Deljenje skrivnosti

↓↓↓
Damjan Strnad

→ **V življenju se pogosto zgodi, da je potrebno zaupno informacijo, imenujmo jo skrivnost, deliti med več oseb na tak način, da vsak posameznik poseduje le del skrivnosti. Pri tem zahtevamo, da posamezni del skrivnosti ne zadošča za določitev celotne skrivnosti, pač pa je potrebno za njeno rekonstrukcijo zbrati vsaj določeno število delov, ne pa nujno vseh.**

Na slednji način lahko tudi zagotovimo, da skrivnost ne bo nedosegljiva ali celo izgubljena, če bo pogrešan katerikoli posamezni del. Praktičen primer potrebe po deljenju skrivnosti je npr. delitev varnostne kode za uporabo jedrskega orožja med skupino pooblaščenih oseb, od katerih jih mora svoj del kode prispevati vsaj polovica, da se varnostna koda lahko sestavi in orožje uporabi. Podobna primera uporabe sta delitev kombinacije trezorja ali gesla za dešifriranje zaupnih dokumentov. V določenih praktičnih primerih skrivnosti ne pozna nihče (npr. šifrirni ključ se naključno tvori med samim postopkom delitve), v drugih pa je lahko vsebina skrivnosti znana vsem pooblaščenim osebam in gre pri njenem deljenju samo za zaščito pred nepooblaščeno osebo, ki mora pridobiti vsaj k delov skrivnosti za njeno rekonstrukcijo. Tretja možnost je uporaba zaupne osebe, imenovane **delivec**, ki izvede deljenje skrivnosti in posreduje dele pooblaščenim osebam.

V tem prispevku bomo opisali Shamirjev algoritem, ki je relativno preprosta, a učinkovita in dokaj varna metoda deljenja skrivnosti. Predpostavili bomo, da je skrivnost S predstavljena kot pozitivno celo število. V primeru, da je izvorna skrivnost besedilo, ga je potrebno najprej pretvoriti v številsko obliko. Daljša besedila je pri tem potrebno razdeliti na krajše odseke, ki jih zatem obravnavamo kot ločene skrivnosti.

Naj bo n število delov, na katere želimo skrivnost S razdeliti, k pa minimalno število delov, ki jih potrebujemo za rekonstrukcijo S. Takšni obliki delitve skrivnosti bomo rekli **shema** (k, n). Vrednosti k in n sta javno znani in odvisni od praktičnih potreb. Če želimo, da so vsi udeleženci enako pomembni, potem bomo vsaki od pooblaščenih oseb dodelili natanko en del. Lahko pa določenim pooblaščencem priredimo višjo prioriteto s tem, da jim dodelimo večje število delov skrivnosti od ostalih. Kombinacijo bančnega trezorja bi lahko, recimo, delili po shemi $(3, 4)$, nato pa predsedniku banke dodelili dva dela skrivnosti, vsakemu od njegovih dveh pomočnikov pa po enega. Za odprtje trezorja bi potem zadostovala prisotnost predsednika in kateregakoli od pomočnikov, medtem ko niti predsednik sam niti oba pomočnika skupaj ne bi imeli dostopa do trezorja.

V nadaljevanju bomo opisali poenostavljeno različico originalne Shamirjeve metode, ki ima določene pomanjkljivosti, a uporablja samo običajno aritmetiko in je zato enostavnejša za razumevanje. Metoda za razdelitev skrivnosti uporabi polinom stopnje $k - 1$, ki ga lahko zapišemo kot $p(x) = a_{k-1}x^{k-1} + \ldots + a_2x^2 + a_1x + a_0$. Koeficienti $a_1, \ldots, a_{k-1}$ so skrita naravna števila, ki jih naključno izbere algoritem deljenja skrivnosti, medtem ko vrednost konstantnega člena a_0 postavimo na S. Shamirjeva metoda temelji na matematičnem dejstvu, da za enolično določitev polinoma stopnje $k - 1$ potrebujemo vsaj k njegovih točk – za določitev premice potrebujemo dve točki, za določitev parabole tri točke in tako naprej. Če kot dele skrivnosti izberemo točke $d_i = (x_i, p(x_i))$, kjer je $x_i = i$ za $i \in \{1, 2, \ldots, n\}$, potem bo potrebnih vsaj k ali več delov za enolično določitev neznanih koeficientov polinoma in s tem izračun skrivnosti.

Opisani postopek deljenja skrivnosti lahko strnemo v naslednjem algoritmu:

```
function Deli_Skrivnost(n, k, S)
  a_0 = S
  for i = 1, 2, ..., k − 1 do
    a_i = randint()
  end for
  D = {}
  for i = 1, 2, ..., n do
    p_i = 0
    for j = 0, 1, ..., k − 1 do
      p_i = p_i + a_j · i^j
    end for
    D = D ⋃ {(i, p_i)}
  end for
  return D
end function
```

Klic funkcije `randint` v zgornji kodi vrača naključno celo število. V praksi največjo vrednost koeficienta polinoma omejimo, da ne pride do prekoračitve obsega predstavitve celih števil. Vhodni podatki algoritma so javno znani vrednosti n in k ter vrednost skrivnosti S, rezultat algoritma pa je množica D delov skrivnosti, od katerih je vsak zapisan kot par $(x, p(x))$.

Ostane še vprašanje rekonstrukcije skrivnosti, če je znanih katerihkoli njenih k delov $(x_j, p(x_j))$ za $j \in \{1, 2, \ldots, k\}$. Vrednost prostega člena polinoma, ki predstavlja iskano skrivnost, lahko izračunamo neposredno po naslednji enačbi:

- $$S = p(0) = \sum_{j=1}^{k} p(x_j) \prod_{\substack{u=1 \\ u \neq j}}^{k} \frac{x_u}{x_u - x_j}.$$

Pri naivni implementaciji zgornje enačbe lahko pri računanju ulomkov prihaja do zaokrožitvenih napak, kar lahko omilimo tako, da produkta števcev in imenovalcev izračunavamo ločeno ter deljenje izvedemo šele na koncu.

Oglejmo si sedaj zgled deljenja in rekonstrukcije skrivnosti $S = 9672$ po shemi $(3, 5)$. Ker je $k = 3$, tvorimo naključen polinom druge stopnje. Denimo, da sta naključno izbrana koeficienta $a_1 = 32731$ in $a_2 = 53929$. Skupaj z $a_0 = 9672$ nam to da naslednjo enačbo polinoma:

- $p(x) = 53929x^2 + 32731x + 9672.$

Če enačbo polinoma ovrednotimo pri $x \in \{1, 2, 3, 4, 5\}$, dobimo naslednje dele skrivnosti:

- $d_1 = (1, 96332)$
  $d_2 = (2, 290850)$
  $d_3 = (3, 593226)$
  $d_4 = (4, 1003460)$
  $d_5 = (5, 1521552)$

Poskusimo sedaj rekonstruirati skrivnost iz delov d_1, d_3 in d_4:

- $$S = 96332 \cdot \frac{2}{2-1} \cdot \frac{4}{4-1} +$$
  $$+ 290850 \cdot \frac{1}{1-2} \cdot \frac{4}{4-2} +$$
  $$+ 1003460 \cdot \frac{1}{1-4} \cdot \frac{2}{2-4} =$$
  $$= \frac{770656}{3} - \frac{1163400}{2} + \frac{2006920}{6} = 9672$$

Na podoben način z rekonstrukcijo iz delov d_2, d_3 in d_5 dobimo:

- $$S = 290850 \cdot \frac{3}{3-2} \cdot \frac{5}{5-2} +$$
  $$+ 593226 \cdot \frac{2}{2-3} \cdot \frac{5}{5-3} +$$
  $$+ 1521552 \cdot \frac{2}{2-5} \cdot \frac{3}{3-5} =$$
  $$= \frac{4362750}{3} - \frac{5932260}{2} + \frac{9129312}{6} = 9672$$

Bralec se lahko prepriča, da tudi vsaka druga trojica delov omogoča rekonstrukcijo začetne skrivnosti.

Kot smo na začetku omenili, je opisan postopek v resnici poenostavitev originalne Shamirjeve metode, ki za izračun delov skrivnosti uporablja modularno aritmetiko. Pomanjkljivost opisane metode je v tem, da lahko z vsakim pridobljenim delom skrivnosti dodatno omejimo nabor možnih vrednosti koeficientov polinoma. Z zadostnim številom pridobljenih delov lahko zato nepridipravu uspe zalogo vrednosti koeficientov skrčiti do te meje, da lahko skrivnost izračuna z grobo metodo, tj. s preizkušanjem vseh

možnih kombinacij. Dobra novica je, da lahko varnost metode povečamo s preprosto razširitvijo, pri kateri izberemo veliko praštevilo m, za katerega velja $m > S$ in $m > n$. Vrednost m mora biti javno znana. Naključne vrednosti koeficientov polinoma potem omejimo na $a_i < m$ in dele skrivnosti določimo kot $d_i = (x_i, p(x_i) \textbf{ mod } m)$, kjer **mod** predstavlja ostanek pri celoštevilskem deljenju. Zaradi tega se nekoliko zaplete tudi postopek rekonstrukcije skrivnosti, vendar sedaj nepooblaščena oseba s prilastitvijo dodatnih delov skrivnosti, vse dokler jih skupaj nima vsaj k, ne pridobi dodatne informacije za rekonstrukcijo celotne skrivnosti.

Literatura

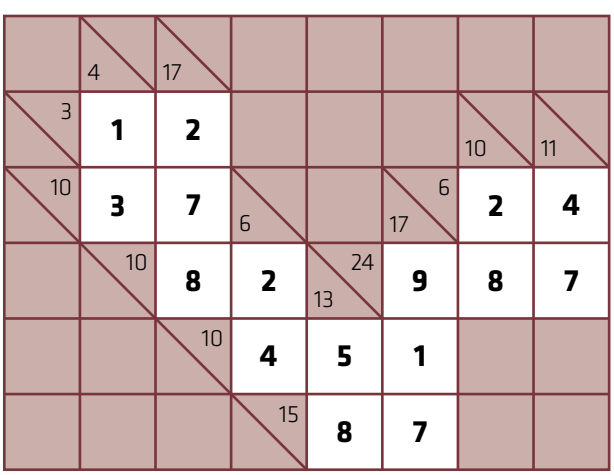
[1] A. Shamir, *How to Share a Secret*, Communications of the ACM, 1979.

× × ×

Križne vsote

REŠITEV S STRANI 8

↓↓↓

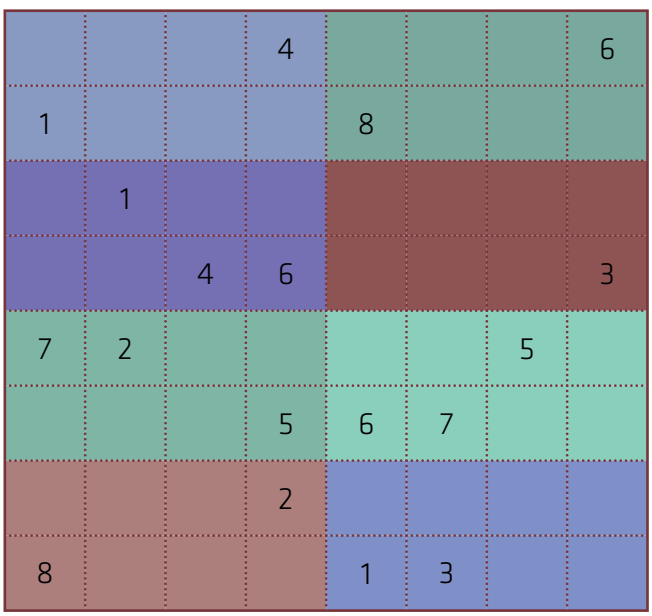


× × ×

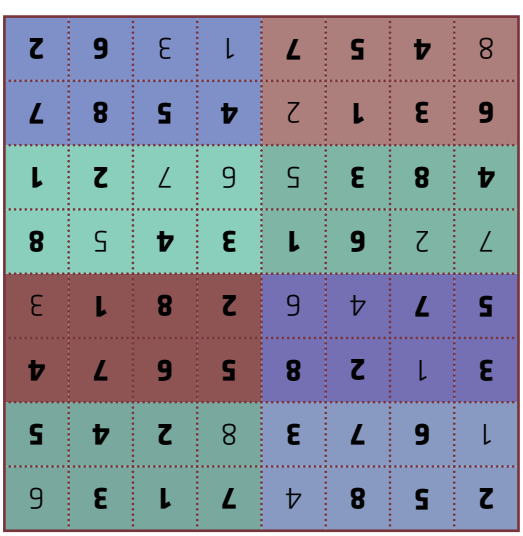
Barvni sudoku

↓↓↓

→ V 8×8 kvadratkov moraš vpisati začetna naravna števila od 1 do 8 tako, da bo v vsaki vrstici, v vsakem stolpcu in v kvadratkih iste barve (pravokotnikih 2×4) nastopalo vseh 8 števil.



→→→ REŠITEV BARVNI SUDOKU



× × ×