Część 3

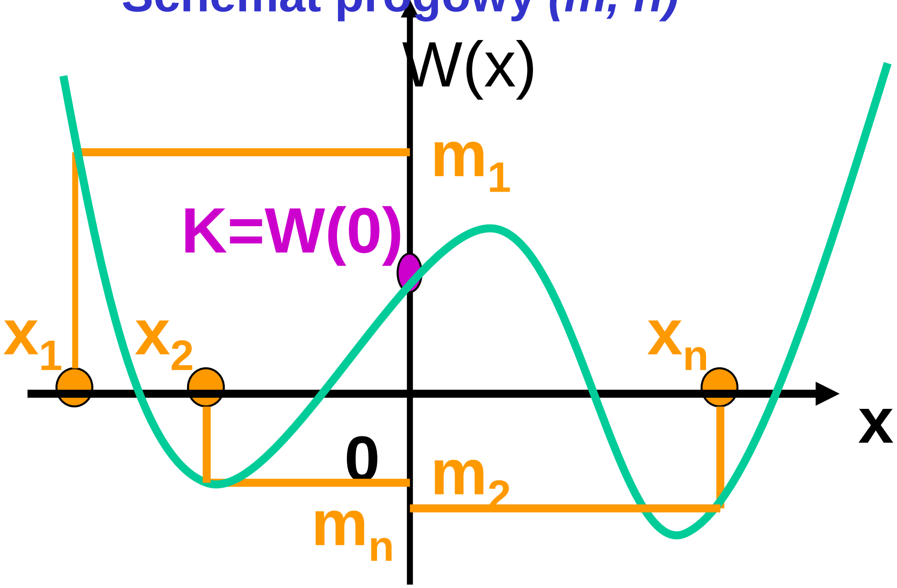
Progowe metody podziału sekretów i obliczenia grupowe

Przypomnienie

Metody progowe podziału sekretu **powinny spełniać następujące warunki:**

- **znajomość $m \le n$ cieni sekretu umożliwia łatwe odtworzenie sekretu;**
- **odtworzenie sekretu na podstawie znajomości $i < m$ cieni jest problemem trudnym obliczeniowo.**

Schemat progowy *(m, n)*



$$W(x) = a_{m-1} x^{m-1} + \dots + a_1 x + K$$

Schemat progowy *(n, n)*

Sekret K - k-bitowy ciąg binarny

Utworzyć (n-1) k-bitowych ciągów losowych S(1), S(2), ..., S(n-1)

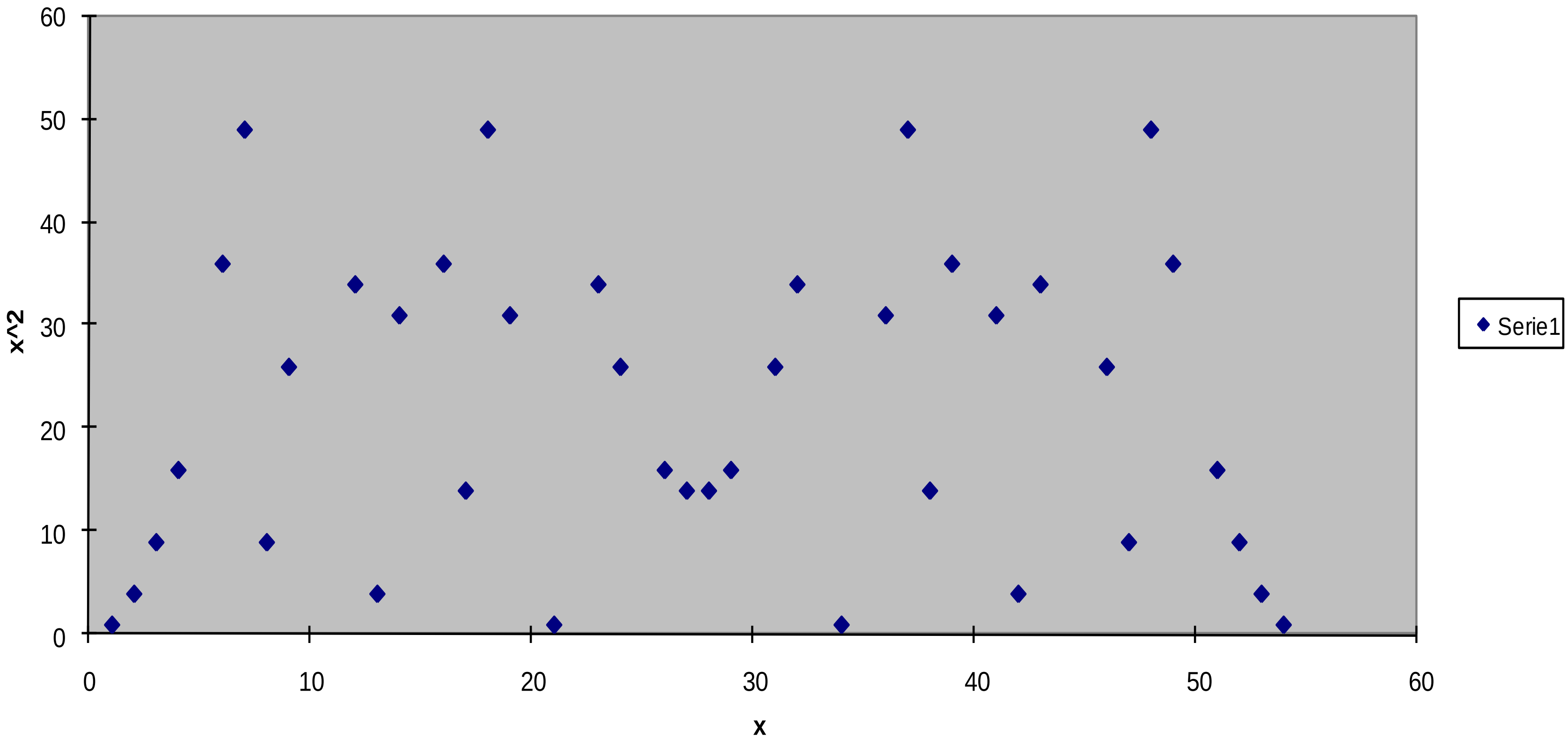
S(1) -> P1
S(2) -> P2
....
S(n-1) -> Pn-1
$S(n) = K \oplus S(1) \oplus S(2) \oplus \dots \oplus S(n-1)$ -> Pn

Odtwarzanie sekretu:

$K = S(1) \oplus S(2) \oplus \dots \oplus S(n-1) \oplus S(n)$

Praktyka – klasyczne komputery przetwarzają ciągi binarne, które można utożsamiać z liczbami całkowitymi

Wartości funkcji $f(x) = x^2$ w grupie multiplikatywnej Z^*_{55}



Metoda wielomianu interpolacyjnego Lagrange'a (A.Shamir) Schemat progowy (m, n)

Określa się wielomian stopnia $m-1$ o losowych współczynnikach a_i :

$$W(x) = (a_{m-1} x^{m-1} + \dots + a_1 x + a_0) \bmod p,$$

gdzie p jest liczbą pierwszą większą niż M i n, zaś $a_0 = M$ jest wartością liczbową „ukrywanego” metodą progową sekretu;

$$W(0) = M \bmod p = M$$

Arbitralnie (np. wykorzystując generator liczb losowych) wybiera się n różnych liczb x_i (często rezygnuje się z „losowości” wybierając kolejne liczby naturalne 1, 2, ..., n).

Cienie (udziały) wiadomości M określa się z zależności :

$$m_i = W(x_i) \bmod p$$

UWAGA: Atak brutalny pozornie wymaga sprawdzenia $\approx p^m$ wariantów.

W rzeczywistości wiedząc, że $M < p$ wystarczy $\approx p/2$ prób.

Metoda wielomianu interpolacyjnego Lagrange'a (A.Shamir)
Schemat progowy (m, n) - Rekonstrukcja sekretu

$$W(x) = \sum_{s=1}^{m} m_{is} \prod_{j=1, m \neq s}^{m} (x - x_{ij}) / (x_{is} - x_{ij}) \mod p$$

UWAGA 1: Interesująca jest wyłącznie wartość W(x) dla x = 0.

UWAGA 2: jeśli –p < (a – b) < 0, to (a – b) mod p = (a – b + p).

UWAGA 3: 1/a należy interpretować jako a^{-1} mod p.

Wykorzystanie rozszerzonego algorytmu Euklidesa do obliczania a^{-1} mod n

```
long inverse_modulo(long a, long n)
{
long g[3], u[3], v[3];
long x, y;
 g[0] = n;
 g[1] = a;
 u[0] = v[1] = 1L;
 u[1] = v[0] = 0L;
 while(g[1])
         {
         y = g[0] / g[1];
         g[2] = g[0] - y * g[1];
         u[2] = u[0] - y * u[1];
         v[2] = v[0] - y * v[1];
         g[0] = g[1];
         g[1] = g[2];
         u[0] = u[1];
         u[1] = u[2];
         v[0] = v[1];
         v[1] = v[2];
         }
 x = v[0];
 if(x < 0) x += n;
 return(x);
}
```

Przykład:

Niech wartość sekretu $M = 11$, ilość cieni $n = 5$, zaś wartość progowa $m = 3$.

Po określeniu modułu przekształcenia $p = 13$ i losowo wybranych współczynników:

$$a_2 = 7 \quad \text{i} \quad a_1 = 8$$

odpowiedni wielomian Lagrange'a przyjmuje postać:

$$W(x) = 7\,x^2 + 8\,x + 11 \ (\,mod\ 13)$$

Wyznacza się pięć *cieni*:

dla	$x_1 = 1$	$m_1 = W(1)\ mod\ 13 = 26\ (mod\ 13) = 0$
dla	$x_2 = 2$	$m_2 = W(2)\ mod\ 13 = 55\ (mod\ 13) = 3$
dla	$x_3 = 3$	$m_3 = W(3)\ mod\ 13 = 98\ (mod\ 13) = 7$
dla	$x_4 = 4$	$m_4 = W(4)\ mod\ 13 = 155\ (mod\ 13) = 12$
dla	$x_5 = 5$	$m_5 = W(5)\ mod\ 13 = 226\ (mod\ 13) = 5$

Odtworzenie sekretu na podstawie znajomości m_2, m_3 i m_5:

$$W(x) = m_2 [(x - x_3) / (x_2 - x_3)] [(x - x_5) / (x_2 - x_5)] +$$
$$+ m_3 [(x - x_2 / (x_3 - x_2)] [(x - x_5) / (x_3 - x_5)] +$$
$$+ m_5 [(x - x_2) / (x_5 - x_2)] [(x - x_3) / (x_5 - x_3)] \pmod p$$

$$M = W(0) = m_2 [-x_3 / (x_2 - x_3)] [-x_5 / (x_2 - x_5)] +$$
$$+ m_3 [-x_2 / (x_3 - x_2)] [-x_5 / (x_3 - x_5)] +$$
$$+ m_5 [-x_2 / (x_5 - x_2)] [-x_3 / (x_5 - x_3)] \pmod p$$

$M = W(0) =$

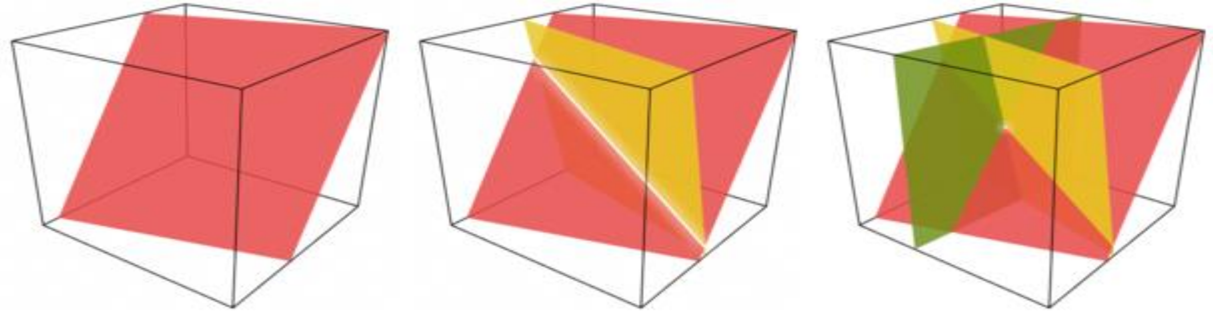
$$= 3[-3/(-1)][-5/(-3)] + 7[-2/1][-5/(-2)] + 5[-2/3][-3/2] \pmod{13} =$$

$-1 \bmod 13 = 12$	$-2 \bmod 13 = 11$	$-3 \bmod 13 = 10$	$-5 \bmod 13 = 8$
$1^{-1} \bmod 13 = 1$	$2^{-1} \bmod 13 = 7$	$3^{-1} \bmod 13 = 9$	
$10^{-1} \bmod 13 = 4$	$11^{-1} \bmod 13 = 6$	$12^{-1} \bmod 13 = 12$	

$$= 3*(10*12)*(8*4) + 7*(11*1)*(8*6) + 5*(11*9)*(10*7) \pmod{13} =$$
$$= [3*(120 \bmod 13)*(32 \bmod 13) + 7*(11 \bmod 13)*(48 \bmod 13) +$$
$$+ 5*(99 \bmod 13)*(70 \bmod 13)] \pmod{13} =$$
$$= [(3*3*6)\pmod{13} + (7*11*9)\pmod{13} + (5*8*5)\pmod{13}] \pmod{13} =$$
$$= [54 \bmod 13 + 693 \bmod 13 + 200 \bmod 13] \pmod{13} =$$
$$= [2 + 4 + 5] \pmod{13} = 11 \bmod 13 = 11$$

Metoda wektorów w przestrzeniach m-wymiarowych (G.R.Blakley)

Metoda w swej koncepcji zbliżona do metody wielomianu Lagrange'a-Shamira, wykorzystująca fakt, że jeżeli *m* hiperpłaszczyzn w przestrzeni *m*-wymiarowej przecina się w jednym punkcie, to punkt ten jest wyznaczony jednoznacznie (wszystkie współczynniki równań hiperpłaszczyzn oraz współrzędne punktów są elementami *GF(q)*).
W metodzie tej *cieniem* jest równanie płaszczyzny *(m-1)*-wymiarowej, zawierającej ten punkt.



Znajomość *i < m* cieni umożliwia jedynie stwierdzenie, że poszukiwany punkt (sekret) należy do określonej hiperpłaszczyzny o wymiarze *(m – i)*.
Dla wartości *m = 2* metoda jest praktycznie równoważna metodzie wielomianu Lagrange'a.
W interpretacji geometrycznej zbiór *cieni* jest wówczas dowolnym skończonym podzbiorem *n* prostych należących do pęku prostych zdefiniowanego przez punkt o współrzędnych *(0, M)* odpowiadający sekretowi.

Metoda Asmutha - Blooma

W celu rozdzielenia sekretu *M* wybiera się dużą liczbę pierwszą $p > M$, a następnie wybiera się *n* liczb wzajemnie względnie pierwszych i uporządkowanych rosnąco (także względnie pierwszych z *p*):

$$d_1, d_2, \ldots, d_n \quad (d_i < d_{i+1} \text{ dla każdego } i = 1, 2, \ldots, n-1)$$

Wartość progową *m* określa zależność :

$$\prod_{i=1}^{m} d_i > p \prod_{i=n-m+2}^{n} d_i$$

Parametry p, d_1, d_2, ... , d_n są parametrami publicznymi.

Cienie **sekretu *M* określa się na podstawie zależności :**

$$m_i = (M + rp) \bmod d_i \,,$$

gdzie *r* jest liczbą losową spełniającą warunek : $\prod_{i=n-m+2}^{n} d_i < rp + M < \prod_{i=1}^{m} d_i$

Z dowolnego zestawu m cieni można odtworzyć sekret M, wykorzystując „chińskie twierdzenie o resztach”, gdyż znajdując jedyne rozwiązanie x układu równań:

$$\begin{cases} x \equiv m_{i1} \mod d_{i1} \\ \dots\dots\dots\dots\dots\dots\dots \\ x \equiv m_{im} \mod d_{im} \end{cases}$$

mamy:

$$x \bmod p = (M + rp) \bmod p = M.$$

Przykład:

Niech wartość sekretu $M = 11$, ilość cieni $n = 5$, zaś wartość progowa $m = 3$.

Wybierzmy odpowiednio:

$$p = 13,\ d_1 = 17,\ d_2 = 19,\ d_3 = 21,\ d_4 = 22,\ d_5 = 23.$$

$d_1\, d_2\, d_3 = 6783$ $d_4\, d_5 = 506$ $p\, d_4\, d_5 = 6578 < 6783$

$r = 510$ $M + rp = 6641$

Cienie sekretu:

$$m_1 = 6641 \bmod 17 = 11$$
$$m_2 = 6641 \bmod 19 = 10$$
$$m_3 = 6641 \bmod 21 = 5$$
$$m_4 = 6641 \bmod 22 = 19$$
$$m_5 = 6641 \bmod 23 = 17$$

Załóżmy, że zgromadzono cienie $m_1 = 11$, $m_2 = 10$ **i** $m_5 = 17$.

Rozwiązaniem układu równań:

$$\begin{cases} x \equiv 11 \bmod 17 \\ x \equiv 10 \bmod 19 \\ x \equiv 17 \bmod 23 \end{cases}$$

jest $x = 6641$.

Odtworzenie sekretu: $M = 6641 \bmod 13 = 11$.

Metoda KGH (E.Karnin, J.Greene, M.Hellman)

W metodzie tej sekret jest przedstawiany w postaci iloczynu macierzowego :

$$M = U\ V_o$$

gdzie :

U **jest wektorem wierszowym (*dim U = m + 1*);**

V_o , V_1 , ... , V_n to wektory o wymiarze *m*, ale wybrane tak, by każda możliwa macierz utworzona z tych wektorów miała wymiar *m* x *m*.

Cieniami **w tej metodzie są iloczyny macierzowe $m_i = U\ V_i$ (dla każdego *i = 1,..., n*).**
Odtwarzanie sekretu *M* polega na rozwiązaniu systemu *m* x *m* równań liniowych, w których niewiadomymi są współczynniki wektora *U*, a następnie wyznaczeniu sekretu na podstawie znajomości wektora V_o.
Do konstrukcji systemu *KGH* wykorzystuje się dowolne ciało skończone *GF(q)*, zaś liczba udziałowców $n \leq q-1$, dzięki czemu metoda jest uważana za bardziej efektywną, niż „klasyczna" metoda Shamira.

(Szczegóły np.: `https://clever-geek.github.io/articles/2641917/index.html`)

Metoda KGH (E.Karnin, J.Greene, M.Hellman)

W metodzie tej sekret jest przedstawiany w postaci iloczynu macierzowego :

$$M = U\,V_o$$

gdzie :

U jes ... 1):

cienie (udziały)

$$\begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_r \\ s_{r+1} \end{bmatrix} = \begin{bmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{t-1} \\ 1 & \alpha_2 & \alpha_2^2 & \cdots & \alpha_2^{t-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha_r & \alpha_r^2 & \cdots & \alpha_r^{t-1} \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} k \\ a_1 \\ a_2 \\ \vdots \\ a_{t-1} \end{bmatrix}$$

sekret

t – progowa liczba cieni

α - generator $GF(q)$

$\alpha_i = \alpha^i$

$$n = r + 1, r \le q^m - 1$$

V_o , V_1 , ... , V_n to wekt ... każda możliwa macierz utw ... m x m.

Cieniami **w tej metod ... dla każdego $i = 1,\dots, n$).**
Odtwarzanie sekretu ... m równań liniowych, w których niewiadom ... stępnie wyznaczeniu sekretu na podstawie

Do konstrukcji systemu *KGH* wykorzystuje się dowolne ciało skończone *GF(q)*, zaś liczba udziałowców $n \le q-1$, dzięki czemu metoda jest uważana za bardziej efektywną, niż „klasyczna" metoda Shamira.

(Szczegóły np.: `https://clever-geek.github.io/articles/2641917/index.html`)

Metody progowe „w sytuacji konkurujących stronnictw”

Niech w systemie podziału sekretu uczestniczy K grup uczestników, liczących po n_k członków każda ($k = 1, 2, \ldots, K$). Grupy te mają współdzielić pewien sekret M, zaś kryterium odtworzenia sekretu niech będzie brzmiało następująco :

„Do odtworzenia sekretu niezbędna jest obecność co najmniej m_k spośród n_k członków z każdej grupy k”.

Sposób rozwiązania tego problemu zostanie pokazany na przykładzie metody wielomianu Lagrange’a, lecz podobne podejście można zastosować w przypadku dowolnej innej metody progowej.

Dokonuje się rozkładu sekretu M na K czynników (dowolnych) :

$$M = M_1 * M_2 * \ldots * M_K$$

(lub w sposób „klasyczny”, wykorzystując $K-1$ ciągów losowych, przedstawia się M w postaci:

$$M = M_1 \oplus M_2 \oplus \ldots \oplus M_K).$$

Dla k-tej grupy określa się wielomian stopnia $m_k - 1$ o losowych współczynnikach $a_{i,k}$:

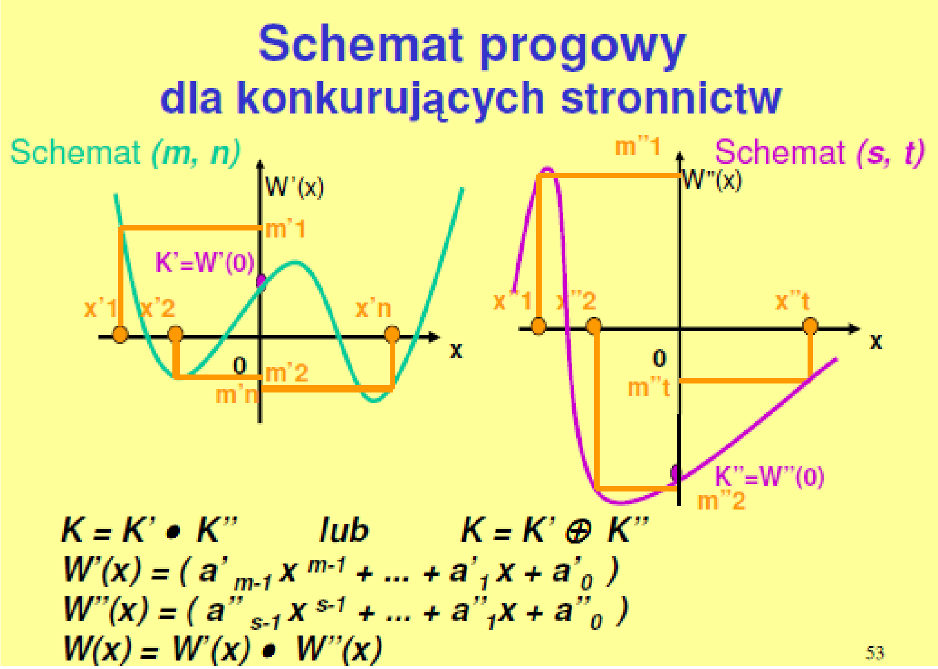
$$W_k(x) = (a_{m_k-1,k}\, x^{m_k-1} + \ldots + a_{1,k}\, x + a_{0,k}) \bmod p_k \,,$$

gdzie p_k jest liczbą pierwszą większą od M_k, zaś $a_{0,k} = M_k$ jest jednym z czynników/składników „ukrywanego” metodą progową sekretu.

Cienie **wiadomości M_k określa się z zależności :**

$$m_{i,k} = W_k(x_i) \bmod p.$$

Każda grupa jest w stanie odtworzyć jedynie „swój” czynnik sekretu M_k, wykorzystując m_k swych cieni, lecz nie jest w stanie odtworzyć całego sekretu.
Do jego odtworzenia niezbędna jest odpowiednia liczba cieni z każdej grupy.



Proaktywne współdzielenie sekretów (na przykładzie wielomianów Shamira-Lagrange'a)

Dany jest wielomian „pierwotny”, służący do wytworzenia cieni m_i sekretu M:

$$W(x) = (a_{m-1}\, x^{m-1} + \ldots + a_1 x + a_0) \bmod p.$$

Cienie (udziały) określone są na podstawie arbitralnie wybranych $x_i \neq 0$:

$$m_i = W(x_i) \bmod p \qquad\qquad M = W(0)$$

Dodanie do wielomianu $W(x)$ losowego wielomianu $d(x)$ takiego, że:

$$d(0) = 0 \bmod p \quad \text{i} \quad \deg(d(x)) \leq \deg(W(x)),$$

tworzy *odświeżony* wielomian $W'(x) = W(x) + d(x)$ chroniący ten sam sekret, gdyż:

$$W'(0) = [W(0) + d(0)] \bmod p = (M + 0) \bmod p = M.$$

Nowe, odświeżone wartości cieni: $m'_i = W'(x_i) \bmod p$.

Załóżmy, że cienie są rozdane między n serwerów zdolnych do samodzielnego wygenerowania wielomianów (oczywiście muszą one znać wartości x_i służące do utworzenia pierwotnych cieni m_i).

Każdy z serwerów niezależnie generuje swój własny wielomian odświeżający $d_k(x)$ i wyznacza na jego podstawie tzw. „*fragmenty odnowy sekretu*”:

$$s_{k\,i} = d_k(x_i) \bmod p \qquad d_k(0) = 0 \bmod p$$

i wysyła je do pozostałych serwerów (indywidualnie i zabezpieczonymi kanałami !) .

Następnie każdy z serwerów odświeża swój własny cień sekretu:

$$m'_k = m_k + \sum_{j=1}^{n} s_{j\,k}$$

Postępowanie to jest prawidłowe, gdyż odświeżony wielomian:

$$W'(x) = W(x) + \sum_{j=1}^{n} d_j(x) \pmod p$$

nadal spełnia warunek: $W'(0) = M \bmod p = M$

Warunkowo bezpieczny podział sekretów
(oparty na wielomianie Shamira-Lagrange'a)

Arbiter dokonujący podziału sekretu w schemacie progowym (t,n) wybiera losowy wielomian stopnia $(t - 1)$ o współczynnikach z $\mathrm{GF}(q)$:

$$f(x) = a_0 + a_1 x + \ldots + a_{t-1} x^{t-1},$$

gdzie $q = 2^l$, zaś $q\text{-}1 = p > 3$ jest „szczególną" liczbą pierwszą (tzw. ***liczbą pierwszą Mersenne'a***).

Następnie ***kanałem bezpiecznym*** dystrybuuje tzw. ***udziały trwałe*** podmiotów P_i:

$$s_i = f(x_i) \qquad (x_i \in \mathrm{GF}(q) \text{ – wartości jawne})$$

Arbiter wybiera ponadto losowo jeden z generatorów grupy cyklicznej $g \in \mathrm{GF}(q)$ i rozsyła go do podmiotów P_i ***kanałem jawnym (publicznym)***.

Współdzielonym sekretem jest wartość $k = g^{f(0)}$, zaś każdy z podmiotów P_i oblicza swą wartość ***udziału przejściowego***:

$$c_i = g^{s_i} = g^{f(x_i)}.$$

W schemacie podziału sekretu wykorzystana jest funkcja:

$$F(x) = g^{f(x)} = g^{a_0}(g^{a_1})^x \dots (g^{a_{t-1}})^{x^{t-1}} = g_0 g_1^x \dots g_{t-1}^{x^{t-1}},$$

gdzie $g_i = g^{a_i}$ dla $i = 1, 2, \dots, (t\text{-}1)$, zaś $k = F(0)$.

Odtworzenie sekretu wymaga zgromadzenia *t* *udziałów przejściowych*:

$$c_{i_1} = g^{f(x_{i_1})}, \dots, c_{i_t} = g^{f(x_{i_t})}.$$

Sekret $k = g^{f(0)} = \prod_{j=1}^{t} (c_{i_j})^{b_j}$,

gdzie $c_i = g^{s_i}$, zaś $b_j = \prod_{\substack{1 \le l \le t \\ l \ne j}} \frac{x_{i_l}}{x_{i_l} - x_{i_j}} \mathit{mod}\ p$.

Udziały trwałe $s_i = f(x_i)$ *nie są nigdy ujawniane* (zakłada się trudność wyznaczenia logarytmów dyskretnych).
Załóżmy,że część *udziałów przejściowych* została ujawniona (skompromitowana), ale nie ujawniono *udziałów trwałych*.

Po uzyskaniu takiej informacji ***arbiter*** unieważnia wszystkie dotychczasowe udziały przejściowe, a następnie rozsyła podmiotom P_i ***kanałem publicznym*** (ale w sposób zapewniający ***integralność*** i ***autentyczność***) nową wartość generatora g' grupy cyklicznej ciała $\mathrm{GF}(q)$.

Każdy z podmiotów P_i oblicza swą nową wartość ***udziału przejściowego***:

$$c'_i = g'^{s_i} = g'^{f(x_i)}.$$

Chcąc uniemożliwić podmiotom P_i samodzielne odtworzenie sekretu, podmiot do tego uprawniony może także ustalić nową wartość generatora jako „tajną" potęgę poprzedniego generatora $g' = g^R$, gdzie R jest zachowywaną w tajemnicy liczbą losową z przedziału $(1, q\text{-}1)$.

Odtwarzanie sekretu wymaga wówczas najpierw obliczenia $k' = g'^{f(0)} \neq k$, a następnie:

$$k = g^{f(0)} = g^{Rf(0)R^{-1}}.$$

W poprzednich przykładach sekret jest odtwarzany przez podmiot „gromadzący” odpowiednią liczbę cieni/udziałów w celu wykonania finalnej operacji kryptograficznej za pomocą tego sekretu (szyfrowanie symetryczne, podpis kluczem prywatnym, itp.).
Inne podejście zakłada, że „udziałowcy” są nie tylko „kolekcjonerami” cieni sekretu, ale wykonują także „cząstkowe” operacje kryptograficzne, a ich rezultaty przesyłają podmiotowi odtwarzającemu na podstawie cząstkowych rezultatów „właściwą” operację kryptograficzną.

Przykład: Deszyfrowanie progowe (Desmedt, Frankel - 1991)

INICJALIZACJA SCHEMATU

- **„Dealer” ustanawia system RSA z publicznym modułem $N=pq$, publicznym wykładnikiem K i prywatnym wykładnikiem k;**

- **„Dealer” tworzy schemat progowy (t,n) z wielomianem Shamira-Lagrange’a $f(x)$ stopnia (t-1) nad $Z_{\lambda(N)}$, gdzie:**
  $\lambda(N) = lcm(p-1,q-1) = 2p'q'$ (p i q – silne liczby pierwsze);
  współrzędne x_i, spełniające rolę „identyfikatorów” udziałowców P_i, są nieparzyste i publiczne ($\mathcal{P}$ – zbiór wszystkich n udziałowców);
  udziały (cienie) $s_i = f(x_i)\alpha_i^{-1} \bmod \lambda(N)$ są parzyste (i tajne), gdzie:

$$\alpha_i = \prod_{P_j \in \mathcal{P}:\, j \neq i} (x_i - x_j)$$

  zaś sekret $f(-1) = k$-1.

SZYFROWANIE

- **Podmiot szyfrujący po pozyskaniu klucza publicznego z wiarygodnego źródła szyfruje wiadomość jawną $m \in Z_N$, tworząc kryptogram:**

$$c = m^{K} \bmod N.$$

DESZYFROWANIE

- **Każdy z udziałowców $P_i \in B$; $|B| = t$, oblicza swój kryptogram cząstkowy:**

$$c_i = c^{s_i} \bmod N;$$

- **Odtwarzający wiadomość jawną łączy kryptogramy cząstkowe z kryptogramem właściwym:**

$$\hat{c}_i \equiv c_i^{\prod_{P_j \in P-B:\, j \neq i} (x_i - x_j) \prod_{P_j \in B:\, j \neq i} (-1 - x_j)} \bmod N$$

$$\prod_{P_i \in B} \hat{c}_i \bullet c = c^{f(-1)+1} = c^{k} \equiv m \pmod N$$

Inne rodzaje obliczeń grupowych

Metody progowe są jednym z przykładów obliczeń grupowych, lecz ich cechą charakterystyczną jest niezbędny współudział progowej liczby uczestników w realizacji operacji kryptograficznej.

Do obliczeń grupowych wykorzystujących mechanizmy kryptograficzne należą także różne propozycje systemów anonimowego głosowania, czy kryptograficzne protokoły gier hazardowych (np. poker przez telefon).

Niemniej jednak najczęściej kryptograficzne obliczenia grupowe rozumiane są jako systemy kryptografii asymetrycznej, w których finalnej operacji kryptograficznej „w imieniu grupy” dokonuje niezależnie jeden z członków tej grupy bez konieczności współudziału pozostałych członków grupy, przy czym z reguły nie ujawnia on swojej tożsamości (pozostaje anonimowy, jako członek „identyfikowalnej” grupy, choć każdy „przyzwoity” system uwzględnia rolę „arbitra”, który w określonej sytuacji tą tożsamość może wskazać).

Podpisy grupowe

Schemat podpisu grupowego powinien mieć następujące własności:

- **podpisy mogą składać wyłącznie członkowie grupy;**
- **weryfikator podpisu może stwierdzić jego ważność (jako podpisu złożonego przez członka określonej grupy), lecz nie może ustalić tożsamości członka tej grupy, który złożył zweryfikowany ważny podpis;**
- **w przypadku sporu musi istnieć możliwość ujawnienia tożsamości podmiotu, który złożył podpis, przy czym w procesie ujawniania, w zależności od wariantu schematu, może być konieczne współdziałanie innych członków uprawnionej grupy, bądź też nie.**

W zależności od tego, czy skład grupy jest ustalany raz podczas inicjacji systemu, czy też możliwe jest dołączanie nowych członków do grupy, bądź wykluczanie z grupy określonych członków, grupy takie określane są jako statyczne albo dynamiczne (co wpływa na mechanizmy stosowane w systemie).

Schemat podpisów grupowych dla grup statycznych (M. Bellare, D. Micciancio, B. Warinschi - 2003)

Schemat obejmuje cztery podstawowe algorytmy:

- **randomizowany algorytm generowania kluczy podpisujących dla podmiotów tworzących grupę (GKg);**
- **randomizowany algorytm składania podpisu przez dowolnego członka grupy (GSig);**
- **deterministyczny algorytm weryfikacji podpisu grupowego (GVf);**
- **deterministyczny algorytm „otwarcia”, czyli ujawnienia tożsamości podmiotu składającego podpis (Open).**

GKg - **Algorytm (randomizowany) generowania kluczy podpisujących i weryfikującego, oraz klucza prywatnego „menedżera grupy”.**

Wejście:

- **parametr bezpieczeństwa** 1^k;
- **rozmiar grupy** n **(liczba podmiotów tworzących grupę).**

Wyjście:

- **klucz publiczny grupy** *gpk*;
- **klucz prywatny „menedżera grupy”** *gmsk*;
- **wektor kluczy prywatnych/podpisujących podmiotów tworzących grupę** *gsk*[ ].

GSig - **Algorytm (randomizowany) składania podpisu przez *i*-tego członka grupy**

Wejście:

- **wiadomość podpisywana *m*;**
- **klucz podpisujący *i*-tego członka grupy *gsk*[*i*].**

Wyjście:

- **podpis *s* = GSig(*gsk*[*i*], *m*).**

GVf - **Algorytm (deterministyczny) weryfikacji podpisu grupowego**

Wejście:

- **klucz publiczny grupy *gpk*;**
- **wiadomość podpisana *m*;**
- **weryfikowany „kandydat” na podpis σ;**

Wyjście:

- **PRAWDA/FAŁSZ = GVf(*gpk*, *m*, σ).**

Open - Algorytm (deterministyczny) ujawniania „tożsamości" podmiotu podpisującego

Wejście:

- **klucz prywatny „menedżera grupy"** ***gmsk*****;**
- **wiadomość podpisana** ***m*****;**
- **weryfikowany „kandydat" na podpis σ;**

Wyjście:

- **„tożsamość" podmiotu podpisującego** ***i*****/informacja o błędzie = Open (*gmsk, m,* σ).**

Istotnym wymaganiem związanym z bezpieczeństwem schematu jest uniemożliwienie „menedżerowi grupy" (ani pozostałym członkom grupy) tworzenia „fałszywych" podpisów w imieniu innego członka grupy.
Autorzy zaproponowali uzupełnienie schematu o „klasyczne" schematy podpisu cyfrowego i szyfrowania asymetrycznego (w sensie realizacji usługi poufności) oraz protokoły wiedzy zerowej, a także rozdzielenie roli „menedżera grupy" (pełniącego w tym schemacie rolę podmiotu ujawniającego tożsamość podpisującego) od roli podmiotu odpowiedzialnego za generowanie kluczy (z jednoczesnym generowaniem certyfikatów kluczy publicznych odpowiadających kluczom podpisującym członków grupy).

Koniec części 3