A diszkrét logaritmus probléma és a DLP-n alapuló nyilvános kulcsú kriptorendszer

2011. március 21.

A diszkrét logaritmus probléma

Diszkrét logaritmus probléma (DLP)

Legyen p prím és jelölje $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ a p elemű testet, valamint $\mathbb{F}_p^*$ ennek egységcsoportját. Tegyük fel, hogy adott egy α primitív gyök modulo p, azaz $\mathbb{F}_p^*$ egy generátora, valamint egy $\beta \in \mathbb{F}_p^*$ elem. Határozzuk meg azt az egyértelműen létező $e \leq p-2$ pozitív egész számot, melyre

$$\alpha^e \equiv \beta \pmod{p}.$$

- A fenti e számot a számelméletben a β szám α alapú indexének hívjuk modulo p.
- A DLP által vázolt probléma lényegében egy b szám a alapú logaritmusának definíciójához hasonlít, bár megjegyezzük, hogy meghatározása egész más eszközöket igényel, és általában is nehezebb probléma.

Az általánosított diszkrét logaritmus probléma

A fenti gondolatmenet sokkal általánosabban, bármely ciklikus csoportra értelmezhető.

Általánosított diszkrét logaritmus probléma (DLP)

Legyen G egy n rendű ciklikus csoport, α generátora G-nek, valamint egy $\beta \in G$ elem. Határozzuk meg azt az egyértelműen létező $e \leq n-1$ pozitív egész számot, melyre

$$\alpha^e = \beta.$$

A Silver-Pohlig-Hellman algoritmus

Legyen α az $\mathbb{F}_p^*$ egy generátora, $\beta \in \mathbb{F}_p^*$, és tegyük fel, hogy

$$p - 1 = \prod_{i=1}^{r} p_j^{a_j}, \qquad a_j \in \mathbb{N},$$

ahol $p_j,\ j = 1, \ldots, r$ különböző prímek.
Az $e = \log_\alpha \beta$ kiszámításához

- Számítsuk ki $e \pmod{p_j^{a_j}}$ értékét
  - Ehhez határozzuk meg $e \pmod{p_j^{a_j}}$ alakját a p_j alapú számrendszerben

$$e = \sum_{i=0}^{a_j - 1} b_i^{(j)} p_j^i, \qquad 0 \le b_i^{(j)} \le p_j - 1, \text{ ahol } 0 \le i \le a_j - 1.$$

  - A $b_i^{(j)}$ meghatározását ld. a következő fólián.
- Alkalmazzuk a kínai maradék tételt

A Silver-Pohlig-Hellman algoritmus – a $b_0^{(j)}$ értékek meghatározása

- $\beta_0^{\frac{p-1}{p_j}} \equiv \alpha^{\frac{p-1}{p_j} b_0^{(j)}} \pmod{p}$

Valóban, mivel $\beta_0 = \beta$, $\alpha^{p-1} \equiv 1 \pmod{p}$, és $p_j \mid e - b_0^{(j)}$

$$\beta^{\frac{p-1}{p_j}} \equiv \alpha^{\frac{p-1}{p_j} e} \equiv \alpha^{\frac{p-1}{p_j} b_0^{(j)}} \cdot \alpha^{\frac{p-1}{p_j}(e - b_0^{(j)})} \equiv \alpha^{\frac{p-1}{p_j} b_0^{(j)}} \pmod{p}$$

- Kiszámítjuk $\alpha^{\frac{(p-1)k}{p_j}} \pmod{p}$ értékét $k = 1, 2, \ldots, p_j - 1$, addig míg végül

$$\alpha^{\frac{(p-1)k}{p_j}} \equiv \beta_0^{\frac{p-1}{p_j}} \pmod{p}$$

teljesül. Ez a k szükségképpen $b_0^{(j)}$ lesz.

A Silver-Pohlig-Hellman algoritmus – a $b_i^{(j)}$ meghatározása

- Tegyük fel, hogy $b_0^{(j)}, \ldots, b_{i-1}^{(j)}$ már ismert. Legyen

  $\beta_i := \beta \cdot \alpha^{-\sum_{k=0}^{i-1} b_k^{(j)} p_j^k}$ és $x_i := \sum_{k=i}^{a_i-1} b_k^{(j)} p_j^k$

- Ekkor $\beta_i^{\frac{p-1}{p_j^{i+1}}} \equiv \alpha^{\frac{p-1}{p_j} b_i^{(j)}} \pmod{p}$

Valóban, mivel $\beta_i = \alpha^{x_i}$, $\alpha^{p-1} \equiv 1 \pmod{p}$, és $p_j^{i+1} \mid x_i - b_i^{(j)} p_j^k$

$$\beta_i^{\frac{p-1}{p_j^{i+1}}} \equiv \alpha^{\frac{p-1}{p_j^{i+1}} x_i} \equiv \alpha^{\frac{p-1}{p_j^{i+1}} b_i^{(j)} p_j^k} \cdot \alpha^{\frac{p-1}{p_j^{i+1}} (x_i - b_i^{(j)} p_j^k)} \equiv \alpha^{\frac{p-1}{p_j} b_i^{(j)}} \pmod{p}$$

- Kiszámítjuk $\alpha^{\frac{(p-1)k}{p_j}} \pmod{p_j}$ értékét $k = 1, 2, \ldots, p_j - 1$, míg

$$\alpha^{\frac{(p-1)k}{p_j}} \equiv \beta_i^{\frac{p-1}{p_j^{i+1}}} \pmod{p}$$

teljesül. Ez a k szükségképpen $b_0^{(j)}$ lesz.

Megjegyzés.

Ha $n = p - 1$, és adva van $p - 1$ kanonikus alakja, akkor a Silver-Pohlig-Hellman algoritmus futási ideje

$$O\left(\sum_{j=1}^{r} a_j(\ln n + \sqrt{p_j})\right).$$

Ez azt mutatja, hogy a Silver-Pohlig-Hellman algoritmus csak akkor hatékony, ha $p - 1$ prímfaktorai kicsik.

Példa

Legyen $\alpha = 5$, $\beta = 68$ és $p = 73$. Keressük azt az x értéket, melyre $5^x \equiv 68 \pmod{73}$. Ebben a példában minden kongruencia modulo 73 értendő.

- $p - 1 = 72 = 2^3 \cdot 3^2 = p_1^{a_1} p_2^{a_2}$

Példa megoldása: $p_1 = 2$ esetén

k	0	1
$\alpha^{(p-1)k/p_1}$	1	$5^3 6 \equiv 72$

i	0	1	$2 = a_1 - 1$
β_i	68	$68 \cdot 5^{-1} \equiv 72$	$68 \cdot 5^{-1} \equiv 72$
$\beta_i^{(p-1)/p_1^{i+1}}$	$68^{36} \equiv 72$	$72^{18} \equiv 1$	$72^9 \equiv 72$
$b_i^{(1)}$	1	0	1

Így $\log_5 68 \pmod 8$ alakja a 2 alapú számrendszerben:

$$\sum_{i=0}^{a_1-1} b_i^{(1)} p_1^i = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 \equiv 5 \pmod 8.$$

Példa megoldása: $p_2 = 3$ esetén

k	0	1	2
$\alpha^{(p-1)k/p_2}$	1	$5^2 4 \equiv 8$	$5^{24\cdot 2} \equiv 64$

i	0	$1 = a_2 - 1$
β_i	68	$68 \cdot 5^{-1} \equiv 72$
$\beta_i^{(p-1)/p_2^{i+1}}$	$68^{24} \equiv 8$	$72^8 \equiv 1$
$b_i^{(2)}$	1	0

Így $\log_5 68 \pmod 9$ alakja a 3 alapú számrendszerben:

$$\sum_{i=0}^{a_2-1} b_i^{(2)} p_2^i = 1 \cdot 3^0 + 0 \cdot 3^1 \equiv 1 \pmod 9.$$

Példa megoldása:

Kínai maradék tétellel megoldjuk a

$$\log_5 68 \equiv 5 \pmod 8$$
$$\log_5 68 \equiv 1 \pmod 9$$

és kapjuk, hogy

$$\log_5 68 = 37 \quad \mathbb{F}_{73}\text{-ban}.$$

A Baby-Step Giant-Step algoritmus – D. Shanks

Legyen G egy n rendű ciklikus csoport, α generátora a G csoportnak, $\beta \in G$. Keressük azt az egyetlen $x \in \mathbb{N}$ értéket, melyre $x < n$ és $\alpha^x = \beta$.

1. Legyen $s := [\sqrt{n}]$
2. **Baby-Step:** Számítsuk ki a $(j, \alpha^j \beta)$ párokat $j = 0, 1, \ldots, s-1$ esetén, és rendezzük a listát a második koordináta szerint.
3. **Giant-Step:** Számítsuk ki a (α^{is}, i) párokat $i = 1, 2, \ldots, s+1$ esetén, és rendezzük a listát az első koordináta szerint.
4. **Keresés és összehasonlítás:** A 2, 3 pontokban létrehozott listákban keresve, találunk olyan $\alpha^j \beta$ elemet a 2, és α^{is} elemet a 3 pontból, hogy $\alpha^j \beta = \alpha^{is}$. Ekkor $x \equiv is - j \pmod{n}$.

Példa

Legyen $\alpha = 5$, $\beta = 71$ és $p = 167$ (n=166). Keressük azt az x értéket, melyre $5^x \equiv 71 \pmod{167}$, és $x < 166$.

- $s = [166] = 12$.
- A Baby-Step: $(j, 5^j \cdot 71 \pmod{167})$, ahol $j = 0, 1, \ldots 12$

j	8	1	3	0	9	10	5	2	4	11	7	6
$5^j 71$	17	21	24	71	85	91	99	105	120	121	137	161

- A Giant-Step: (α^{is}, i), ahol $i = 1, 2, \ldots, s$

5^{12i}	8	24	44	47	54	56	58	75	130	132	141	152	162
i	10	4	9	11	13	6	2	8	12	3	5	1	7

- A két táblázatban közös a 24, azaz $\alpha^3\beta \equiv \alpha^{4 \cdot 12} \pmod{167}$. Tehát $x = 4 \cdot 12 - 3 = 45$.

Az index kalkulus algoritmus – Előzetes számítások

Legyen p egy prím, α az $\mathbb{F}_p^*$ egy generátora, $\beta \in \mathbb{F}_p^*$. Keressük azt az $x \in \mathbb{N}$ számot, amelyre $x < p-1$ és $\alpha^x \equiv \beta \pmod{p}$.

1. Válasszunk egy $\mathcal{B} = \{p_1, \ldots, p_B\}$ faktorbázist, ami az első B prímből áll, és B úgy lett megválasztva, hogy $\mathbb{F}_p^*$ elemeinek egy "jelentős része" felírható $\mathcal{B}$-beli elemek hatványainak szorzataként.
2. Sorra választunk $k < p-1$ véletlen számokat, és kiszámítjuk $\alpha^k \pmod{p}$ értékét, majd ezt megpróbáljuk felírni a faktorbázisban $\prod_{j=1}^{B} p_j^{k_j}$ alakban. Minden sikeres felírás egy

$$k \equiv \sum_{j=1}^{B} k_j \log_\alpha p_j \pmod{p-1} \tag{1}$$

alakú relációt ad. Addig folytatjuk k választását, amíg legalább B darab (1) relációnk lesz.

Az index kalkulus algoritmus – A logaritmus kiszámítása

3. Megoldva az (1) relációkból álló moduláris egyenletrendszert, meghatározzuk a $\log_\alpha p_j$, $j = 1, \ldots, B$ értékeket.
4. Válasszunk egy véletlen $t < p - 1$ számot, és számítsuk ki $\beta\alpha^t \pmod p$ értékét.
5. Próbáljuk $\beta\alpha^t \pmod p$ értékét felírni a faktorbázisban

$$\beta\alpha^t \equiv \prod_{j=1}^{B} p_j^{t_j} \pmod p \quad (t_j \geq 0) \tag{2}$$

Ha nem sikerül így felírni, akkor a 4 pontban, új t számot választunk, ha sikerül, akkor

$$\log_\alpha \beta + t \equiv \sum_{j=1}^{B} t_j \log_\alpha p_j \pmod{p-1}$$

megadja $x = \log_\alpha \beta$ értékét.

Példa - Előzetes számítás

Legyen $p = 3361$, $\alpha = 22$, $\beta = 4$ és $\mathcal{B} = \{2, 3, 5, 7\}$. Számítsuk ki $\log_{22} 4$ értékét $\mathbb{F}^*_{3361}$-ban.

- Véletlenszerűen válasszuk $k = 48, 100, 186, 2986$

$$22^{48} \equiv 2^5 \cdot 3^2 \pmod{3361}, \quad 22^{100} \equiv 2^6 \cdot 7 \pmod{3361},$$
$$22^{186} \equiv 2^9 \cdot 5 \pmod{3361}, \quad 22^{2986} \equiv 2^3 \cdot 3 \cdot 5^2 \pmod{3361}.$$

- Így az alábbi relációkat kapjuk

$$\begin{aligned}
48 &\equiv 5\log_{22} 2 + 2\log_{22} 5 \pmod{3360} \\
100 &\equiv 6\log_{22} 2 + \log_{22} 7 \pmod{3360} \\
186 &\equiv 9\log_{22} 2 + \log_{22} 5 \pmod{3360} \\
2986 &\equiv 3\log_{22} 2 + \log_{22} 3 + 2\log_{22} 5 \pmod{3360}
\end{aligned}$$

Példa – Logaritmus kiszámítása

- Megoldjuk a fenti rendszert:
  $\log_{22} 2 = 1100$, $\log_{22} 3 = 2314$, $\log_{22} 5 = 366$, $\log_{22} 7 = 220$
- Véletlenszerűen a $t = 754$ értéket választjuk, és kiszámítjuk a következőt:

$$\beta\alpha^t = 4 \cdot 22^{754} \equiv 2 \cdot 3^2 \cdot 5 \cdot 7 \pmod{3361},$$

és megkapjuk, hogy

$$\log_{22} 4 + 754 \equiv \log_{22} 2 + 2\log_{22} 3 + \log_{22} 5 + \log_{22} 7 \pmod{3360}$$

azaz $\log_{22} 4 = 2200$.

A Pohlig-Hellman szimmetrikus hatvány-titkosító

1. Választunk egy titkos p prímet és egy titkos $e \in \mathbb{N}$ titkosító kulcsot, melyre $e \leq p-2$.
2. Kiszámítunk egy titkos d visszafejtő kulcsot, melyre $ed \equiv 1 \pmod{p-1}$.
3. Az m nyilvános szöveg-egység titkosítása az alábbi képlettel történik:
$$c \equiv m^e \pmod{p}.$$
4. A visszafejtéshez az $m \equiv c^d \pmod{p}$ összefüggést használjuk.

Mivel e és p ismeretében d kiszámítása könnyű, így p és e egyaránt titkos.
A kriptorendszer biztonsága a DLP megoldásának nehéz voltán múlik.

Az ElGamal kriptorendszer – Kulcsgenerálás

Alice szeretne egy $m \in \{0, 1, \ldots, p-1\}$ üzenetet elküldeni Bobnak.

1. Bob választ egy nagy p véletlen prímet (amely esetén a DLP megoldása $\mathbb{Z}_p$-ben "szinte lehetetlen"), és egy α primitív gyököt modulo p.
2. Bob ezután választ egy véletlen $2 \leq a < p-1$ természetes számot, és kiszámolja $\beta := \alpha^a \pmod{p}$ értékét.
3. Bob nyilvános kulcsa: (p, α, β)
   Bob titkos kulcsa: a

Az ElGamal kriptorendszer

Titkosító fázis

1. Alice megszerzi Bob nyilvános kulcsát: (p, α, β)
2. Alice ezután választ egy véletlen $2 \leq b < p - 1$ természetes számot
3. Alice kiszámolja $\alpha^b \pmod p$ és $m\alpha^{ab} = m\beta^b \pmod p$ értékét
4. Alice elküldi a $c = (\alpha^b, m\alpha^{ab})$ kriptoszöveget Bobnak

Visszafejtő fázis

1. Bob felhasználva saját titkos kulcsát kiszámítja az $(\alpha^b)^{-a} \equiv (\alpha^b)^{p-1-a} \pmod p$ értéket.
2. Bob ezután kiszámítja $(\alpha^b)^{-a} m\alpha^{ab} \equiv m \pmod p$ értéket, azaz visszanyeri m értékét.

z ElGamal kriptorendszer – Megjegyzések

Megjegyzések

- Az ElGamal kriptorendszer egyik fő előnye, hogy nem determinisztikus, azaz a b (Alice által választott) véletlen szám miatt, egyazon nyílt szöveg többszöri elküldése esetén, a kriptoszöveg különböző lesz.
- Látható, hogy az ElGamal kriptorendszer esetén a kriptoszöveg körülbelül kétszer olyan hosszú, mint a nyílt szöveg. Ezt a jelenséget hívják "üzenet expanziónak", és ez az ElGamal kriptorendszer egyik jelentős hátránya.
- Ha az üzenet numerikus kódja $m > p$, akkor az RSA-nál ismertetett módon az eredeti nyílt szöveget blokkokra kell bontani.

Az általánosított ElGamal kriptorendszer – Kulcsgenerálás

Alice és Bob közösen rögzítenek egy G ciklikus csoportot,

- melyben a DLP megoldása "szinte lehetetlen",
- melyben a csoportművelet hatékonyan elvégezhető,
- melynek rendje n kellően nagy, és generátora α.

Alice szeretne egy $m \in G$ üzenetet elküldeni Bobnak.

1. Bob ezután választ egy véletlen $2 \leq a < n$ természetes számot, és kiszámolja $\beta := \alpha^a$ csoportelemet.
2. Bob nyilvános kulcsa: (α, β)
   Bob titkos kulcsa: a

Az általánosított ElGamal kriptorendszer

Titkosító fázis

1. Alice megszerzi Bob nyilvános kulcsát: (α, β)
2. Alice ezután választ egy véletlen $2 \leq b < n$ természetes számot
3. Alice kiszámolja α^b és $m\alpha^{ab} = m\beta^b$ elemeket
4. Alice elküldi a $c = (\alpha^b, m\alpha^{ab})$ kriptoszöveget Bobnak

Visszafejtő fázis

1. Bob felhasználva saját titkos kulcsát kiszámítja az $(\alpha^b)^{-a}$ elemet.
2. Bob ezután kiszámítja $(\alpha^b)^{-a} m\alpha^{ab}$ elemet, azaz visszanyeri m értékét.

Példa

- Alice és Bob rögzíti az alábbiakat: Legyen $G = \mathbb{F}_{5^3}^*$, melynek elemeit $\mathbb{F}_{5^3}^* \cong \mathbb{F}_5[x]/(r(x))$ elemeiként írjuk fel, ahol $r(x) = x^3 + x + 1 \in \mathbb{F}_5[x]$. $\mathbb{F}_{5^3}$ elemeit a megfelelő polinomok együtthatóiból álló számhármassal reprezentáljuk.
- A G csoport generátoraként választhatjuk az $\alpha = 2x^2 + 2$ elemet, amely $\alpha = (2, 0, 2)$ alakot ölt.
- Ha Bob az $a = 9$ titkos kulcsot választja, akkor kiszámítja a $\beta = \alpha^9 = (1, 2, 3)$ elemet, és
  - nyilvános kulcsa: $(\alpha, \alpha^9) = ((2, 0, 2), (1, 2, 3))$
  - titkos kulcsa: 9
- Alice a $(4, 3, 2)$ üzenetet akarja Bobnak elküldeni.
- Alice véletlenszerűen választja a $b = 96$ értéket, és kiszámítja $\alpha^b = \alpha^{96} = (1, 4, 2)$ és $m\alpha^{ab} = (4, 3, 2)(1, 2, 3)^{96} = (4, 4, 2)$.
- Elküldi a $c = (\alpha^b, m\alpha^{ab}) = ((1, 4, 2), (4, 4, 2))$ kriptoszöveget Bobnak.
- Bob kiszámolja $\alpha^{-ab} = (1, 4, 2)^{-9} = (2, 4, 3)$, és végül az $m = \alpha^{-ab} m\alpha^{ab} = (2, 4, 3)(4, 4, 2) = (4, 3, 2)$ nyílt szöveget.

A Massey-Omura kriptorendszer

Legyen p egy prím és $n \in \mathbb{N}$. Tegyük fel, hogy Alice egy $m \in \mathbb{F}_{p^n}^*$ nyílt szöveget akar Bobnak elküldeni. Alice és Bob a következő lépéseket hajtja végre:

1. Alice és Bob egymástól függetlenül választanak egy-egy $2 \leq e_A, e_B < p^n - 1$ egész számot, melyekre $(e_a, p^n - 1) = 1$ és $(e_b, p^n - 1) = 1$.
2. Alice és Bob kiszámolja a $d_A \equiv e_A^{-1} \pmod{p^n - 1}$, illetve a $d_B \equiv e_B^{-1} \pmod{p^n - 1}$ elemeket a kiterjesztett Euklideszi algoritmus segítségével.
3. Alice elküldi az m^{e_A} elemet Bobnak
4. Bob visszaküldi az $m^{e_A e_B}$ elemet Alicenek
5. Alice elküldi az $m^{e_A e_B d_A} = m^{e_B}$ elemet Bobnak
6. Bob kiszámítja az $(m^{e_B})^{d_B} = m$ elemet.

A Massey-Omura kriptorendszer – Megjegyzések

- A Massey-Omura kriptorendszer nem szigorúan vett nyilvános kulcsú kriptorendszer, hiszen nincs se nyilvános kulcs, se megosztott titkos kulcs.
- Az első lépés után Bob nem tudja m^{e_A} segítségével meghatározni az m üzenetet, mert nem ismeri a d_A kitevőt.
- Bob m^{e_A} segítségével (sőt később m ismeretében sem) nem tudja meghatározni az e_A kitevőt, hacsak nem tudja megoldani a DLP-t az $\mathbb{F}_{p^n}^*$ csoportban.
- Alice m^{e_B} és m segítségével nem tudja meghatározni az e_B kitevőt, hacsak nem tudja megoldani a DLP-t az $\mathbb{F}_{p^n}^*$-ben.
- Ettől függetlenül, minden kommunikációhoz új kulcsot használnak.
- A Massey-Omura kriptorendszer egyik hátránya, hogy három üzenetküldésre van szükség
- Másik, ennél súlyosabb probléma, hogy ez a kriptorendszer teljesen védtelen a "man in the middle" támadással szemben

A Massey-Omura kriptorendszer – Példa

- Alice és Bob rögzíti az alábbiakat: Legyen $G = \mathbb{F}_{5^3}^*$, melynek elemeit $\mathbb{F}_{5^3} \cong \mathbb{F}_5[x]/(r(x))$ elemeiként írjuk fel, ahol $r(x) = x^3 + x + 1 \in \mathbb{F}_5[x]$. $\mathbb{F}_{5^3}$ elemeit a megfelelő polinomok együtthatóiból álló számhármassal reprezentáljuk.
- Alice a Massey-Omura kriptorendszer segítségével a $(1, 2, 3)$ üzenetet akarja Bobnak elküldeni.
- Alice az $e_a = 25$, $d_A = 5$, míg tőle függetlenül Bob a $d_B = 3$ és $d_B = 83$ kulcsokat generálta.
- Alice elküldi az $m^{e_A} = (1, 2, 3)^{25} = (3, 3, 1)$ elemet Bobnak
- Bob visszaküldi az $m^{e_A e_B} = (3, 3, 1)^3 = (4, 0, 3)$ elemet Alicenek
- Alice elküldi az $m^{e_A e_B d_A} = m^{e_B} = (4, 0, 3)^5 = (2, 2, 0)$ elemet Bobnak
- Bob kiszámítja az $m = (m^{e_B})^{d_B} = (2, 2, 0)^{83} = (1, 2, 3)$ elemet.