Teorija kodiranja in kriptografija 2013/2014

Digitalni podpis

Arjana Žitnik

Univerza v Ljubljani, Fakulteta za matematiko in fiziko

Ljubljana, 15. 4. 2014

Vsebina

- Digitalni podpis
- Digitalni podpis z RSA
- Varnost digitalnega podpisa
- Digitalni podpis in zgoščevalne funkcije
- DSA

Digitalni podpis

Digitalni podpis je nadomestek za lastnoročni podpis pri elektronski izmenjavi in digitalnemu hranjeju podatkov.

V Sloveniji je to urejeno z *Zakonom o elektronskem poslovanju in elektronskem podpisovanju - ZEPEP*.

Digitalni podpis omogoča dokazati neodvisni tretji stranki

- izvor sporočila ali entitete,
- pristnost (celovitost) sporočila
- preprečevanje tajenja

Uporablja se v mnogih protokolih.

Lastnosti digitalnega podpisa

- Digitalni podpis dokazuje, da je podpisnik zares podpisal dokument.
- Vsebine digitalno podpisanega dokumenta ni mogoče spreminjati.
- Podpisa ni mogoče kopirati in ponarejati.
- Podpisnik kasneje ne more zanikati, da je podpisal dokument.

Algoritem digitalnega podpisa

Algoritem digitalnega podpisa je sestavljen iz treh delov:

1. algoritma generiranja ključa,
2. algoritma generiranja digitalnega podpisa sig_A sporočilu priredi podpis osebe A,
3. algoritma preverjanja digitalnega podpisa ver_A sporočilu in podpisu priredi vrednost “veljaven” oziroma “neveljaven”.

Sistem za digitalno podpisovanje

Formalno je sistem za digitalno podpisovanje peterka $(\mathcal{B}, \mathcal{A}, \mathcal{K}, \mathcal{S}, \mathcal{V})$, za katero velja

- $\mathcal{B}$ je končna množica besedil,
- $\mathcal{A}$ je končna množica podpisov,
- $\mathcal{K}$ je končna množica ključev,
- za vsak ključ $K \in \mathcal{K}$ obstajata algoritem za podpisovanje

$$\mathrm{sig}_K \in \mathcal{S}, \quad \mathrm{sig}_K : \mathcal{B} \to \mathcal{A}$$

in algoritem za preverjanje podpisa

$$\mathrm{ver}_K \in \mathcal{V}, \quad \mathrm{ver}_K : \mathcal{B} \times \mathcal{A} \to \{\mathrm{true}, \mathrm{false}\}.$$

Zahteve

Funkciji sig_K in ver_K imata to lastnost, da za vsako besedilo $x \in \mathcal{B}$ in vsak podpis $y \in \mathcal{A}$ velja

$$\mathrm{ver}_K(x, y) = \begin{cases} \text{true}, & \text{če } y = \mathrm{sig}_K(x) \\ \text{false}, & \text{če } y \neq \mathrm{sig}_K(x) \end{cases}$$

Zahteve:

- algoritma sig_K in ver_K imata polinomsko časovno zahtevnost
- sig_K je znan le podpisniku
- ver_K je splošno znan
- računsko mora biti nemogoče ponarediti podpis

Podpisovanje z algoritmom RSA

Algoritem RSA lahko uporabimo tudi za podpisovanje. Naj bo $n = pq$, kjer sta p in q praštevili.

Naj bo (n, d) zasebni ključ in (n, e) javni ključ.
Potem definiramo:

$$\begin{aligned} \mathrm{sig}_K(x) &= d_K(x) = x^d \pmod{n} \\ \mathrm{ver}_K(x, y) &= \mathrm{true} \iff x = e_K(y) = y^e \pmod{n} \end{aligned}$$

za $x, y \in \mathbb{Z}_n$.

Varnost podpisovanja

Kaj zmore napadalec:

- ima na voljo javni ključ (algoritem za preverjanje podpisa);
- ima na voljo podpisana sporočila $(x_1, y_1), \ldots, (x_t, y_t)$, kjer je $y_i = \mathrm{sig}_K(x_i)$;
- ima za določen čas na voljo algoritem za podpisovanje.

Varnost podpisovanja - cilji

Cilji napadalca so lahko naslednji:

- izračun zasebnega ključa za podpisovanje;
- podpis izbranih sporočil (z nezanemarljivo verjetnostjo);
- generirati veljaven par (sporočilo, podpis),
  torej podpis naključnih sporočil.

Podpis naključnih sporočil

Pri uporabi osnovne variante RSA za podpis je mogoče ponarediti podpis naključnih sporočil.

Ponarejevalec najprej izbere podpis y in nato izračuna

$$x \equiv y^e \pmod{n}.$$

Možnosti takega ponarejanja se izognemo, če

- namesto sporočila podpišemo izvleček sporočila (dobljen z varno zgoščevalno funkcijo);
- zahtevamo, da ima sporočilo x določen pomen.

Multiplikativna lastnost RSA

Podpisi z osnovno varianto RSA niso varni pred napadom z izbranim besedilom.

Ideja: Če sta $s_1 = x_1^d \pmod{n}$ in $s_2 = x_2^d \pmod{n}$ veljavna RSA podpisa, je tudi

$$s_1 \cdot s_2 = x_1^d \cdot x_2^d \pmod{n}$$

veljaven RSA podpis.

Napad: Nasprotnik zahteva podpis sporočil x_1 in x_2.
Potem lahko izračuna podpis sporočila $x_1 \cdot x_2$.

Digitalni podpis in zgoščevalne funkcije

Prejšnjim dvem napadom se lahko izognemo, če namesto sporočila podpišemo izvleček sporočila.

Potrebne lastnosti zgoščevalnih funkcij:

- odpornost 2-praslik (sicer lahko za dani par (besedilo, podpis) poiščemo drugo besedilo z istim podpisom);
- odpornost na trke (sicer lahko napadalec poišče dve sporočili x, x' z istim izvlečkom. Podpisa x in x' sta enaka!)
- odpornost praslik (sicer lahko ponaredimo podpis naključnega sporočila);

Šifriranje in podpisovanje

Kakšen je pravi vrstni red?

- Najprej sporočilo podpišemo (s svojim zasebnim ključem), potem vse skupaj šifriramo (s prejemnikovim javnim ključem).
- Najprej sporočilo šifriramo (s prejemnikovim javnim ključem), potem podpišemo (s svojim zasebnim ključem).

V drugem primeru lahko napadalec Cene zamenja Anitin podpis s svojim in Bojan bo mislil, da je sporočilo prišlo od Ceneta.

Zato se priporoča najprej podpisovanje in nato šifriranje.

Šifriranje in podpisovanje - težave

V primeru algoritma RSA je potrebno pri zaporednem podpisovanju in šifriranju paziti na velikosti modulov (*reblocking problem*).

Naj bo n_A Anitin in n_B Bojanov javni ključ.
Če je $n_A > n_B$, se lahko zgodi, da Bojan ne bo mogel razvozlati sporočila.

Zgled:

Anitin ključ: $(n_A, e_A, d_A) = (62894113, 5, 37726937)$,
Bojanov ključ: $(n_B, e_B, d_B) = (55465219, 5, 44360237)$.

Anita podpiše sporočilo $x = 1368797$ in podpis zašifrira:

1. $s = x^{d_A} \bmod n_A = 59847900$,
2. $y = s^{e_B} \bmod n_B = 38842235$.

Bojan izračuna

1. $\widehat{s} = y^{d_B} \bmod n_B = 4382681$,
2. $\widehat{x} = \widehat{s}^{e_A} \bmod n_A = 54383568$.

Ker je $s > n_B$, je $\widehat{x} \neq x = 1368797$.

Verjetnost tega dogodka je

$$\frac{n_A - n_B}{n_A}.$$

ElGamalov sistem za digitalno podpisovanje

Za razliko od algoritma RSA je ElGamalov sistem namenjen predvsem digitalnemu podpisovanju, čeprav se ga da v posebnih primerih uporabiti tudi za šifriranje.

Podpis je nedeterminističen (odvisen od naključnega števila), torej sploh ni natanko določen.

Generiranje ključa

Naj bo p takšno praštevilo, da je v $\mathbb{Z}_p$ težko izračunati diskretni algoritem in $\alpha \in \mathbb{Z}_p^*$ primitivni element.

Potem je $\mathcal{B} = \mathbb{Z}_p^*$, $\mathcal{A} = \mathbb{Z}_p^* \times \mathbb{Z}_{p-1}$ in

$$\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod{p}\}.$$

Število a je skrito (zasebno),

Števila p, α in β pa so javno znana.

Podpisovanje

Podpisnik s ključem $K = (p, \alpha, a, \beta)$ izbere naključno skrito število $k \in \mathbb{Z}^*_{p-1}$ in določi

$$\mathrm{sig}_K(x, k) = (\gamma, \delta),$$

kjer je

$$\gamma \equiv \alpha^k \pmod{p}$$

in

$$\delta \equiv (x - a\gamma)k^{-1} \pmod{p-1}.$$

Preverjanje podpisa

Za to potrebujemo p, α in β, ki so javni:

$$\mathrm{ver}_K(x, \gamma, \delta) = \mathrm{true} \iff \beta^\gamma \gamma^\delta \equiv \alpha^x \pmod{p}.$$

ElGamalov podpis - zgled

1. Bojan izbere ključ: $p = 467, \alpha = 2$ in $a = 127$ in izračuna $\beta \equiv \alpha^a \bmod p = 132$.
2. Bojan želi podpisati sporočilo $x = 100$.
   Izbere še nakjučni $k = 213$.
   Podpis je enak (γ, δ), kjer je

$$\gamma \equiv 2^{213} \bmod 467 = 29$$

   in

$$\delta \equiv (100 - 127 \cdot 29)\, 431 \bmod 466 = 51.$$

3. Anita preveri podpis. Izračuna

$$132^{29}\, 29^{51} \equiv 189 \pmod{467} \quad \text{in}$$

$$2^{100} \equiv 189 \pmod{467}.$$

   Zadnji vrednosti se ujemata, zato je podpis pravi.

Varnost ElGamalovega sistema za podpisovanje

Kako bi lahko ponaredili podpis, ne da bi vedeli za vrednost skritega števila a?

I. Za dano sporočilo x iščemo podpis, t.j. par (γ, δ), da bo veljalo $\beta^\gamma \gamma^\delta \equiv \alpha^x \pmod{p}$, torej
   - če izberemo γ: rabimo $\delta = \log_\gamma \alpha^x \beta^{-\gamma} \pmod{p}$,
   - če izberemo δ: glede na γ je potrebno rešiti enačbo $\beta^\gamma \gamma^\delta \equiv \alpha^x \pmod{p}$ (zaenkrat hiter postopek za reševanje te enačbe ni znan),
   - hkrati računamo γ in δ.

II. Za izbran podpis (γ, δ) iščemo sporočilo x:

$$x = \log_\alpha \beta^\gamma \gamma^\delta \pmod{p}.$$

III. Hkratno računanje x, γ in δ: naj bosta i in j takšni števili, da velja $0 \leq i, j \leq p-2$ in $D(j, p-1) = 1$. Potem števila

$$\begin{aligned} \gamma &\equiv \alpha^i \beta^j \pmod{p}, \\ \delta &\equiv -\gamma j^{-1} \pmod{p-1}, \\ x &\equiv -\gamma i j^{-1} \pmod{p-1} \end{aligned}$$

zadoščajo enačbi $\beta^\gamma \gamma^\delta \equiv \alpha^x \pmod{p}$.

Zgled: naj bo $p = 467, \alpha = 2$ in $\beta = 132$. Potem z izbiro $i = 99$ in $j = 179$, dobimo veljaven podpis $(117, 41)$ za sporočilo 331.

IV. Pri veljavnem podpisu (γ, δ) za x iščemo podpis za neko drugo sporočilo x'. To se da narediti.

Naj bodo h, i in j takšna števila, da zanje velja $0 \leq h, i, j \leq p-2$ in $D(h\gamma - j\delta, p-1) = 1$.
Potem je par (λ, μ) veljaven podpis za x', kjer je

$$\begin{aligned} \lambda &= \gamma^h \alpha^i \beta^j \pmod{p}, \\ \mu &= \delta\lambda(h\gamma - j\delta)^{-1} \pmod{p-1}, \\ x' &= \lambda(hx + i\delta)(h\gamma - j\delta)^{-1} \pmod{p-1}. \end{aligned}$$

Ti napadi zaenkrat ne predstavljajo resnične nevarnosti, ker še vedno ne znamo podpisati izbranega sporočila.

Nevarnosti pri napačni uporabi ElGamalovega sistema

1. Če naključno število k ne ostane skrito in $\gcd(\gamma, p-1) = 1$, lahko iz $\delta = (x - a\gamma) \cdot k^{-1}$ izračunamo

$$a = (x - k\delta)\gamma^{-1} \pmod{p-1}.$$

2. Število k lahko uporabimo le enkrat, sicer ga je mogoče zlahka izračunati.

Digital Signature Standard

Digital signature algorithm (DSA) je izboljšava ElGamalovega sistema za podpisovanje.

S strani NIST je bil predlagan za ameriški standard leta 1991, sprejet pa leta 1993. (FIPS 186-1, FIPS 186-2).

Algoritem za digitalni podpis (DSA)

Generiranje ključa:

1. Izberi 160-bitno praštevilo q.
2. Izberi 1024-bitno praštevilo p, da $q|p-1$.
3. Izberi element $h \in \mathbb{Z}_p^*$ in izračunaj $\alpha = h^{(p-1)/q}$ mod p; ponavljaj, dokler $\alpha \neq 1$. (α je generator natanko določene ciklične grupe reda q v $\mathbb{Z}_p^*$.)
4. Izberi naključno naravno število a manjše od q.
5. Izračunaj $\beta = \alpha^a$ mod p.
6. Javni ključ osebe A je (p, q, α, β), zasebni ključ pa je a.

Opomba: red elementov α, β in γ je enak q.

DSA - podpis sporočila x

1. Izberi naključno naravno število k, ki je manjše od q.
2. Izračunaj $\gamma = (\alpha^k \bmod p) \bmod q$.
3. Izračunaj $k^{-1} \bmod q$.
4. Izračunaj $\delta = k^{-1}(h(x) + a\gamma) \bmod q$, kjer je $h(x)$ povzetek sporčila x, dobljen z zgoščevalno funkcija SHA-1.
5. Če je $\gamma = 0$ ali $\delta = 0$, potem začni ponovno s korakom (1).
6. Podpis sporočila x je par (γ, δ).

DSA - preverjanje podpisa

1. Priskrbi si overjeno kopijo javnega ključa (p, q, α, β) osebe, katere podpis preverjaš.
2. Izračunaj $w = \delta^{-1} \bmod q$ in $h(x)$.
3. Izračunaj $e_1 = h(x)w \bmod q$ in $e_2 = \gamma w \bmod q$.
4. Izračunaj $v = (\alpha^{e_1}\beta^{e_2} \bmod p) \bmod q$.
5. Sprejmi podpis, če in samo če je $v = \gamma$.