, x_n \in \mathbb{Z}_q$ distincte, fiecare x_i fiind comunicat lui P_i.

Schema lui Shamir

Fie q ($q \geq n+1$) prim și $\mathcal{K} = \mathbb{Z}_q$, $\mathcal{S} = \mathbb{Z}_q$.
Fiecare participant P_i cunoaște un punct (x_i, y_i) de pe graficul unui polinom generat aleator, de grad cel mult $k-1$ și cu termen liber S.

1. D alege $x_1, \ldots, x_n \in \mathbb{Z}_q$ distincte, fiecare x_i fiind comunicat lui P_i.
2. Dacă D vrea să repartizeze secretul $S \in \mathbb{Z}_q$, va genera aleator $a_1, \ldots, a_{k-1} \in \mathbb{Z}_q$ și va construi

$$a(X) = S + \sum_{j=1}^{k-1} a_j X^j \pmod{q}.$$

Schema lui Shamir

Fie q ($q \geq n+1$) prim și $\mathcal{K} = \mathbb{Z}_q$, $\mathcal{S} = \mathbb{Z}_q$.
Fiecare participant P_i cunoaște un punct (x_i, y_i) de pe graficul unui polinom generat aleator, de grad cel mult $k-1$ și cu termen liber S.

1. D alege $x_1, \ldots, x_n \in \mathbb{Z}_q$ distincte, fiecare x_i fiind comunicat lui P_i.
2. Dacă D vrea să repartizeze secretul $S \in \mathbb{Z}_q$, va genera aleator $a_1, \ldots, a_{k-1} \in \mathbb{Z}_q$ și va construi

$$a(X) = S + \sum_{j=1}^{k-1} a_j X^j \pmod{q}.$$

3. D calculează $y_i = a(x_i)$ și-l comunică lui P_i $(1 \leq i \leq n)$.

Fie o submulțime $\{P_{i_1}, \ldots, P_{i_k}\}$ de participanți care doresc să reconstituie secretul.
Ei știu valorile x_{i_j} și $y_{i_j} = a(x_{i_j})$, $(1 \leq j \leq k)$

$a(X) \in \mathbb{Z}_q[X]$ este polinomul (secret) folosit de D.

Fie o submulțime $\{P_{i_1}, \ldots, P_{i_k}\}$ de participanți care doresc să reconstituie secretul.
Ei știu valorile x_{i_j} și $y_{i_j} = a(x_{i_j})$, $(1 \leq j \leq k)$

$a(X) \in \mathbb{Z}_q[X]$ este polinomul (secret) folosit de D.

Cum gradul lui este cel mult $k-1$, putem scrie

$$a(X) = a_0 + a_1 X + \cdots + a_{k-1} X^{k-1}$$

unde $a_0, \ldots, a_{k-1} \in \mathbb{Z}_q$ sunt necunoscute.
Ele se află rezolvând sistemul liniar de k ecuații
$y_{i_j} = a(x_{i_j})$, $1 \leq j \leq k$.

Fie o submulțime $\{P_{i_1}, \ldots, P_{i_k}\}$ de participanți care doresc să reconstituie secretul.
Ei știu valorile x_{i_j} și $y_{i_j} = a(x_{i_j}),\ (1 \leq j \leq k)$

$a(X) \in \mathbb{Z}_q[X]$ este polinomul (secret) folosit de D.

Cum gradul lui este cel mult $k-1$, putem scrie

$$a(X) = a_0 + a_1 X + \cdots + a_{k-1} X^{k-1}$$

unde $a_0, \ldots, a_{k-1} \in \mathbb{Z}_q$ sunt necunoscute.
Ele se află rezolvând sistemul liniar de k ecuații
$y_{i_j} = a(x_{i_j}),\ \ 1 \leq j \leq k$.

Dacă ecuațiile sunt independente, soluția este unică, iar valoarea lui a_0 este chiar secretul S.

Exemplu

$q = 17,\ k = 3,\ n = 5$ și $x_i = i\ (1 \leq i \leq 5)$.
Dacă mulțimea $B = \{P_1, P_3, P_5\}$ de participanți vrea să afle secretul, fiecare participant aducând informațiile 8, 10 și respectiv 11, ei vor scrie polinomul general

$$a(X) = a_0 + a_1 X + a_2 X^2$$

și vor reduce problema la rezolvarea în $\mathbb{Z}_{17}$ a sistemului

$$\begin{cases} a(1) = a_0 + a_1 + a_2 = 8 \\ a(3) = a_0 + 3a_1 + 9a_2 = 10 \\ a(5) = a_0 + 5a_1 + 8a_2 = 11 \end{cases}$$

Exemplu

$q = 17,\ k = 3,\ n = 5$ și $x_i = i\ (1 \leq i \leq 5)$.
Dacă mulțimea $B = \{P_1, P_3, P_5\}$ de participanți vrea să afle secretul, fiecare participant aducând informațiile 8, 10 și respectiv 11, ei vor scrie polinomul general

$$a(X) = a_0 + a_1 X + a_2 X^2$$

și vor reduce problema la rezolvarea în $\mathbb{Z}_{17}$ a sistemului

$$\begin{cases} a(1) = a_0 + a_1 + a_2 = 8 \\ a(3) = a_0 + 3a_1 + 9a_2 = 10 \\ a(5) = a_0 + 5a_1 + 8a_2 = 11 \end{cases}$$

Acesta admite soluția unică în $\mathbb{Z}_{17}$:

$$a_0 = 13,\ a_1 = 10,\ a_2 = 2$$

Abordare bazată pe polinoame de interpolare Lagrance

Construcția schemei de partajare (n, k) - majoritară:

1. Se alege un număr prim $q > max\{S, n\}$.

Abordare bazată pe polinoame de interpolare Lagrance

Construcția schemei de partajare (n, k) - majoritară:

1. Se alege un număr prim $q > max\{S, n\}$.
2. Se definește polinomul

$$a(X) = a_{k-1}X^{k-1} + \cdots + a_1X + a_0 \in \mathbb{Z}_q[X]$$

cu $a_0 = S$ și a_i $(1 \leq i \leq k-1)$ arbitrari în $\mathbb{Z}_q$.

Abordare bazată pe polinoame de interpolare Lagrance

Construcția schemei de partajare (n, k) - majoritară:

1. Se alege un număr prim $q > max\{S, n\}$.
2. Se definește polinomul

$$a(X) = a_{k-1}X^{k-1} + \cdots + a_1X + a_0 \in \mathbb{Z}_q[X]$$

cu $a_0 = S$ și a_i $(1 \leq i \leq k-1)$ arbitrari în $\mathbb{Z}_q$.

3. Se determină componentele

$$S_i = a(x_i), \quad 1 \leq i \leq n$$

unde $x_1, \ldots, x_n \in \mathbb{Z}_q$ sunt valori publice arbitrare, distincte două câte două.
Aceste componente se trimit celor n participanți.

Abordare bazată pe polinoame de interpolare Lagrance

Construcția schemei de partajare (n, k) - majoritară:

1. Se alege un număr prim $q > max\{S, n\}$.
2. Se definește polinomul

$$a(X) = a_{k-1}X^{k-1} + \cdots + a_1X + a_0 \in \mathbb{Z}_q[X]$$

cu $a_0 = S$ și a_i $(1 \leq i \leq k-1)$ arbitrari în $\mathbb{Z}_q$.

3. Se determină componentele

$$S_i = a(x_i), \quad 1 \leq i \leq n$$

unde $x_1, \ldots, x_n \in \mathbb{Z}_q$ sunt valori publice arbitrare, distincte două câte două.

Aceste componente se trimit celor n participanți.

Secretul S se obține cu formula de interpolare Lagrance.

Fie polinomul

$$P(X) = \sum_{j=1}^{k} S_{i_j} \prod_{\substack{1 \le t \le k \\ t \ne j}} \frac{X - x_{i_t}}{x_{i_j} - x_{i_t}}.$$

El este un polinom de grad cel mult $k-1$, cu

$$S_{i_j} = P(x_{i_j}),\ j = 1, \ldots, k$$

Cum un astfel de polinom este unic, rezultă că el este chiar polinomul $a(X)$.

Fie polinomul

$$P(X) = \sum_{j=1}^{k} S_{i_j} \prod_{\substack{1 \le t \le k \\ t \ne j}} \frac{X - x_{i_t}}{x_{i_j} - x_{i_t}}.$$

El este un polinom de grad cel mult $k-1$, cu

$$S_{i_j} = P(x_{i_j}),\ j = 1, \ldots, k$$

Cum un astfel de polinom este unic, rezultă că el este chiar polinomul $a(X)$.

Un grup B de k participanți poate calcula $a(X)$ pe baza acestei formule.
De fapt este suficient să obțină $S = a(0)$.

Înlocuind în formulă pe X cu 0, avem

$$S = \sum_{j=1}^{k} S_{i_j} \prod_{\substack{1 \leq t \leq k \\ t \neq j}} \frac{x_{i_t}}{x_{i_t} - x_{i_j}}.$$

Înlocuind în formulă pe X cu 0, avem

$$S = \sum_{j=1}^{k} S_{i_j} \prod_{\substack{1 \le t \le k \\ t \neq j}} \frac{x_{i_t}}{x_{i_t} - x_{i_j}}.$$

Dacă definim

$$b_j = \prod_{\substack{1 \le t \le k \\ t \neq j}} \frac{x_{i_t}}{x_{i_t} - x_{i_j}}, \qquad 1 \le j \le k$$

aceste valori pot fi precalculate și făcute publice de către arbitru.

Înlocuind în formulă pe X cu 0, avem

$$S = \sum_{j=1}^{k} S_{i_j} \prod_{\substack{1 \le t \le k \\ t \ne j}} \frac{x_{i_t}}{x_{i_t} - x_{i_j}}.$$

Dacă definim

$$b_j = \prod_{\substack{1 \le t \le k \\ t \ne j}} \frac{x_{i_t}}{x_{i_t} - x_{i_j}}, \qquad 1 \le j \le k$$

aceste valori pot fi precalculate și făcute publice de către arbitru. Secretul este o combinație liniară de k componente:

$$S = \sum_{j=1}^{k} b_j S_{i_j}.$$

Exemplu

Fie $q = 31,\ x_1 = 20,\ x_2 = 6,\ x_3 = 11$ componentele publice. Componentele secrete sunt

$$S_1 = 12, \quad S_2 = 25, \quad S_3 = 27.$$

Exemplu

Fie $q = 31,\ x_1 = 20,\ x_2 = 6,\ x_3 = 11$ componentele publice. Componentele secrete sunt

$$S_1 = 12, \quad S_2 = 25, \quad S_3 = 27.$$

Dacă mulțimea autorizată $\{P_2, P_3\}$ dorește să afle secretul S, va calcula

$$S = S_2 \cdot \frac{x_3}{x_3 - x_2} + S_3 \cdot \frac{x_2}{x_2 - x_3} = 4 \ (mod\ 31)$$

Exemplu

Fie $q = 31,\ x_1 = 20,\ x_2 = 6,\ x_3 = 11$ componentele publice. Componentele secrete sunt

$$S_1 = 12, \quad S_2 = 25, \quad S_3 = 27.$$

Dacă mulțimea autorizată $\{P_2, P_3\}$ dorește să afle secretul S, va calcula

$$S = S_2 \cdot \frac{x_3}{x_3 - x_2} + S_3 \cdot \frac{x_2}{x_2 - x_3} = 4 \ (mod\ 31)$$

La același rezultat se ajunge dacă se rezolvă sistemul

$$\begin{cases} 6a_1 + a_0 = 25 \\ 11a_1 + a_0 = 27 \end{cases}$$

cu soluția $a_1 = 19,\ a_0 = 4$.

Teoremă

Schema de partajare Shamir este perfectă.

Proof.

Dacă avem numai $k-1$ componente, sistemul de $k-1$ ecuații

$$\begin{cases} a_{k-1}x_{i_1}^{k-1} + \cdots + a_1 x_{i_1} &= S_{i_1} - a_0 \\ a_{k-1}x_{i_2}^{k-1} + \cdots + a_1 x_{i_2} &= S_{i_2} - a_0 \\ & \vdots \\ a_{k-1}x_{i_{k-1}}^{k-1} + \cdots + a_1 x_{i_{k-1}} &= S_{i_{k-1}} - a_0 \end{cases}$$

cu necunoscutele $\{a_1, \ldots, a_{k-1}\}$ are soluție unică pentru fiecare a_0. □

1. Mărimea fiecărei componente S_i nu depășește mărimea secretului S (schema este "ideală").

1. Mărimea fiecărei componente S_i nu depășește mărimea secretului S (schema este "ideală").
2. Pentru o valoare fixată a lui k se pot adăuga sau elimina dinamic componente S_i (de exemplu, prin venirea sau ieșirea din sistem a noi participanți) fără a afecta celelalte componente.

1. Mărimea fiecărei componente S_i nu depășește mărimea secretului S (schema este "ideală").
2. Pentru o valoare fixată a lui k se pot adăuga sau elimina dinamic componente S_i (de exemplu, prin venirea sau ieșirea din sistem a noi participanți) fără a afecta celelalte componente.
3. Componentele S_i pot fi modificate fără a schimba secretul S: se alege un nou polinom $a(X)$ având S ca termen liber.

   Acest lucru se recomandă a se efectua periodic, pentru a păstra nivelul de securitate al sistemului.

Pentru $n = k$ există o variantă simplificată a algoritmului Shamir, definită pentru $\mathcal{K} = \mathbb{Z}_m$, $\mathcal{S} = \mathbb{Z}_m$ (m nu este obligatoriu număr prim; este posibil ca $m \leq n$).

1. D alege aleator elementele $y_1, \ldots, y_{n-1} \in \mathbb{Z}_m$;

Pentru $n = k$ există o variantă simplificată a algoritmului Shamir, definită pentru $\mathcal{K} = \mathbb{Z}_m$, $\mathcal{S} = \mathbb{Z}_m$ (m nu este obligatoriu număr prim; este posibil ca $m \leq n$).

1. D alege aleator elementele $y_1, \ldots, y_{n-1} \in \mathbb{Z}_m$;
2. D calculează

$$y_n = S - \sum_{i=1}^{n-1} y_i \pmod m$$

Pentru $n = k$ există o variantă simplificată a algoritmului Shamir, definită pentru $\mathcal{K} = \mathbb{Z}_m$, $\mathcal{S} = \mathbb{Z}_m$ (m nu este obligatoriu număr prim; este posibil ca $m \leq n$).

1. D alege aleator elementele $y_1, \ldots, y_{n-1} \in \mathbb{Z}_m$;
2. D calculează
$$y_n = S - \sum_{i=1}^{n-1} y_i \pmod m$$
3. Fiecare element y_i este transmis (prin canal securizat) lui P_i, $(i = 1, \ldots, n)$.

Cei n participanți pot determina

$$S = \sum_{i=1}^{n} y_i \pmod{m}.$$

Evident, $n-1$ participanți nu-l pot obține pe S.

Cei n participanți pot determina

$$S = \sum_{i=1}^{n} y_i \pmod{m}.$$

Evident, $n-1$ participanți nu-l pot obține pe S.

Chiar dacă pun în comun componentele lor, ei pot afla doar $S - y$, unde y este componenta celui care lipsește.
Cum y este o valoare aleatoare din $\mathbb{Z}_m$, nu se va obține nici o informație suplimentară referitoare la secret.

Avem deci o structură de acces (n, n) - majoritară.

Schema Mignotte

Se bazează pe secvențe de numere întregi numite *șiruri Mignotte*.

Definiție

Fie n $(n \geq 2)$ un număr întreg și $2 \leq k \leq n$. Un șir (n, k) - Mignotte este o secvență de numere întregi pozitive $p_1 < p_2 < \cdots < p_n$ prime două câte două, cu proprietatea

$$\prod_{i=0}^{k-2} p_{n-i} < \prod_{i=1}^{k} p_i.$$

Exemplu

Șirul

$$5, 7, 9, 11, 13$$

formează o secvență $(5, 3)$ - Mignotte.

Cele cinci numere sunt prime (deci și prime între ele), iar inegalitatea din definiție este

$$p_4 \cdot p_5 < p_1 \cdot p_2 \cdot p_3$$

verificată evident.

1. D alege un șir (n, k) - Mignotte și calculează

$$\alpha = \prod_{i=1}^{k} p_i, \qquad \beta = \prod_{i=0}^{k-2} p_{n-i}.$$

1. D alege un șir (n, k) - Mignotte și calculează

$$\alpha = \prod_{i=1}^{k} p_i, \qquad \beta = \prod_{i=0}^{k-2} p_{n-i}.$$

2. D alege $S \in (\beta, \alpha)$ (în general, S este generat aleator).

1. D alege un șir (n, k) - Mignotte și calculează

$$\alpha = \prod_{i=1}^{k} p_i, \qquad \beta = \prod_{i=0}^{k-2} p_{n-i}.$$

2. D alege $S \in (\beta, \alpha)$ (în general, S este generat aleator).
3. D calculează $S_i = S \ (mod \ p_i)$ și trimite fiecărui utilizator P_i perechea (p_i, S_i), $\quad i = 1, \ldots, n$.

1. D alege un șir (n, k) - Mignotte și calculează

$$\alpha = \prod_{i=1}^{k} p_i, \qquad \beta = \prod_{i=0}^{k-2} p_{n-i}.$$

2. D alege $S \in (\beta, \alpha)$ (în general, S este generat aleator).
3. D calculează $S_i = S \ (mod \ p_i)$ și trimite fiecărui utilizator P_i perechea (p_i, S_i), $\quad i = 1, \ldots, n$.

Fiind cunoscute k componente distincte $S_{i_1}, \ldots, S_{i_k}$, secretul S poate fi aflat pe baza Teoremei Chineze a Restului, fiind soluția (unică) modulo $p_{i_1} p_{i_2} \ldots p_{i_k}$ a sistemului

$$\begin{cases} x \equiv S_{i_1} & (mod \ p_{i_1}) \\ \quad \vdots & \\ x \equiv S_{i_k} & (mod \ p_{i_k}) \end{cases}$$

Exemplu

Folosind șirul Mignotte anterior se determină
$\alpha = 5 \cdot 7 \cdot 9 = 315, \quad \beta = 11 \cdot 13 = 143.$
Să considerăm secretul $285 \in (143, 315)$. Componentele sunt

$$S_1 = S \ (mod\ 5) = 0, \qquad S_2 = S \ (mod\ 7) = 5,$$
$$S_3 = S \ (mod\ 9) = 6, \qquad S_4 = S \ (mod\ 11) = 10,$$
$$S_5 = S \ (mod\ 13) = 12$$

Exemplu

Folosind șirul Mignotte anterior se determină
$\alpha = 5 \cdot 7 \cdot 9 = 315, \quad \beta = 11 \cdot 13 = 143.$
Să considerăm secretul $285 \in (143, 315)$. Componentele sunt

$$S_1 = S \ (mod\ 5) = 0, \qquad S_2 = S \ (mod\ 7) = 5,$$
$$S_3 = S \ (mod\ 9) = 6, \qquad S_4 = S \ (mod\ 11) = 10,$$
$$S_5 = S \ (mod\ 13) = 12$$

Pentru grupul autorizat $\{P_1, P_3, P_4\}$ trebuie rezolvat sistemul

$$\begin{cases} x \equiv 0 \ (mod\ 5) \\ x \equiv 6 \ (mod\ 9) \\ x \equiv 10 \ (mod\ 11) \end{cases}$$

a cărui soluție este 285.

De remarcat că – deoarece $\frac{\beta - \alpha}{\beta} = 1.2$ – în intervalul (β, α) există puține numere care pot fi luate drept secret S accesibil oricărui grup autorizat.

De remarcat că – deoarece $\frac{\beta - \alpha}{\beta} = 1.2$ – în intervalul (β, α) există puține numere care pot fi luate drept secret S accesibil oricărui grup autorizat.
De exemplu, pentru $S = 300$, participanții P_3 și P_4 posedă aceeași componentă:

$$S_3 = S_4 = 3$$

Deci nu există nici un grup autorizat de forma $\{P_3, P_4, x\}$ care să aibă acces la secretul S.

Scheme majoritar ponderate

Într-o schemă de partajare majoritar ponderată, fiecărui utilizator i se asociază un număr pozitiv ("pondere").

Secretul poate fi aflat dacă și numai dacă suma ponderilor participanților este cel puțin egală cu o valoare limită fixată.

Scheme majoritar ponderate

Într-o schemă de partajare majoritar ponderată, fiecărui utilizator i se asociază un număr pozitiv ("pondere").

Secretul poate fi aflat dacă și numai dacă suma ponderilor participanților este cel puțin egală cu o valoare limită fixată.

Shamir le definește, luând ca exemplu o companie în care secretul poate fi acoperit de 3 directori, de doi directori și un vice-președinte, sau de către președinte.
Ideea este de a acorda mai multe componente utilizatorilor mai importanți (președintele primește 3 componente, fiecare vice-președinte are câte două componente, iar un director deține numai o componentă a secretului).

Definiție

Fie $n \geq 2$, $x = (x_1, \ldots, x_n)$ *un vector de numere întregi pozitive și numărul întreg* $w \in \left(2, \sum_{i=1}^{n} x_i\right)$. *Structura de acces*

$$\mathcal{A} = \left\{ A \in 2^{\mathcal{P}} \mid \sum_{P_i \in A} x_i \geq w \right\}$$

se numește structură (x, w, n) - *majoritar ponderată.*

Într-o astfel de schemă, un grup $\{P_{i_1}, \ldots, P_{i_t}\}$ este autorizat dacă și numai dacă $\{i_1, \ldots, i_t\}$ este mulțime de acces într-o structură (x, w, n) - majoritar ponderată: $\sum_{j=1}^{t} x_{i_j} \geq w$.

Parametrii $x_1, \ldots, x_n$ se numesc ponderi iar w este limita schemei de partajare.

Într-o astfel de schemă, un grup $\{P_{i_1}, \ldots, P_{i_t}\}$ este autorizat dacă și numai dacă $\{i_1, \ldots, i_t\}$ este mulțime de acces într-o structură (x, w, n) - majoritar ponderată: $\sum_{j=1}^{t} x_{i_j} \geq w$.

Parametrii $x_1, \ldots, x_n$ se numesc ponderi iar w este limita schemei de partajare.

Dacă $\mathcal{A}$ este o structură de acces (x, w, n) - majoritar ponderată, orice sistem de partajare construit pe baza ei se numește schemă de partajare a secretelor (x, w, n) - majoritar ponderată.

Există structuri de acces care nu sunt majoritar ponderate

Fie $n = 4$ și $\mathcal{A}_{min} = \{\{P_1, P_2\}, \{P_3, P_4\}\}$
Presupunem că lucrăm cu o structură de acces majoritar ponderată, cu ponderile x_1, x_2, x_3, x_4 și limita w. Deci

$$x_1 + x_2 \geq w, \quad x_3 + x_4 \geq w.$$

Prin adunare se obținem $x_1 + x_2 + x_3 + x_4 \geq 2w$, deci

$$2 \cdot max(x_1, x_2) + 2 \cdot max(x_3, x_4) \geq 2w$$

sau $max(x_1, x_2) + max(x_3, x_4) \geq w$.
Deci una din mulțimile

$$\{P_1, P_3\}, \{P_1, P_4\}, \{P_2, P_3\}, \{P_2, P_4\}$$

este mulțime autorizată de acces, dar nu este generată de $\mathcal{A}_{min}$.

Cea mai simplă metodă de construcție a unei scheme (x, w, n) - majoritar ponderate constă în utilizarea unei scheme de partajare (N, w) - majoritare, în care $N = \sum_{i=1}^{n} x_i$.

Cea mai simplă metodă de construcție a unei scheme (x, w, n) - majoritar ponderate constă în utilizarea unei scheme de partajare (N, w) - majoritare, în care $N = \sum_{i=1}^{n} x_i$.

Detaliere:

Fie $s_1, \ldots, s_N$ componentele corespunzătoare unui secret S în raport cu o schemă arbitrară de partajare a secretelor (N, w) - majoritară. Considerăm o partiție oarecare $\{X_1, \ldots, X_n\}$ a mulțimii $\{1, 2, \ldots, N\}$, cu $card(X_i) = x_i$, $(1 \leq i \leq n)$.

Cea mai simplă metodă de construcție a unei scheme (x, w, n) - majoritar ponderate constă în utilizarea unei scheme de partajare (N, w) - majoritare, în care $N = \sum_{i=1}^{n} x_i$.

Detaliere:

Fie $s_1, \ldots, s_N$ componentele corespunzătoare unui secret S în raport cu o schemă arbitrară de partajare a secretelor (N, w) - majoritară. Considerăm o partiție oarecare $\{X_1, \ldots, X_n\}$ a mulțimii $\{1, 2, \ldots, N\}$, cu $card(X_i) = x_i$, $(1 \leq i \leq n)$.

Definim componentele structurii de acces majoritar ponderate prin

$$S_i = \{s_j \mid j \in X_i\}, \quad i = 1 \ldots n$$

Scheme de partajare unanime

În cazul $\mathcal{A} = \mathcal{A}_{min} = \{P_1, P_2, \ldots, P_n\}$, o $\mathcal{A}$ - schemă de partajare a secretelor se numește unanimă de ordin n.
Pentru astfel de scheme, secretul este aflat numai prin participarea tuturor utilizatorilor implicați.

Scheme de partajare unanime

În cazul $\mathcal{A} = \mathcal{A}_{min} = \{P_1, P_2, \ldots, P_n\}$, o $\mathcal{A}$ - schemă de partajare a secretelor se numește unanimă de ordin n.
Pentru astfel de scheme, secretul este aflat numai prin participarea tuturor utilizatorilor implicați.

Evident, o schemă de partajare unanimă de ordin n este echivalentă cu o schemă de partajare (n, n) - majoritară și – reciproc – orice schemă de partajare a secretelor (n, n) - majoritară poate fi utilizată în realizarea unei scheme de partajare unanime.

Schema Karnin - Greene - Hellman

1. Secretul S este un număr aleator din $\mathbb{Z}_q$ ($q > 2$ număr arbitrar fixat).
2. D generează aleator componentele $S_i \in \mathbb{Z}_q$, $(1 \leq i \leq n-1)$. După aceea determină

$$S_n = S - \sum_{i=1}^{n-1} S_i \ (mod\ q)$$

Schema Karnin - Greene - Hellman

1. Secretul S este un număr aleator din $\mathbb{Z}_q$ ($q > 2$ număr arbitrar fixat).
2. D generează aleator componentele $S_i \in \mathbb{Z}_q$, $(1 \leq i \leq n-1)$. După aceea determină

$$S_n = S - \sum_{i=1}^{n-1} S_i \ (mod\ q)$$

3. D trimite fiecărui participant P_i componenta $S_i \quad i = 1, \ldots, n$

Schema Karnin - Greene - Hellman

1. Secretul S este un număr aleator din $\mathbb{Z}_q$ ($q > 2$ număr arbitrar fixat).
2. D generează aleator componentele $S_i \in \mathbb{Z}_q$, $(1 \leq i \leq n-1)$. După aceea determină

$$S_n = S - \sum_{i=1}^{n-1} S_i \ (mod\ q)$$

3. D trimite fiecărui participant P_i componenta $S_i \quad i = 1, \ldots, n$

Secretul S poate fi reconstruit cu relația

$$S = \sum_{i=1}^{n} S_i \ (mod\ q)$$

Exemplu

Să considerăm $n = 20,\ q = 15$, și fie secretul $S = 4 \in \mathbb{Z}_{15}$.

Dacă se definesc componentele $S_i = i,\ (1 \leq i \leq 19)$, ultima componentă va fi

$$S_{20} = 4 - \sum_{i=1}^{19} i = 4 - 190 = -186 = 9\ (mod\ 15)$$

Secretul S poate fi recompus numai prin însumarea celor 20 componente:

$$S = \sum_{i=1}^{20} S_i = 1 + 2 + \cdots + 19 + 9 = 199 = 4\ (mod\ 15)$$

Scheme bazate pe grafuri

O structură în care orice mulțime minimală de acces are două elemente se numește *structură de acces* 2 - *omogenă* sau structură de acces bazată pe grafuri.

Definiție

Un graf $G = (V, E)$ *este multipartit complet dacă* V *se poate partiționa în submulțimile* $V_1, \ldots, V_s$ *astfel încât* $\{x, y\} \in E$ *dacă și numai dacă* $x \in V_i,\ y \in V_j$ *cu* $i \neq j$.

Scheme bazate pe grafuri

O structură în care orice mulțime minimală de acces are două elemente se numește *structură de acces* 2 - *omogenă* sau structură de acces bazată pe grafuri.

Definiție

Un graf $G = (V, E)$ este multipartit complet dacă V se poate partiționa în submulțimile $V_1, \ldots, V_s$ astfel încât $\{x, y\} \in E$ dacă și numai dacă $x \in V_i$, $y \in V_j$ cu $i \neq j$.
Mulțimile V_i se numesc componente.

Scheme bazate pe grafuri

O structură în care orice mulțime minimală de acces are două elemente se numește *structură de acces* 2 - *omogenă* sau structură de acces bazată pe grafuri.

Definiție

Un graf $G = (V, E)$ este multipartit complet dacă V se poate partiționa în submulțimile $V_1, \ldots, V_s$ astfel încât $\{x, y\} \in E$ dacă și numai dacă $x \in V_i$, $y \in V_j$ cu $i \neq j$.
Mulțimile V_i se numesc componente.
Pentru $card(V_i) = n_i$ $(1 \leq i \leq s)$, graful este notat $K_{n_1,\ldots,n_s}$.
Graful multipartit complet $K_{1,\ldots,1}$ cu s componente este de fapt un graf complet și se notează K_s.

Teoremă

Fie G un graf conex. Există o schemă ideală de partajare a secretelor pentru structura de acces specificată de G dacă și numai dacă G este un graf multipartit complet.

Teoremă

Fie G un graf conex. Există o schemă ideală de partajare a secretelor pentru structura de acces specificată de G dacă și numai dacă G este un graf multipartit complet.

Stinson dă o schemă ideală de partajare a secretelor bazată pe structura de acces specificată de graful $K_{n_1,n_2,\ldots,n_s} = (V, E)$:

1. Fie $q > m = card(V)$ un număr prim și $V_1, \ldots, V_s$ componentele grafului $K_{n_1,n_2,\ldots,n_s}$ (având nodurile numerotate $1, 2, 3, \ldots, m$).
2. D generează s numere aleatoare distincte $x_1, \ldots, x_s \in \mathbb{Z}_q$.

1. Fie $q > m = card(V)$ un număr prim și $V_1, \ldots, V_s$ componentele grafului $K_{n_1,n_2,\ldots,n_s}$ (având nodurile numerotate $1, 2, 3, \ldots, m$).
2. D generează s numere aleatoare distincte $x_1, \ldots, x_s \in \mathbb{Z}_q$.
3. Dacă $S \in \mathbb{Z}_q$ este un secret, componentele sale se definesc prin

$$S_i = x_j \cdot S + r \ (mod\ q)$$

pentru $i \in V_j$ $(1 \leq j \leq s)$ și $r \in \mathbb{Z}_q$ arbitrar fixat.

1. Fie $q > m = card(V)$ un număr prim și $V_1, \ldots, V_s$ componentele grafului $K_{n_1,n_2,\ldots,n_s}$ (având nodurile numerotate $1, 2, 3, \ldots, m$).
2. D generează s numere aleatoare distincte $x_1, \ldots, x_s \in \mathbb{Z}_q$.
3. Dacă $S \in \mathbb{Z}_q$ este un secret, componentele sale se definesc prin
$$S_i = x_j \cdot S + r \ (mod\ q)$$
pentru $i \in V_j$ $(1 \leq j \leq s)$ și $r \in \mathbb{Z}_q$ arbitrar fixat.
4. D trimite fiecărui participant $u_i \in V_j$ componenta (S_i, x_j), $(1 \leq j \leq s)$.

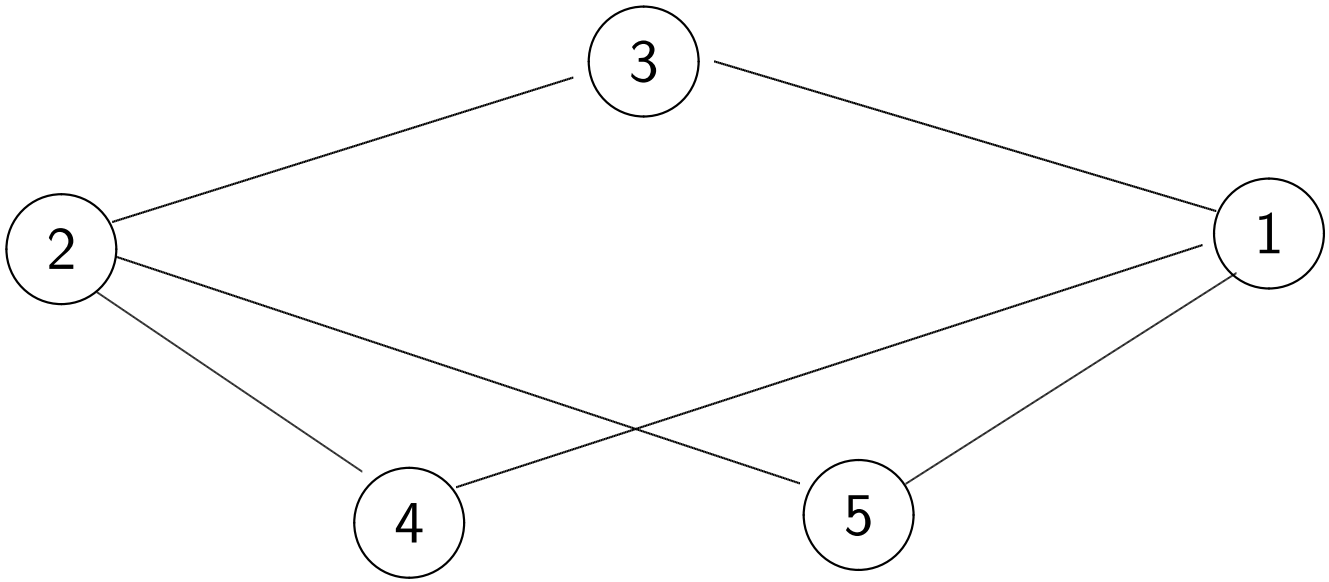
Deci $\mathcal{K} = \mathbb{Z}_q$, $\mathcal{S} = \mathbb{Z}_q \times \mathbb{Z}_q$.

1. Fie $q > m = card(V)$ un număr prim și $V_1, \ldots, V_s$ componentele grafului $K_{n_1,n_2,\ldots,n_s}$ (având nodurile numerotate $1, 2, 3, \ldots, m$).
2. D generează s numere aleatoare distincte $x_1, \ldots, x_s \in \mathbb{Z}_q$.
3. Dacă $S \in \mathbb{Z}_q$ este un secret, componentele sale se definesc prin
$$S_i = x_j \cdot S + r \ (mod\ q)$$
pentru $i \in V_j$ $(1 \leq j \leq s)$ și $r \in \mathbb{Z}_q$ arbitrar fixat.
4. D trimite fiecărui participant $u_i \in V_j$ componenta (S_i, x_j), $(1 \leq j \leq s)$.

Deci $\mathcal{K} = \mathbb{Z}_q$, $\mathcal{S} = \mathbb{Z}_q \times \mathbb{Z}_q$.
Oricare doi utilizatori $(u_1, u_2) \in V_{j_1}, \times V_{j_2}$, $(j_1 \neq j_2)$ pot recompune secretul S:
$$S = \frac{S_{u_1} - S_{u_2}}{x_{j_1} - x_{j_2}} \quad (mod\ q).$$

Exemplu

Să considerăm graful multipartit complet $K_{2,3}$:



Avem $s = 2$ și $V_1 = \{P_1, P_2\},\ V_2 = \{P_3, P_4, P_5\}$.
Fie $q = 11$ și valorile aleatoare $x_1 = 7,\ x_2 = 4,\ r = 8$.

Avem $s = 2$ și $V_1 = \{P_1, P_2\},\ V_2 = \{P_3, P_4, P_5\}$.
Fie $q = 11$ și valorile aleatoare $x_1 = 7,\ x_2 = 4,\ r = 8$.

Pentru secretul $S = 10$, componentele sale sunt:

$$S_1 = S_2 = x_1 \cdot S + r = 7 \cdot 10 + 8\ (mod\ 11) = 1,$$

$$S_3 = S_4 = S_5 = x_2 \cdot S + r = 4 \cdot 10 + 8\ (mod\ 11) = 4.$$

Avem $s = 2$ și $V_1 = \{P_1, P_2\},\ V_2 = \{P_3, P_4, P_5\}$.
Fie $q = 11$ și valorile aleatoare $x_1 = 7,\ x_2 = 4,\ r = 8$.

Pentru secretul $S = 10$, componentele sale sunt:

$$S_1 = S_2 = x_1 \cdot S + r = 7 \cdot 10 + 8\ (mod\ 11) = 1,$$

$$S_3 = S_4 = S_5 = x_2 \cdot S + r = 4 \cdot 10 + 8\ (mod\ 11) = 4.$$

Dacă participanții P_2 și P_3 vor să recompună secretul, ei vor calcula

$$S = \frac{S_1 - S_4}{x_1 - x_2} = \frac{1 - 4}{7 - 4} = 10\ (mod\ 11)$$

Construcție cu circuite monotone

Se construiește un circuit combinațional care "vede" structura de acces și generează un sistem de partajare a secretului.

Construcție cu circuite monotone

Se construiește un circuit combinațional care "vede" structura de acces și generează un sistem de partajare a secretului.

Fie $\mathbf{C}$ un circuit combinațional cu n intrări notate prin variabilele booleene $x_1, \ldots, x_n$ (asociate celor n participanți $P_1, \ldots, P_n$) și o ieșire booleană $y = \mathbf{C}(x_1, \ldots, x_n)$.
Presupunem că la construcția circuitului sunt folosite numai porți *AND* și *OR*.

Construcție cu circuite monotone

Se construiește un circuit combinațional care "vede" structura de acces și generează un sistem de partajare a secretului.

Fie $\mathbf{C}$ un circuit combinațional cu n intrări notate prin variabilele booleene $x_1, \ldots, x_n$ (asociate celor n participanți $P_1, \ldots, P_n$) și o ieșire booleană $y = \mathbf{C}(x_1, \ldots, x_n)$.
Presupunem că la construcția circuitului sunt folosite numai porți *AND* și *OR*.

Un astfel de circuit este "monoton" dacă modificarea valorii unei intrări din 0 în 1 nu va implica niciodată transformarea ieșirii y din 1 în 0.

Se notează

$$B(x_1, \ldots, x_n) = \{P_i \mid x_i = 1\}$$

mulțimea participanților. Dacă circuitul $\mathbf{C}$ este monoton:

$$\Gamma(\mathbf{C}) = \{B(x_1, \ldots, x_n) \mid \mathbf{C}(x_1, \ldots, x_n) = 1\}$$

o mulțime monotonă de părți ale lui $\mathcal{P}$.

Se notează

$$B(x_1, \ldots, x_n) = \{P_i \mid x_i = 1\}$$

mulțimea participanților. Dacă circuitul $\mathbf{C}$ este monoton:

$$\Gamma(\mathbf{C}) = \{B(x_1, \ldots, x_n) \mid \mathbf{C}(x_1, \ldots, x_n) = 1\}$$

o mulțime monotonă de părți ale lui $\mathcal{P}$.

Fiind dată o mulțime monotonă $\mathcal{A} \subseteq \mathcal{P}$, se poate construi ușor un circuit monoton $\mathbf{C}$ cu $\Gamma(\mathbf{C}) = \mathcal{A}$.

Exemplu de construcție circuit monoton

Fie $\mathcal{A}_{min}$ o bază a lui $\mathcal{A}$. Construim expresia booleană (în forma normal disjunctivă)

$$\bigvee_{B \in \mathcal{A}_{min}} \left(\bigwedge_{P_i \in B} P_i \right)$$

Fiecare clauză din această formă normală este legată printr-o poartă *AND*, iar disjuncția finală corespunde unei porți *OR*.

Numărul total de porți folosite este

$$1 + card(\mathcal{A}_{min})$$

Fie **C** un circuit monoton care recunoaște $\mathcal{A}$.
Mulțimea secretelor este $\mathcal{K} = \mathbb{Z}_q$ (q prim).

1. $x_{out} \leftarrow S$;

Fie **C** un circuit monoton care recunoaște $\mathcal{A}$.

Mulțimea secretelor este $\mathcal{K} = \mathbb{Z}_q$ (q prim).

1. $x_{out} \leftarrow S$;
2. pentru orice poartă G din care iese un arc marcat x, iar arcele care intră sunt nemarcate, execută:
   1. Dacă G este o poartă OR, atunci $x_V \leftarrow x$ pentru orice arc V care intră în G;

Fie **C** un circuit monoton care recunoaște $\mathcal{A}$.
Mulțimea secretelor este $\mathcal{K} = \mathbb{Z}_q$ (q prim).

1. $x_{out} \leftarrow S$;
2. pentru orice poartă G din care iese un arc marcat x, iar arcele care intră sunt nemarcate, execută:
   1. Dacă G este o poartă OR, atunci $x_V \leftarrow x$ pentru orice arc V care intră în G;
   2. Dacă G este o poartă AND și $V_1, \ldots, V_n$ sunt arcele care intră în G, atunci
      1. Alege aleator $x_{V,1}, \ldots, x_{V,n-1} \in \mathbb{Z}_q$;
      2. Calculează
      $$x_{V,n} = x - \sum_{i=1}^{n-1} x_{V,i} \pmod{q}$$
      3. Marchează arcul V_i cu $x_{V,i}$ $(1 \leq i \leq n)$.

Fie **C** un circuit monoton care recunoaște $\mathcal{A}$.
Mulțimea secretelor este $\mathcal{K} = \mathbb{Z}_q$ (q prim).

1. $x_{out} \leftarrow S$;
2. pentru orice poartă G din care iese un arc marcat x, iar arcele care intră sunt nemarcate, execută:
   1. Dacă G este o poartă OR, atunci $x_V \leftarrow x$ pentru orice arc V care intră în G;
   2. Dacă G este o poartă AND și $V_1, \ldots, V_n$ sunt arcele care intră în G, atunci
      1. Alege aleator $x_{V,1}, \ldots, x_{V,n-1} \in \mathbb{Z}_q$;
      2. Calculează
      $$x_{V,n} = x - \sum_{i=1}^{n-1} x_{V,i} \pmod q$$
      3. Marchează arcul V_i cu $x_{V,i}$ ($1 \leq i \leq n$).
3. D distribuie fiecărui participant P_i componenta S_i definită $S_i = \{x_{V,i} \mid V \text{ arc ce intră într-o poartă } AND\}$.

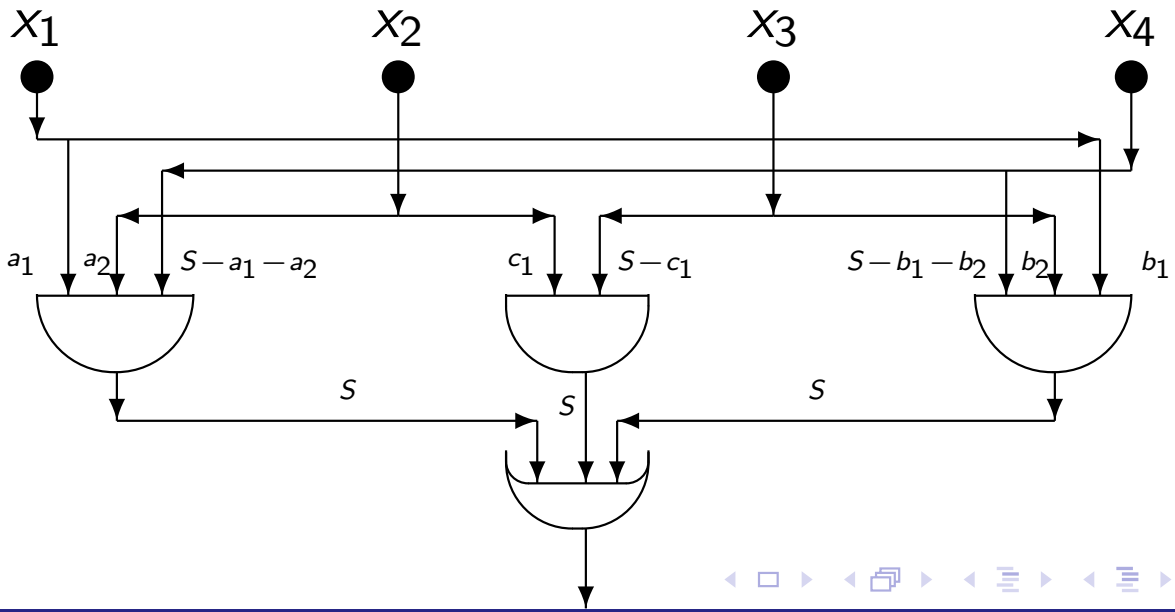
Exemplu

Pentru

$$\mathcal{A}_{min} = \{\{P_1, P_2, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3\}\}$$

se poate asocia expresia booleană

$$(P_1 \wedge P_2 \wedge P_4) \vee (P_1 \wedge P_3 \wedge P_4) \vee (P_2 \wedge P_3)$$



$a_1, a_2, b_1, b_2, c_1, c_2 \in \mathbb{Z}_q$ sunt alese aleator.
Fiecare participant primește drept componentă două numere:

1. a_1 și b_1 pentru P_1,
2. a_2 și c_1 pentru P_2,
3. b_2 și $S - c_1$ pentru P_3,
4. $S - a_1 - a_2$ și $S - b_1 - b_2$ pentru P_4.

Fiecare mulțime autorizată poate calcula (modulo q) valoarea lui S.

Despre mulțimile neautorizate

Este suficient să arătăm că mulțimile maximale neautorizate nu pot determina secretul, folosind informațiile pe care le dețin.

Despre mulțimile neautorizate

Este suficient să arătăm că mulțimile maximale neautorizate nu pot determina secretul, folosind informațiile pe care le dețin.

Exemplu

În exemplul anterior, mulțimile maximale neautorizate sunt

$$\{P_1, P_2\}, \{P_1, P_3\}, \{P_1, P_4\}, \{P_2, P_4\}, \{P_3, P_4\}$$

În fiecare caz, pentru determinarea secretului S lipsește o informație definită aleator.

Despre mulțimile neautorizate

Este suficient să arătăm că mulțimile maximale neautorizate nu pot determina secretul, folosind informațiile pe care le dețin.

Exemplu

În exemplul anterior, mulțimile maximale neautorizate sunt

$$\{P_1, P_2\}, \{P_1, P_3\}, \{P_1, P_4\}, \{P_2, P_4\}, \{P_3, P_4\}$$

În fiecare caz, pentru determinarea secretului S lipsește o informație definită aleator.
De exemplu, grupul $\{P_1, P_2\}$ deține informațiile a_1, a_2, b_1 și c_1.
Pentru a reconstitui S ar avea nevoie cel puțin de numărul $S - a_1 - a_2$, sau de $S - c_1$.

Teoremă

Fie $\mathbf{C}$ *un circuit combinațional monoton. El definește o schemă perfectă de partajare a secretului, a cărui structură de acces este* $\mathcal{A} = \Gamma(\mathbf{C})$.

Teoremă

Fie $\mathbf{C}$ *un circuit combinațional monoton. El definește o schemă perfectă de partajare a secretului, a cărui structură de acces este* $\mathcal{A} = \Gamma(\mathbf{C})$.

Când un grup autorizat B dorește aflarea secretului S, el trebuie să știe circuitul utilizat de D pentru construirea schemei și să deducă de aici ce componente sunt necesare pentru parcurgerea arcelor respective.

Această informație trebuie să fie publică; numai valorile componentelor S_i trebuie să fie secrete.

Rata de informație

Fie $\mathcal{P}$ o mulțime de participanți și $\mathcal{S}$ spațiul tuturor componentelor posibile ale secretului S. O distribuție de componente este o funcție

$$f : \mathcal{P} \longrightarrow \mathcal{S}$$

$f(P_i)$ va fi componenta distribuită participantului P_i $(1 \leq i \leq n)$.

Rata de informație

Fie $\mathcal{P}$ o mulțime de participanți și $\mathcal{S}$ spațiul tuturor componentelor posibile ale secretului S. O distribuție de componente este o funcție

$$f : \mathcal{P} \longrightarrow \mathcal{S}$$

$f(P_i)$ va fi componenta distribuită participantului P_i $(1 \leq i \leq n)$. Pentru fiecare $S \in \mathcal{K}$, fie $\mathcal{F}_S$ mulțimea tuturor distribuțiilor posibile ale lui S. Definim ansamblul complet al tuturor distribuțiilor posibile de secrete

$$\mathcal{F} = \bigcup_{S \in \mathcal{K}} \mathcal{F}_S$$

Rata de informație

Fie $\mathcal{P}$ o mulțime de participanți și $\mathcal{S}$ spațiul tuturor componentelor posibile ale secretului S. O distribuție de componente este o funcție

$$f : \mathcal{P} \longrightarrow \mathcal{S}$$

$f(P_i)$ va fi componenta distribuită participantului P_i $(1 \leq i \leq n)$. Pentru fiecare $S \in \mathcal{K}$, fie $\mathcal{F}_S$ mulțimea tuturor distribuțiilor posibile ale lui S. Definim ansamblul complet al tuturor distribuțiilor posibile de secrete

$$\mathcal{F} = \bigcup_{S \in \mathcal{K}} \mathcal{F}_S$$

Rolul arbitrului este de a selecta aleator un element $f \in \mathcal{F}_S$ și de a distribui componentele în conformitate cu această alegere.

Pentru o submulțime $B \subseteq \mathcal{P}$ (autorizată sau nu) de participanți, se definește

$$S(B) = \{f^B \mid f \in \mathcal{F}\},$$

unde funcția

$$f^B : B \longrightarrow \mathcal{S}$$

este restricția distribuției de componente f la submulțimea B; este definită

$$f^B(P_i) = f(P_i), \quad \forall P_i \in B.$$

Pentru o submulțime $B \subseteq \mathcal{P}$ (autorizată sau nu) de participanți, se definește

$$S(B) = \{f^B \mid f \in \mathcal{F}\},$$

unde funcția

$$f^B : B \longrightarrow \mathcal{S}$$

este restricția distribuției de componente f la submulțimea B; este definită

$$f^B(P_i) = f(P_i), \quad \forall P_i \in B.$$

Deci $S(B)$ este mulțimea tuturor distribuțiilor posibile ale componentelor secretului S la participanții din B.

Evaluare performanțe

În cazul unei scheme de partajare (n, k) - majoritare, circuitul boolean construit pe baza expresiei scrise în forma normal disjunctivă are $1 + C_n^k$ porți.
Fiecare participant primește o componentă formată din C_{n-1}^{k-1} numere din $\mathbb{Z}_p$.

Evaluare performanțe

În cazul unei scheme de partajare (n, k) - majoritare, circuitul boolean construit pe baza expresiei scrise în forma normal disjunctivă are $1 + C_n^k$ porți.
Fiecare participant primește o componentă formată din C_{n-1}^{k-1} numere din $\mathbb{Z}_p$.

Această partajare este foarte slabă comparativ cu schema Shamir (n, k) - majoritară, care oferă același rezultat folosind componente formate dintr-un singur număr.

Pentru măsurarea performanțelor sistemelor perfecte de partajare a secretelor se folosește rată de informație.

Pentru măsurarea performanțelor sistemelor perfecte de partajare a secretelor se folosește rată de informație.

Definiție

Fie un sistem perfect de partajare a secretelor cu structura de acces $\mathcal{A}$. Rata de informație a unui participant P_i este

$$\rho_i = \frac{log_2(card(\mathcal{K}))}{log_2(card(S(P_i)))}.$$

unde $S(P_i) \subseteq \mathcal{S}$ este mulțimea componentelor posibile pe care le poate primi participantul P_i.
Rata de informație a sistemului este

$$\rho = min\{\rho_i \mid 1 \leq i \leq n\}$$

Dacă se construiește un sistem de partajare a secretelor plecând de la un circuit monoton $\mathbf{C}$, rata sa de informație se obține folosind teorema:

Teoremă

Fie $\mathbf{C}$ un circuit combinațional monoton. Există atunci o schemă perfectă de partajare a secretelor, cu structura de acces $\mathcal{A} = \Gamma(\mathbf{C})$, care admite ca rată de informație

$$\rho = \max_{1 \leq i \leq n} \left\{ \frac{1}{r_i} \right\}$$

unde r_i este numărul arcelor de intrare din circuit (pentru valorile x_i).

Evident, este preferabilă o rată de informație cât mai mare.
Valoarea ei este însă limitată superior.

Teoremă

Pentru orice schemă perfectă de partajare a secretelor cu structura de acces $\mathcal{A}$, rata de informație verifică inegalitatea $\rho \leq 1$.

O schemă de partajare cu $\rho = 1$ va fi numită "ideală".

Ca un exemplu, schema de partajare majoritară Shamir are $\rho = 1$, deci este o schemă ideală.

O schemă de partajare cu $\rho = 1$ va fi numită "ideală".

Ca un exemplu, schema de partajare majoritară Shamir are $\rho = 1$, deci este o schemă ideală.

În schimb, rata de informație pentru o schemă de partajare (n, k) - majoritară bazată pe circuite monotone construite cu forma normal disjunctivă este $\frac{1}{C_{n-1}^{k-1}}$, extrem de ineficientă când $1 < k < n$.

Schema de partajare Brickell

Cunoscută și sub numele de construcția vectorială a lui Brickell.

Fie $\mathcal{A}$ o structură de acces, q prim și $d \geq 2$. Definim funcția

$$\phi : \mathcal{P} \longrightarrow \mathbb{Z}_q^d$$

cu proprietatea $(1, 0, \ldots, 0) \in \langle \phi(P_i) \mid P_i \in B \rangle \iff B \in \mathcal{A}$. (1)

Schema de partajare Brickell

Cunoscută și sub numele de construcția vectorială a lui Brickell.

Fie $\mathcal{A}$ o structură de acces, q prim și $d \geq 2$. Definim funcția

$$\phi : \mathcal{P} \longrightarrow \mathbb{Z}_q^d$$

cu proprietatea $(1, 0, \ldots, 0) \in \langle \phi(P_i) \mid P_i \in B \rangle \iff B \in \mathcal{A}$. (1)

Altfel spus, vectorul $(1, 0, \ldots, 0)$ este o combinație liniară de vectori din mulțimea

$$\{\phi(P_i) \mid P_i \in B\}$$

dacă și numai dacă B este o mulțime autorizată.

Pe baza lui ϕ se construiește o schemă de partajare a secretelor cu $\mathcal{K} = S(P_i) = \mathbb{Z}_q$ $(1 \leq i \leq n)$.
Pentru orice $\mathbf{a} = (a_1, \ldots, a_d) \in \mathbb{Z}_p^d$ definim o funcție de distribuție a componentelor $f_{\mathbf{a}} : \mathcal{P} \longrightarrow \mathcal{S}$ prin $f_{\mathbf{a}}(x) = \mathbf{a} \cdot \phi(x)$.

Pe baza lui ϕ se construiește o schemă de partajare a secretelor cu $\mathcal{K} = S(P_i) = \mathbb{Z}_q$ $(1 \leq i \leq n)$.
Pentru orice $\mathbf{a} = (a_1, \ldots, a_d) \in \mathbb{Z}_p^d$ definim o funcție de distribuție a componentelor $f_{\mathbf{a}} : \mathcal{P} \longrightarrow \mathcal{S}$ prin $f_{\mathbf{a}}(x) = \mathbf{a} \cdot \phi(x)$.
Schema Brickell de partajare a secretelor este:

1. Pentru $1 \leq i \leq n$, D atribuie lui P_i vectorul $\phi(P_i) \in \mathbb{Z}_q^d$. Acești vectori sunt publici.

Pe baza lui ϕ se construiește o schemă de partajare a secretelor cu $\mathcal{K} = S(P_i) = \mathbb{Z}_q$ $(1 \leq i \leq n)$.
Pentru orice $\mathbf{a} = (a_1, \ldots, a_d) \in \mathbb{Z}_p^d$ definim o funcție de distribuție a componentelor $f_{\mathbf{a}} : \mathcal{P} \longrightarrow \mathcal{S}$ prin $f_{\mathbf{a}}(x) = \mathbf{a} \cdot \phi(x)$.
Schema Brickell de partajare a secretelor este:

1. Pentru $1 \leq i \leq n$, D atribuie lui P_i vectorul $\phi(P_i) \in \mathbb{Z}_q^d$. Acești vectori sunt publici.
2. Pentru partajarea secretului $S \in \mathbb{Z}_q$, D alege $d-1$ elemente aleatoare $a_2, \ldots, a_d \in \mathbb{Z}_q$.
3. Folosind vectorul $\mathbf{a} = (S, a_2, \ldots, a_d)$, arbitrul calculează componentele

$$S_i = \mathbf{a} \cdot \phi(P_i) \quad (1 \leq i \leq n)$$

Pe baza lui ϕ se construiește o schemă de partajare a secretelor cu $\mathcal{K} = S(P_i) = \mathbb{Z}_q$ $(1 \leq i \leq n)$.
Pentru orice $\mathbf{a} = (a_1, \ldots, a_d) \in \mathbb{Z}_p^d$ definim o funcție de distribuție a componentelor $f_{\mathbf{a}} : \mathcal{P} \longrightarrow \mathcal{S}$ prin $f_{\mathbf{a}}(x) = \mathbf{a} \cdot \phi(x)$.
Schema Brickell de partajare a secretelor este:

1. Pentru $1 \leq i \leq n$, D atribuie lui P_i vectorul $\phi(P_i) \in \mathbb{Z}_q^d$. Acești vectori sunt publici.
2. Pentru partajarea secretului $S \in \mathbb{Z}_q$, D alege $d-1$ elemente aleatoare $a_2, \ldots, a_d \in \mathbb{Z}_q$.
3. Folosind vectorul $\mathbf{a} = (S, a_2, \ldots, a_d)$, arbitrul calculează componentele

$$S_i = \mathbf{a} \cdot \phi(P_i) \quad (1 \leq i \leq n)$$

4. Pentru $i = 1, 2, \ldots, n$ arbitrul D trimite componenta S_i participantului P_i.

Teoremă

Dacă ϕ verifică proprietatea (1), mulțimea distribuțiilor de componente $\mathcal{F}_S$, $S \in \mathcal{K}$ formează o schemă perfectă de partajare a secretelor, cu structura de acces $\mathcal{A}$.

Teoremă

Dacă ϕ verifică proprietatea (1), mulțimea distribuțiilor de componente $\mathcal{F}_S$, $S \in \mathcal{K}$ formează o schemă perfectă de partajare a secretelor, cu structura de acces $\mathcal{A}$.

Schema de partajare (n, k) - majoritară a lui Shamir este un caz particular.

Într-adevăr, fie $d = k$ și

$$\phi(P_i) = (1, x_i, x_i^2, \ldots x_i^{k-1})$$

pentru $1 \leq i \leq n$, unde x_i este coordonata x dată de P_i.
Sistemul obținut este echivalent cu sistemul din schema lui Shamir.

Scheme de partajare fără arbitru

Există posibilitatea ca un anumit secret să fie partiționat fără a face apel la arbitru.
În acest caz, secretul va fi considerat implicit și va fi aflat doar atunci când este reconstituit de o mulțime autorizată de acces.

Scheme de partajare fără arbitru

Există posibilitatea ca un anumit secret să fie partiționat fără a face apel la arbitru.
În acest caz, secretul va fi considerat implicit și va fi aflat doar atunci când este reconstituit de o mulțime autorizată de acces.

Ideea construirii unei astfel de partajări de secrete apare prima oară la C. Meadows, dar o schemă funcțională este propusă de Ingermarsson și Simmons.
Aici, utilizatorul P_i alege un număr S_i care va fi una din cele n componente ale unui secret S, pe care îl partajează pentru ceilalți utilizatori.

Se obține astfel o schemă de partajare $(n, n-1)$ - majoritară.

1. Fiecare participant P_i alege un număr aleator $S_i \in \mathbb{Z}_q$ (q număr prim fixat și public);

1. Fiecare participant P_i alege un număr aleator $S_i \in \mathbb{Z}_q$ (q număr prim fixat și public);
2. P_i generează aleator componentele $S_{i,j}$ astfel ca

$$S_i = \sum_{(j=1)\&(j\neq i)}^{n} S_{i,j} \pmod q$$

1. Fiecare participant P_i alege un număr aleator $S_i \in \mathbb{Z}_q$ (q număr prim fixat și public);
2. P_i generează aleator componentele $S_{i,j}$ astfel ca

$$S_i = \sum_{(j=1)\&(j\neq i)}^{n} S_{i,j} \pmod q$$

3. P_i trimite fiecărui participant P_j ($j \neq i$) componenta $S_{i,j}$.

Fiecare participant P_i va dispune de componenta

$$(S_i, S_{1,i}, \ldots, S_{i-1,i}, S_{i+1,i}, \ldots, S_{n,i}).$$

1. Fiecare participant P_i alege un număr aleator $S_i \in \mathbb{Z}_q$ (q număr prim fixat și public);
2. P_i generează aleator componentele $S_{i,j}$ astfel ca

$$S_i = \sum_{(j=1)\&(j\neq i)}^{n} S_{i,j} \pmod q$$

3. P_i trimite fiecărui participant P_j ($j \neq i$) componenta $S_{i,j}$.

Fiecare participant P_i va dispune de componenta

$$(S_i, S_{1,i}, \ldots, S_{i-1,i}, S_{i+1,i}, \ldots, S_{n,i}).$$

Dacă primii $n-1$ participanți vor să recompună secretul, vor calcula

$$S = \sum_{i=1}^{n-1} S_i + \sum_{i=1}^{n-1} S_{n,i} = \sum_{i=1}^{n-1} S_i + S_n = S \pmod q$$

Generalizare la o mulțime de acces arbitrară $\mathcal{A}$.

1. Se folosește o schemă unanimă de ordin k pentru a construi componentele $S_1, \ldots, S_k$ ale unui secret S;
2. Utilizatorul P_i $(1 \leq i \leq k)$ împarte S_i (considerat ca un secret al cărui arbitru este) în componente, pentru o mulțime de acces $\mathcal{A}_i$, apoi împarte aceste componente utilizatorilor din $\mathcal{A}_i$.

Pentru construcția mulțimilor de acces $\mathcal{A}_i$:

1. Se ia mulțimea $\mathcal{A} \cup \{\{P_1\}\}$ și se consideră baza ei; fie $\mathcal{A}_{1,min}$ această bază;
2. $\mathcal{A}_1$ este mulțimea generată de $\mathcal{A}_{1,min}$;

Pentru construcția mulțimilor de acces $\mathcal{A}_i$:

1. Se ia mulțimea $\mathcal{A} \cup \{\{P_1\}\}$ și se consideră baza ei; fie $\mathcal{A}_{1,min}$ această bază;
2. $\mathcal{A}_1$ este mulțimea generată de $\mathcal{A}_{1,min}$;
3. Pentru $i = 2, \ldots, k$:
   1. Se elimină utilizatorii $P_1, \ldots, P_{i-1}$ din $\mathcal{P}$; fie $\mathcal{A}'$ noua mulțime de acces;

Pentru construcția mulțimilor de acces $\mathcal{A}_i$:

1. Se ia mulțimea $\mathcal{A} \cup \{\{P_1\}\}$ și se consideră baza ei; fie $\mathcal{A}_{1,min}$ această bază;
2. $\mathcal{A}_1$ este mulțimea generată de $\mathcal{A}_{1,min}$;
3. Pentru $i = 2, \ldots, k$:
   1. Se elimină utilizatorii $P_1, \ldots, P_{i-1}$ din $\mathcal{P}$; fie $\mathcal{A}'$ noua mulțime de acces;
   2. Se construiește $\mathcal{A}' \cup \{\{P_i\}\}$ și se determină baza ei: $\mathcal{A}_{i,min}$;
   3. $\mathcal{A}_i$ este mulțimea de acces generată de $\mathcal{A}_{i,min}$.

Exemplu

Să construim o schemă de partajare $(4, 3)$ - majoritară, fără arbitru, pentru mulțimea de acces având baza

$$\mathcal{A}_{min} = \{\{P_1, P_2, P_3\}, \{P_1, P_2, P_4\}, \{P_1, P_3, P_4\}, \{P_2, P_3, P_4\}\}$$

Cele trei baze sunt:
$\mathcal{A}_{1,min} = \{\{P_1\}, \{P_2, P_3, P_4\}\}$,
$\mathcal{A}_{2,min} = \{\{P_2\}, \{P_3, P_4\}\}$,
$\mathcal{A}_{3,min} = \{\{P_3\}, \{P_4\}\}$

continuare

Utilizatorul P_1:

1. Generează aleator S_1;
2. Construiește – prin partajare majoritară – componentele $S_{1,2}, S_{1,3}, S_{1,4}$ ale secretului S_1;
3. Distribuie $S_{1,j}$ participantului P_j $(j = 2, 3, 4)$.

continuare

Utilizatorul P_1:

1. Generează aleator S_1;
2. Construiește – prin partajare majoritară – componentele $S_{1,2}, S_{1,3}, S_{1,4}$ ale secretului S_1;
3. Distribuie $S_{1,j}$ participantului P_j $(j = 2, 3, 4)$.

Utilizatorul P_2:

1. Generează aleator S_2;
2. Construiește – prin partajare majoritară – componentele $S_{2,3}, S_{2,4}$ ale secretului S_2;
3. Distribuie $S_{2,j}$ participantului P_j $(j = 3, 4)$.

continuare

Utilizatorul P_1:

1. Generează aleator S_1;
2. Construiește – prin partajare majoritară – componentele $S_{1,2}, S_{1,3}, S_{1,4}$ ale secretului S_1;
3. Distribuie $S_{1,j}$ participantului P_j $(j = 2, 3, 4)$.

Utilizatorul P_2:

1. Generează aleator S_2;
2. Construiește – prin partajare majoritară – componentele $S_{2,3}, S_{2,4}$ ale secretului S_2;
3. Distribuie $S_{2,j}$ participantului P_j $(j = 3, 4)$.

Utilizatorul P_3:

1. Generează aleator S_3;
2. Trimite $S_{3,4} = S_3$ participantului P_4.

continuare

Deci,secretul este $S = S_1 + S_2 + S_3$ cu
$S_1 = S_{1,2} + S_{1,3} + S_{1,4}$,
$S_2 = S_{2,3} + S_{2,4}$,
$S_3 = S_{3,4}$

continuare

Deci,secretul este $S = S_1 + S_2 + S_3$ cu
$S_1 = S_{1,2} + S_{1,3} + S_{1,4}$,
$S_2 = S_{2,3} + S_{2,4}$,
$S_3 = S_{3,4}$
În plus,
P_1 deține $\{S_1\}$,
P_2 deține $\{S_2, S_{1,2}\}$
P_3 deține $\{S_3, S_{1,3}, S_{2,3}\}$,
P_4 deține $\{S_{1,4}, S_{2,4}, S_{3,4}\}$.

continuare

Deci,secretul este $S = S_1 + S_2 + S_3$ cu
$S_1 = S_{1,2} + S_{1,3} + S_{1,4}$,
$S_2 = S_{2,3} + S_{2,4}$,
$S_3 = S_{3,4}$
În plus,
P_1 deține $\{S_1\}$,
P_2 deține $\{S_2, S_{1,2}\}$
P_3 deține $\{S_3, S_{1,3}, S_{2,3}\}$,
P_4 deține $\{S_{1,4}, S_{2,4}, S_{3,4}\}$.
Dacă – de exemplu – mulțimea autorizată de acces $\{P_1, P_3, P_4\}$ vrea să găsească secretul, va calcula

$$S = S_1 + S_3 + S_{2,3} + S_{2,4}$$

Scheme de partajare verificabile

În toate schemele prezentate s-a presupus că părțile implicate (arbitrul și participanții) se comportă onest.

Scheme de partajare verificabile

În toate schemele prezentate s-a presupus că părțile implicate (arbitrul și participanții) se comportă onest.

Cazul când arbitrul D trișează este studiat prima oară de Chor, Goldwasser, Micali și Awerbuch; ei introduc și noțiunea de *schemă de partajare verificabilă*, unde fiecare participant poate verifica dacă primit o componentă validă.

Schema de partajare Feldman

Verificare a sistemului de partajare Shamir.

1. Se generează numerele prime p, q astfel ca $q|(p-1)$; fie $\alpha \in \mathbb{Z}_p^*$ un element de ordin q. Toate sunt publice.

Schema de partajare Feldman

Verificare a sistemului de partajare Shamir.

1. Se generează numerele prime p, q astfel ca $q|(p-1)$; fie $\alpha \in \mathbb{Z}_p^*$ un element de ordin q. Toate sunt publice.
2. D generează polinomul

$$a(X) = a_0 + a_1 X + \cdots + a_{k-1} X^{k-1} \in \mathbb{Z}_q[X]$$

cu $a_0 = S$; face publice $\alpha_i = \alpha^{a_i} \ (mod\ p)$, $(i \leq k-1)$;

Schema de partajare Feldman

Verificare a sistemului de partajare Shamir.

1. Se generează numerele prime p, q astfel ca $q|(p-1)$; fie $\alpha \in \mathbb{Z}_p^*$ un element de ordin q. Toate sunt publice.
2. D generează polinomul

$$a(X) = a_0 + a_1 X + \cdots + a_{k-1} X^{k-1} \in \mathbb{Z}_q[X]$$

cu $a_0 = S$; face publice $\alpha_i = \alpha^{a_i} \ (mod\ p)$, $(i \leq k-1)$;
3. D distribuie (prin canal securizat) către fiecare participant P_i componenta $S_i = a(i) \quad (i = 1, \ldots, n)$;

Schema de partajare Feldman

Verificare a sistemului de partajare Shamir.

1. Se generează numerele prime p, q astfel ca $q|(p-1)$; fie $\alpha \in \mathbb{Z}_p^*$ un element de ordin q. Toate sunt publice.
2. D generează polinomul
$$a(X) = a_0 + a_1 X + \cdots + a_{k-1} X^{k-1} \in \mathbb{Z}_q[X]$$
cu $a_0 = S$; face publice $\alpha_i = \alpha^{a_i} \ (mod\ p)$, $(i \leq k-1)$;
3. D distribuie (prin canal securizat) către fiecare participant P_i componenta $S_i = a(i) \quad (i = 1, \ldots, n)$;

P_i verifică corectitudinea componentei primite S_i testând

$$\alpha^{S_i} \ (mod\ p) \ = \ \prod_{j=0}^{k-1} \alpha_j^{ij} \ (mod\ p)$$

Schema lui Pedersen

1. Fie p, q prime cu $q|(p-1)$ și $g, h \in \mathbb{Z}_p^*$ de ordin q (toate publice).

Schema lui Pedersen

1. Fie p, q prime cu $q|(p-1)$ și $g, h \in \mathbb{Z}_p^*$ de ordin q (toate publice).
2. D calculează $E_0 = g^S h^t \ (mod\ p)$ unde $t \in \mathbb{Z}_q$ este arbitrar;

Schema lui Pedersen

1. Fie p, q prime cu $q|(p-1)$ și $g, h \in \mathbb{Z}_p^*$ de ordin q (toate publice).
2. D calculează $E_0 = g^S h^t \ (mod\ p)$ unde $t \in \mathbb{Z}_q$ este arbitrar;
3. D generează
   $a(X) = S + a_1 X + \cdots + a_{k-1} X^{k-1} \in \mathbb{Z}_[X]$,
   $b(X) = t + b_1 X + \cdots + b_{k-1} X^{k-1} \in \mathbb{Z}_q[X]$,
   calculează valorile $E_i = g^{a_i} h^{b_i} \ (mod\ p)$ și face public vectorul $(E_0, E_1, \ldots, E_n)$;

Schema lui Pedersen

1. Fie p, q prime cu $q|(p-1)$ și $g, h \in \mathbb{Z}_p^*$ de ordin q (toate publice).
2. D calculează $E_0 = g^S h^t \ (mod\ p)$ unde $t \in \mathbb{Z}_q$ este arbitrar;
3. D generează
   $a(X) = S + a_1 X + \cdots + a_{k-1} X^{k-1} \in \mathbb{Z}_[X]$,
   $b(X) = t + b_1 X + \cdots + b_{k-1} X^{k-1} \in \mathbb{Z}_q[X]$,
   calculează valorile $E_i = g^{a_i} h^{b_i} \ (mod\ p)$ și face public vectorul $(E_0, E_1, \ldots, E_n)$;
4. D distribuie (prin canal securizat) către fiecare participant componenta

$$S_i = (a(i), b(i)), \quad i = 1, \ldots, n$$

Fiecare utilizator P_i poate testa corectitudinea componentei $S_i = (s_i, t_i)$ primite, verificând egalitatea

$$g^{s_i} h^{t_i} = \prod_{j=0}^{k-1} E_j^{i^j} \pmod{p}$$

Fiecare utilizator P_i poate testa corectitudinea componentei $S_i = (s_i, t_i)$ primite, verificând egalitatea

$$g^{s_i} h^{t_i} = \prod_{j=0}^{k-1} E_j^{i^j} \pmod p$$

Schema Pedersen are proprietăți de liniaritate: astfel, dacă secretele A_1, A_2 sunt definite prin componentele $(s_{i,j}, t_{i,j})$, $i = 1, \ldots, n$, $j = 1, 2$, atunci, pentru orice valori $a_1, a_2 \in \mathbb{Z}_p$ secretul $a_1 A_1 + a_2 A_2$ este definit de componentele

$$(a_1 \cdot s_{i,1} + a_2 \cdot s_{i,2}), \quad i = 1, 2, \ldots, n$$

toate calculele fiind efectuate în $\mathbb{Z}_q$.

The END