UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO
Podiplomski študij: matematika – izobraževalna smer

DOKAZI BREZ RAZKRITJA ZNANJA: DISKRETNI LOGARITEM

(seminarska naloga)

Aljoša Brlogar,
Kranj, september 2004

KAZALO

1. UVOD....3

2. CILJI....4

3. OSNOVE DOKAZOV BREZ RAZKRITJA ZNANJA....5

3. 1. LASTNOSTI DOKAZOV BREZ RAZKRITJA ZNANJA....5
3. 2. PRIMER: DOKAZ O ZNANJU O KVADRATNIH OSTANKIH....6

4. OSNOVNI PROTOKOL ZA DISKRETNI LOGARITEM....8

4. 1. MOTIVACIJA....8
4. 2. OPIS IN ANALIZA PROTOKOLA....8

5. POSPLOŠITEV PROTOKOLA ZA DISKRETNI LOGARITEM....13

5. 1. PREDSTAVITEV PROBLEMA....13
5. 2. OPIS IN ANALIZA PROTOKOLA....13

6. DISKRETNI LOGARITEM IN IDENTIFIKACIJSKE SHEME....15

6. 1. IDENTIFIKACIJSKE SHEME....15
6. 2. SCHNORR-OV IDENTIFIKACIJSKI PROTOKOL....16
6. 3. FIAT-SHAMIR-JEV IDENTIFIKACIJSKI PROTOKOL....16
6. 4. PRIMERJAVA SCHNORR-OVEGA IN FIAT-SHAMIR-JEVEGA IDENTIFIKACIJSKEGA PROTOKOLA....17

7. ZAKLJUČEK....19

8. REFERENCE....20

1. UVOD

Kot že samo ime pove je bistvo teh dokazov, da ena oseba prepriča drugo osebo o nekem dejstvu, ne da pri tem izdala katerokoli skrivno informacijo. Oglejmo si naslednji primer.

Recimo, da imajo vsi agenti neke tajne agencije rezervne kopije svojih tajnih dokumentov, ki pa se nahajajo na vsem agentom dostopnem mestu, skladišču rezervnih kopij. Vsi ti dokumenti so šifrirani s pomočjo enkripcije z njihovim javnim ključem. V nekem trenutku želi agent **A** razkriti agentu **B** celotno vsebino enega izmed svojih tajnih dokumentov, ki se nahaja med njegovimi rezervnimi kopijami. Prva rešitev, ki nam pade na pamet, je, da bi agent **A** ta dokument enostavno poslal agentu **B**. Vendar se kmalu pojavi problem. Agent **B** ne more preveriti, ali je poslan dokument res iz skladišča rezervnih kopij ali pa je popolnoma poljuben dokument. Agent **A** lahko dokaže pristnost dokumenta, s tem da odkrije svoj osebni enkripcijski ključ. Vendar bi s tem dal agentu **B** možnost, da si lahko ogleda tudi preostale njegove tajne dokumente, kar pa si seveda agent **A** ne želi. Postavi se vprašanje, ali lahko agent **A** prepriča agenta **B**, da mu je poslal pravi dokument, ne da bi mu pri tem razkril kakršno koli informacijo (»znanje«).

S takim načinom dokazovanja so se srečali že v 16. stoletju v Italiji, ko je Tartaglia našel rešitev enačbe 3. stopnje, vendar je ni želel objaviti. Nek drug matematik Fior mu ni verjel, zato mu je poslal 30 nalog. Tartaglia je rešil vseh 30 enačb, mu poslal rezultate, in pri tem ni razkril postopka reševanja.

V splošnem se vprašamo, ali obstaja dokaz za neko izjavo, da pri dokazu njene veljavnosti, ne bi razkrili nobenih drugih informacij oziroma »znanja« o tej izjavi. Takšne dokaze imenujemo dokazi brez razkritja znanja, seveda če obstajajo.
Torej nam dokazi brez razkritja znanja povedo le, ali je izjava pravilna ali ne.

Protokoli, ki temeljijo na dokazih brez razkritja znanja se danes uporabljajo za preverjanje identitete, digitalno podpisovanje, izmenjavo ključev ipd. Ti protokoli onemogočajo, da bi med komuniciranjem, bila razkrita kakršna koli skrivna informacija.

2. CILJI

V tej seminarski nalogi bom skušal na kratko opisati lastnosti in pomen dokazov brez razkritja znanja, saj so zelo zanimiva tema. Podal bom nekaj praktičnih primerov: Rubikova kocka, preštevanje listov ter šahovski problem. Največji poudarek bo na diskretnem logaritmu, in sicer bom opisal, kako lahko nekoga prepričamo, da poznamo diskretni logaritem, ne da bi mu ga razkrili. Na enostavnem protokolu za diskretni logaritem, bom dokazali varnost in polnost tega protokola, medtem ko bom dokaz za posplošeni diskretni algoritem izpustili, saj je natančno opisan v enem izmed člankov, ki sem jih uporabil pri izdelavi te projektne naloge.

Prav tako me je zanimala uporaba tega protokola, ki se imenuje tudi Schnorr-ov protokol, v identifikacijskih shemah. Ker se za identifikacijske sheme uporabljajo tudi drugi protokol, sem se odločil, da bom ta protokol primerjal s katerim bolj znanim. Za primerjavo sem si izbral Fiat-Shamir-jev protokol. Da pa ju bo lažje primerjati, ga bom tudi na kratko opisal.

Na začetku sem bil mnenja, da je Schnorr-ov protokol dober za identifikacijske sheme, ker temelji na diskretnem logaritmu, katerega ne druga oseba (Viktor) v interakciji ne neka tretja oseba, ki se pojavlja kot napadalec ne moreta kar tako izračunati. Kajti pri Fiat-Shamir-jevem protokolu gre le za računanje kvadratnega korena. Izkaže se, da je drugi protokol primernejši zaradi enostavnejših operacij, saj je časovno manj zahteven in zato poteka interakcija med dvema stranema precej hitreje.

3. OSNOVE DOKAZOV BREZ RAZKRITJA ZNANJA

V protokolu dokaza nastopata nujno naslednji dve osebi:

Petra (dokazovalka oz. »Prover«) ima oz. pozna nekaj informacij in bi rada Viktorju dokazala, da jih ima oz. jih pozna, ne da bi mu izdala kakršnokoli skrivnost.

Viktor (preverjevalec oz. »Verifier«) postavlja Petri vprašanja, s katerimi skuša ugotovit, ali Petra skrivne informacije. Navkljub različnim vprašanjem Viktor ne mora odkriti nobene skrivnosti.

Recimo, da Petra trdi, da zna rešiti Rubikovo kocko, vendar načina noče razkriti Viktorju. V tem primeru Viktor pokaže premešano kocko **A** in zahteva, da jo sestavi. Petra pokaže Viktorju drugo premešano kocko **B**. Sedaj ima Viktor lahko zahteva dve različni stvari od Petre:
Petra mora premešati situacijo s kocke **B** v situacijo z začetne kocke **A**, ali da mora rešiti kocko **B**.
Če Petra zna premešati kocke tako, da pride iz situacije na kocki **B** do situacije na kocki **A**, je uspešna v obeh primerih. Če kocke ne zna rešiti, ima 50% možnosti da uspe, saj bo uspešna le pri drugi nalogi.

3. 1. LASTNOSTI DOKAZOV BREZ RAZKRITJA ZNANJA

Dokazi brez razkritja znanja imajo posebne lastnosti:
Viktor se iz samega dokaza ne nauči nič novega. Edino kar se lahko nauči, se nauči na podlagi lastnih spoznanj brez Petrine pomoči.

Petra ne more goljufati Viktorja. Če Petre ne pozna skrivnosti lahko uspe, le če ima srečo, vendar lahko Viktor po nekaj ponovitvah protokola to ugotovi in ga prekine.

Viktor ne more goljufati Petre. Če se Viktor odloči, da ne bo sledil protokolu, ne izve nobene skrivnosti. Edino kar se lahko prepriča je, ali Petra res pozna skrivnost.

Viktor se ne more tretji osebi izdajati za dokazovalca. To enostavno sledi iz že velikokrat ponovljenega dejstva, da od Petre ne izve nobene skrivne informacije.

Poglejmo si še en primer:
Petra trdi, da lahko v nekaj sekundah prešteje liste na zelo velikem drevesu. Kako lahko prepriča Viktorja, ne da mu izdala način preštevanja in število listov?
Zapre oči in sedaj ima Viktor zopet dve možnosti: odtrga en list ali pa ne stori nič. Ko Petra pogleda na drevo, mu pove, kaj je naredil.
Vidimo, da ta primer zadošča prvi, tretji in četrti lastnosti., saj Viktor ne izve števila listov in ne načina preštevanja le-teh.
Če Petra goljufa lahko ugiba. Četudi bi imela srečo pri ugibanju, bi po večkratnih ponovitvah Viktor ugotovil, da goljufa. Torej velja tudi druga lastnost.

Dokazi brez razkritja znanja morajo ustrezati naslednjim zahtevam:

- **veljavnost:** zahteva, da postopek preverjanja ne omogoča sprejema napačne trditve. Veljavnost varuje preverjevalca (Viktorja) pred tem, da bi bil prepričan v veljavnost nepravilne izjave ne glede na to, kakšne trike pri tem uporablja dokazovalka (Petra).
- **polnost:** omogoča dokazovalcu, da prepriča preverjevalca v veljavnost pravilne izjave.
- **»Zero knowledge« (»znanje ni razkrito«)**: med potekom dokaza ne more ne Viktor ne neka tretja oseba odkriti skrivnosti.

3. 2. PRIMER: DOKAZ O ZNANJU O KVADRATNIH OSTANKIH

Dokazovalka (Petra) bi rada preverjevalcu (Viktorju) dokazala znanje (poznavanje) kvadratnega ostanka $r \equiv s^2 \pmod{N}$. Kot dokaz mora večkrat ponoviti naslednji protokol:

1. Petra izbere naključno število q, modul N in pošlje $t \equiv q^2 \pmod{N}$ Viktorju.
2. Viktor izbere naključno število $b \in \{0,1\}$ in ga pošlje Petri.
3. Petra odgovori s $p \equiv q \cdot s^b \pmod{N}$.
4. Viktor izračuna $p^2 \pmod{N}$ in ga primerja z vrednostjo $t \cdot r^b \pmod{N}$ ter se tako prepriča o resničnosti Petrine trditve.

Če Petra ne pozna kvadratnega ostanka, potem ima vsakič le 50 %, da ji uspe. Torej, če bi protokol ponovili *k*-krat, bi nek napadalec uspel z verjetnostjo 2^{-k}. Takšen protokol se s pomočjo Fiat – Shamirjeve identifikacijske sheme, katero si bomo ogledali v nadaljevanju lahko uporabi za dokazovanje identitete, pri čemer se lažno predstavljanje zgodi z verjetnostjo največ 2^{-k}.

4. OSNOVNI PROTOKOL ZA DISKRETNI LOGARITEM

4. 1. MOTIVACIJA

Petra trdi, da pozna rešitev problema diskretnega logaritma. To pomeni, da za določena števila α, β in N pozna število x (eksponent), ki reši enačbo $\alpha^x \equiv \beta (\mathrm{mod}\, N)$. Svoje »znanje« o eksponentu x, bi rada dokazala Viktorju, vendar mu ne želi razkriti števila x. Kako lahko to stori?

Vemo, da je diskretni logaritem lahko definirati v katerikoli končni grupi. Protokol, ki ga bomo opisali v nadaljevanju je izvedljiv v katerikoli končni grupi G, v kateri tako Petra kot tudi Viktor lahko vse potrebne operacije zaključita v polinomskem času.

Jasno je, da je protokol smiseln le, če ne obstaja noben učinkovit algoritem za izračun diskretnega logaritma v tej grupi G.

4. 2. OPIS IN ANALIZA PROTOKOLA

Preden si ogledamo ta protokol, je potrebno še razjasniti pomen varnostnega parametra t.
Definicija: O varnostnem parametru t Petra in Viktor dogovorita preden začneta s komunikacijo. Pove na število ponovitev celotnega protokola. Če število ponovitev večamo, se verjetnost uspešnega goljufanja zmanjšuje eksponentno.

Na začetku interakcije se Petra in Viktor dogovorita o številih α, β in N, kjer N lahko praštevilo ali pa produkt dveh praštevil, število α je element grupe Z_N^*, število β pa je element grupe generirane z α.
Protokol, ki ga ponovita t-krat, je sledeči:

1. Petra si izbere naključno število $r \in \{1,...,\phi(N)\}$ ter izračuna $\gamma \equiv \alpha^r (\mathrm{mod}\, N)$, ki ga nato pošlje Viktorju.

2. Viktor si izbere naključni bit $b \in \{0,1\}$ in ga pošlje Petri.
3. Petra odgovori s $y \equiv r + bx (\text{mod } \phi(\text{N}))$.
4. Za potrditev Petrine trditve, da res pozna diskretni logaritem, Viktor preveri, če velja $\alpha^y \equiv \gamma\beta^b (\text{mod } N)$.

Najprej preverimo, če je protokol dobro definiran.
Ko je $b = 0$, Petra v tretjem koraku izračuna $y = r$. Torej je $\alpha^y \equiv \alpha^r \equiv \gamma (\text{mod } N)$, kar pomeni, da Viktor sprejme dokaz, saj je $\beta^b = \beta^0 = 1$.
V primeru, da je $b = 1$, je $y = r + x$ ter α^y ??α^r??α^r α^x γ $\beta (\text{mod } N)$. S tem je Petra zopet uspešno prepričala Viktorja, da pozna diskretni logaritem. Pokazali smo, da je protokol dobro definiran.

V nadaljevanju bomo dokazali veljavnost in varnost tega protokola. Varnost protokola nam zagotavlja, da Viktor ne pridobi nobenega znanja o diskretnem logaritmu, četudi goljufa. Za dokaz varnosti bomo pokazali, da obstaja simulator, ki simulira komunikacijo med Petro in Viktorjem in uporablja samo vhodne podatke, ki jih Viktor pozna na samem začetku protokola, to so α, β in *N*.
V prvem koraku Viktor prejme število γ od Petre. V drugem koraku generira naključno število *b*. Če goljufa, potem tega števila ne izbere naključno, ampak v odvisnosti od prejšnjih sporočil, ki jih je prejel oziroma poslal Petri. Predpostavljamo, da za takšen način izbire števila *b* uporablja nek algoritem. Med samo izvedbo tega algoritma lahko Viktor pridobi neke vmesne rezultate. Zaradi želje po razkritju znanja si lahko te vmesne rezultate shranjuje, recimo jim *B*. V tretjem koraku prejme Viktor do Peter še število *y*. Torej po prvi ponovitvi Viktor pridobi četverico (γ, B, b, y). Po *t* ponovitvah ima množico četveric $W_V = \{(\gamma_1, B_1, b_{1,} y_1), ..., (\gamma_t, B_t, b_{t,} y_t)\}$, ki vsebuje vse podatek, katere je Viktor pridobil med ponavljanjem protokola. W_V je slučajna spremenljivka, katere porazdelitev je odvisna od začetnih podatkov $I_P = (\alpha, \beta, N, x)$.

V kriptografskih protokolih Petra in Viktorju za komunikacijo uporabljata vsak svoj interaktivni (Turingov) stroj, recimo T_P in T_V. Zato bomo rekli, da množica W_V vsebuje vse

podatke, ki jih med ponavljanjem protokola shranil interaktivni stroj T_V. Že od preje vemo, da so to lahko vsa poslana ali prejeta sporočila, ter nekateri vmesni in končni rezultati.

Privzemimo, da za poljubne začetne podatke I_P, vrednosti W_V ležijo v neki števni množici Ω. S P_{I_P} označimo porazdelitev spremenljivke W_V po množici Ω.

Definicija: *P*-simulator, osnovan na interaktivnem stroju T_V, je Turingov stroj, ki producira četverice množice W_V^* s skoraj enako porazdelitvijo kot je porazdeljena slučajna spremenljivka W_V. Slučajna spremenljivka W_V^* je odvisna le od začetnih podatkov $I_P^* = (\alpha, \beta, N)$. Povedano z drugimi besedami: če s $P_{I_P^*}$ označimo porazdelitev slučajne spremenljivke W_V^*, potem za vse začetne podatke I_P velja:

$$\forall \omega \in \Omega \left| P_{I_P}(W_V = \omega) - P_{I_P^*}(W_V^* = \omega) \right| \leq C^{-1},$$

kjer je *C* neka poljubna konstanta in $C > 1$.

Trditev: Če Viktor ne goljufa in Petra ne pozna diskretnega logaritma *x*, potem je verjetnost, da Viktor ugotovi, da Petra goljufa večja kvečjemu enaka $1 - 2^{-t}$.

Dokaz:
Če Petra res ne pozna diskretnega logaritma, potem v 3. koraku Viktorju ne pošlje pravilnega števila *y* vsaj enem primeru vrednosti bita *b*. V vsaki ponovitvi protokola jo Viktor zasači z verjetnostjo enako vsaj $\frac{1}{2}$. Kar pomeni, da je po *t* ponovitvah ta verjetnost večja ali enaka $1 - 2^{-t}$.
S tem smo dokazali veljavnost protokola.

Trditev: Če Petra ne goljufa, potem za poljuben interakcijski stroj, ki deluje naključno s polinomsko časovno zahtevnostjo, obstaja *P*-simulator s polinomsko časovno zahtevnostjo.

Dokaz:

Naj bo T_V interakcijski stroj, ki deluje naključno s polinomsko časovno zahtevnostjo, katerega uporablja Viktor. Privzemimo, da je varnostni parameter t = 1, torej gledamo samo eno ponovitev. Imamo naslednje podatke $I_P = (\alpha, \beta, N, x)$, $I_P^* = (\alpha, \beta, N)$ ter $W_V = (\gamma, B, b, y)$, kjer: je γ sporočilo, ki ga Viktor sprejme v 1. koraku; b je bit, ki ga generira T_V **;** B vsebuje vmesne korake določanja števila b, ki jih shranjuje T_V ; in y je število, ki ga Viktor prejme v 3. koraku. Definirajmo *P*-simulator s polinomsko časovno zahtevnostjo:

Naslednje korake ponovi največ L – krat, kjer je $L := \log N / \log 2$.

1. Izberi si naključen bit $c \in \{0,1\}$.
2. Izberi si naključno število $y \in \{0,1,..., N-2\}$.
3. Izračunaj $\gamma = \alpha^y \beta^{-c}$.
4. Z uporabo interakcijskega stroja T_V določi $b \in \{0,1\}$, B naj vsebuje shranjene vmesne rezultate.
5. Če je $b = c$, izpiši $W_v^* = (\gamma, B, b, y)$.

Ponavljaj, dokler je $b = c$.

Če je $b \neq c$ izpiši W_V^* = »neuspešno«.

Simulator vse operacije opravi v polinomskem času.

Predpostavimo, da je N praštevilo. Oglejmo si eno ponovitev vseh petih korakov. Število γ je element ciklične grupe generirane z α, števili γ in c sta medsebojno neodvisni. Ker je bit b generiran v odvisnosti od γ, sta zato tudi b in c neodvisna. Zato je verjetnost, da sta enaka $\frac{1}{2}$. Torej je verjetnost, da je $b = c$ vsaj v eni izmed L-tih ponovitev algoritma enaka najmanj $1 - N^{-1}$. Naj bo Ω množica vseh vrednosti, ki jih lahko zavzame W_V^*, vključno z sporočilom »neuspešno«. Za vsak $\omega \in \Omega$ razen $\omega =$ »neuspešno« velja

$$P_{I_P^*}(W_V^* = \omega \mid W_V^* \neq \text{"neuspešno"}) = P_{I_P}(W_V = \omega).$$

Če še upoštevamo dejstvo, da je $P_{I_P^*}(W_V^* = \text{"neuspešno"}) \leq N^{-1}$, dobimo da je

$$S := \sum_{\omega \in \Omega} \left| P_{I_P}(W_V = \omega) - P_{I_P^*}(W_V^* = \omega) \right| \leq 2N^{-1}.$$

Ker je dolžina niza I_P sorazmerna z $\log N$, je trditev dokazana, če je N praštevilo.

Recimo, da je *N* sestavljeno število, $N = P_1 P_2$, kjer sta praštevili P_1 in P_2 reda velikosti $N^{1/2}$. Potem $N-1$ ni večkratnik reda števila $\alpha \in \mathbf{Z}_N^*$, zato ni nujno, da število γ, ki ga računamo v 3. koraku algoritma element ciklične grupe generirane z α. Vendar, vsi zgoraj omenjeni argumenti držijo, če upoštevamo, da je $0 \leq y \leq \phi(N)-1$. Če upoštevamo, da je $P_{I_P^*}(\phi(N) \leq y \leq N-2) = O(N^{-1/2})$, potem od tod sledi, da je *S* navzgor omejena z $O(N^{-1/2})$.

V primeru da je varnostni parameter $t > 1$, potem se zgoraj opisani simulator ponovi *t*–krat. Torej se čas delovanje poveča za faktor *t*, vsota *S* pa za faktor, ki je manjši kvečjemu enak parametru *t*. Ker pa je število ponovitev navzgor omejeno $\log N$, imamo še vedno polinomsko časovno zahtevnost. Torej je trditev in s tem tudi polnost dokazana.

5. POSPLOŠITEV PROTOKOLA ZA DISKRETNI LOGARITEM

5. 1. PREDSTAVITEV PROBLEMA

Poznamo več posplošitev protokola. Mi si bomo ogledali posplošitev za naslednjo situacijo. Imamo naravno število *N*, praštevilo ali sestavljeno, ter poljubno število iz grupe ostankov pri deljenju z *N*, $\alpha \in \mathbf{Z}_N^*$. Vzemimo števila $\beta_1,...,\beta_k$, ki so elementi grupe generirane z α. Iščemo takšne eksponente $x_1,\ldots,x_k$ (diskretne logaritme), ki rešijo naslednje enačbe: $\alpha^{x_1} \equiv \beta_1 (\text{mod } N),...,\alpha^{x_k} \equiv \beta_k (\text{mod } N)$.

5. 2. OPIS IN ANALIZA PROTOKOLA

Podobno kot pri osnovnem protokolu se morata Petra in Viktor dogovoriti o začetnih oziroma vhodnih podatkih: $\alpha, \beta_1,..., \beta_k, N$. Petra bi rada Viktorja prepričala, da pozna diskretne logaritme, vendar mu jih noče izdati.
Varnostni *t* parameter poznamo že od prej, pove nam kolikokrat Petra in Viktor ponovita protokol::

1. Petra si izbere naključno število $r \in \{1,...,\phi(N)\}$ ter izračuna $\gamma \equiv \alpha^r (\text{mod } N)$, ki ga nato pošlje Viktorju.
2. Viktor si izbere *k* naključnih bitov $b_i \in \{0,1\}$ in jih pošlje Petri.
3. Petra izračuna $y \equiv r + b_1 x_1 + \ldots + b_k x_k (\text{mod } \phi(\text{N}))$ in ga sporoči Viktorju.
4. Za potrditev Petrine trditve, da res pozna diskretni logaritem, Viktor preveri, če velja $\alpha^y \equiv \gamma \beta_1^{b_1} \ldots \beta_k^{b_k} (\text{mod } N)$.

Da je protokol dobro definiran, ne bomo preverjali, saj je podobno kot pri osnovnem protokolu.

Trditev: Če Viktor ne goljufa in Petra ne pozna vsaj enega izmed diskretnih logaritmov x_1, K, x_k, potem je verjetnost, da Viktor ugotovi, da Petra goljufa večja kvečjemu enaka $1 - 2^{-t}$.

Trditev: Če Petra ne goljufa, potem za poljuben interakcijski stroj, ki deluje naključno s polinomsko časovno zahtevnostjo, katerega uporablja Viktor, obstaja *P*-simulator s polinomsko časovno zahtevnostjo.

Dokaza teh dveh trditev sta posplošitvi dokazov trditev iz prejšnjega poglavja. Tako, da bomo kar brez dokaza privzeli, da trditvi zanima. Sicer pa lahko oba dokaza najdemo v članku An improved protocol for demonstrating possession of discrete logarithms and some generalizations (za več glejte reference).

6. DISKRETNI LOGARITEM IN IDENTIFIKACIJSKE SHEME

6. 1. IDENTIFIKACIJSKE SHEME

Identifikacijske sheme so uporabne v velikih sistemih, kjer uporabniki niso seznanjeni eden z drugim. Vsakdanji primer je situacija potrošnik – prodajalec, ali pa v računalniškem svetu elektronska pošta.
V tem poglavju si bomo pogledali identifikacijske shem, ki temeljijo na dokazih brez razkritja znanja. Vemo, da obstajajo tudi druge vrste, nekatere bazirajo samo na medsebojni komunikaciji s skrivnimi ključi, pri nekaterih pa se pojavijo tudi javni ključi.

Identifikacijsko shemo sestavljata javna datoteka, ki vsebuje zapise o vseh uporabnikih in identifikacijski protokol. Vsak zapis sestavlja ime oziroma identiteta uporabnika in pomožne informacije o identiteti., ki se nato uporabi pri zagonu identifikacijskega protokola.
Preden si ogledamo še kak identifikacijski protokol, si oglejmo kako to sploh poteka.
Recimo, da Petra želi dokazati Viktorju, da zares komunicira z njo. Petra zato pokliče identifikacijski protokol z zapisom, ki ustreza njenemu imenu, kot parametrom. Viktor preveri, da parameter, ki je v uporabi, resnično ustreza Petrinemu javnemu zapisu in nato nadaljuje s protokolom. Pri tem zahtevamo, da lahko Petra vedno prepriča Viktorja (da je to zares ona), medtem ko nihče drug ne more pretentati Viktorja, da bi verjel, da komunicira s Petro. Neka tretja oseba se nikoli ne more predstaviti kot Petra, tudi po prejetju polinomsko mnogo dokazov o identiteti.

Identifikacijske sheme morajo torej zagotoviti, da se nihče neki tretji osebi ne more predstaviti kot Petra ali Viktor.
Vendar kljub temu ne moremo zagotoviti popolne varnosti. Petra lahko proda ali posodi svoj privatni ključ neki tretji osebi. To pa deluje v vsakem primeru, saj pri dokazih brez razkritja znanja potrebujemo le del informacije. Verjetno je to največji problem identifikacijskih shem.

6. 2. SCHNORR-ov IDENTIFIKACIJSKI PROTOKOL

Protokol, ki smo ga opisali v 4. poglavju imenujemo tudi Schnorrov identifikacijski protokol. Vemo že, da ta protokol temelji na poznavanju diskretnega logaritma. Uporaba teh protokolov je zanimiva le v primerih, ko sta števili *t* ter 2^k omejeno z $\log N$, kjer je *t* varnostni parameter oziroma število ponovitev protokola, *k* pa je število eksponentov iz protokola v 4. poglavju.

Schnorrov identifikacijski protokol je mogoče uporabiti pri identifikacijskih shemah. Za to potrebujemo nek varnostni center, ki si naključno izbere števila $x_1,\mathsf{K}\,,x_k$, in jih sporoči samo Petri. Prav tako ta center izračuna vrednosti $\alpha^{x_1} \equiv \beta_1 (\mathrm{mod}\, N),...,\alpha^{x_k} \equiv \beta_k (\mathrm{mod}\, N)$, kjer sta števili α in *N* javna parametra, števila $\beta_1,...,\beta_k$ pa so Petrin javni ključ. Pri tem je potrebno opozoriti, da se lahko zraven pojavijo še kake zgoščevalne funkcije, ki pa niso predmet tega sestavka. Viktor ima dostop do števil $\beta_1,...,\beta_k$, pozna oba javna parametra α in *N*, ne pozna pa eksponentov $x_1,\mathsf{K}\,,x_k$. Z uporabo protokola, ki smo ga opisali v 4. poglavju, lahko sedaj Petra prepriča Viktorja, da res komunicira z njo oziroma se mu identificira. Pri tem mu ne razkrije svojega skrivnega ključa, prav tako pa se nihče drug ne more lažno predstavljati kot Petra. Seveda lahko Viktorja nadomestimo z bančnim avtomatom (sicer ne delujejo na ta način), z raznimi sistemi, do katerih lahko dostopimo le z identifikacijo, ipd.

6. 3. FIAT-SHAMIR-jev IDENTIFIKACIJSKI PROTOKOL

Na kratko si bomo ogledali še Fiat-Shamir-jev identifikacijski protokol, ker naj bi bil bolj uporaben pri identifikacijskih shemah kot pa zgoraj omenjeni Schnorr-ov protokol. Fiat-Shamirjev identifikacijski protokol je v bistvu posplošitev dokaza o znanju o kvadratnih ostankih, ki smo ga opisali v razdelku **3. 2.**

Varnostni center objavi javni parameter *N*, ki je sestavljeno število. Razcep tega števila je skrivnost, ni objavljen nikjer, poznan je samo varnostnemu centru. Zaradi lažje primerjave bomo uporabili podobne oznake kot pri Schnorr-ovem protokolu.

Varnostni center generira števila $x_1, \ldots, x_k$, ki jih sporoči samo Petri. Pri tem lahko upošteva njene osebne podatke (ime, priimek, naslov,…). Ta števila so Petrin zasebni ključ. Center nato izračuna Petrin javni ključ $\beta_1 \equiv x_1^2 \pmod N, \ldots, \beta_k \equiv x_k^2$, ki jih shrani v nek javno dostopen dokument. Če se hoče Viktor sedaj prepričati, da je na drugi strani res Petra, požene naslednji protokol, ki v *i*-ti ponovitvi vsebuje naslednje korake:

1. Petra si izbere naključno število $r_i \in [0, n)$ in izračuna $\gamma_i \equiv r_i^2 \pmod N$ ter ga pošlje Viktorju.
2. Viktor si izbere naključni binarni vektor $(b_{i1}, \ldots, b_{ik})$, kjer je $b_{ij} \in \{0,1\}$, ter ga pošlje Petri.
3. Petra izračuna $y_i \equiv r_i \prod_{e_{ij}=1} x_j \pmod N$ in ga pošlje Viktorju.
4. Viktor sprejme dokaz za pravilen, če velja $y_i^2 \equiv \gamma_i \prod_{e_{ij}=1} \beta_j \pmod N$.

Da je protokol dobro definiran, ne bomo dokazovali. Omenimo samo to, da Viktor zopet nič ne izve o kvadratnih ostankih. Pri zadnjem preverjanju računa s števili, ki mu jih je poslala Petra oziroma z njenim javnim ključem.

6. 4. PRIMERJAVA SCHNORR-ovega IN FIAT-SHAMIR-jevega IDENTIFIKACIJSKEGA PROTOKOLA

Glavno razliko med njima smo opazili že pri samem opisu, prvi temelji na računanju diskretnega logaritma, drugi pa na kvadratnem ostanku oziroma na računanju kvadratnega korena. V obeh primerih nam nastopata tako javni ključ, $\beta_1, \ldots, \beta_k$, kot tudi zasebni ključ, eksponenti oziroma koreni $x_1, \ldots, x_k$.

V Fiat-Shamir-jevi identifikacijski shemi je vhodni oziroma začetni podatek *N* sestavljeno število, torej mora varnostni center poskrbeti, da njegov razcep ostane skriven. Poskrbeti mora, da nihče ne more dostopiti do njega. Kar je nekoliko manj varno, kot pa pri Schnorr-vi shemi, ki lahko uporablja za modul praštevila.

Po drugi strani Fiat in Shamir trdita, da v njuni shemi varnostni center lahko generira javni ključ s pomočjo kake javne funkcije v odvisnosti od Petrinih osebnih podatkov (ime, priimek, …), katere Petra sporoči. Torej lahko vsak preverjevalec sam generira Petrin javni ključ, če pozna javno funkcijo in njene podatke. Zato sploh ni potrebno, da je Petrin javni ključ

shranjen v nekem javno dostopnem dokumentu. Javna funkcija mora biti takšna, da lahko samo varnostni center, ki pozna razcep modula, izračuna kvadratne korene nekaterih izhodnih podatkov, ki jih le-ta generira.

Izkaže se, da je Fiat-Shamir-jeva shema učinkovitejša ter uporabnejša koz Schnorr-ova, saj zahteva le operacijo kvadriranja, medtem ko druga temelji na potenciranju, kar pa je časovno zahtevnejše.

7. ZAKLJUČEK

Oba protokola, ki smo jih opisali in medsebojno primerjali se danes uporabljata poleg identifikacijskih shem uporabljata tudi pri digitalnem podpisovanju. Pri tem gre za še eno zahtevo več kot pri identifikacijskih shemah. Omenili smo, da se Viktor ne more neke 3. osebi izdajati za Petro. Digitalno podpisovanje pa zahteva še, da tudi se tudi samemu sebi ne more predstaviti kot Petra. Če želimo identifikacijsko shemo nadgraditi v shemo za podpisovanje, moramo vlogo Viktorja, kjer generira določene podatke, nadomestiti z neko funkcijo. Ponavadi se zato uporabljajo zgoščevalne funkcije.
Zgoščevalne funkcije je možno uporabiti tudi pri Fiat-Shamir-jevi shemi za računanje začetnih oziroma vhodnih podatkov. Bolj natančno za Petrin javni ključ.

Ta dva protokola seveda nista edina, ki se lahko uporabljajo v identifikacijskih shemah. Če Fiat-Shamir-jeva shema temelji na računanju kvadratnega korena. Poznamo tudi Guillou-Quisquater-jevo shemo, katere osnova je računanje poljubnega korena in je v bistvu posplošitev prve sheme. Možno je posplošiti tudi protokol za diskretni logaritem, in sicer da namesto ene osnove uporabimo več različnih
Seveda je možno te sheme še bolj posplošiti. Namesto naključnih binarnih števil lahko vzamemo katera druga, ki niso binarna.
Še bolj zanimiva posplošitev pa je, da se lahko za identifikacijsko shemo uporabi katerikoli primer NP polnega problema. Seveda pa se tu poraja vprašanje njihove praktične uporabnosti.

8. REFERENCE

- A. Fiat, A. Shamir, How to prove yourself: Practical solutions to identification and signature problems, Crypto '86 Proceedings, Springer-Verlag, Berlin Heidelberg, 1986

- D. Chaum, J.H. Evertse, J. van de Graaf, An improved protocol for demonstrating possession of discrete logarithms and some generalizations, Eurocrypt '87, Springer-Verlag, Berlin Heidelberg, 1987

- B. Škrlep, Dokazi brez razkritja znanja, diplomsko delo, Ljubljana, 2000

- D. R. Stinson, Cryptography: theory and practice, Boca Raton: CRC, 1995

- A. Jurišić, Dokazi brez razkritja znanja (tečaj iz kriptografije), http://valjhun.fmf.uni-lj.si/~ajurisic, 1999

- N. Li, J. Li, Cryptography CS 555, Lecture 18 (študijsko gradivo) http://www.cs.purdue.edu/homes/ninghui/courses/Spring04/lectures/lect18.pdf, 2004

- G. Dirschl, Zero-knowledge protocols, Advanced Systems (seminar) http://i30www.ira.uka.de/teaching/coursedocuments/84/Dirschl_Zero-Knowledge-Protocols.pdf, 2003

- H. A. Aronsson, Network security: Zero knowledge protocols and small systems, http://www.tml.hut.fi/Opinnot/Tik-110.501/1995/zeroknowledge, Helsinki

- A. Giani, Identification with zero knowledge protocols, http://www.sans.org/rr/papers/20/719.pdf, SANS Institute 2001