UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO
FAKULTETA ZA MATEMATIKO IN FIZIKO

Janoš Vidali

SKUPINSKI PODPISI

Diplomska naloga
na univerzitetnem študiju

Mentor: prof. dr. Aleksandar Jurišić

Ljubljana, 2008

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Namesto te strani **vstavite** original izdane teme diplomskega dela s podpisom mentorja in dekana ter žigom fakultete, ki ga diplomant dvigne v študentskem referatu, preden odda izdelek v vezavo!

Zahvala

Rad bi se zahvalil svojemu mentorju Aleksandru Jurišiću za nasvete in pomoč pri izdelavi te diplomske naloge, ter Matjažu Urlepu in Jerneju Tonejcu, ki sta me opozorila na marsikatero nedoslednost. Prav tako bi se rad zahvalil staršem in sošolcem za izkazano podporo.

Kazalo

Seznam uporabljenih kratic in simbolov

RSA	Rivest, Shamir, Adleman (asimetrični kriptosistem)
$\mathbb{N}$	množica naravnih števil
$\mathbb{Z}$	množica celih števil
$\mathbb{R}$	množica realnih števil
$\mathbb{P}$	množica praštevil
U	množica članov skupine (angl. *users*)
$\mathcal{H}$	zgoščevalna funkcija (angl. *hash function*)
Pr	verjetnost (angl. *probability*)
exp	poskus [2.3.4] (angl. *experiment*)
Adv	prednost [2.3.5] (angl. *advantage*)
anon	poskus napada na anonimnost [2.3.6] (angl. *anonymity*)
bu-anon	poskus napada na anonimnost z vzvratno nepovezljivostjo [4.2.4] (angl. *backwards unlinkability anonymity*)
trace	poskus napada na sledljivost [2.3.9] (angl. *traceability*)
ftrace	poskus napada na polno sledljivost [4.2.6] (angl. *full traceability*)
nf	poskus napada na nepodtakljivost [2.3.10] (angl. *non-frameability*)
unforg	poskus napada na neponaredljivost [3.3.1] (angl. *unforgeability*)
ind-cca	poskus napada na nerazločljivost pri napadu z izbranim tajnopisom [3.4.1] (angl. *indistinguishability under chosen ciphertext attack*)
pk	javni ključ (angl. *public key*)
sk	zasebni ključ (angl. *secret key*)
gpk	javni ključ skupine (angl. *group public key*)
gsk	tajni ključ skupine (angl. *group secret key*)

SPK	podpis znanja [3.5.1] (angl. *signature based on a proof of knowledge*)
gcd	največji skupni deljitelj (angl. *greatest common divisor*)
φ	Eulerjeva funkcija [1]
ord	red elementa grupe (angl. *order*)
$\top$	uspeh (angl. *true*)
$\bot$	neuspeh (angl. *false*, tj. obrnjen $\top$)

Povzetek

Skupinski podpisi omogočajo članom skupine podpisovanje v imenu skupine, pri čemer ostane identiteta podpisnika skrita za javnost. Razkrije ga lahko le nadzornik skupine, ki poseduje tajni ključ za odpiranje podpisov. Definirali bomo shemo za skupinske podpise in predstavili varnostne zahteve anonimnosti, sledljivosti, neponaredljivosti, odpornosti proti koalicijam in nepovezljivosti ter njihove formalizirane različice. Nato bomo pogledali kriptografske primitive, ki nam služijo kot sestavni deli shem za skupinske podpise. Predstavili bomo nekatere računske modele in predpostavke o računski zahtevnosti, nato pa navedli varnostne zahteve za sheme za digitalne podpise in asimetrične šifrirne sheme, ki jih lahko uporabimo pri gradnji sheme za skupinske podpise. Pokazali bomo še nekaj neinteraktivnih dokazov brez razkritja znanja. Sledila bo predstavitev dveh shem za skupinske podpise. Prva, Camenisch-Michelsova shema, je ena od prvih varnih in učinkovitih shem za skupinske podpise, ki pa ne omogoča odstranjevanja članov iz skupine. Tako bomo predstavili še drugo, Zhou-Linovo shemo, ki to omogoča in je še učinkovitejša, a pri tem žrtvuje nekaj varnosti. Nazadnje bomo pogledali še nekaj možnih variant skupinskih podpisov in predlagali tudi nov tip sheme za skupinske podpise.

Ključne besede:

asimetrična kriptografija, varnost, anonimnost, digitalni podpis, skupinski podpis, dokaz brez razkritja znanja

Abstract

Group signatures make it possible for group members to sign messages on behalf of the group, while not revealing the signer's identity. Only the group's revocation manager, who is in possession of a secret key for signature opening, can identify the signer. We define a group signature scheme and present the security notions of anonymity, traceability, unforgeability, coalition resistance, unlinkability and their formalized counterparts. We also take a look at the cryptographic primitives that the group signature schemes consist of. We present some computational models and computational hardness assumptions, and then we give the security notions for digital signature schemes and asymmetric encryption schemes that can be used for building a group signature scheme. We then show some non-interactive zero-knowledge proofs. An overview of two group signature schemes follows. The first scheme by Camenisch and Michels was among the first secure and efficient group signature schemes, but it does not support removing group members. Next we present the scheme by Zhou and Lin, which supports group member revocation and is even more efficient, but its security is somewhat reduced. Finally, we take a look at possible versions of group signatures and propose a new group signature scheme.

Keywords:

asymmetric cryptography, security, anonymity, digital signature, group signature, zero-knowledge proof

Poglavje 1

Uvod

Že od začetkov asimetrične kriptografije [13] poznamo koncept **digitalnega podpisa** (angl. *digital signature*). Pri njem ima vsak uporabnik svoj javni ključ pk, ki je dostopen vsakomur, in zasebni ključ sk, ki ga obdrži zase. Če želi Anita podpisati sporočilo m, namenjeno Borisu, uporabi algoritem $\mathrm{Sig}(sk, m)$ in tako dobi podpis σ, ki ga skupaj s sporočilom m pošlje Borisu. Če se želi ta prepričati, da je sporočilo res prišlo od Anite in ali ga ni morda kdo spreminjal, uporabi algoritem $\mathrm{Ver}(pk, m, \sigma)$, ki mu pove, ali je podpis σ veljaven podpis sporočila m. Če je temu tako, je lahko Boris prepričan, da je sporočilo m res prišlo od Anite v nespremenjeni obliki. Danes poznamo več vrst digitalnih podpisov, kot so slepi podpisi, enkratni podpisi, podpisi brez možnosti zanikanja, *fail-stop* podpisi, ...

Denimo, da imamo podjetje, v katerem lahko zaposleni podpisujejo dokumente v imenu podjetja, pri čemer nočemo, da stranke vedo, kdo je podpisal posamezen dokument. V primeru zlorabe s strani zaposlenega pa vseeno želimo, da ga lahko direktor identificira. Običajnih digitalnih podpisov v tem primeru ne moremo uporabljati, saj bi ti izdali identiteto pospisnika. Zato uvedemo nov tip digitalnega podpisa – **skupinski podpis**.

Koncept skupinskega podpisa (angl. *group signature*) sta podala David Chaum in Eugene van Heyst leta 1991 v [11]. Glavna ideja je podpisovanje v imenu skupine, kjer lahko zunanji preverjevalec za nek podpis ugotovi le, da ga je podpisal član skupine, identiteto podpisnika pa lahko razkrije le določena oseba – **nadzornik skupine** (angl. *revocation manager*).

Še ena možnost uporabe skupinskih podpisov je pri spletnih dražbah. Denimo, da se udeleženci zavežejo, da bodo za predmet dejansko plačali, če bodo ponudili najvišjo ceno. Poleg tega želimo, da so ponudbe anonimne. Tako vzpostavimo skupino, katere člani so udeleženci dražbe, njen nadzornik pa

je vodja dražbe. Vsaka ponudba se tako podpiše s skupinskim podpisom in anonimno objavi. Ko se dražba konča, vodja odpre podpis najvišje ponudbe in tako izve, kdo jo je oddal.

Skupinski podpisi se uporabljajo tudi v druge namene, na primer za anonimno avtentikacijo [14] ali elektronski denar [19, 25]. Če se lahko znebimo nadzornika skupine, jih lahko uporabljamo tudi za elektronske volitve [21, 18].

V nadaljevanju si bomo pogledali definicijo sheme za skupinske podpise in varnostne zahteve, ki jim mora zadoščati. Sledila bo predstavitev osnovnih gradnikov, uporabljenih v shemah za skupinske podpise, nato pa bomo videli še dva konkretna primera shem za skupinske podpise in njuno varnostno analizo. Zaključili bomo s še nekaj možnimi variantami skupinskih podpisov in predlagali nov tip sheme za skupinske podpise.

Notacija in definicije

Uporabljali bomo sledečo notacijo:

- imena algoritmov pišemo s tiskanimi črkami,
- ključi so označeni s *poševnimi* črkami,
- množice bomo ponavadi označevali z velikimi pisanimi črkami,
- za ostale vrednosti uporabljamo *kurzivne* ali grške črke,
- postopki so navedeni v okvirjih,
- v korakih, kjer preverjamo določen pogoje, s simboli $\stackrel{?}{=}$, $\stackrel{?}{\neq}$ in $\stackrel{?}{\in}$ zaporedoma označimo preverjanje enakosti, neenakosti in vsebovanosti v množici,
- če pogoj velja, se postopek nadaljuje, sicer pa se prekine z neuspehom (torej vrne $\perp$).

Naj bo $\mathcal{M}$ množica. Potem je $\mathcal{PM}$ njena potenčna množica, torej množica vseh njenih podmnožic. Simboli $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$ in $\mathbb{P}$ zaporedoma označujejo množice naravnih, celih in realnih števil ter množico praštevil. Naj bo $[a..b]$ pri $a, b \in \mathbb{Z}$, $a \leq b$, interval celih števil od a do b, torej $[a..b] = [a, b] \cap \mathbb{Z}$. Za naravno število n je $\mathbb{Z}_n$ kolobar z elementi iz $[0..(n-1)]$ in aritmetiko po modulu n. Z $\gcd(a, b)$ označimo največji skupni deljitelj naravnih števil a in b. V kolobarju $\mathbb{Z}_n$ definirajmo grupo $\mathbb{Z}_n^*$, torej $\mathbb{Z}_n^* = \{ x \in [1..(n-1)] \mid \gcd(x, n) = 1 \}$.

Naj bo $\varphi(n)$ Eulerjeva funkcija, katere vrednost je enaka velikost grupe $\mathbb{Z}_n^*$. Z G bomo označevali grupe. Naj bo g element grupe. Potem je $\mathrm{ord}(g)$ oznaka za njegov **red**, torej najmanjši tak $r \in \mathbb{N}$, da je $g^r = 1$. Z $\langle g \rangle$ označimo ciklično grupo, ki jo generira element g, torej so njeni elementi $g, g^2, \ldots g^r$, kjer je $r = \mathrm{ord}(g)$. Tedaj je $\log_g a$ za $a, g \in G$ **diskretni logaritem elementa** a **pri osnovi** g, torej je $x = \log_g a$ tak $x \in \mathbb{Z}_r$, da velja $g^x = a$.

Za $a \in \mathbb{Z}$ in $p \in \mathbb{P}$ definirajmo **Legendrov simbol** $\left(\frac{a}{p}\right)$:

$$\left(\frac{a}{p}\right) = \begin{cases} 0; & a \text{ je večkratnik } p, \\ 1; & \exists x \in \mathbb{Z}_p^* : x^2 \equiv a \pmod{p}, \\ -1; & \text{sicer.} \end{cases}$$

Za naravno število n s praštevilskim razcepom $n = \prod_{i=1}^{k} p_i^{e_i}$ definirajmo še **Jacobijev simbol**:

$$\left(\frac{a}{n}\right) = \prod_{i=1}^{k} \left(\frac{a}{p_i}\right)^{e_i}.$$

Naj bo A verjetnostni algoritem. Z $[\mathrm{A}(p_1, \ldots, p_n)]$ označimo množico vseh možnih izhodov pri izvajanju algoritma A s parametri $p_1, \ldots, p_n$. Z $x := \mathrm{A}(p_1, \ldots, p_n)$ označimo priredbo izhoda algoritma spremenljivki x, torej x dobi vrednost iz $[\mathrm{A}(p_1, \ldots, p_n)]$. Če je $\mathcal{M}$ končna množica, $x \in_R \mathcal{M}$ pomeni, da element x naključno (angl. *random*) izberemo iz množice $\mathcal{M}$ po enakomerni porazdelitvi.

Naj bo $\mathcal{S}$ množica znakov. Če je $k \in \mathbb{N}_0$, je $\mathcal{S}^k$ množica vseh besed dolžine k, sestavljenih iz znakov v množici $\mathcal{S}$. Definirajmo še $\mathcal{S}^* = \bigcup_{k=0}^{\infty} \mathcal{S}^k$, torej je $\mathcal{S}^*$ množica besed poljubne končne dolžine, sestavljenih iz znakov v $\mathcal{S}$. Če sta w_1 in w_2 besedi, označimo z $w_1 \| w_2$ njuno združitev. Kot množico znakov bomo navadno vzeli $\mathcal{S} = \{0, 1\}$. Včasih se bodo spremenljivke, definirane kot besede, pojavljale v aritmetičnih izrazih – tedaj jih interpretiramo kot števila v dvojiškem zapisu. Nasprotno bomo tudi notacijo z $\|$ zlorabili za združevanje neznakovnih vrednosti, predvsem za elemente grup, ki jih tedaj razumemo kot njihovo dvojiško predstavitev.

Poglavje 2

Osnovni pojmi in varnostne zahteve

Za uporabo skupinskih podpisov je potrebno najprej izbrati primerno shemo in vzpostaviti skupino. Vsak član skupine mora izvesti protokol za pridružitev skupine, pri čemer pridobi svoj zasebni ključ, s katerim lahko podpisuje sporočila v imenu skupine. Nadzornik skupine pa si pri vzpostavitvi pridobi tajni ključ, s katerim lahko identificira avtorja posameznega podpisa. Objavi se še javni ključ skupine, s katerim lahko kdorkoli za skupinski podpis ugotovi, ali je veljaven in ga je torej opravil eden od članov skupine.

Za dosego ciljev uporabe skupinskih podpisov morajo sheme zanje ustrezati določenim varnostnim zahtevam. Nekatere sledijo že iz definicije skupinskega podpisa, druge pa so bile predstavljene v kasnejši literaturi. Ker pa se te zahteve ponavadi podajajo v neformalni obliki, bomo tako začeli tudi mi, nato pa predstavili še njihovo formalizacijo iz [2] in [4].

2.1 Definicija sheme za skupinske podpise

Definirajmo sledeče množice:

- $\mathcal{M}$ je množica sporočil,
- $\mathcal{S}$ je množica podpisov,
- $\mathcal{K}_{gp}$ je množica javnih ključev skupine (angl. *group public keys*),
- $\mathcal{K}_{gs}$ je množica tajnih ključev nadzornika skupine (angl. *group secret keys*),

- $\mathcal{K}_{us}$ je množica zasebnih ključev članov skupine (angl. *user secret keys*),
- $\mathcal{U}$ je množica potencialnih članov skupine (angl. *universum*),
- $U \subseteq \mathcal{U}$ je trenutna skupina (angl. *users*).

Naj bo $k \in \mathbb{N}$ varnostni parameter. Potem je shema za skupinske podpise peterica algoritmov

$$(\text{Gen}, \text{Issue}, \text{GSig}, \text{GVer}, \text{Open}),$$

za katero velja:

- $\text{Gen} : \mathbb{N} \to \mathcal{K}_{gp} \times \mathcal{K}_{gs}$ je verjetnostni algoritem za generiranje ključev skupine,
- $\text{Issue} : \mathcal{K}_{gp} \times \mathcal{K}_{gs} \times \mathcal{U} \to \mathcal{K}_{us}$ je algoritem za izdajanje članskih ključev,
- $\text{GSig} : \mathcal{K}_{gp} \times \mathcal{K}_{us} \times \mathcal{M} \to \mathcal{S}$ je algoritem za podpisovanje sporočila,
- $\text{GVer} : \mathcal{K}_{gp} \times \mathcal{M} \times \mathcal{S} \to \{\top, \bot\}$ je algoritem za preverjanje podpisa,
- $\text{Open} : \mathcal{K}_{gp} \times \mathcal{K}_{gs} \times \mathcal{PU} \times \mathcal{M} \times \mathcal{S} \to \mathcal{U} \cup \{\bot\}$ je algoritem za odstiranje avtorja podpisa.

Naj bosta gpk in gsk javni in tajni ključ skupine, dobljena z algoritmom $\text{Gen}(k)$, ter sk_u zasebni ključ člana u skupine U, dobljen z uporabo algoritma $\text{Issue}(gpk, gsk, u)$. Veljati morata naslednja kriterija:

1. za vsak $m \in \mathcal{M}$ velja:

$$\sigma \in [\text{GSig}(gpk, sk_u, m)] \Leftrightarrow \text{GVer}(gpk, m, \sigma) = \top \Leftrightarrow$$

$$\Leftrightarrow \text{Open}(gpk, gsk, U, m, \sigma) = u,$$

2. za vsaka $m \in \mathcal{M}$ in $\sigma \in \mathcal{S}$ velja:

$$\text{GVer}(gpk, m, \sigma) = \bot \Leftrightarrow \text{Open}(gpk, gsk, U, m, \sigma) = \bot.$$

Pri podani formalni definiciji niso jasno razvidne posamezne vloge v skupini. V najbolj osnovnem primeru imamo le nadzornika skupine, ki ima tajni ključ skupine gsk, in člane skupine s svojimi zasebnimi ključi sk_u. Pogosto se zahteva, da se vloga nadzornika skupine razdeli [4, 8], najpogosteje na:

- izdajatelja – osebo, ki skrbi za članstvo v skupini, in
- odpiralca – osebo, ki lahko identificira podpisnika.

V tem primeru ima vsak od njiju svoj tajni ključ, ki ga potrebuje za svojo vlogo.

Še ena pomembna lastnost, ki ni razvidna iz definicije, je preverjanje pravilnosti izvajanja operacij. To velja tako za generiranje skupine, kot tudi za dodajanje člana v skupino in odpiranje podpisa. Pri prvih dveh operacijah sodelujejo le člani skupine in avtoritete, zato jih lahko izpeljemo kot protokole z dokazi brez razkritja znanja (glej 3.5). Pri odpiranju podpisa pa želimo javnosti oziroma neki tretji osebi dokazati pravilnost odprtja, zato pridejo tukaj v upoštev neinteraktivni dokazi brez razkritja znanja.

Nazadnje povejmo še to, da je shema za skupinske podpise lahko **statična** ali **dinamična**. Pri statičnih shemah se skupina določi ob nastanku in je kasneje ni več mogoče spreminjati. Dinamične sheme ločimo na delno dinamične oziroma monotono rastoče sheme, kjer lahko člane v skupino le dodajamo, in popolnoma dinamične sheme, kjer lahko člane tudi izločamo iz skupine.

2.2 Neformalne varnostne zahteve

Že iz same ideje skupinskih podpisov sledi zahteva o **anonimnosti** (angl. *anonymity*): brez tajnega ključa nadzornika skupine ni mogoče identificirati podpisnika. Iz definicije sledi tudi zahteva o **sledljivosti** (angl. *traceability*), ki pravi, da lahko nadzornik skupine identificira avtorja vsakega veljavnega skupinskega podpisa. Že iz navadnih digitalnih podpisov pa imamo zahtevo o **neponaredljivosti** (angl. *unforgeability*), torej da brez zasebnega ključa člana skupine ni mogoče ponarediti veljavnega skupinskega podpisa.

Zaželena lastnost sheme za skupinske podpise je tudi **odpornost proti koalicijam** (angl. *coalition resistance*) – nobena koalicija članov skupine in njenega nadzornika ne more ponarediti skupinskega podpisa nekega člana skupine, ki ni del koalicije. Poseben primer te zahteve, ko je koalicija sestavljena iz enega samega člana (angl. *exculpability*), bi lahko umestili tudi med osnovne zahteve. Drugi poseben primer imamo, ko koalicijo sestavljajo nadzornik in vsi člani skupine, z izjemo enega, katerega podpis se poskuša ponarediti (angl. *framing*).

Zadnja zahteva, ki pa ni univerzalna, je **nepovezljivost** (angl. *unlinkability*). Ta pravi, da brez tajnega ključa nadzornika skupine za dva veljavna skupinska podpisa ni mogoče povedati, ali prihajata od istega člana ali ne. Ta

lastnost je sicer ponavadi zaželena, v nekaterih primerih, kot so elektronske volitve, pa temu ni tako – takrat namreč želimo vedeti, ali ni morda kdo glasoval večkrat.

2.3 Formalizacija varnostnih zahtev

Vse navedene neformalne zahteve je mogoče formalizirati v dve zahtevi pri statičnih skupinah [2] oziroma tri zahteve pri dinamičnih skupinah [4]. Preden si jih pogledamo, uvedimo še nekaj osnovnih pojmov.

Definicija 2.3.1. Funkcija $f : \mathbb{N} \to \mathbb{R}$ je (asimptotično) **zanemarljiva**, če za vsak neničelni polinom p obstaja tako število m, da

$$\text{za vsak } n \in \mathbb{N},\ n > m,\ \text{velja } |f(n)| < \frac{1}{|p(n)|}.$$

Če je funkcija f zanemarljiva, pišemo $f(n) < \varepsilon$.

Definicija 2.3.2. (Polinomski) **orakelj** je funkcija, ki lahko v polinomskem času odgovori na nekatere poizvedbe, za katere uporabnik te funkcije ne pozna učinkovitega algoritma.

Definicija 2.3.3. (Polinomski) **nasprotnik** N je oseba, ki algoritmom s polinomsko časovno zahtevnostjo zagotavlja dostop do določenih podatkov in orakljev.

Definicija 2.3.4. Poskus $\exp_N$ je algoritem, ki ga izvede nasprotnik N z namenom, da ogrozi neko varnostno lastnost kriptografske sheme.

Definicija 2.3.5. Prednost $\mathrm{Adv}_N^{\exp}$ nasprotnika N, ki izvaja poskus $\exp_N$, je razlika med verjetnostjo, da $\exp_N$ uspe, in verjetnostjo, da je cilj dosežen z naključnim ugibanjem.

V sledečih varnostnih zahtevah, ki veljajo za dinamične skupine, bodo kot zanemarljive funkcije nastopale prednosti nasprotnika. Njihov argument bo implicitni varnostni parameter k, ki določa lastnosti sheme, kot sta dolžina ključev in velikost podpisa. V primerih, ko je verjetnost, da bi dosegli cilj z naključnim ugibanjem, zanemarljiva, jo bomo pri definicijah posameznih prednosti izpuščali.

Anonimnost. Nasprotnik N si izbere sporočilo m ter člana skupine, ki ju označimo z i_0 in i_1. Naj bo $b \in \{0, 1\}$. Član i_b izda podpis σ sporočila

m. Poskus nasprotnika N, da identificira podpisnika sporočila σ, označimo z anon_N^b. Poskus vrne x, če je kot podpisnika identificiral člana i_x, $x \in \{0,1\}$.

Predpostavljamo, da ima pri poskusu anon_N^b nasprotnik N dostop do orakljev za:

- identifikacijo podpisnika,
- dodajanje članov v skupino,
- spreminjanje podatkov o članih skupine,

poleg tega pozna tudi:

- vse zasebne ključe sk_u članov u skupine U.

Oraklja za identifikacijo podpisnika nasprotnik N ne sme poklicati za izbrano sporočilo m in njegov podpis σ, sicer poskus ne uspe.

Definicija 2.3.6. Shema za skupinske podpise je **anonimna**, če je prednost $\text{Adv}_N^{\text{anon}}$ polinomskega nasprotnika N zanemarljiva:

$$\text{Adv}_N^{\text{anon}} = \Pr(\text{anon}_N^1 = 1) - \Pr(\text{anon}_N^0 = 1) < \varepsilon.$$

Opomba 2.3.7. Prednost $\text{Adv}_N^{\text{anon}}$ smo definirali kot vsoto prednosti pri posameznem poskusu anon_N^b, $b \in \{0,1\}$. Pri vsakem poskusu je verjetnost uspeha ob naključnem ugibanju enaka $\frac{1}{2}$. Če verjetnosti seštejemo, dobimo:

$$\text{Adv}_N^{\text{anon}} = \Pr(\text{anon}_N^1 = 1) + \Pr(\text{anon}_N^0 = 0) - 1.$$

Če upoštevamo še $\Pr(\text{anon}_N^0 = 1) = 1 - \Pr(\text{anon}_N^0 = 0)$, dobimo ravno formulo iz 2.3.6.

Opomba 2.3.8. Za verjetnosti $\Pr(\text{anon}_N^b = b)$, $b \in \{0,1\}$ lahko privzamemo, da sta večji ali enaki kot $\frac{1}{2}$, saj sicer lahko skonstruiramo tak poskus anon'^b_N, ki vrača $1 - \text{anon}_N^b$ in je tako njegova verjetnost uspeha večja od $\frac{1}{2}$. Prednost $\text{Adv}_N^{\text{anon}}$ je tako vedno nenegativna.

Sledljivost. Nasprotnik N si izbere sporočilo m in poskuša ponarediti veljaven podpis σ (torej $\text{GVer}(gpk, m, \sigma) = \top$), za katerega velja ena od naslednjih trditev:

- za podpis σ ni mogoče identificirati podpisnika:

$$\mathrm{Open}(gpk, gsk, U, m, \sigma) = \perp,$$

- za identificiranega podpisnika ne obstaja veljaven dokaz.

Poskus nasprotnika N, da ponaredi tak podpis σ, označimo s trace_N. Če je ponarejanje podpisa uspešno, poskus vrne 1, sicer pa 0.

Predpostavljamo, da ima pri poskusu trace_N nasprotnik N dostop do orakljev za:

- dodajanje članov v skupino,
- branje podatkov o članih skupine,

poleg tega pa pozna tudi:

- vse zasebne ključe sk_u članov u skupine U,
- tajni ključ gsk nadzornika skupine za identifikacijo podpisnika.

Definicija 2.3.9. Shema za skupinske podpise je **sledljiva**, če je prednost $\mathrm{Adv}_N^{\mathrm{trace}}$ polinomskega nasprotnika N zanemarljiva:

$$\mathrm{Adv}_N^{\mathrm{trace}} = \Pr(\mathrm{trace}_N = 1) < \varepsilon.$$

Nepodtakljivost. Nasprotnik N si izbere podmnožico članov skupine $\mathcal{C} \subseteq U$ in sporočilo m ter poskuša ponarediti njegov podpis σ in dokaz, da je podpisnik član i skupine U, ki ga ni v množici $\mathcal{C}$. Poskus nasprotnika N, da ponaredi tak podpis σ, označimo z nf_N. Če je ponarejeni podpis veljaven in dokaz pravilen, poskus vrne 1, sicer pa 0.

Predpostavljamo, da ima pri poskusu nf_N ima nasprotnik N dostop do orakljev za:

- podpisovanje z zasebnimi ključi članov skupine,
- dodajanje članov v skupino,
- spreminjanje podatkov o članih skupine,

poleg tega pa pozna:

- zasebne ključe sk_u za izbrano podmnožico članov $u \in \mathcal{C}$,
- tajni ključ *gsk* nadzornika skupine za identifikacijo podpisnika.

Oraklja za podpisovanje nasprotnik N ne sme poklicati za sporočilo m in člana i skupine U, sicer poskus ne uspe.

Definicija 2.3.10. Shema za skupinske podpise je **nepodtakljiva**, če je prednost $\mathrm{Adv}_N^{\mathrm{nf}}$ polinomskega nasprotnika N zanemarljiva:

$$\mathrm{Adv}_N^{\mathrm{nf}} = \Pr(\mathrm{nf}_N = 1) < \varepsilon.$$

V primeru statičnih skupin nimamo orakljev za dodajanje članov v skupino in za spreminjanje podatkov o članih, tako da se v tem primeru sledljivost in nepodtakljivost združita v eno samo varnostno zahtevo – **polno sledljivost**. Analogno tudi zahtevo o anonimnosti v tem primeru imenujemo **polna anonimnost**.

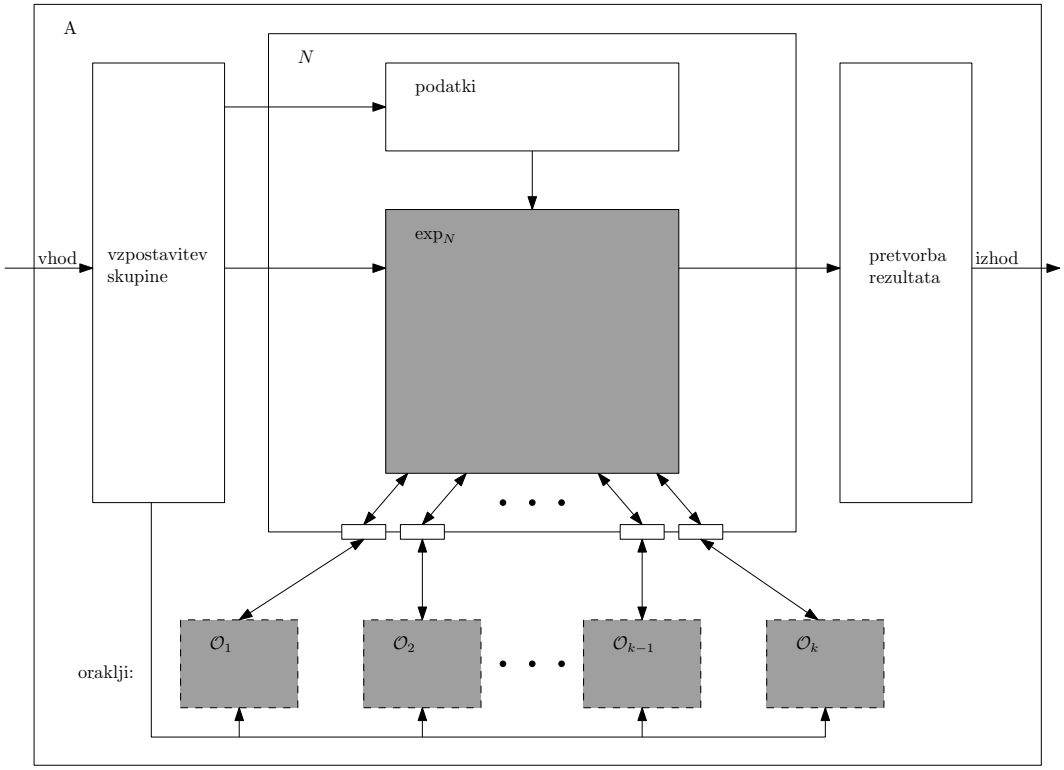
Če bi želeli dokazati, da ima shema za skupinske podpise katero od naštetih varnostnih lastnosti, za nek težek ali domnevno težek problem skonstruiramo polinomski verjetnostni algoritem A, ki komunicira z nasprotnikom N in odgovarja na njegove poizvedbe orakljem, ko nasprotnik N izvaja poskus $\exp_N$ (slika 2.1). Ker vemo ali domnevamo, da tak algoritem A ne obstaja, potem tudi prednost nasprotnika N pri izvajanju poskusa $\exp_N$ ne more biti zanemarljiva. Nasprotno, če želimo katero od varnostnih lastnosti ovreči, skonstruiramo polinomski algoritem $\exp_N$ z nezanemarljivo prednostjo uspeha.

2.4 Razmerja med neformalnimi in formaliziranimi zahtevami

Pokažimo, da shema, ki ustreza formaliziranim varnostnim zahtevam, ustreza tudi neformalnim zahtevam.

Trditev 2.4.1. *Shema, ki je (polno) anonimna, ustreza tudi neformalni zahtevi o anonimnosti.*

Dokaz. Dokažimo s protislovjem. Če bi lahko nasprotnik N identificiral podpisnika danega podpisa z verjetnostjo, ki ni zanemarljivo različna od $\frac{1}{2}$, potem bi tudi poskusa anon_N^0 in anon_N^1 uspevala z verjetnostjo, ki ni zanemarljivo različna od $\frac{1}{2}$. Po 2.3.8 sta ti verjetnosti večji od $\frac{1}{2}$. Prednost $\mathrm{Adv}_N^{\mathrm{anon}}$ nasprotnika N tako ne bi bila zanemarljiva, torej shema ne bi bila (polno) anonimna. □



Slika 2.1: Model z nasprotnikom N. Če želimo dokazati neobstoj polinomskega poskusa $\exp_N$ z nezanemarljivo verjetnostjo uspeha, ga prevedemo na algoritem A za domnevno težek problem.

Trditev 2.4.2. *Shema, ki je (polno) sledljiva, ustreza tudi neformalni zahtevi o sledljivosti.*

Dokaz. Dokažimo s protislovjem. Če bi lahko nasprotnik N z nezanemarljivo verjetnostjo ponaredil veljaven podpis, za katerega ne bi bilo mogoče ugotoviti podpisnika, potem bi poskus trace_N uspel z nezanemarljivo verjetnostjo in shema ne bi bila (polno) sledljiva. □

Trditev 2.4.3. *Shema, ki je nepodtakljiva oziroma polno sledljiva, ustreza tudi neformalni zahtevi o neponaredljivosti.*

Dokaz. Dokažimo s protislovjem. Če bi lahko nasprotnik N z nezanemarljivo verjetnostjo ponaredil podpis člana u skupine U, za katerega ne pozna zasebnega ključa, potem bi poskus nf_N uspel z nezanemarljivo verjetnostjo, saj bi lahko s pomočjo ključa nadzornika skupine tvoril tudi dokaz, da je podpisnik

član u. Shema tako ne bi bila nepodtakljiva in tako tudi ne bi bila polno sledljiva. □

Trditev 2.4.4. *Shema, ki je nepodtakljiva oziroma polno sledljiva, ustreza tudi neformalni zahtevi o odpornosti proti koalicijam.*

Dokaz. Dokažimo s protislovjem. Če bi lahko nasprotnik N z nezanemarljivo verjetnostjo ponaredil podpis člana u skupine U, ki ni član koalicije $\mathcal{C}$ in torej zanj ne pozna zasebnega ključa, potem bi poskus nf_N uspel z nezanemarljivo verjetnostjo, saj bi lahko s pomočjo ključa nadzornika skupine nasprotnik N tvoril tudi dokaz, da je podpisnik član u. Shema tako ne bi bila nepodtakljiva in tako tudi ne bi bila polno sledljiva. □

Trditev 2.4.5. *Shema, ki je (polno) anonimna, ustreza tudi neformalni zahtevi o nepovezljivosti.*

Dokaz. Če bi lahko nasprotnik N za dva različna podpisa z verjetnostjo, ki ni zanemarljivo različna od $\frac{1}{2}$, ugotovil, ali je njun avtor isti, bi lahko pri poskusu anon_N^b storil sledeče: izbral bi člana skupine i_0 in i_1, nato pa bi z zasebnim ključem enega izmed njiju izdal podpis σ' izbranega sporočila. Za podpis σ sporočila m, ki ga je izdal član i_b skupine U, bi nato nasprotnik N ugotovil, ali prihaja od istega člana kot σ'. Poskus anon_N^b bi tako uspel z verjetnostjo, ki ni zanemarljivo različna od $\frac{1}{2}$ in shema ne bi bila (polno) anonimna. □

Poglavje 3

Osnovni gradniki shem

Kakor bomo videli v naslednjem poglavju, sheme za skupinske podpise sestavljajo različni kriptografski primitivi. Najprej si bomo pogledali računske modele in nekatere predpostavke o računski zahtevnosti, uporabljene pri shemah za skupinske podpise, nato pa še splošne predpostavke za gradnike in nekaj primerov.

3.1 Računski modeli

Osnovni model, ki ga ponavadi privzamemo v kriptografiji, je **standardni model** [3]. Pri njem predpostavljamo, da je nasprotnik omejen le s časom in računsko močjo, ki ju ima na voljo. Shema je varna v standardnem modelu, če njeno varnost lahko dokažemo le s predpostavkami o računski zahtevnosti.

Ker je dokaze v standardnem modelu pogosto težko najti, si zato pomagamo tako, da nadomestimo določene kriptografske primitive z njihovimi idealiziranimi različicami. Tipičen primer takega modela je **model z naključnim orakljem** [3]. Tukaj namesto zgoščevalnih funkcij uporabimo naključnega oraklja (angl. *random oracle*) – funkcijo, ki vrača naključen izhod, vendar je ta izhod pri istem vhodu vedno enak. Za večino kriptografskih shem, ki so dokazano varne v modelu z naključnim orakljem, velja prepričanje, da so varne tudi v standardnem modelu. Sicer pa je mogoče sestaviti sheme, ki so varne v modelu z naključnim orakljem, a se jih da trivialno razbiti v standardnem modelu [10].

Še en model, ki se uporablja predvsem pri neinteraktivnih dokazih brez razkritja znanja, je **model s skupnim referenčnim nizom** [5]. Pri njem predpostavljamo, da imajo vsi dostop do skupnega niza z (angl. *common reference string*), ki je bil izbran po neki določeni distribuciji. Shema, ki je varna po tem modelu, je varna tudi v standardnem modelu, če se niz z izbere

pravično, neodvisno od vseh vpletenih.

3.2 Predpostavke o računski zahtevnosti

Kakor pri večini javne kriptografije, ki se dandanes uporablja, tudi varnost večine shem za skupinske podpise temelji na predpostavkah o zahtevnosti problema razcepa števil in problema diskretnega logaritma oziroma sorodnih problemov. Opisali bomo nekaj takih, ki so uporabljenih pri shemah, predstavljenih v naslednjem poglavju.

Razcep naravnega števila. Cilj problema razcepa naravnega števila je za dano naravno število n najti njegov praštevilski razcep, torej $n = \prod_{i=1}^{k} p_i^{e_i}$, kjer so $p_i \in \mathbb{P}$ in $e_i \in \mathbb{N}$ za $i = 1, \ldots, k$. Predpostavljamo, da je ta problem težek. Najboljši znani algoritmi za razcep naravnega števila imajo namreč podeksponentno, a nadpolinomsko časovno zahtevnost. Izkaže se, da je problem najtežji, ko je n produkt dveh približno enako velikih praštevil.

Velja opomniti, da za razcep naravnega števila obstaja kvantni algoritem, ki teče v polinomskem času [24]. Vendar pa to ne ogroža predpostavke o težkosti problema, saj je največje število, katerega razcep je uspel na kvantnem računalniku, enako 15 [27]. Za vsak n, ki bi ga želeli faktorizirati, bi namreč s trenutno tehnologijo morali sestaviti nov kvantni računalnik, kar pa bi bilo izjemno drago in težko izvedljivo.

Krepka RSA predpostavka. Ta predpostavka temelji na krepkem RSA problemu, katerege cilj je za dano grupo G in njen element z najti tak par $(u, e) \in G \times (\mathbb{N}\backslash\{1\})$, da velja $u^e = z$. Predpostavka pravi, da obstaja verjetnostni algoritem K, tako da je za vsak polinomski verjetnostni algoritem A sledeča pogojna verjetnost zanemarljiva glede na red velikosti generirane grupe:

$$\Pr\left(z = u^e \wedge e > 1 \mid (G, z) := \mathrm{K}(), (u, e) := \mathrm{A}(G, z)\right) < \varepsilon.$$

V [8] je uporabljena varianta krepke RSA predpostavke, pri kateri omejimo možne vrednosti e na nek interval oblike

$$\left[(2^a - 2^b)..(2^a + 2^b)\right],\ b < a < \ell_g,$$

kjer je ℓ_g dolžina reda grupe G.

Problem diskretnega logaritma. Diskretni logaritem $\log_g a$, kjer sta a in g elementa grupe G in je $\mathrm{ord}(g) = r$, je tako število $x \in \mathbb{Z}_r$, da velja $g^x = a$.

Predpostavljamo, da je računanje diskretnega logaritma težko. Tako kot za razcep naravnega števila imajo namreč tudi za ta problem najboljši algoritmi podeksponentno, a nadpolinomsko časovno zahtevnost. Prav tako zanj obstaja polinomski kvantni algoritem [24].

Diffie-Hellmanova odločitvena predpostavka. Naj bo G grupa in n deljitelj njenega reda. Naj bo $\mathcal{DH}(G)$ množica takih četveric (g_1, g_2, y_1, y_2), za katere velja:

$$\mathrm{ord}(g_1) = \mathrm{ord}(g_2) = n,$$

$$\log_{g_1} y_1 = \log_{g_2} y_2,$$

$\mathcal{Q}(G)$ pa naj bo množica takih četveric, kjer velja le:

$$\mathrm{ord}(g_1) = \mathrm{ord}(g_2) = \mathrm{ord}(y_1) = \mathrm{ord}(y_2) = n.$$

Predpostavka pravi, da obstaja verjetnostni algoritem K, tako da sta za vsak polinomski verjetnostni algoritem A pogojni verjetnosti v naslednjem izrazu računsko nerazločljivi, torej je izraz zanemarljiv glede na red velikosti generirane grupe $G := \mathrm{K}()$:

$$\Pr\left(a = 1 \mid T \in_R \mathcal{DH}(G), a := \mathrm{A}(T)\right) +$$

$$+ \Pr\left(a = 1 \mid T \in_R \mathcal{Q}(G), a := \mathrm{A}(T)\right) - 1 < \varepsilon. \tag{3.2.1}$$

Če povemo še z besedami, predpostavka pravi, da za primerno izbrano grupo G noben polinomski algoritem A ne bo znal zanesljivo ločiti med četvericami iz $\mathcal{DH}(G)$ in $\mathcal{Q}(G)$. Podobno kot v 2.3.8 lahko privzamemo, da sta verjetnosti v 3.2.1 večji ali enaki $\frac{1}{2}$ in je tako izraz nenegativen.

V [28] je uporabljena varianta Diffie-Hellmanove odločitvene predpostavke, ki jo imenujemo tridelna Diffie-Hellmanova odločitvena predpostavka. Pri njej imamo namesto četveric šesterice, saj dodamo še elementa y_3 in y_4. Pri šestericah iz $\mathcal{DH}(G)$ tako velja:

$$\mathrm{ord}(g_1) = \mathrm{ord}(g_2) = n,$$

$$\log_{g_1} y_4 = \log_{g_1} y_1 \log_{g_1} y_2 \log_{g_2} y_3,$$

pri $\mathcal{Q}(G)$ pa ima vseh šest elementov enak red n. V splošnem lahko g_1 in g_2 pripadata različnim grupam, pomembno je le, da sta istega reda. Če sta grupi ciklični s praštevilskim redom, potem lahko za g_1 in g_2 implicitno vzamemo kar njuna generatorja.

Velja opomniti, da Diffie-Hellmanova odločitvena predpostavka ne velja za vse grupe. Primer take grupe, da predpostavka ne velja, je $\mathbb{Z}_{pq}^*$, $p, q \in \mathbb{P}$. Tudi brez poznavanja faktorizacije pq lahko namreč Jacobijev simbol pove, da velja $\log_{g_1} y_1 \neq \log_{g_2} y_2$.

q-krepka Diffie-Hellmanova predpostavka. Za vsako ciklično grupo G z generatorjem g praštevilskega reda p in vsak polinomski verjetnostni algoritem A je sledeča pogojna verjetnost zanemarljiva glede na red velikosti grupe G:

$$\Pr\left(u = g^{(\gamma+e)^{-1}} \;\middle|\; \gamma \in_R \mathbb{Z}_p^*, (u, e) := \mathrm{A}\left(g, g^{\gamma}, \ldots, g^{\gamma^q}\right)\right) < \varepsilon.$$

Problemu iskanja para (u, e), za katerega velja $u = g^{(\gamma+e)^{-1}}$, pravimo q-krepki Diffie-Hellmanov problem.

3.3 Sheme za digitalne podpise

Prvi kriptografski primitiv, ki ga bomo predstavili, so sheme za digitalne podpise. Najpogosteje se navadni digitalni podpisi v shemah za skupinske podpise pojavljajo pri certifikatih o članstvu, lahko pa so tudi sestavni del samega skupinskega podpisa.

Naj bo $\mathcal{M}$ množica sporočil, $\mathcal{S}$ množica podpisov, $\mathcal{K}_p$ množica javnih ključev, $\mathcal{K}_s$ množica zasebnih ključev ter $k \in \mathbb{N}$ varnostni parameter. Shema za digitalne podpise je trojica algoritmov

$$(\mathrm{Gen}, \mathrm{Sig}, \mathrm{Ver}),$$

za katero velja:

- $\mathrm{Gen} : \mathbb{N} \to \mathcal{K}_p \times \mathcal{K}_s$ je verjetnostni algoritem za generiranje ključev,
- $\mathrm{Sig} : \mathcal{K}_s \times \mathcal{M} \to \mathcal{S}$ je algoritem za podpisovanje sporočila,
- $\mathrm{Ver} : \mathcal{K}_p \times \mathcal{M} \times \mathcal{S} \to \{\top, \bot\}$ je algoritem za odpiranje podpisa.

Naj bosta pk in sk javni in zasebni ključ, dobljena z algoritmom $\mathrm{Gen}(k)$. Potem mora za vsak $m \in \mathcal{M}$ veljati:

$$\sigma \in [\mathrm{Sig}(sk, m)] \Leftrightarrow \mathrm{Ver}(pk, m, \sigma) = \top.$$

Da je uporaba sheme za digitalne podpise varna, mora biti **neponaredljiva.** Nasprotnik N si izbere sporočilo m in poskuša ponarediti njegov podpis

σ. Poskus nasprotnika N, da ponaredi podpis σ, označimo z unforg_N. Če je ponarejeni podpis veljaven, poskus vrne 1, sicer pa 0.

Predpostavljamo, da ima pri poskusu unforg_N nasprotnik N dostop do oraklja za podpisovanje z zasebnim ključem *sk*. N tega ključa nima, pač pa ima javni ključ *pk*, s katerim lahko preverja podpise. Oraklja za podpisovanje nasprotnik N ne sme poklicati za sporočilo m, sicer poskus ne uspe.

Definicija 3.3.1. Shema za digitalne podpise je **neponaredljiva**, če je prednost $\text{Adv}_N^{\text{unforg}}$ polinomskega nasprotnika N zanemarljiva:

$$\text{Adv}_N^{\text{unforg}} = \Pr(\text{unforg}_N = 1) < \varepsilon.$$

3.4 Asimetrične šifrirne sheme

Asimetrične šifrirne sheme se v shemah za skupinske podpise uporabljajo predvsem za skrivanje identitete podpisnika. Identiteta je tako lahko zašifrirana z javnim ključem nadzornika skupine in se pri odpiranju podpisa odšifrira.

Naj bo $\mathcal{M}$ množica sporočil, $\mathcal{C}$ množica tajnopisov, $\mathcal{K}_p$ množica javnih ključev, $\mathcal{K}_s$ množica zasebnih ključev ter $k \in \mathbb{N}$ varnostni parameter. Asimetrična šifrirna shema je trojica algoritmov

$$(\text{Gen}, \text{Enc}, \text{Dec}),$$

za katero velja:

- $\text{Gen} : \mathbb{N} \to \mathcal{K}_p \times \mathcal{K}_s$ je verjetnostni algoritem za generiranje ključev,
- $\text{Enc} : \mathcal{K}_p \times \mathcal{M} \to \mathcal{C}$ je algoritem za šifriranje sporočila,
- $\text{Dec} : \mathcal{K}_s \times \mathcal{C} \to \mathcal{M}$ je algoritem za odšifriranje tajnopisa.

Naj bosta *pk* in *sk* javni in zasebni ključ, dobljena z algoritmom $\text{Gen}(k)$. Potem mora za vsak $m \in \mathcal{M}$ veljati:

$$\text{Dec}(sk, \text{Enc}(pk, m)) = m.$$

Da je uporaba asimetrične šifrirne sheme varna, mora zadostovati zahtevi o **nerazločljivosti pri napadu z izbranim tajnopisom**. Nasprotnik N si izbere sporočili m_0 in m_1. Naj bo $b \in \{0, 1\}$. Nasprotnik dobi tajnopis c, ki

je zašifrirano sporočilo m_b. Poskus nasprotnika N, da identificira sporočilo, ki ustreza tajnopisu c, označimo z ind-cca^b_N. Poskus vrne x, če je kot čistopis identificiral sporočilo m_x, $x \in \{0, 1\}$.

Predpostavljamo, da ima pri poskusu ind-cca^b_N nasprotnik N dostop do oraklja za odšifriranje tajnopisov, šifriranih z javnim ključem pk. Nasprotnik N pozna javni ključ pk, ne pa tudi ustreznega zasebnega ključa sk. Oraklja za odšifriranje za tajnopis c nasprotnik N ne sme poklicati, sicer poskus ne uspe.

Definicija 3.4.1. Asimetrična šifrirna shema ustreza **zahtevi o nerazločljivosti pri napadu z izbranim tajnopisom**, če je prednost $\mathrm{Adv}^{\text{ind-cca}}_N$ polinomskega nasprotnika N zanemarljiva:

$$\mathrm{Adv}^{\text{ind-cca}}_N = \Pr(\text{ind-cca}^1_N = 1) - \Pr(\text{ind-cca}^0_N = 1) < \varepsilon.$$

Tako kot v 2.3.8 lahko privazmemo, da sta uspeha poskusov ind-cca^b_N, $b \in \{0, 1\}$, večji ali enaki kot $\frac{1}{2}$ in tako je prednost $\mathrm{Adv}^{\text{ind-cca}}_N$ nenegativna.

3.5 Dokazi brez razkritja znanja

Zadnji primitiv, ki si ga bomo pogledali, so dokazi brez razkritja znanja. Čeprav ponavadi pod tem imenom mislimo na interaktivne protokole za dokazovanje, ki lahko pridejo prav tudi pri shemah za skupinske podpise (na primer pri pridruževanju skupini), pa se bomo tukaj osredotočili na neinteraktivne dokaze brez razkritja znanja.

Dokaz brez razkritja znanja je protokol, v katerem sodelujeta dokazovalec in preverjevalec. Dokazovalec želi preverjevalca prepričati, da velja neka trditev, a ne želi izdati nobene druge informacije. Rečemo, da sta dokazovalec in preverjevalec **poštena**, če sledita protokolu. Nasprotno je vsak dokazovalec ali preverjevalec, ki protokolu ne sledi, **goljufiv**. Za dokaze brez razkritja znanja morajo veljati naslednje lastnosti:

- **polnost** (angl. *completeness*): če trditev velja, se bo pošteni preverjevalec prepričal o njeni veljavnosti,
- **uglašenost** (angl. *soundness*): če trditev ne velja, je verjetnost, da bi goljufivi dokazovalec prepričal poštenega preverjevalca o njeni veljavnosti, zanemarljiva,

- **brez razkritja znanja** (angl. *zero-knowledge*): če je trditev pravilna, goljufivi preverjevalec iz dokaza ne pridobi nobene druge informacije.

Da bi dokazali, da pri dokazu goljufivi preverjevalec res ne pridobi nobene informacije, sestavimo **simulator** protokola – algoritem, ki ga izvede preverjevalec, katerega izhod je računsko nerazločljiv od prepisa interakcije med dokazovalcem in preverjevalcem pri dejanski izvedbi protokola za dokaz brez razkritja znanja. Tako se namreč prepričamo, da pri izvajanju protokola preverjevalec ne izve ničesar takega, česar ne bi že vedel.

Dokazu brez razkritja znanja, pri katerem pride le do enega samega prenosa podatkov (torej od dokazovalca k preverjevalcu), pravimo **neinteraktivni dokaz brez razkritja znanja** (angl. *non-interactive zero-knowledge proof*). Ker preverjevalec pri takem dokazu brez razkritja znanja ne sodeluje aktivno, je mogoče isti dokaz brez razkritja znanja uporabiti večkrat. Takim dokazom brez razkritja znanja tako včasih pravimo tudi **podpisi znanja** (angl. *signatures of knowledge*), saj lahko, namesto da bi ga dokazovalec poslal preverjevalcu, dokaz brez razkritja znanja objavi in tako ga lahko preveri kdorkoli.

Neinteraktivni dokazi brez razkritja znanja niso mogoči v standardnem modelu [16], zato jih podajamo v modelu s skupnim referenčnim nizom (glej 3.1). Zahtevi, da je skupni referenčni niz izbran neodvisno od vseh vpletenih, se lahko približamo tako, da zanj vzamemo kar sporočilo, ki ga podpisujemo. Tako lahko tudi kakršenkoli digitalni podpis razumemo kot neinteraktivni dokaz brez razkritja znanja o poznavanju zasebnega ključa.

Definicija 3.5.1. Naj bodo x_i, $i = 1, 2, \ldots, n$ vrednosti, ki jih poznata tako dokazovalec kot preverjevalec, in m skupni referenčni niz. Neinteraktivni dokaz brez razkritja znanja π, da dokazovalec pozna vrednosti α_j, $j = 1, 2, \ldots, t$, za katere je predikat

$$P(x_1, x_2, \ldots, x_n, \alpha_1, \alpha_2, \ldots, \alpha_t)$$

resničen, označimo kot:

$$\pi = \mathrm{SPK}\left\{(\alpha_1, \alpha_2, \ldots, \alpha_t) \mid P(x_1, x_2, \ldots, x_n, \alpha_1, \alpha_2, \ldots, \alpha_t)\right\}(m)\,.$$

Opomba 3.5.2. V neinteraktivnih dokazih brez razkritja znanja bomo uporabljali grške črke za vrednosti, katerih poznavanje želi dokazovalec pokazati. Z latinskimi črkami bomo označevali vrednosti, ki so znane tako dokazovalcu, kot tudi preverjevalcu.

Opisali bomo nekaj primerov neinteraktivnih dokazov brez razkritja znanja, uporabljenih pri shemah za skupinske podpise, predstavljenih v naslednjem poglavju. Pri vseh primerih predpostavljamo, da imamo na voljo zgoščevalno funkcijo $\mathcal{H} : \{0,1\}^* \to \{0,1\}^k$. Dolžino reda grupe G, v kateri računamo, označimo z ℓ_g, z $\epsilon \in \mathbb{R}$ pa označimo varnostni parameter, za katerega velja $\epsilon > 1$.

Če želi preverjevalec preveriti, da je neinteraktiven dokaz brez razkritja znanja veljaven, preveri, ali njegove komponente ustrezajo definiciji posameznega dokaza.

Dokazovanje poznavanja diskretnega logaritma.

Trditev 3.5.3. *Naj bodo $g, y \in G$, $s \in [(-2^{\ell_g+k})..(2^{\epsilon(\ell_g+k)})]$ in $c \in \{0,1\}^k$. Če velja $c = \mathcal{H}(g\|y\|g^s y^c\|m)$, je (c,s) dokaz brez razkritja znanja, da izdajatelj dokaza pozna tak α, da velja $\alpha = \log_g y$, torej:*

$$(c,s) = \mathrm{SPK}\left\{(\alpha) \mid y = g^\alpha\right\}(m)\,.$$

Računanje dokaza brez razkritja znanja (c,s):

1. izberi $r \in_R \{0,1\}^{\epsilon(\ell_g+k)}$,
2. izračunaj $c := \mathcal{H}(g\|y\|g^r\|m)$,
3. izračunaj $s := r - c\alpha$ v $\mathbb{Z}$.

Dokaz. Velja:

$$g^s y^c = g^{r-c\alpha} g^{c\alpha} = g^r,$$

zato res velja $c = \mathcal{H}(g\|y\|g^s y^c\|m)$ in dokaz brez razkritja znanja je tako poln.

Po predpostavki iz modela z naključnim orakljem je zgoščevalna funkcija $\mathcal{H}$ ekvivalentna naključni funkciji, tako da vrednosti c ni mogoče dobiti iz izbrane vrednosti s. Pri obratnem pristopu pa je očitno s mogoče izračunati le ob poznavanju α. Dokaz brez razkritja znanja je tako uglašen.

Sestavimo še simulator za dokaz brez razkritja znanja. Ta izbere $c' \in_R \{0,1\}^k$ in $s' \in_R \{0,1\}^{\epsilon(\ell_g+k)}$ ter izračuna $t' := g^{s'} y^{c'}$. Zadostuje pokazati, da je razlika med verjetnostnima porazdelitvama vrednosti s in s' zanemarljiva. Velja:

$$\Pr(s=x) \begin{cases} = 0; & s < (2^k-1)(2^{\ell_g}-1) \\ \le 2^{-\epsilon(\ell_g+k)}; & (2^k-1)(2^{\ell_g}-1) \le s < 0 \\ = 2^{-\epsilon(\ell_g+k)}; & 0 \le s \le 2^{\epsilon(\ell_g+k)} - (2^k-1)(2^{\ell_g}-1) \\ \le 2^{-\epsilon(\ell_g+k)}; & 2^{\epsilon(\ell_g+k)} - (2^k-1)(2^{\ell_g}-1) < s \le 2^{\epsilon(\ell_g+k)} - 1 \\ = 0; & 2^{\epsilon(\ell_g+k)} - 1 < s \end{cases}$$

Tako lahko izračunamo:

$$\sum_{x\in\mathbb{Z}} |\Pr(s=x)-\Pr(s'=x)| \leq \frac{2(2^k-1)(2^{\ell_g}-1)}{2^{\epsilon(\ell_g+k)}} \leq \frac{2^{\ell_g+k+1}}{2^{\epsilon(\ell_g+k)}} = \frac{2}{2^{(\epsilon-1)(\ell_g+k)}} < \varepsilon$$

Verjetnostni porazdelitvi vrednosti s in s' sta torej zanemarljivo različni in tako računsko nerazločljivi. Posledično to velja tudi za t in t', medtem ko sta verjetnostni porazdelitvi c in c' kar enaki. Pri dokazu brez razkritja znanja torej res ne izdamo nobene druge informacije. □

Dokaz brez razkritja znanja je mogoče posplošiti na dokazovanje poznavanja takih $\alpha_1, \ldots, \alpha_n$, da velja $y = \prod_{i=1}^{n} g_i^{\alpha_i}$. V tem primeru namesto enega izberemo n števil $r_1, \ldots, r_n \in_R \{0,1\}^{\epsilon(\ell_g+k)}$, v c vključimo vse g_i, $i = 1, \ldots, n$, namesto s pa imamo $s_i = r_i - c\alpha_i$ za $i = 1, \ldots, n$.

Dokazovanje poznavanja kvocienta dveh diskretnih logaritmov.

Trditev 3.5.4. *Naj bodo $g, h, y_1, y_2 \in G$, $s_1, s_2 \in [(-2^{\ell_g+k})..(2^{\epsilon(\ell_g+k)})]$ in $c \in \{0,1\}^k$. Če velja $c = \mathcal{H}(g\|h\|y_1\|y_2\|y_1^c g^{s_1}\|y_2^c h^{s_2}\|m)$, je (c, s_1, s_2) dokaz brez razkritja znanja, da izdajatelj pozna taka α in β, da velja $\beta = \log_h y_2$ in $\alpha = \beta(\log_g y_1)^{-1}$, torej:*

$$(c, s_1, s_2) = \mathrm{SPK}\left\{(\alpha,\beta) \;\middle|\; y_1^\alpha = g^\beta \wedge y_2 = h^\beta\right\}(m)\,.$$

□

Računanje dokaza brez razkritja znanja (c, s_1, s_2):

1. izberi $r_1, r_2 \in_R \{0,1\}^{\epsilon(\ell_g+k)}$,
2. izračunaj $c := \mathcal{H}(g\|h\|y_1\|y_2\|g^{r_1}\|h^{r_2}\|m)$,
3. izračunaj $z := \beta\alpha^{-1}$ v G,
4. izračunaj $s_1 := r_1 - cz$ v $\mathbb{Z}$,
5. izračunaj $s_2 := r_2 - c\beta$ v $\mathbb{Z}$.

Dokaz izpustimo, saj pri njem postopamo na enak način, kot pri prejšnjem primeru.

V posebnem primeru, ko je znano, da je $\alpha = 1$ in dokazujemo le ekvivalenco dveh diskretnih logaritmov, lahko uporabimo $r_1 = r_2 = r$ in tako $s_1 = s_2 = s$. Dokaz brez razkritja znanja je tedaj (c, s).

Dokazovanje intervala, v katerem leži diskretni logaritem.

Trditev 3.5.5. *Naj bodo* $g, y \in G$, $a, b \in \mathbb{N}$, $s \in [(-2^{b+k})..(2^{\epsilon(b+k)})]$ *in* $c \in \{0,1\}^k$ *taki, da velja* $\epsilon(b+k) < a < \ell_g$. *Če velja* $c = \mathcal{H}(g\|y\|g^{s-c2^a}y^c\|m)$, *je* (c,s) *dokaz brez razkritja znanja, da izdajatelj pozna tak* α *na intervalu* $[(2^a - 2^{\epsilon(b+k)+1} + 1)..(2^a + 2^{\epsilon(b+k)+1} - 1)]$, *da velja* $\alpha = \log_g y$, *torej:*

$$(c,s) = \mathrm{SPK}\left\{(\alpha) \;\middle|\; y = g^\alpha \wedge 2^a - 2^{\epsilon(b+k)+1} < \alpha < 2^a + 2^{\epsilon(b+k)+1}\right\}(m)\,.$$

□

Računanje dokaza brez razkritja znanja (c,s), če je $\alpha \in [(2^a)..(2^a+2^b)]$:
1. izberi $r \in_R \{0,1\}^{\epsilon(b+k)}$,
2. izračunaj $c := \mathcal{H}(g\|y\|g^r\|m)$,
3. izračunaj $s := r - c(\alpha - 2^a)$ v $\mathbb{Z}$.

Celoten dokaz je tudi tukaj podoben kot v prejšnjih primerih. Prepričajmo se le v polnost dokaza brez razkritja znanja, ko velja $\alpha \in [(2^a)..(2^a+2^b)]$. Tedaj velja $0 \le c(\alpha - 2^a) < 2^{b+k}$, torej s res leži v zahtevanem intervalu. Preverjevalec se ne more prepričati, ali α res leži v $[(2^a)..(2^a+2^b)]$, saj morda obstajajo take vrednosti izven tega intervala, da bo s ležal v zahtevanem intervalu. Lahko pa se prepriča, da mora veljati $2^a - 2^{\epsilon(b+k)+1} < \alpha < 2^a + 2^{\epsilon(b+k)+1}$.

Če velja

$$\alpha \le 2^a - 2^{\epsilon(b+k)+1},$$

je

$$s \ge r + c2^{\epsilon(b+k)+1}.$$

Če velja $c > 0$, je torej

$$s \ge 2^{\epsilon(b+k)+1} > 2^{\epsilon(b+k)}$$

in zato (c,s) ni veljaven dokaz. Podobno pokažemo, če je

$$\alpha \ge 2^a + 2^{\epsilon(b+k)+1}.$$

Tedaj je

$$s \le r - c2^{\epsilon(b+k)+1}.$$

Pri $c > 0$ velja

$$s < 2^k - 2^{\epsilon(b+k)+1} < -2^{\epsilon(b+k)} < -2^{(b+k)}$$

in tako (c, s) ni veljaven dokaz.

Zahtevi, da je $c > 0$, lahko ugodimo tako, da dokaze (c, s), kjer je $c = 0$, zavrnemo. Lahko pa to možnost tudi zanemarimo, saj je verjetnost, da je $c = 0$, enaka 2^{-k} in je torej zanemarljiva.

Dokazovanje, da je število produkt dveh praštevil.

Trditev 3.5.6. *Naj bodo $n \in \mathbb{N}$ in $y, r, x \in \mathbb{Z}_n^*$. Če velja $y^n \equiv x \pmod{n}$ in $r^2 \bmod n \in \{\pm x \bmod n, \pm 2x \bmod n\}$, je (y, r) dokaz brez razkritja znanja, da izdajatelj pozna taka $\alpha, \beta \in \mathbb{P}$, da velja $n = \alpha\beta$, $\alpha, \beta \not\equiv 1 \pmod 8$, $\alpha \not\equiv \beta \pmod 8$, torej:*

$$(y, r) = \mathrm{SPK}\left\{(\alpha, \beta) \mid n = \alpha\beta \ \wedge \ \alpha, \beta \in \mathbb{P} \wedge \right.$$

$$\left. \wedge\, \alpha, \beta \not\equiv 1 \,(\mathrm{mod}8) \ \wedge \ \alpha \not\equiv \beta \,(\mathrm{mod}8)\right\}(x)\,.$$

Računanje dokaza brez razkritja znanja (y, r)
1. izračunaj $m := n^{-1} \bmod (\alpha - 1)(\beta - 1)$,
2. izračunaj $y := x^m \bmod n$,
3. izračunaj $r := \sqrt{s} \bmod n$ za $s \in \{\pm x, \pm 2x\}$.

Pokažimo, čemu služita vrednosti y in r v dokazu brez razkritja znanja.

Trditev 3.5.7. *Vrednost y, ki ustreza definiciji, je dokaz brez razkritja znanja, da izdajatelj pozna praštevilski razcep n, ki je produkt samih različnih praštevil.*

Dokaz. Če izdajatelj pozna praštevilski razcep n, potem lahko izračuna $m = n^{-1} \bmod \varphi(n)$ in tako velja $y^n \equiv x^{mn} \equiv x \bmod n$. Dokaz brez razkritja znanja je torej poln.

Naj bo $d = \gcd(n, \varphi(n))$. Če obstaja tak $p \in \mathbb{P}$, da p^2 deli n, potem velja $d > 1$ in m tedaj ni mogoče izračunati. Tako je verjetnost, da velja $x \equiv y^n \pmod n$ za nek $y \in \mathbb{Z}_n^*$, največ $\frac{1}{d}$ in tako zanemarljiva glede na red velikosti najmanjšega prafaktorja števila n. Dokaz brez razkritja znanja je tako uglašen.

Sestavimo še simulator. Ta izbere $y' \in_R \mathbb{Z}_n^*$ in izračuna $x' := y'^n$. Vrednosti x' in y' sta enakomerno porazdeljeni na $\mathbb{Z}_n^*$, kar velja tudi za x in y. Pri dokazu brez razkritja znanja tako dokazovalec ne izda nobene druge informacije. □

Trditev 3.5.8. *Vrednost r, ki ustreza definiciji, je dokaz brez razkritja znanja, da je n produkt dveh potenc praštevil.*

Dokaz. Ker velja

$$\left(\frac{-1}{\alpha}\right) = 1 \Leftrightarrow \alpha \equiv 1 \pmod 4,$$

$$\left(\frac{2}{\alpha}\right) = 1 \Leftrightarrow \alpha \equiv \pm 1 \pmod 8$$

in podobno za β, je pri

$$\alpha, \beta \not\equiv 1 \,(\mathrm{mod}8) \ \wedge \ \alpha \not\equiv \beta \,(\mathrm{mod}8) \tag{3.5.1}$$

natanko eden od $\pm x$, $\pm 2x$ kvadratni ostanek in izdajatelj dokaza lahko zanj izračuna kvadratni koren r. Tako velja

$$r^2 \bmod n \in \{\pm x \bmod n, \pm 2x \bmod n\}$$

in dokaz brez razkritja znanja je poln.

Če je n produkt več kot dveh različnih praštevil, potem je naključni element $\mathbb{Z}_n^*$ kvadratni ostanek z verjetnostjo največ $\frac{1}{8}$, torej je v množici kvadratni ostanek z verjetnostjo največ $\frac{1}{2}$. Če pa velja ne velja pogoj 3.5.1, pa je eden od -1, 2 in -2 kvadratni ostanek in tako je v množici kak kvadratni ostanek z verjetnostjo $\frac{1}{2}$. Če velja

$$\alpha \equiv \beta \equiv 1 \pmod 8,$$

so -1, 2 in -2 vsi kvadratni ostanki in zato je verjetnost celo le $\frac{1}{4}$. S t-kratnim ponavljanjem dokaza brez razkritja znanja je tako verjetnost prevare manjša od $\frac{1}{2^t}$ in tako zanemarljiva v t. Dokaz brez razkritja znanja je torej uglašen.

Sestavimo še simulator. Ta izbere $r' \in_R \mathbb{Z}_n^*$ in $b \in_R \{\pm 1, \pm 2\}$ ter izračuna $x' := r'^2 b^{-1} \pmod n$. Vrednosti x' in r' sta enakomerno porazdeljeni na $\mathbb{Z}_n^*$, kar velja tudi za x in r. Pri dokazu brez razkritja znanja tako dokazovalec ne izda nobene druge informacije. □

Za računanje kvadratnega korena po praštevilskem modulu p obstajata učinkovita deterministična algoritma za primera, ko velja $p \equiv 3 \pmod 4$ ali $p \equiv 5 \pmod 8$ [20]. Tako je mogoče izračunati tudi kvadratni koren po modulu n za vse take n, za katere lahko izdamo dokaz brez razkritja znanja. Izdajatelj izračuna kvadratni koren za tisto število $z \in \{\pm x, \pm 2x\}$, za katerega velja $\left(\frac{z}{\alpha}\right) = \left(\frac{z}{\beta}\right) = 1$.

Dokaz brez razkritja znanja je mogoče razširiti na dokaz, da je število n produkt dveh kvazi-varnih (tipa $p = 2q^e + 1$, $p, q \in \mathbb{P}$, $e \in \mathbb{N}$) [15] oziroma varnih praštevil (tipa $p = 2q + 1$, $p, q \in \mathbb{P}$, $e \in \mathbb{N}$) [9].

Poglavje 4

Primeri shem za skupinske podpise

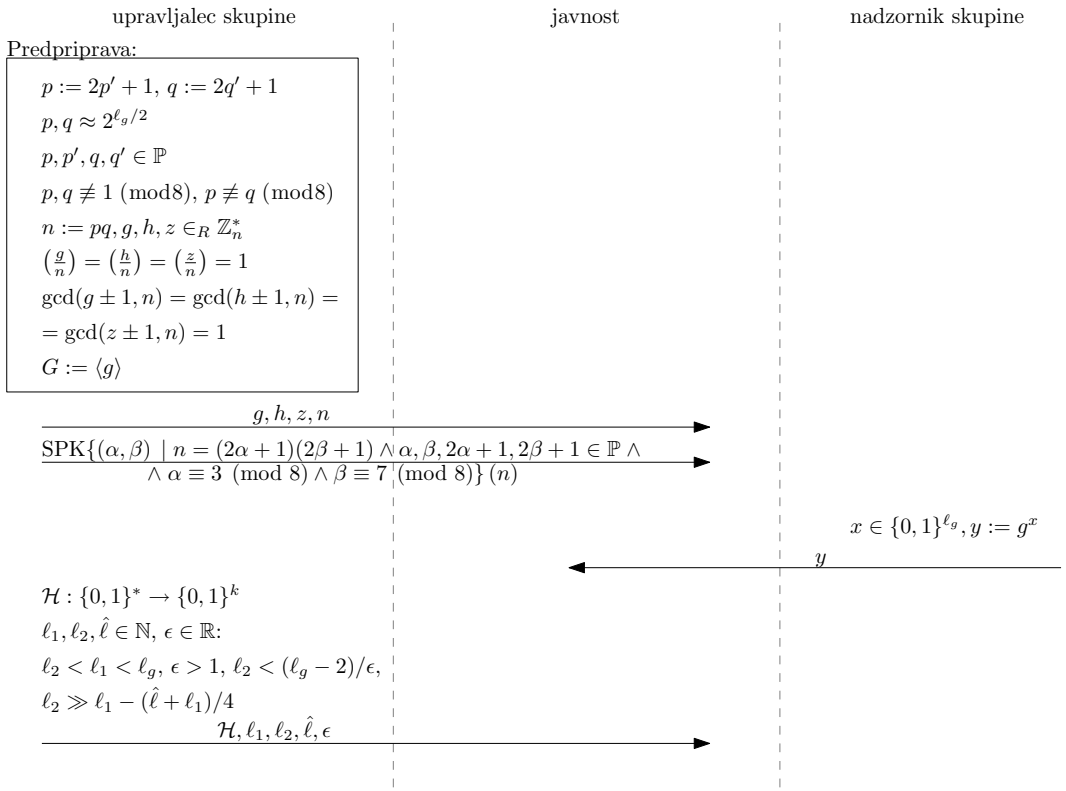
Idealna shema za skupinske podpise bi morala biti funkcionalna, varna in učinkovita. Izkaže se, da si te zahteve pogosto nasprotujejo med seboj, kar je morda največja ovira za splošnejšo uveljavitev skupinskih podpisov.

Tako je potrebno pri načrtovanju shem sklepati kompromise. Predstavili bomo dve shemi za skupinske podpise: Camenisch-Michelsovo in Zhou-Linovo shemo. Prva je dinamična shema, vendar brez možnosti odstranjevanja članov iz skupine, njena velika pomanjkljivost pa je še velikost podpisa. Druga je v osnovi statična shema, ki pa omogoča odstranjevanje članov iz skupine, a pri tem žrtvuje nekaj varnosti, je pa zato zelo učinkovita.

4.1 Camenisch-Michelsova shema

Camenisch-Michelsova shema, predstavljena v [8], temelji na varianti krepke RSA predpostavke, problemu diskretnega logaritma in Diffie-Hellmanovi odločitveni predpostavki. Je prva učinkovita shema za skupinske podpise, za katero je bilo mogoče dokazati, da je odporna proti koalicijam. Izboljšava Camenisch-Michelsove sheme je sledila v [1].

Poleg nadzornika skupine, ki skrbi za odpiranje podpisov, ima pri tej shemi posebno vlogo še **upravljalec skupine**. Ta skrbi za vzpostavitev skupine in dodajanje članov vanjo. Pri vzpostavitvi upravljalec skupine izbere grupo $G = \langle g \rangle$ ter taka naključna elementa grupe z in h, tako da sta istega reda kot g. Ta red je približno 2^{ℓ_g} in ne sme biti praštevilo, kar mora upravljalec skupine tudi dokazati. V grupi G mora biti problem diskretnega logaritma težek, prav tako mora zanjo veljati krepka Diffie-Hellmanova predpostavka.



Slika 4.1: Protokol za vzpostavitev Camenisch-Michelsove sheme, če je G podgrupa v $\mathbb{Z}_n^*$

Možna izbira grupe G je podgrupa v $\mathbb{Z}_n^*$, tako da velja $n = pq$, kjer sta p in q varni praštevili, torej velja:

$$p = 2p' + 1,\ q = 2q' + 1,$$

$$p, p', q, q' \in \mathbb{P}.$$

Vrednosti p in q sta približno $2^{\ell_g/2}$, poleg tega mora veljati še:

$$p, q \not\equiv 1 \pmod 8,\ p \not\equiv q \pmod 8,$$

$$\left(\frac{g}{n}\right) = 1,$$

tako da za grupo velja Diffie-Hellmanova odločitvena predpostavka. Nadzornik skupine objavi n, s čimer opiše grupo in tako pokaže tudi njen približen red. Za dokaz, da je n res produkt dveh varnih praštevil, se lahko uporabi metoda

iz [9]. Da so redi elementov g, z in h vsaj $p'q'$, lahko vsakdo preveri tako, da preveri:

$$g \not\equiv \pm 1 \pmod{n},$$
$$\gcd(g \pm 1, n) = 1,\ \left(\frac{g}{n}\right) = 1$$

in podobno še za z, h. Ker grupa $\mathbb{Z}_n^*$ ni ciklična, je maksimalen red elementa $2p'q'$, torej imajo kvadratni ostanki red največ $p'q'$.

Ko se izbere grupa in dokaže njena primernost, nadzornik skupine naključno izbere $x \in [0..(2^{\ell_g} - 1)]$, ki bo služil kot tajni ključ za odpiranje podpisov. Nato izračuna $y = g^x$ in ga objavi kot javni ključ skupine. Nazadnje upravljalec skupine izbere še zgoščevalno funkcijo $\mathcal{H} : \{0,1\}^* \to \{0,1\}^k$ in varnostne parametre ℓ_1, ℓ_2, $\hat{\ell}$, ϵ, tako da velja:

$$\ell_2 < \ell_1 < \ell_g,\ \epsilon > 1,$$
$$\ell_2 < (\ell_g - 2)/\epsilon,\ \ell_2 \gg (\hat{\ell} + \ell_1)/4.$$

Celoten protokol za vzpostavitev skupine, ko za G izberemo podgrupo v $\mathbb{Z}_n^*$, je prikazan na sliki 4.1.

Pridruževanje skupini

Denimo, da se želi Anita pridružiti skupini. Tedaj izbere praštevili:

$$\hat{e} \in_R [(2^{\hat{\ell}-1})..(2^{\hat{\ell}} - 1)] \cap \mathbb{P},\ e \in_R [(2^{\ell_1})..(2^{\ell_1} + 2^{\ell_2} - 1)] \cap \mathbb{P}$$
$$\hat{e}, e \not\equiv 1 \ (\mathrm{mod}8), \hat{e} \not\equiv e \ (\mathrm{mod}8).$$

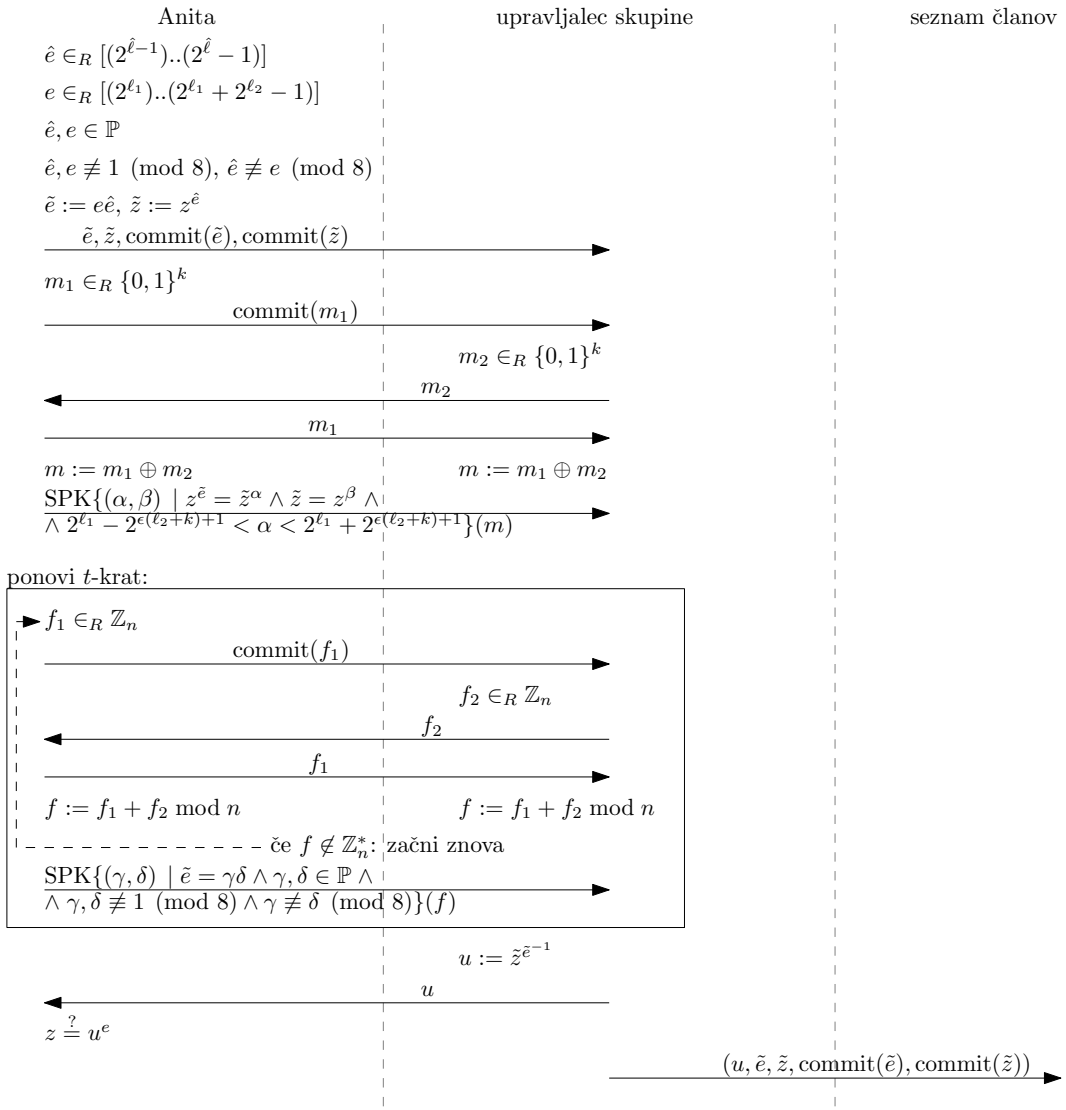
Nato izračuna $\tilde{e} := e\hat{e}$ in $\tilde{z} := z^{\hat{e}}$ in zapriseže vrednosti $\tilde{e}$ in $\tilde{z}$. To lahko stori tako, da ju podpiše z zasebnim ključem iz običajne sheme za digitalne podpise, ki je neodvisna od sheme za skupinske podpise, kateri skupini se Anita pridružuje. Vrednosti $\tilde{e}$, $\tilde{z}$ in zaprisegi nato pošlje upravljalcu skupine. Da dokaže pravilnost poslanih podatkov, Anita in upravljalec skupine izvedeta protokol, ki ustreza sledečima dokazoma brez razkritja znanja:

$$\mathrm{SPK}\left\{(\alpha, \beta) \ \middle|\ z^{\tilde{e}} = \tilde{z}^{\alpha} \wedge \tilde{z} = z^{\beta} \wedge 2^{\ell_1} - 2^{\epsilon(\ell_2+k)+1} < \alpha < 2^{\ell_1} + 2^{\epsilon(\ell_2+k)+1}\right\}(m),$$

$$\mathrm{SPK}\left\{(\gamma, \delta) \mid \tilde{e} = \gamma\delta \wedge \gamma, \delta \in \mathbb{P} \wedge \gamma, \delta \not\equiv 1 \ (\mathrm{mod}8) \wedge \gamma \not\equiv \delta \ (\mathrm{mod}8)\right\}(f).$$

Da bi spremenili neinteraktivni dokaz brez razkritja znanja v interaktivnega, Anita lahko prepusti izbiro m in f upravljalcu skupine, ali pa ju izbereta skupaj, tako da eden od njiju najprej naključno izbere

$$m_1 \in \{0,1\}^k \text{ oziroma } f_1 \in \mathbb{Z}_n$$



Slika 4.2: Protokol za pridruževanje skupini, če je G podgrupa $\mathbb{Z}_n^*$

in pošlje drugemu zaprisego te vrednosti, nato pa drugi izbere

$$m_2 \in \{0,1\}^k \text{ oziroma } f_2 \in \mathbb{Z}_n.$$

Za tem se razkrije vrednost m_1 oziroma f_1 in se izračuna

$$m := m_1 \oplus m_2 \text{ oziroma } f := f_1 + f_2 \bmod n.$$

Pri tem mora veljati $f \in \mathbb{Z}_n^*$, tako da se lahko zgodi, da je treba izbiro f ponoviti. Sicer pa je potrebno sam protokol za dokazovanje, da je n produkt dveh praštevil, večkrat ponoviti, saj lahko pri posamezni ponovitvi verjetnost prevare doseže $\frac{1}{2}$. Če protokol ponovimo t-krat, je verjetnost prevare kvečjemu $1 - 2^{-t}$.

Ko se upravljalec skupine prepriča, da je Anita izbrala $\tilde{e}$ in $\tilde{z}$ na pravilen način, izračuna Anitin javni certifikat članstva $u := \tilde{z}^{\tilde{e}^{-1}}$. Anita preveri, da velja $\tilde{z} = u^{\tilde{e}}$, kar je ekvivalentno $z = u^e$. Upravljalec skupine doda $(u, \tilde{e}, \tilde{z})$ in zaprisegi na seznam članov skupine, Anita pa shrani e kot svoj zasebni ključ za podpisovanje v imenu skupine.

Celoten protokol za pridruževanje skupini, ko za G izberemo podgrupo v $\mathbb{Z}_n^*$, je prikazan na sliki 4.2.

Podpisovanje in preverjanje podpisa

Skupinski podpis sporočila m po Camenisch-Michelsovi shemi je neinteraktiven dokaz brez razkritja znanja:

$$(c, s_1, s_2, s_3, a, b, d) = \mathrm{SPK}\{(\eta, \theta, \xi) \mid zy^\theta = b^\eta \ \wedge \ a^\eta = g^\theta \ \wedge \ a = g^\xi \wedge$$

$$\wedge\, d = g^\eta h^\xi \ \wedge \ 2^{\ell_1} - 2^{\epsilon(\ell_2+k)+1} < \eta < 2^{\ell_1} + 2^{\epsilon(\ell_2+k)+1}\}\,(m)\,.$$

Podpisovanje sporočila m v imenu skupine:

1. izberi $w \in_R \{0,1\}^{\ell_g}$,
2. izračunaj $a := g^w$, $b := uy^w$ in $d := g^e h^w$,
3. izberi $r_1 \in_R \{0,1\}^{\epsilon(\ell_2+k)}$, $r_2 \in_R \{0,1\}^{\epsilon(\ell_g+\ell_1+k)}$ in $r_3 \in_R \{0,1\}^{\epsilon(\ell_g+k)}$,
4. izračunaj $t_1 := b^{r_1} y^{-r_2}$, $t_2 := a^{r_1} g^{-r_2}$, $t_3 := g^{r_3}$, $t_4 := g^{r_1} h^{r_3}$,
5. izračunaj $c := \mathcal{H}(g\|h\|y\|z\|a\|b\|d\|t_1\|t_2\|t_3\|t_4\|m)$,
6. izračunaj v $\mathbb{Z}$:

$$s_1 := r_1 - c(e - 2^{\ell_1}), \qquad s_2 := r_2 - cew,$$
$$s_3 := r_3 - cw,$$

7. izdaj podpis $\sigma := (c, s_1, s_2, s_3, a, b, d)$.

Preverjanje podpisa $\sigma = (c, s_1, s_2, s_3, a, b, d)$ sporočila m:

1. preveri:

$$\begin{aligned} c &\stackrel{?}{\in} \{0,1\}^k \\ s_1 &\stackrel{?}{\in} [(-2^{\ell_2+k})..(2^{\epsilon(\ell_2+k)})], \\ s_2 &\stackrel{?}{\in} [(-2^{\ell_g+\ell_1+k})..(2^{\epsilon(\ell_g+\ell_1+k)})], \\ s_3 &\stackrel{?}{\in} [(-2^{\ell_g+k})..(2^{\epsilon(\ell_g+k)})], \\ a, b, d &\stackrel{?}{\in} G. \end{aligned}$$

2. izračunaj:

$$\begin{aligned} t_1' &:= z^c b^{s_1 - c2^{\ell_1}} y^{-s_2}, & t_2' &:= a^{s_1 - c2^{\ell_1}} g^{-s2}, \\ t_3' &:= a^c g^{s_3}, & t_4' &:= d^c g^{s_1 - c2^{\ell_1}} h^{s_3}, \end{aligned}$$

3. preveri $c \stackrel{?}{=} \mathcal{H}(g\|h\|y\|z\|a\|b\|d\|t_1'\|t_2'\|t_3'\|t_4'\|m)$.

Trditev 4.1.1. *Preverjanje podpisa je pravilno izvedeno.*

Dokaz. Ob predpostavki, da je zgoščevalna funkcija $\mathcal{H}$ brez trkov, zadošča preveriti, da velja $t_i = t_i'$ za $i = 1, 2, 3, 4$:

$$t_1' = z^c b^{s_1 - c2^{\ell_1}} y^{-s_2} = z^c b^{r_1 - ce} y^{-r_2 + cew} = u^{ce} b^{r_1} u^{-ce} y^{-cew} y^{-r_2 + cew} = b^{r_1} y^{-r_2} = t_1$$
$$t_2' = a^{s_1 - c2^{\ell_1}} g^{-s2} = a^{r_1 - ce} g^{-r_2 + cew} = a^{r_1} g^{cew} g^{-r_2 + cew} = a^{r_1} g^{-r_2} = t_2$$
$$t_3' = a^c g^{s_3} = g^{cw} g^{r_3 - cw} = g^{r_3} = t_3$$
$$t_4' = d^c g^{s_1 - c2^{\ell_1}} h^{s_3} = g^{ce} h^{cw} g^{r_1 - ce} h^{r_3 - cw} = g^{r_1} h^{r_3} = t_4$$

□

Odpiranje podpisa

Da bi nadzornik skupine razkril podpisnika veljavnega skupinskega podpisa $\sigma = (c, s_1, s_2, s_3, a, b, d)$, izračuna $u' := ba^{-x}$, najde člana skupine s certifikatom članstva u' ter razkrije njegovo identiteto, $\tilde{e}$, $\tilde{z}$ in zaprisegi zanju. Kot dokaz, da je razkritje podpisnika pravilno, izda še podpis znanja:

$$P = \mathrm{SPK}\left\{(\alpha) \;\middle|\; y = g^\alpha \wedge bu'^{-1} = a^\alpha \right\}(u'\|\sigma\|m)\,.$$

Trditev 4.1.2. *Razkritje certifikata članstva je pravilno.*

Dokaz. Pokažimo, da velja $u' = u$:

$$u' = ba^{-x} = uy^w g^{-xw} = uy^w y^{-w} = u.$$

□

Boris se s preverjanjem podpisa znanja prepriča, da ga je izdal nekdo, ki pozna x, torej nadzornik skupine, in da je razkriti certifikat članstva pravi. Identiteto podpisnika Boris preveri tako, da preveri, ali je zaprisegi za $\tilde{e}$ in $\tilde{z}$ izdal razkriti podpisnik, pri čemer mora veljati $\tilde{z} = u'^{\tilde{e}}$.

Varnost sheme

Pogledali si bomo varnostno analizo sheme glede na formalizirane varnostne zahteve. Pri Camenisch-Michelsovi shemi bomo dokaze varnostnih lastnosti le skicirali. Za bolj točne dokaze bi morali uporabiti nekatere tehnike, ki jih bomo videli pri dokazu varnostnih lastnosti Zhou-Linove sheme.

Trditev 4.1.3. *Camenisch-Michelsova shema je anonimna, če velja Diffie-Hellmanova odločitvena predpostavka.*

Dokaz. Denimo, da prednost $\mathrm{Adv}_N^{\mathrm{anon}}$ polinomskega nasprotnika N ni zanemarljiva, kar je ekvivalentno zahtevi, da verjetnost, da poskus anon_N^b pri $b \in \{0, 1\}$ uspe, ni zanemarljivo različna od $\frac{1}{2}$. Pri poskusu N izbere člana u_0 in u_1 skupine U, nato pa mu je dan podpis σ, ki ga izda član u_b. Poskus uspe, če je član u', za katerega N trdi, da je podpisnik, res podpisnik sporočila.

Po predpostavki iz modela z naključnim orakljem je zgoščevalna funkcija $\mathcal{H}$ ekvivalentna naključni funkciji. Ker so tudi r_1, r_2 in r_3 izbrani naključno, nasprotnik N iz delov podpisa c, s_1, s_2 in s_3 ne more dobiti nobene koristne informacije. Ker je $u' = u_b$ z verjetnostjo, ki ni zanemarljivo različna od $\frac{1}{2}$, tudi verjetnost, da je odločitev, ali velja

$$\log_g a = \log_y(bu'^{-1}) = \log_h(dg^{-e}),$$

ni zanemarljivo različna od $\frac{1}{2}$, kar je v nasprotju z Diffie-Hellmanovo odločitveno predpostavko. Camenisch-Michelsova shema je torej anonimna. □

Trditev 4.1.4. *Camenisch-Michelsova shema je sledljiva, če velja krepka RSA predpostavka.*

Dokaz. Pri poskusu trace_N je cilj nasprotnika N izdelati tak veljaven skupinski podpis, za katerega ni mogoče ugotoviti podpisnika oziroma ugotovitve ni mogoče dokazati. To lahko doseže le tako, da ponaredi zasebni ključ e in ustrezni certifikat članstva u, ne da bi izvedel protokol za pridruževanje skupini – veljaven podpis je namreč dokaz, da podpisnik taki števili pozna.

Za ponarejena e in u mora veljati $z = u^e$, toda računanje takega para je po krepki RSA predpostavki računsko nedosegljivo. Shema je torej sledljiva. □

Trditev 4.1.5. *Camenisch-Michelsova shema je nepodtakljiva, če velja krepka RSA predpostavka.*

Dokaz. Podobno kot pri trace_N je tudi pri nf_N cilj nasprotnika N ponarediti veljaven skupinski podpis, le da mora tukaj znati ponarediti še dokaz, da je podpisnik član skupine, za katerega N ne poseduje zasebnega ključa za izdajanje skupinskih podpisov.

Tudi tukaj mora torej N poznati taka e in u, da velja $z = u^e$ in po krepki RSA predpostavki je računanje takega para računsko nedosegljivo, tako da si s spreminjanjem podatkov o članih N lahko pridobi le zanemarljivo prednost. Prav tako je za znan u po predpostavki o težkosti problema diskretnega logaritma računsko nedosegljivo računanje takega e, da velja $z = u^e$. Shema je torej nepodtakljiva. □

Opombe

Predlagana izbira varnostnih parametrov v [8] je:

$$\epsilon = \frac{9}{8},\ \ell_g = \hat{\ell} = 1200,\ \ell_1 = 860,\ \ell_2 = 600,\ k = 160.$$

Pri teh parametrih potrebujeta podpisovanje in preverjanje podpisa nekaj manj kot 13000 množenj po 1200-bitnem modulu, podpis pa je dolg nekaj več kot 1 kB.

Slabost sheme je, da ne omogoča brisanja članov iz skupine. Zaradi polne anonimnosti sheme tudi naiven pristop s preklicnim seznamom ne deluje.

4.2 Zhou-Linova shema

Zhou-Linova shema, predstavljena v [28], je v osnovi statična shema, ki pa omogoča brisanje članov iz skupine. Za dosego tega cilja uporablja preklicni seznam, pri čemer se mora preverjevalec podpisa za vsak vnos v preklicnem

seznamu prepričati, da ne pripada podpisniku. Takemu pristopu pravimo **preklic pri preverjevalcu** (angl. *verifier-local revocation*).

Zhou-Linova shema je izboljšava sheme iz [7], ki prav tako uporablja preklic pri preverjevalcu, vendar pa ima to pomanjkljivost, da z odstranjevanjem člana iz skupine vsi njegovi dotedanji podpisi postanejo neveljavni. Posledično je mogoče odločiti, ali dva podpisa prihajata od istega člana skupine, če je ta bil dodan na preklicni seznam. Zhou-Linova shema to pomanjkljivost odpravlja, hkrati pa ohranja učinkovitost.

Shema temelji na q-krepki Diffie-Hellmanovi predpostavki in tridelni Diffie-Hellmanovi odločitveni predpostavki, poleg tega pa predpostavlja obstoj učinkovitih enosmernih bilinearnih preslikav $e : G \times G \to G$ z $e(g, g) \neq 1$ pri $G = \langle g \rangle$ – veljati mora torej $e(u^a, v^b) = e(u, v)^{ab}$ za vse $u, v \in G$, $a, b \in \mathbb{Z}$. Lahko se je prepričati, da za poljubne $u, v, w \in G$ velja $e(uv, w) = e(u, w)e(v, w)$ in $e(u, vw) = e(u, v)e(u, w)$.

V nasprotju s prejšnjo shemo tukaj nimamo ločenega upravitelja skupine, pač pa za vzpostavitev skupine skrbi kar nadzornik skupine. Ta najprej izbere grupo $G = \langle g \rangle$ z $\mathrm{ord}(g) = p \in \mathbb{P}$, element $\tilde{g} \in_R G$, enosmerno bilinearno preslikavo $e : G \times G \to G$ z $e(g, g) \neq 1$ in zgoščevalno fukcijo $\mathcal{H} : \{0, 1\}^* \to \mathbb{Z}_p^*$ brez trkov. Nato izbere $\gamma \in_R \mathbb{Z}_p$ in $h_j \in_R G$ za $j = 1, \ldots, T$, kjer je T število časovnih obdobij, ter izračuna $w := g^\gamma$. Nadzornik skupine objavi javni ključ skupine $(g, \tilde{g}, w, \{h_1, \ldots, h_T\})$. Določi se skupno število članov n ter za vsakega izmed njih nadzornik skupine izbere $x_i \in_R \mathbb{Z}_p^*$ in izračuna $A_i := g^{(\gamma + x_i)^{-1}}$ za $i = 1, \ldots, n$. Član skupine i dobi (A_i, x_i), ki mu služi kot zasebni ključ za podpisovanje v imenu skupine. Nazadnje nadzornik skupine izračuna še preklicne žetone $B_{ij} = h_j^{x_i}$ za $i = 1, \ldots, n$ in $j = 1, \ldots, T$. Če hoče nadzornik iz skupine izločiti člana i skupine U v časovnem obdobju j, doda preklicni žeton B_{ij} v preklicni seznam RL_j za časovno obdobje j.

Podpisovanje sporočila

Naj bo Anita član i skupine U. Denimo, da želi Anita podpisati sporočilo m, ki vsebuje podatek o časovnem obdobju j, v katerem je nastalo. To lahko dosežemo tako, da neodvisna avtoriteta izda certifikat s časovnim žigom [17] za sporočilo oziroma njegov izvleček. Potem je skupinski podpis sporočila m po Zhou-Linovi shemi neinteraktiven dokaz brez razkritja znanja:

$$(c, s_1, s_2, s_3, s_4, s_5, s_6, s_7, a, b, d, f, u) = \mathrm{SPK}\left\{ (\alpha, \beta, \delta, \xi, \omega) \;\middle|\; a = \omega\tilde{g}^\alpha \wedge \right.$$

$$\left. \wedge\, b = g^\alpha \tilde{g}^\beta \;\wedge\; d = h_j^{\xi\delta} \;\wedge\; f = u^\delta \;\wedge\; e(\omega, wg^\xi) = e(g, g) \right\}(m) = \qquad (4.2.1)$$

$$= \mathrm{SPK}\left\{(\alpha,\beta,\delta,\xi,\zeta,\eta,\theta)\,\middle|\, f^{\xi} = u^{\zeta} \ \wedge\ b = g^{\alpha}\tilde{g}^{\beta} \ \wedge\ d = h_j^{\zeta} \ \wedge\ f = u^{\delta} \wedge \right.$$
$$\left. \wedge\ b^{\xi} = g^{\eta}\tilde{g}^{\theta} \ \wedge\ e(a,w) = e(a^{-\xi}\tilde{g}^{\eta}, g)e(\tilde{g}^{\alpha}, w)e(g,g)\right\}(m). \qquad (4.2.2)$$

Podpisovanje sporočila m v časovnem obdobju j v imenu skupine:

1. izberi $k, \ell, q \in_R \mathbb{Z}_p^*$, $u \in_R G$,
2. izračunaj $a := A_i\tilde{g}^k$, $b := g^k\tilde{g}^{\ell}$, $d := h_j^{x_i q}$, $f := u^q$,
3. izberi $r_1, r_2, r_3, r_4, r_5, r_6, r_7 \in_R \mathbb{Z}_p^*$,
4. izračunaj:

$$\begin{aligned} t_1 &:= f^{r_1}u^{-r_2}, & t_2 &:= g^{r_3}\tilde{g}^{r_4},\\ t_3 &:= h_j^{r_2}, & t_4 &:= u^{r_5},\\ t_5 &:= b^{r_1}g^{-r_6}\tilde{g}^{-r_7}, & t_6 &:= e(a^{-r_1}\tilde{g}^{r_6}, g)e(\tilde{g}^{r_3}, w), \end{aligned}$$

5. izračunaj $c := \mathcal{H}(g\|\tilde{g}\|w\|a\|b\|d\|f\|u\|t_1\|t_2\|t_3\|t_4\|t_5\|t_6\|m)$,
6. izračunaj v $\mathbb{Z}_p$:

$$\begin{aligned} s_1 &:= r_1 - cx_i, & s_2 &:= r_2 - cx_iq,\\ s_3 &:= r_3 - ck, & s_4 &:= r_4 - c\ell,\\ s_5 &:= r_5 - cq, & s_6 &:= r_6 - cx_ik,\\ s_7 &:= r_7 - cx_i\ell, \end{aligned}$$

7. izdaj podpis $\sigma = (c, s_1, s_2, s_3, s_4, s_5, s_6, s_7, a, b, d, f, u)$.

Trditev 4.2.1. *Dokaza brez razkritja znanja v 4.2.1 in 4.2.2 sta ekvivalentna.*

Dokaz. ($\Rightarrow$) Denimo, da izdajatelj pozna take $\alpha, \beta, \delta, \xi, \omega$, ki ustrezajo dokazu brez razkritja znanja v 4.2.1. Potem pozna tudi $\zeta = \xi\delta$, $\eta = \xi\alpha$ in $\theta = \xi\beta$, da velja:

$$f = u^{\delta} \Rightarrow f^{\xi} = u^{\xi\delta} = u^{\zeta},$$
$$d = h_j^{\xi\delta} \Rightarrow d = h_j^{\zeta},$$
$$b = g^{\alpha}\tilde{g}^{\beta} \Rightarrow b^{\xi} = g^{\xi\alpha}\tilde{g}^{\xi\beta} = g^{\eta}\tilde{g}^{\theta}.$$

Dokažimo še veljavnost zadnje dokazovane enačbe:

$$e(g,g) = e(\omega, wg^{\xi}) = e(\omega, w)e(\omega, g^{\xi})$$
$$e(\omega^{-\xi}\tilde{g}^{-\xi\alpha}\tilde{g}^{\xi\alpha}, g)e(\tilde{g}^{\alpha}, w)e(g,g) = e(\omega, w)e(\tilde{g}^{\alpha}, w) = e(\omega\tilde{g}^{\alpha}, w)$$
$$e(a^{-\xi}\tilde{g}^{\eta}, g)e(\tilde{g}^{\alpha}, w)e(g,g) = e(a, w)$$

($\Leftarrow$) Denimo, da izdajatelj pozna take $\alpha, \beta, \delta, \xi, \zeta, \eta, \theta$, ki ustrezajo dokazu brez razkritja znanja v 4.2.2. Iz $f = u^{\delta}$ in $f^{\xi} = u^{\zeta}$ sledi $\zeta = \xi\delta$, iz $b = g^{\alpha}\tilde{g}^{\beta}$ in $b^{\xi} = g^{\eta}\tilde{g}^{\theta}$ pa še $\eta = \xi\alpha$ in $\delta = \xi\beta$. Izdajatelj lahko izračuna še $\omega = a\tilde{g}^{-\alpha}$, tako da velja:

$$\omega = a\tilde{g}^{-\alpha} \Rightarrow a = \omega\tilde{g}^{\alpha}$$
$$d = h_j^{\zeta} \Rightarrow d = h_j^{\xi\delta}$$
$$e(a^{-\xi}\tilde{g}^{\eta}, g)e(\tilde{g}^{\alpha}, w)e(g,g) = e(a,w) \Rightarrow$$
$$\Rightarrow e(g,g) = e(a,w)e(\tilde{g}^{-\alpha}, w)e((a\tilde{g}^{-\alpha})^{\xi}, g) = e(a\tilde{g}^{-\alpha}, w)e(a\tilde{g}^{-\alpha}, g^{\xi}) = e(\omega, wg^{\xi})$$

□

Preverjanje in odpiranje podpisa

Če želi Boris preveriti veljavnost podpisa po Zhou-Linovi shemi, to stori v dveh korakih – v prvem se prepriča, da je bil podpis izdan z obstoječim skupinskim ključem, v drugem pa, da podpisnika ni na preklicnem seznamu.

Preverjanje podpisa $\sigma = (c, s_1, s_2, s_3, s_4, s_5, s_6, s_7, a, b, d, f, u)$ sporočila m v časovnem obdobju j:

Prvi korak:

1. preveri $c \overset{?}{\in} \mathbb{Z}_p^*$, $s_1, s_2, s_3, s_4, s_5, s_6, s_7 \overset{?}{\in} \mathbb{Z}_p$, $a, b, d, f, u \overset{?}{\in} G$,
2. izračunaj:

$$t_1' := f^{s_1}u^{-s_2}, \qquad t_2' := b^c g^{s_3}\tilde{g}^{s_4},$$
$$t_3' := h_j^{s_2}d^c, \qquad t_4' := f^c u^{s_5},$$
$$t_5' := b^{s_1}g^{-s_6}\tilde{g}^{-s_7}, \qquad t_6' := e(a^{-s_1}\tilde{g}^{s_6}g^{-c}, g)e(a^c\tilde{g}^{s_3}, w),$$

3. preveri $c \overset{?}{=} \mathcal{H}(g\|\tilde{g}\|w\|a\|b\|d\|f\|u\|t_1'\|t_2'\|t_3'\|t_4'\|t_5'\|t_6'\|m)$.

Drugi korak:

4. Za vsak $B \in RL_j$ preveri $e(d,u) \overset{?}{\neq} e(B,f)$.

Trditev 4.2.2. *Preverjanje podpisa je pravilno izvedeno.*

Dokaz. Ob predpostavki, da je zgoščevalna funkcija $\mathcal{H}$ brez trkov, pri prvem koraku zadošča preveriti, da velja $t_i = t_i'$ za $i = 1, 2, 3, 4, 5, 6$:

$$\begin{aligned}
t_1' &= f^{s_1}u^{-s_2} = u^{q(r_1-cx_i)}u^{-r_2+cx_iq} = u^{qr_1}u^{-r_2} = f^{r_1}u^{-r_2} = t_1,\\
t_2' &= b^c g^{s_3}\tilde{g}^{s_4} = g^{ck}\tilde{g}^{c\ell}g^{r_3-ck}\tilde{g}^{r_4-c\ell} = g^{r_3}\tilde{g}^{r_4} = t_2,\\
t_3' &= h_j^{s_2}d^c = h_j^{r_2-cx_iq}h_j^{cx_iq} = h_j^{r_2} = t_3,\\
t_4' &= f^c u^{s_5} = u^{cq}u^{r_5-cq} = u^{r_5} = t_4,\\
t_5' &= b^{s_1}g^{-s_6}\tilde{g}^{-s_7} = g^{k(r_1-cx_i)}\tilde{g}^{\ell(r_1-cx_i)}g^{-r_6+cx_ik}\tilde{g}^{-r_7+cx_i\ell} =\\
&\quad = g^k\tilde{g}^{\ell}g^{-r_6}\tilde{g}^{-r_7} = bg^{-r_6}\tilde{g}^{-r_7} = t_5,
\end{aligned}$$

$$\begin{aligned}
t_6' &= e(a^{-s_1}\tilde{g}^{s_6}g^{-c}, g)e(a^c\tilde{g}^{s_3}, w) =\\
&= e(A_i^{-r_1+cx_i}\tilde{g}^{-k(r_1-cx_i)}\tilde{g}^{r_6-cx_ik}g^{-c}, g)e(A_i^c\tilde{g}^{ck}\tilde{g}^{r_3-ck}, g^\gamma) =\\
&= e(A_i^{-r_1}\tilde{g}^{-kr_1}\tilde{g}^{r_6}, g)e(g^{cx_i(\gamma+x_i)^{-1}}g^{-c}, g)e(g^{c(\gamma+x_i)^{-1}}, g^\gamma)e(\tilde{g}^{r_3}, g^\gamma) =\\
&= e(a^{-r_1}\tilde{g}^{r_6}, g)e(g, g)^{c(\gamma+x_i)(\gamma+x_i)^{-1}-c}e(\tilde{g}^{r_3}, w) = e(a^{-r_1}\tilde{g}^{r_6}, g)e(\tilde{g}^{r_3}, w) = t_6.
\end{aligned}$$

Če je nadzornik skupine v časovnem obdobju j iz skupine odstranil podpisnika – naj bo to član i skupine U – potem preklicni seznam RL_j vsebuje preklicni žeton $B_{ij} = h_j^{x_i}$, ki ustreza zgornji enačbi:

$$e(d, u) = e(h_j^{x_iq}, u) = e(h_j^{x_i}, u^q) = e(B_{ij}, f).$$

Očitno je, da noben $B \in RL_j$ ne bo ustrezal enačbi, če član i ni bil odstranjen iz skupine v časovnem obdobju j in torej $B_{ij} \notin RL_j$. □

Odpiranje podpisa poteka po istem postopku kot preverjanje veljavnosti, s to izjemo, da se v drugem koraku namesto elementov preklicnega seznama uporabijo kar vsi preklicni žetoni za časovno obdobje j, v katerem je bil izdan podpis. Kot podpisnika nadzornik skupine identificira tistega člana i skupine U, katerega žeton B_{ij} ustreza enačbi.

Varnost sheme

Ker formalizirane varnostne zahteve iz drugega poglavja ne upoštevajo možnosti, da se lahko člani odstranjujejo iz skupine, bomo tukaj uporabili nekoliko prirejene zahteve, kjer je tudi to predvideno.

Trditev 4.2.3. *Zhou-Linova shema **ni** anonimna.*

Dokaz. Ker ima nasprotnik N pri poskusu anon_N^b dostop do vseh zasebnih ključev članov skupine, lahko izračuna $B_{ij} := h_j^{x_i}$ za poljubnega člana $i = 1, \ldots, n$ skupine U in časovno obdobje $j = 1, \ldots, T$ in tako lahko identificira podpisnika kateregakoli sporočila. □

Vidimo lahko tudi, da lahko vsak član i skupine U izračuna B_{ij} za poljuben $j = 1, \ldots, T$, kar pomeni, da lahko član skupine identificira lastne podpise. To je lahko tudi prednost, saj lahko tako član skupine ugotovi, ali je bil morda njegov zasebni ključ ukraden.

Varnostna lastnost, ki jo bomo dokazali pri tej shemi, je **anonimnost z vzvratno nepovezljivostjo** (angl. *backwards-unlinkability anonymity*). Neformalno to pomeni, da shema ohranja nepovezljivost tudi z odstranjevanjem članov iz skupine. Anonimnost z vzvratno nepovezljivostjo dobimo, če pri anonimnosti poskus anon_N^b nadomestimo z bu-anon_N^b. Pri njem lahko N počne isto kot pri anon_N^b, poleg tega pa lahko za vsakega člana skupine pridobi njegov preklicni žeton za poljubno časovno obdobje. Za izbrana člana i_0 in i_1 mora veljati, da N ni pridobil njunih preklicnih žetonov pred izbranim časovnim obdobjem J, iz katerega se generira podpis, katerega podpisnika mora N identificirati.

Zaradi prej omenjene slabosti bomo dokazali nekoliko šibkejšo različico te zahteve. Namesto dostopa do vseh zasebnih ključev članov skupine naj ima N dostop do oraklja za podpisovanje sporočil, zasebne ključe članov pa lahko pridobi, vendar i_0 in i_1 ne smeta biti med temi člani skupine.

Definicija 4.2.4. Shema za skupinske podpise je anonimna z vzvratno nepovezljivostjo, če je prednost $\text{Adv}_N^{\text{bu-anon}}$ polinomskega nasprotnika N zanemarljiva:

$$\text{Adv}_N^{\text{bu-anon}} = \Pr(\text{bu-anon}_N^1 = 1) - \Pr(\text{bu-anon}_N^0 = 1) < \varepsilon.$$

Trditev 4.2.5. *Zhou-Linova shema je anonimna z vzvratno nepovezljivostjo.*

Dokaz. Denimo, da lahko nasprotnik N z nezanemarljivo prednostjo $\text{Adv}_N^{\text{bu-anon}}$ razloči med skupinskima podpisoma dveh članov skupine. Po predpostavki iz modela z naključnim orakljem N iz c ne more pridobiti nobene informacije, prav tako ne iz s_i, $i = 1, \ldots, 7$, saj so r_i, $i = 1, \ldots, 7$ naključno izbrani. Zato mu morajo zadostovati a, b, d, f in u. Naj bo A algoritem, s katerim poskušamo

razbiti tridelno Diffie-Hellmanovo odločitveno predpostavko. Algoritem A dobi na vhod četverico $(g_1, g_2, g_3, z) \in G^4$, kjer je $G = \langle g \rangle$, $g_1 = g^\alpha$, $g_2 = g^\beta$, $g_3 = g^\delta$. Cilj algoritma A je ugotoviti, ali velja $z = g^{\alpha\beta\delta}$. Algoritem A uporabi nasprotnika N in odgovarja na njegove poizvedbe orakljem.

Najprej algoritem A vzpostavi shemo za skupinske podpise po sledečem postopku:

Vzpostavitev skupine s strani algoritma A:

1. uporabi grupo $G := \langle g \rangle$,
2. izberi $\tilde{g} \in_R G$, $\gamma \in_R \mathbb{Z}_p^*$,
3. izračunaj $w := g^\gamma$, $h_1 := g_1$,
4. izberi $r_j \in_R \mathbb{Z}_p^*$ za $j = 2, \ldots, T$,
5. izračunaj $h_j := g^{r_j}$ za $j = 2, \ldots, T$,
6. izberi $x_i \in_R \mathbb{Z}_p^*$ za $i = 2, \ldots, n$,
7. izračunaj $A_i = g^{(\gamma + x_i)^{-1}}$ za $i = 2, \ldots, n$,
8. izračunaj $B_{ij} := h_j^{x_i}$ za $i = 2, \ldots, n$ in $j = 2, \ldots, T$,
9. izračunaj $B_{i1} := g_1^{x_i}$ za $i = 2, \ldots, n$,
10. izračunaj $B_{1j} := g_2^{r_j}$ za $j = 2, \ldots, T$.

Vrednosti x_1, A_1 in B_{11} ostanejo neznane, saj bi moralo veljati $x_1 = \beta$, $A_1 = g^{(\gamma+\beta)^{-1}}$ in $B_{11} = g^{\alpha\beta}$. Na klice zgoščevalne funkcije $\mathcal{H}$ algoritem A odgovori z naključnim izhodom, pri čemer si vsak vhod zapomni, tako da lahko pri ponovitvi vhoda ponovi tudi izhod.

Če nasprotnik N zahteva skupinski podpis od člana $i \neq 1$ skupine U, potem lahko algoritem B po postopku za podpisovanje izda veljaven podpis, saj ima vse potrebne podatke. Če velja $i = 1$, a je zahtevano časovno obdobje j različno od 1, potem A izbere:

$$a, b, u \in_R G,\ q \in_R \mathbb{Z}_p^*$$

ter izračuna:

$$d := B_{1j}^q,\ f := u^q,$$

če pa velja $i = 1$ in $j = 1$, pa A izbere:

$$a, b \in_R G,\ z_1, z_2 \in_R \mathbb{Z}_p^*$$

ter izračuna:

$$d := g_1^{z_1},\ f := g^{z_1 z_2},\ u = g_2^{z_2}.$$

Očitno v slednjem primeru velja:

$$d = g^{\alpha\beta q} = B_{11}^q,\ f = u^q,$$

kjer je $q = z_1\beta^{-1}$. V obeh primerih nato izbere:

$$s_1, s_2, s_3, s_4, s_5, s_6, s_7 \in_R \mathbb{Z}_p,\ c \in_R \mathbb{Z}_p^*$$

ter izračuna t_1', t_2', t_3', t_4', t_5', t_6' po formulah iz druge točke pri preverjanju podpisa. Če je nasprotnik N že klical

$$\mathcal{H}(g\|\tilde{g}\|w\|a\|b\|d\|f\|u\|t_1'\|t_2'\|t_3'\|t_4'\|t_5'\|t_6'\|m),$$

potem se algoritem A prekine in vrne naključno izbran odgovor. V nasprotnem primeru temu vhodu določi izhod c in izda podpis

$$\sigma = (c, s_1, s_2, s_3, s_4, s_5, s_6, s_7, a, b, d, f, u).$$

Če nasprotnik N zahteva zasebni ključ člana $i = 1$ ali njegov preklicni žeton za obdobje $j = 1$, potem se algoritem A prekine in vrne naključno izbran odgovor. Ko nasprotnik N izbere sporočilo m, člana i_0 in i_1 ter časovno obdobje J, algoritem A naključno izbere $\phi \in \{0, 1\}$ ter se prekine in vrne naključen odgovor, če velja $i_\phi \neq 1$ ali $J \neq 1$. V nasprotnem primeru algoritem A generira podpis tako, da izbere:

$$a, b \in_R G,\ r \in_R \mathbb{Z}_p^*$$

ter postavi:

$$d := z,\ f := g_3^r,\ u := g^r.$$

Preostanek podpisa naredi po istem postopku, kot v prejšnjem odstavku. Če velja $z = g^{\alpha\beta\delta}$, potem sta

$$d = h_1^{x_1\delta},\ f = u^\delta,$$

torej ustrezata skupinskemu podpisu, ki bi ga izdal član $i = 1$ in zato nasprotnik N pravilno identificira podpisnika z verjetnostjo, ki ni zanemarljivo različna

od $\frac{1}{2}$. Če pa velja $z \neq g^{\alpha\beta\delta}$, potem generirani podpis ne ustreza podpisu, ki bi ga izdal kateri od članov i_0 ali i_1, tako da N lahko le ugiba o podpisniku.

Algoritem A vrne, da velja $Z = g^{\alpha\beta\delta}$ natanko tedaj, ko nasprotnik N kot podpisnika generiranega sporočila identificira člana skupine $i = 1$. Če torej res velja $Z = g^{\alpha\beta\delta}$, potem A to pravilno ugotovi z verjetnostjo

$$\frac{1}{2} + \frac{1}{2nT}\,\mathrm{Adv}_N^{\text{bu-anon}}\,,$$

sicer pa je verjetnost pravilnega odgovora enaka $\frac{1}{2}$. Pri tem smo predpostavili, da je število klicev zgoščevalne funkcije $\mathcal{H}$ in oraklja za podpisovanje zanemarljivo v primerjavi z redom grupe p, zato ju nismo upoštevali. Prednost algoritma A pri tridelnem Diffie-Hellmanovem odločitvenem problemu je tako

$$\mathrm{Adv}_{\mathrm{A}} = \frac{1}{2nT}\,\mathrm{Adv}_N^{\text{bu-anon}}$$

in ni zanemarljiva. To je v protislovju s tridelno Diffie-Hellmanovo odločitveno predpostavko, tako da je Zhou-Linova shema anonimna z vzvratno nepovezljivostjo. □

Razširili bomo še eno varnostno zahtevo iz drugega poglavja, in sicer polno sledljivost, torej združeno zahtevo sledljivosti in nepodtakljivosti za statične sheme. Namesto poskusov trace_N in nf_N bo nasprotnik N izvajal poskus ftrace_N. Pri tem poskusu ima dostop do vseh orakljev, do katerih ima dostop pri trace_N in nf_N, ter dostop do zasebnih ključev podmnožice $\mathcal{C}$ članov skupine in do vseh preklicnih žetonov, za katere tudi ve, komu pripadajo. Poleg sporočila m in njegovega podpisa σ nasprotnik N izbere še časovno obdobje j, v katerem naj bi nastal podpis. Poskus uspe, če je σ veljaven skupinski podpis in zanj ni mogoče določiti podpisnika, ali pa je ta izven množice $\mathcal{C}$.

Definicija 4.2.6. Shema za skupinske podpise je polno sledljiva, če je prednost $\mathrm{Adv}_N^{\text{ftrace}}$ polinomskega nasprotnika N zanemarljiva:

$$\mathrm{Adv}_N^{\text{ftrace}} = \Pr(\mathrm{ftrace}_N = 1) < \varepsilon.$$

Trditev 4.2.7. *Zhou-Linova shema je polno sledljiva.*

Dokaz. Denimo, da lahko nasprotnik N z nezanemarljivo verjetnostjo ponaredi tak veljaven podpis σ, za katerega bodisi ni mogoče identificirati podpisnika, ali pa za identificiranega podpisnika nasprotnik N ni pridobil zasebnega ključa

za podpisovanje v imenu skupine. Opisali bomo algoritem A, ki lahko tedaj z nezanemarljivo verjetnostjo reši n-krepki Diffie-Hellmanov problem ali krepki RSA problem. Algoritem A uporabi nasprotnika N in odgovarja na njegove poizvedbe orakljem. Tako kot pri prejšnjem dokazu tudi tukaj algoritem A skrbi tudi za odgovarjanje na klice zgoščevalne funkcije $\mathcal{H}$.

Algoritem A dobi kot vhod $(n+2)$-terico $(\tilde{g}, g', g'^{\gamma}, g'^{\gamma^2}, \ldots, g'^{\gamma^n})$ za neznan $\gamma \in \mathbb{Z}_p^*$, kjer je g' generator grupe G z $\mathrm{ord}(g') = p \in \mathbb{P}$, $\tilde{g}$ pa je element v G. Cilj algoritma A je najti par (u, e), za katerega velja bodisi $u = g'^{(\gamma+e)^{-1}}$, bodisi $\tilde{g} = u^e$. Najprej si poglejmo postopek iz [6], po katerem lahko generiramo $n-1$ parov (u', e'), za katere velja $u' = g^{(\gamma+e')^{-1}}$ za nek $g \in G$.

Generiranje parov (u', e'), $u' = g^{(\gamma+e')^{-1}}$:

1. izberi $x_i \in_R \mathbb{Z}_p^*$ za $i = 1, \ldots, n-1$,
2. postavi $F(x) := \prod_{i=1}^{n-1}(x + x_i) =: \sum_{i=0}^{n-1} \alpha_i x^i$,
3. izračunaj $g := \prod_{i=0}^{n-1}(g'^{\gamma^i})^{\alpha_i} = g'^{F(\gamma)}$,
4. izračunaj $w := \prod_{i=1}^{n}(g'^{\gamma^i})^{\alpha_{i-1}} = g'^{\gamma F(\gamma)} = g^{\gamma}$,
5. postavi
$$F_i(x) := F(x)(x + x_i)^{-1} =: \sum_{j=0}^{n-2} \beta_{ij} x^j$$
za $i = 1, \ldots, n-1$,
6. izračunaj
$$A_i := \prod_{j=0}^{n-2}(g^{\gamma^j})^{\beta_{ij}} = g'^{F_i(\gamma)} = g^{(\gamma+x_i)^{-1}}$$
za $i = 1, \ldots, n-1$.

Predpostavili smo, da za vsak $i \in [1..(n-1)]$ velja $\gamma + x_i \neq 0$. V nasprotnem primeru z zanemarljivo verjetnostjo, ko velja $g = 1$, je $F(\gamma) = 0$, torej je $\gamma = -x_i$ za nek $i \in [1..n-1]$ in tedaj smo rešili n-krepki Diffie-Hellmanov problem.

Algoritem A najprej poskuša uganiti, ali bo nasprotnik N skušal ponarediti podpis, za katerega ni mogoče identificirati podpisnika, ali pa bo poskušal ponarediti podpis nekega člana skupine, ne da bi uporabil njegov zasebni ključ.

V prvem primeru algoritem A vzpostavi skupino z $n-1$ člani. Ker je red g' praštevilski in $g \neq 1$, velja $\mathrm{ord}(g') = \mathrm{ord}(g)$ in zato je tudi g generator grupe G, ki jo tako algoritem A uporabi v shemi. Prav tako uporabi vrednost $\tilde{g}$, dobljeno kot vhod v algoritem, in vrednost w, dobljeno v postopku iz prejšnjega odstavka, pari (x_i, A_i) za $i = 1, \ldots, n-1$ pa služijo kot zasebni ključi članov skupine. Ostale vrednosti algoritem A izbere po običajnem postopku za vzpostavitev skupine, z izjemo γ, ki ostane neznan. V drugem primeru algoritem A vzpostavi shemo z n člani, pri čemer stori isto kot pri prvem primeru, le da poleg tega izbere še $A_n \in_R G$ in $B_{nj} \in_R G$ za $j = 1, \ldots, T$ vrednost x_n pa pusti nedefinirano.

Če se izbira tipa pričakovanega ponarejenega podpisa izkaže za nepravilno, se s tem zmanjša verjetnost uspeha. Če je izbira naključna, bo pravilna v polovici primerov, vendar lahko to verjetnost povečamo, če poznamo verjetnosti za posamezen tip podpisa, ki ga poskuša ponarediti nasprotnik N.

Če želi nasprotnik N pridobiti zasebni ključ člana n skupine U, se algoritem A prekine z neuspehom. Če pa N pokliče oraklja za podpisovanje sporočila m s strani člana n v časovnem obdobju j, potem algoritem A izbere:

$$k, \ell, q \in_R \mathbb{Z}_p^*,\ u \in_R G$$

ter izračuna

$$a := A_n \tilde{g}^k,\ b := g^k \tilde{g}^\ell,\ d := B_{nj}^q,\ f := u^q.$$

Preostanek podpisa izračuna tako kot v prejšnjem dokazu, vključno s prirejanjem zgoščevalne funkcije $\mathcal{H}$. Klici oraklja za ostale člane skupine se izvedejo po navadnem postopku za podpisovanje, saj ima A vse potrebne podatke.

Vsakič, ko nasprotnik N kliče zgoščevalno funkcijo, si algoritem A zapomni trenutno stanje. Ko N izda uspešno ponarejen podpis

$$\sigma = (c, s_1, s_2, s_3, s_4, s_5, s_6, s_7, a, b, d, f, u),$$

algoritem A izračuna t_1', t_2', t_3', t_4', t_5' in t_6' po formulah iz druge točke pri preverjanju podpisa. Nato algoritem A najde stanje, v katerem je nasprotnik N prvič poklical

$$\mathcal{H}(g \| \tilde{g} \| w \| a \| b \| d \| f \| u \| t_1' \| t_2' \| t_3' \| t_4' \| t_5' \| t_6' \| m).$$

V primeru z zanemarljivo verjetnostjo, ko se to ni nikoli zgodilo (torej je nasprotnik N moral uganiti izhod, saj je podpis σ veljaven) ali pa če je A sam nastavil izhod za ta vhod (kar pomeni, da N ni ponaredil podpisa, pač pa ga je pridobil z orakljem in zato po definiciji poskus ne uspe), se algoritem

A prekine z neuspehom. Sicer pa se algoritem A postavi v stanje pred klicem zgoščevalne funkcije, za katero tokrat da drugačen odgovor kot pri prvotnem klicu. Nato nadaljuje izvajanje, pri čemer N izda nov podpis

$$\sigma' = (c', s'_1, s'_2, s'_3, s'_4, s'_5, s'_6, s'_7, a, b, d, f, u).$$

Če je tudi ta uspešno ponarejen, lahko algoritem A izračuna:

$$\tilde{x} := \frac{s_6 - s'_6}{s_3 - s'_3},\ \tilde{k} := \frac{s_3 - s'_3}{c' - c},\ \tilde{A} := a\tilde{g}^{-\tilde{k}}.$$

Naj bo $y \in \mathbb{Z}_p^*$ tako število, da velja $\tilde{g} = g^y$. Tedaj velja:

$$t'_2 = b^c g^{s_3} \tilde{g}^{s_4} = b^{c'} g^{s'_3} \tilde{g}^{s'_4} \Rightarrow 1 = b^{c-c'} g^{s_3 - s'_3} \tilde{g}^{s_4 - s'_4} = b^{c-c'} g^{s_3 - s'_3 + y(s_4 - s'_4)},$$

$$t'_5 = b^{s_1} g^{-s_6} \tilde{g}^{-s_7} = b^{s'_1} g^{-s'_6} \tilde{g}^{-s'_7} \Rightarrow 1 = b^{s_1 - s'_1} g^{s'_6 - s_6} \tilde{g}^{s'_7 - s_7} = b^{s_1 - s'_1} g^{s'_6 - s_6 + y(s'_7 - s_7)},$$

$$\frac{s_1 - s'_1}{c' - c} = \frac{s_6 - s'_6 + y(s_7 - s'_7)}{s_3 - s'_3 + y(s_4 - s'_4)}. \tag{4.2.3}$$

Če lahko iz enačbe 4.2.3 izrazimo vrednost y, je par (g, y) rešitev krepkega RSA problema za vhod $\tilde{g}$. V nasprotnem primeru mora očitno veljati:

$$\frac{s_1 - s'_1}{c' - c} = \frac{s_6 - s'_6}{s_3 - s'_3} = \frac{s_7 - s'_7}{s_4 - s'_4} = \tilde{x}.$$

Tedaj lahko izrazimo:

$$t'_6 = e(a^{-s_1} \tilde{g}^{s_6} g^{-c}, g) e(a^c \tilde{g}^{s_3}, w) = e(a^{-s'_1} \tilde{g}^{s'_6} g^{-c'}, g) e(a^{c'} \tilde{g}^{s'_3}, w)$$

$$e(a^{s'_1 - s_1} \tilde{g}^{s_6 - s'_6} g^{c' - c}, g) e(a^{c-c'} \tilde{g}^{s_3 - s'_3}, w) = 1$$

$$e(a^{(c-c')\tilde{x}} \tilde{g}^{(s_3 - s'_3)\tilde{x}}, g) e(a^{c-c'} \tilde{g}^{s_3 - s'_3}, w) = e(g^{c-c'}, g)$$

$$e(a^{c-c'} \tilde{g}^{s_3 - s'_3}, g^{\tilde{x}} w) = e(g^{c-c'}, g)$$

$$e(a\tilde{g}^{\tilde{k}}, g^{\tilde{x}} w) = e(g, g)$$

$$e(\tilde{A}, g^{\tilde{x}} g^{\gamma}) = e(g, g)$$

$$\tilde{A} = g^{(\gamma + \tilde{x})^{-1}}$$

Pretvorimo sedaj par $(\tilde{A}, \tilde{x})$ v rešitev za prvotni n-krepki Diffie-Hellmanov problem. To lahko storimo, če je $\tilde{x}$ različen od vseh x_i za $x \in [1..(n-1)]$. Racionalno funkcijo $F(x)/(x + \tilde{x})$ tedaj izrazimo kot:

$$\frac{F(x)}{x + \tilde{x}} = \frac{\delta_{-1}}{x + \tilde{x}} + \sum_{i=0}^{q-2} \delta_i x^i.$$

Izračunajmo še:

$$\tilde{w} := \left(\tilde{A} \prod_{i=0}^{q-2} \left(g'^{\gamma^i} \right)^{-\delta_i} \right)^{1/\delta_{-1}} = \left(g'^{F(\gamma)(\gamma+\tilde{x})^{-1}} g'^{(\delta_{-1}-F(\gamma))(\gamma+\tilde{x})^{-1}} \right)^{1/\delta_{-1}} = g'^{(\gamma+\tilde{x})^{-1}}$$

Par $(\tilde{w}, \tilde{x})$ je torej rešitev n-krepkega Diffie-Hellmanovega problema. Če je prednost nasprotnika N pri poskusu ftrace_N enaka $\text{Adv}_N^{\text{ftrace}}$, je red prednosti Adv_A algoritma A pri reševanju n-krepkega Diffie-Hellmanovega problema ali krepkega RSA problema enak $\Omega((\text{Adv}_N^{\text{ftrace}})^2 n^{-2})$, kjer z Ω označimo spodnjo asimptotično mejo. Ker prednost $\text{Adv}_N^{\text{ftrace}}$ ni zanemarljiva, tudi prednost Adv_A ni zanemarljiva. Po n-krepki Diffie-Hellmanovi predpostavki in krepki RSA predpostavki je prednost pri reševanju enega od obeh problemov zanemarljiva, kar pa je v protislovju z dobljenim rezultatom. Zhou-Linova shema je torej polno sledljiva. □

Opombe

Očitna slabost sheme je v tem, da nadzornik skupine poseduje zasebne ključe vseh članov skupine in tako lahko izdaja podpise v imenu kateregakoli od njih. Če lahko to dejstvo zanemarimo, je potem Zhou-Linova shema popolnoma dinamična, saj lahko nadzornik skupine enostavno poveča tako n kot T in izračuna manjkajoče h_j, x_i, A_i in B_{ij}.

Ker pa tega ponavadi nočemo, lahko to preprečimo tako, da pri vzpostavitvi skupine poleg nadzornika sodelujejo vsi člani skupine. Vsak član izbere svoj x_i ter ga pošlje nadzorniku skupine, ki izračuna A_i in ga pošlje nazaj članu skupine. Poleg tega izračuna še B_{ij} za $j = 1, \ldots, T$. Nato se član skupine prepriča, da je nadzornik zavrgel x_i in A_i. Kasnejše dodajanje članov je sicer mogoče po enakem postopku, večji problem pa predstavlja dodajanje novih časovnih obdobij – vsak član bi sicer lahko izračunal $B_{ij} := h_j^{x_i}$ za nova časovna obdobje in s protokolom, ekvivalentnim

$$\text{SPK}\left\{(\alpha) \mid B_{i1} = h_1^\alpha \wedge B_{ij} = h_i^\alpha\right\}(m),$$

dokazal pravilnost izračuna, vendar nadzornik skupine člana, ki bi zavrnil sodelovanje, ne bi mogel izločiti iz skupine, saj bi mu manjkal ravno podatek, ki je za to potreben.

Za Zhou-Linovo shemo je predlagana uporaba grupe na eliptični krivulji praštevilskega reda p, ki je dolg 170 bitov, tako da so elementi G dolgi 171 bitov. Dolžina skupinskega podpisa je tedaj 277 B.

Poglavje 5

Zaključek

Predstavili smo koncept skupinskega podpisa in si pogledali matematično ozadje, s pomočjo katerega dokazujemo varnost kriptografskim shem, nato pa smo si pogledali še dva konkretna primera shem za skupinske podpise, skupaj z dokazi o njihovi varnosti.

Ker idealne univerzalne sheme za skupinske podpise še ne poznamo, se razvoj shem usmerja tudi v namensko rabo. Omenili smo že skupinske podpise brez nepovezljivosti, ki se lahko uporabljajo za elektronske volitve. Za ta namen so še bolj primerni **krožni podpisi** (angl. *ring signatures*). Ti se razlikujejo od skupinskih podpisov po tem, da ne omogočajo odpiranja podpisov, saj skupino ob vsakem podpisovanju izbere podpisnik, lahko tudi brez vednosti ostalih članov. V osnovni različici, predstavljeni v [23], je prava identiteta podpisnika popolnoma skrita, v [18] pa je predstavljena shema za krožne podpise brez nepovezljivosti, namenjena elektronskemu glasovanju.

Še ena razširitev koncepta skupinskega podpisa so **hierarhični skupinski podpisi**, predstavljeni v [26]. Pri njih imamo hierarhično, v drevesu razporejene skupine. Listi predstavljajo posamezne člane, notranja vozlišča pa nadzornike skupin, pri čemer so v skupini nekega nadzornika vsi njegovi otroci v drevesu. Posamezen nadzornik neke skupine lahko identificira podpisnika nekega skupinskega podpisa le tedaj, ko je ta član skupine, ki jo neposredno nadzira. Če pa je podpisnik član neke podskupine globlje v drevesu, pa lahko nadzornik ugotovi le, kateri skupini, ki jo neposredno nadzoruje, pripada podpisnik.

Omenimo še sheme za skupinske podpise, ki temeljijo na kriptografiji na osnovi identitete [22, 12]. Za konec pa si lahko kot izziv postavimo nov tip skupinskih podpisov, kjer odpiranje podpisov ni v rokah le ene pooblaščene osebe, pač pa lahko to stori, po zgledu shem za deljenje skrivnosti, le poljubna

dovolj velika podmnožica članov skupine.

Slike

Literatura

[1] G. Ateniese, J. Camenisch, M. Joye in G. Tsudik, A practical and provably secure coalition-resistant group signature scheme, *CRYPTO '00*, LNCS **1880** (2000) 255–270.
http://www.zurich.ibm.com/~jca/papers/group2000.pdf

[2] M. Bellare, D. Micciancio in B. Warinschi, Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions, *EUROCRYPT*, LNCS **2656** (2003) 614–629.
http://www-cse.ucsd.edu/users/mihir/papers/gs.pdf

[3] M. Bellare in P. Rogaway, Random oracles are practical: a paradigm for designing efficient protocols, *CCS '93* (1993) 62–73.
http://www-cse.ucsd.edu/users/mihir/papers/ro.pdf

[4] M. Bellare, Shi H. in Zhang C., Foundations of group signatures: The case of dynamic groups, *CT-RSA '05*, LNCS **3376** (2005) 136–153.
http://www-cse.ucsd.edu/users/mihir/papers/bsz.pdf

[5] M. Blum, P. Feldman in S. Micali, Non-interactive zero-knowledge and its applications, *STOC '88* (1988) 103–112.
http://portal.acm.org/ft_gateway.cfm?id=62222&type=pdf

[6] D. Boneh in X. Boyen, Short signatures without random oracles, *EUROCRYPT*, LNCS **3027** (2004) 56–73.
http://robotics.stanford.edu/~xb/eurocrypt04a/bbsigs.ps

[7] D. Boneh in H. Shacham, Group signatures with verifier-local revocation, *CCS '04* (2004) 168–177.
http://www.hovav.net/dist/preteripsistic.pdf

[8] J. Camenisch in M. Michels, A group signature scheme based on an RSA-variant, *Research Series* **27**, BRICS, Department of Computer Science, University of Aarhus (1998).
http://www.brics.dk/RS/98/27/BRICS-RS-98-27.pdf

[9] J. Camenisch in M. Michels, Proving in zero-knowledge that a number is the product of two safe primes, *EUROCRYPT*, LNCS **1592** (1999) 107–122.
http://www.springerlink.com/content/blqm17fy9wr5n1xx/fulltext.pdf

[10] R. Canetti, O. Goldreich in Sh. Halevi, The random oracle methodology, revisited (preliminary version, *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing* (1998) 209–218.
http://arxiv.org/pdf/cs/0010019v1

[11] D. Chaum in E. van Heyst, Group signatures, *EUROCRYPT*, LNCS **547** (1991) 257–265.
http://www.springerlink.com/content/yrk497a8yjge84fx/fulltext.pdf

[12] Chen X., Zhang F. in Kim K., A new ID-based group signature scheme from bilinear pairings, *Proceedings of International Workshop on Information Security Applications (WISA)*, LNCS **2908** (2003) 585–592.
http://eprint.iacr.org/2003/116.pdf

[13] W. Diffie in M. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory* **22** (1976) 644–654.

[14] Funabiki N., Nakanishi T., Takahashi H., Miki K. in Kawashima J., A proposal of anonymous IEEE802.1X authentication protocol for wireless networks, *Second Workshop on Secure Network Protocols (NPSec)* (2006) 26–31.

[15] R. Gennaro, D. Micciancio in T. Rabin, An efficient non-interactive statistical zero-knowledge proof system for quasi-safe prime products, *CCS '98* (nov. 1998) 67–72.
http://www.cs.ucsd.edu/users/daniele/papers/GMR.pdf

[16] O. Goldreich in Y. Oren, Definitions and properties of zero-knowledge proof systems, *Journal of Cryptology* **7** (1994) 1–32.
http://www.wisdom.weizmann.ac.il/~oded/PS/oren.ps

[17] S. Haber in W.S. Stornetta, How to time-stamp a digital document, *Journal of Cryptology* **3** (1991) 99–111.
http://www.cs.utk.edu/~dunigan/cs594-cns/timestamp.pdf

[18] J.K. Liu, V.K. Wei in D.S. Wong, Linkable spontaneous anonymous group signature for ad hoc groups, *ACISP '04*, LNCS **3108** (2004) 325–335.
http://www.springerlink.com/content/7dp5c5cjwq5cg7eq/fulltext.pdf

[19] G. Maitland in C. Boyd, Fair electronic cash based on a group signature scheme, *ICICS '01*, LNCS **2229** (2001) 461–465.
http://sky.fit.qut.edu.au/~boydc/papers/GOC-ECash.pdf

[20] A.J. Menezes, S.A. Vanstone in P.C. van Oorschot, *Handbook of Applied Cryptography* (1996). CRC Press, Inc., Boca Raton, FL, USA.

[21] Nakanishi T., Fujiwara T. in Watanabe H., A linkable group signature and its application to secret voting, *Transactions of Information Processing Society of Japan* **40** (1999) 3085–3096.

[22] C. Popescu, An efficient ID-based group signature scheme, *Studia Univ. Babeş-Bolyai, Informatica* **47** (2002) 29–36.
http://www.cs.ubbcluj.ro/~studia-i/2002-2/4-Popescu.pdf

[23] R.L. Rivest, A. Shamir in Y. Tauman, How to leak a secret: Theory and applications of ring signatures, *Essays in Memory of Shimon Even*, LNCS **3895** (2006) 164–186.
http://www.cs.uwaterloo.ca/~bssadjad/courses/crypto/leak_secret_rivest.pdf

[24] P.W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J.SCI.STATIST.COMPUT.* **26** (1997) 1484.
http://arxiv.org/pdf/quant-ph/9508027v2

[25] Su Y. in Zhu Y., A fair off-line e-cash system with group signature, *Wuhan University Journals Press* **9** (2004) 745–748.
http://www.springerlink.com/content/4651642r50577218/fulltext.pdf

[26] M. Trolin in D. Wikström, Hierarchical group signatures, *Automata, Languages and Programming* (2005) 446–458.
http://eprint.iacr.org/2004/311.pdf

[27] L.M.K. Vandersypen, M. Steffen, G. Breyta, C.S. Yannoni, M.H. Sherwood in I.L. Chuang, Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance, *Nature* **414** (2001) 883–887.

[28] Zhou S. in Lin D., A shorter group signature with verifier-location revocation and backward unlinkability. *Cryptology ePrint Archive, Report 2006/100* (2006).
http://eprint.iacr.org/2006/100.pdf

Izjava

Izjavljam, da sem diplomsko nalogo izdelal samostojno pod vodstvom mentorja izr. prof. dr. Aleksandra Jurišića. Izkazano pomoč drugih sodelavcev sem v celoti navedel v zahvali.

Ljubljana, 9.9.2008 Janoš Vidali