DIGITALNI PODPISI V KRIPTOGRAFIJI, 4. del

V tretjem delu smo predstavili koncept **kriptosistemov z javnimi ključi**, ki so ga leta 1976 odkrili W. Diffie, M. Hellman in R. Merkle (slednji je bil takrat še študent!). Ob njem smo predstavili tudi idejo **digitalnega podpisa**.

> *Z digitalnim podpisom potrjujemo izvor podatkov ali pa nekoga prepričamo, da je podpis opravil lastnik ustreznega zasebnega ključa.*

V tem sestavku bomo predstavili nekaj konkretnih kriptografskih protokolov. Po razmisleku o razlikah lastnoročnega in digitalnega podpisa bomo opisali ElGamalovo shemo za digitalni podpis.

> *Konceptualno se način zapisovanja informacij ni drastično spremenil. Medtem ko smo prej shranjevali in prenašali informacije na papirju, jih sedaj hranimo na magnetnih in drugih medijih ter jih prenašamo preko telekomunikacijskih sistemov (tudi brezžičnih). Bistveno pa se je spremenila možnost kopiranja, prenašanja in spreminjanja informacij. Zlahka naredimo na tisoče kopij neke digitalne informacije in jih v trenutku spravimo na različne konce sveta, pri tem pa se nobena od kopij prav nič ne razlikuje od originala. Z informacijo na papirju je vse to precej težje, če že ne nemogoče.*

Družba, v kateri so informacije spravljene in prenašane v digitalni obliki, mora poskrbeti za to, da ne bo varnost informacij odvisna od fizičnega medija, ki jih je zapisal ali prenesel. Temeljiti mora izključno na digitalni vsebini. Eno izmed osrednjih orodij pri zaščiti informacij je **podpis**.

Le-ta preprečuje poneverjanje in je dokaz o izvoru, identifikaciji, pričanju. Podpis naj bi bil unikat vsakega posameznika, saj se z njim predstavimo, nekaj potrdimo, nekoga pooblastimo,... Z razvojem digitalne informacije moramo ponovno obdelati tudi idejo oziroma koncept podpisa. Torej imamo enkratno priložnost, da s pravo idejo vplivamo na tok zgodovine.

1. (STARI) LASTNOROČNI PODPIS

Pogled v bližnjo prihodnjost nam razkrije nekaj pomanjkljivosti takega podpisovanja:

J: "Danes sem dobil podpis v šoli," pove Janezek sestri.
S: "Kakšno neumnost si pa zopet naredil?"
J: "Te nič ne briga, raje mi svetuj, kako naj prelisičim mamo."
S: "Nad učiteljev podpis skopiraj obvestilo o športnem dnevu."
J: "Odlična ideja, na ta način se bom rešil zagate,
pa še smučat grem lahko, namesto da bi šel v šolo."

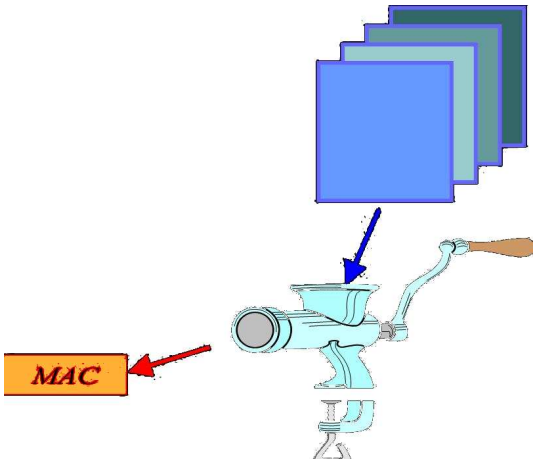
Janezek to naredi in mama pod digitalno obvestilo o smučarskem dnevu pripne svoj "digitalni" podpis. Preden pa ga na ključku odnese v šolo, Janezek zopet zamenja obvestilo o športnem dnevu s starim tekstom. V resnici je škoda za starše še večja, saj ima sedaj Janezek mamin "digitalni" podpis in ga bo odslej lahko uporabljal po mili volji.

Kot kaže, mora biti postopek digitalnega podpisa dobro premišljen, saj moramo preprečiti spreminjanje vsebine podpisanega sporočila in ponarejanje oziroma kopiranje podpisa. Elektronski podpis ni več unikat, ki enolično določa podpisnika, kajti elektronsko kopiranje podpisa je tako lahko, da je skoraj trivialno na nepodpisan dokument pripeti poljuben podpis. Potrebujemo protokole, ki imajo podobne lastnosti kot trenutni "papirni protokoli". Družba ima enkratno priložnost, da vpelje nove in učinkovitejše načine, ki nam bodo zagotovili varnost informacij.

Digitalni podpis naj bi bil nadomestek za lastnoročni podpis pri elektronski izmenjavi in digitalnemu hranjenju podatkov.

Digitalni podpis mora imeti vse lastnosti, ki veljajo za ročni podpis, poleg tega pa mora tudi veljati, da vsebine digitalno podpisanega dokumenta ni mogoče spreminjati in podpisa ni mogoče kopirati in ponarejati. Med obema oblikama podpisovanja pa predvidimo še nekaj bistvenih razlik:

- navadni podpis je fizično del podpisanega dokumenta;
- navadni podpis preverjamo s primerjanjem, digitalnega pa z algoritmom katerega rezultat je odvisen od ključa in dokumenta;
- digitalnega podpisa ne moremo razlikovati od njegove kopije, zato si želimo podpis, ki bo vedno drugačen (četudi podpišemo isto stvar);
- digitalni podpis je odvisen od dokumenta, ki ga podpisujemo (ker so dokumenti poljubno veliki, običajno iz dokumenta skonstruiramo izvleček fiksne dolžine in nato podpišemo le izvleček).



2. (NOVI) DIGITALNI PODPIS

Algoritem digitalnega podpisa je sestavljen iz treh delov: iz algoritma za generiranje ključa, algoritma za generiranje digitalnega podpisa in algoritma za preverjanje digitalnega podpisa. Za lažje razumevanje definirajmo nekaj osnovnih pojmov.

- Pravilo sig_A, ki sporočilu priredi podpis, imenujemo *funkcija za podpis* osebe A.
  Ključ zanjo varuje oseba A in jo uporablja za podpisovanje sporočil.
- Pravilu ver_A, ki sporočilu in podpisu priredi vrednost "veljaven" oziroma "neveljaven", rečemo *funkcija za preverjanje podpisov* osebe A. Ključ zanjo je javno znan. Uporabljajo se za preverjanje podpisov, ki jih je opravila oseba A.

Postopek pošiljanja digitalno podpisanega sporočila med Anito in Bojanom je sledeč.

1. Najprej si Bojan izbere svoj zasebni ključ ter sporoči Aniti ustrezni javni ključ.
2. Da bi podpisal sporočilo x, Bojan uporabi algoritem za generiranje digitalnega podpisa s svojim zasebnim ključem in izračuna podpis $\text{sig}_B(x)$.
3. Nato pošlje sporočilo skupaj z njegovim digitalnim podpisom Aniti.
4. Ker Anita pozna Bojanov javni ključ, lahko uporabi algoritem za preverjanje digitalnega podpisa ver_B ter tako preveri pristnost podpisa.

Vrstni red šifriranja in digitalnega podpisovanja je pomemben. Če hoče Anita poslati Bojanu podpisano, zašifrirano sporočilo, potem danemu čistopisu x najprej izračuna svoj podpis $y = \text{sig}_{Anita}(x)$, nato zašifrira x in y z Bojanovo javno šifrirno funkcijo e_{Bojan} in dobi $z = e_{Bojan}(x, y)$. Tajnopis z pošlje Bojanu. Ta ga odšifrira s svojo zasebno odšifrirno funkcijo d_{Bojan} in dobi par (x, y). Potem uporabi Anitino javno funkcijo ver_{Anita}, da preveri, če res velja $\text{ver}_{Anita}(x, y) =$ veljaven.

Če pa bi Anita najprej zašifrirala sporočilo x in potem podpisala rezultat, bi izračunala $z = e_{Bojan}(x)$ in $y = \text{sig}_{Anita}(z)$ ter par (z, y) poslala Bojanu. Bojan bi z odšifriranjem tajnopisa z dobil x, nato bi preveril podpis y na x z uporabo funkcije ver_{Anita}. V tem primeru pride do problema, če Oskar prestreže takšen par (z, y), saj lahko zamenja Anitin podpis s svojim $y' = \text{sig}_{Oskar}(z)$, čeprav ne pozna čistopisa x. Ko pošlje (z, y') Bojanu, bo ta mislil, da mu je sporočilo x poslal Oskar. Zato se priporoča najprej podpisovanje in nato šifriranje sporočil.

Če je Anita edina, ki lahko podpiše sporočilo s svojim podpisom, torej edina, ki zna pri danem x izračunati y, tako da velja $\text{ver}_{Anita}(x, y) =$ veljaven in če se je Bojan sposoben prepričati, ali gre za Anitin podpis ali pa gre za ponaredek, rečemo, da je algoritem digitalnega podpisa *varen*.

3. ELGAMALOV DIGITALNI PODPIS

Kot primer opišimo shemo za podpis, ki temelji na problemu diskretnega logaritma in Diffie-Hellmanovega dogovora o ključu. Razvil jo je Taher ElGamal leta 1985.

ElGamalov algoritem. *Naj bo p tako praštevilo, da je v $\mathbb{Z}_p$ težko izračunati diskretni logaritem in α generator multiplikativne grupe $\mathbb{Z}_p^*$, tj. $\mathbb{Z}_p^* = \{\alpha^0, \alpha^1, \ldots, \alpha^{p-2}\}$. Naj bo še $\mathcal{P} = \mathbb{Z}_p^*$ množica sporočil in*

$$\mathcal{K} = \{(p, \alpha, a, \beta);\ \beta = \alpha^a \bmod p\}$$

množica ključev. Število a je skrito (zasebno), števila p, α in β pa so javno znana.

Podpisovanje: *podpisnik izbere naključno skrito število $k \in \mathbb{Z}_{p-1}^*$ in s ključem $K = (p, \alpha, a, \beta)$ izračuna $\mathrm{sig}_K(x, k) = (\gamma, \delta)$, kjer je*

$$\gamma = \alpha^k \bmod p \qquad \text{in} \qquad \delta = (x - a\gamma)k^{-1} \bmod (p-1).$$

Preverjanje podpisa *(samo z javnimi števili p, α in β):*

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

Slika 1: Taher ElGamal.

SASA2JASMINA: Tule bi se spodobilo, da preverimo, zakaj je takšno preverjanje podpisa pravilno! Hkrati pojasni tudi zakaj enkrat po mod p in enkrat po mod $p-1$.

Zgornji podpis je nedeterminističen (odvisen od naključnega števila k), torej sploh ni natanko določen.

Vidimo, da podpisnik izračuna podpis z uporabo tako tajne vrednosti a, ki je del ključa kot tajnega naključnega števila k, ki se sme uporabiti samo za podpis enega sporočila x. (Preverjanje pa je opravljeno samo z uporabo javnih informacij.) Če namreč naključno število k ne ostane skrito, ali pa se isto število k uporabi v podpisih dveh različnih sporočil (v tem primeru ga je možno zlahka izračunati), lahko napadalec iz druge enačbe algoritma za podpisovanje izračuna tajno vrednost $a = (x - k\delta)\gamma^{-1} \bmod (p-1)$ in ponaredi podpis. ElGamalovo shemo za podpis smatramo za varno. Do danes namreč še nikomur ni uspelo učinkovito izračunati par (γ, δ) brez računanja diskretnega logaritma. Lahko pa se zgodi, da bomo enkrat ugotovili, da se pri iskanju para (γ, δ) problemu diskretnega logaritma sploh ne moremo izogniti.

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

1. Za dano sporočilo x je potrebno najti tak par (γ, δ), da bo veljalo $\beta^\gamma \gamma^\delta \equiv \alpha^x \pmod p$, torej
   - če izberemo γ: rabimo $\delta = \log_\gamma \alpha^x \beta^{-\gamma} \pmod p$,
   - če izberemo δ: glede na γ je potrebno rešiti enačbo $\beta^\gamma \gamma^\delta \equiv \alpha^x \pmod p$,
   - hkrati računamo γ in δ (zaenkrat ni še nihče odkril hitrega postopka za reševanje zgornje enačbe).
2. Za podpis (γ, δ) je potrebno najti ustrezno sporočilo x: $x = \log_\alpha \beta^\gamma \gamma^\delta \pmod p$.

NALOGE

1. Podpisovalec ni bil pazljiv in je ponesreči izgubil naključno število k, ki ga je uporabil pri ElGamalovem podpisu. Uporabi njegovo napako za izračun zasebnega kluča a?

2. Generator naključnih število je tako počasen, da se je podpisovalec odločil uporabiti število k dvakrat. Uporabi njegovo napako za izračun zasebnega kluča a?

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

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

   zadoščajo enačbi za preverjanje ElGamalovega podpisa: $\beta^\gamma \gamma^\delta \equiv \alpha^x \pmod{p}$.

4. Ali lahko pri veljavnem podpisu (γ, δ) za x najdemo še kakšen podpis za neko drugo sporočilo x'?

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

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

5. še veliko možnosti

Aleksandar Jurišić in Jasmina Veselinović

4. CERTIFIKATI

Še večji problem pa je s pristnostjo javnih ključev.

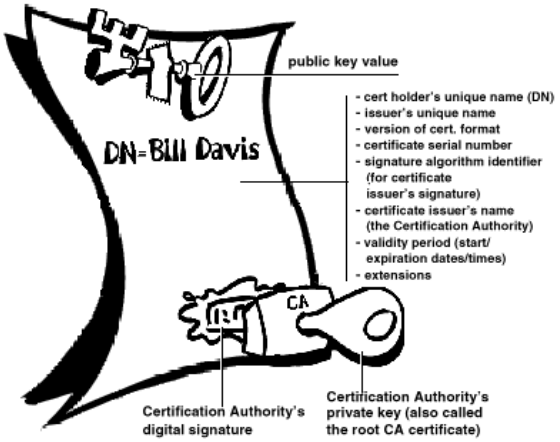
> *Če bi vam nekdo med spanjem v spominu zamenjal sliko vaše babice, potem bi bili zjutraj pripravljeni spustiti v stanovanje povsem tujo ženico, misleč, da gre za babico.*

V drugem delu smo pisali o DH-dogovoru o ključu in spoznali, da je občutljiv na napad srednjega moža. Videli smo, da se v protokolu ne da preveriti identitete udeležencev, kajti ne moremo biti prepričani o avtentičnosti njihovih javnih ključev. Ta pomanjkljivost omogoči srednjemu napadalcu, da prestreže podatke in jih zamenja s svojimi. Javne ključe moramo nekako povezati z lastniki zasebnih ključev. Ta problem rešimo tako, da se udeleženci in javne vrednosti, ki nastopajo v protokolu, ustrezno overijo z digitalnimi podpisi.

Certifikat je digitalno potrdilo, ki poveže javni ključ z identiteto osebe, ki ima pripadajoči zasebni ključ. Vsebuje podatke osebe, njen javni ključ, omejitve uporabe ter druge podatke in je podpisan s strani agencije, ki ga je izdala. Takšni zaupanja vredni agenciji rečemo **certifikatna agencija** in jo označimo s CA. Le-ta preveri resničnost podatkov navedenih v certifikatu in jih s svojim podpisom potrdi. Uporabniki sistema pa preverjajo podpise certifikatne agencije in v ta namen potrebujejo njen javni ključ. Šele nato zaupajo certifikatu. Javni ključi certifikatnih agencij so dostopni vsem. Nekatere najbolj priznane certifikatne agencije pa imajo svoje javne ključe že vgrajeni v spletne brskalnike in operacijske sisteme. Na ta način lahko vsakdo preveri veljavnost certifikata. Podatkom na certifikatu torej zaupamo, če zaupamo certifikatni agenciji, ki ga je podpisala (glej nadaljevanje zgodbe o volku in sedmih kozličkih).

Bodimo bolj natančni. Z $\mathrm{ID}(U)$ označimo informacijo, ki enolično identificira osebo U (npr. ime, e-pošta, telefonska številka, itd.). Oseba U si naključno izbere svoj zasebni ključ a_U in z njim izračuna še svoj javni ključ b_U. Agencija CA si na začetku izbre shemo za digitalni podpis. S tem je določen algoritem za podpisovanje $\mathrm{sig}_{\mathrm{CA}}$, ki uporablja master ključ. Le-ta je najbolj varovana skrivnost. Po možnosti nikoli ne zapusti varne sobe, ali pa zanj uporabimo algoritme za deljenje skrivnosti (glej Presek, letnik 29, številka 6). Določen pa je tudi algoritem za preverjanje podpisov $\mathrm{ver}_{\mathrm{CA}}$, ki uporablja javni ključ agencije. Agencija izda osebi U certifikat

$$C(U) = (\mathrm{ID}(U),\, b_U,\, \mathrm{sig}_{\mathrm{CA}}(\mathrm{ID}(U),\, b_U))$$



5. OVERJEN DH-DOGOVOR O KLJUČU

V drugem delu smo opisali DH-dogovor o ključu, tokrat pa bomo predstavili **overjen Diffie-Hellmanov dogovor o ključu**, v katerega je vključena tudi certifikatna agencija. Ta overi javne podatke uporabnikom, tako da lahko vsak par uporabnikov kasneje izračuna ključ, ki ni poznan ostalim, hkrati pa nikomur ne more biti vsiljen podtaknjen ključ. Shema tega dogovora temelji na problemu diskretnega logaritma. Za demonstracijo uporabimo kar multiplikativno grupo $\mathbb{Z}_p^*$, kjer je p zadosti veliko praštevilo in α njen generator. Privzamemo lahko, da sta vrednosti p in α javno znani. Anita in Bojan si naključno izbereta tudi vsak svoj zasebni ključ, Anita a_A, Bojan pa a_B, kar simbolično zapišemo na naslednji način $a_A, a_B \in_R \{1, \ldots, p-2\}$, ter nato izračunata še vsak svoj javni ključ:

$$b_A = \alpha^{a_A} \bmod p \qquad \text{in} \qquad b_B = \alpha^{a_B} \bmod p,$$

Na osnovi identifikacijskih vrednosti Anite in Bojana, tj. $\mathrm{ID}(A)$ in $\mathrm{ID}(B)$, jima agencija izda naslednja certifikata:

$$C(A) = (\mathrm{ID}(A),\, b_A,\, \mathrm{sig}_{\mathrm{CA}}(\mathrm{ID}(A),\, b_A)) \quad \text{in} \quad C(B) = (\mathrm{ID}(B),\, b_B,\, \mathrm{sig}_{\mathrm{CA}}(\mathrm{ID}(B),\, b_B))\,.$$

Tako Bojan kot Anita lahko potem, ko sta si pridobila certifikata drug od drugega in preverila veljavnost podpisa CA, izračunata skupni tajni ključ, on z uporabo javnega ključa b_A iz Anitinega certifikata in s svojim zasebnim ključem a_B, ona pa z uporabo javnega ključa b_B iz Bojanovega certifikata in s svojim zasebnim ključem a_A:

$$(b_A)^{a_B} = (\alpha^{a_A})^{a_B} \bmod p \quad \text{in} \quad (b_B)^{a_A} = (\alpha^{a_B})^{a_A} \bmod p.$$

Sedaj, ko imata oba skupni tajni ključ $K_{A,B} = \alpha^{a_B a_A}$, si lahko pošiljata z njim šifrirana sporočila in jih po odšifritanju bereta.

Razmislimo o varnosti te sheme. Podpis agencije CA na certifikatih uporabnikov preprečuje Oskarju, da spreminja certifikate. Ključno vprašanje je: ali lahko Oskar izračuna $K_{A,B}$? Z drugimi besedami: ali je mogoče izračunati $\alpha^{a_A a_B} \bmod p$, če poznamo p, α, $\alpha^{a_A} \bmod p$ in $\alpha^{a_B} \bmod p$, ne pa tudi a_A ali a_B? Temu rečemo **Diffie-Hellmanov problem**. Očitno je overjen Diffie-Hellmanov dogovor o ključu varen natanko tedaj, ko je varen (oz. težko rešljiv) Diffie-Hellmanov problem.

Če Oskar lahko določi a_A iz b_A ali a_B iz b_B, potem lahko izračuna $K_{A,B}$. Oba izračuna pa sta primera problema diskretnega logaritma in če poskrbimo, da je problem diskretnega logaritma v $\mathbb{Z}_p$ težko rešljiv, je overjen Diffie-Hellmanov dogovor o ključev varen pred takimi vrstami napadov.

Anita in Bojan bi bila rada prepričana, da pri medsebojni izmenjavi sporočil ni prišlo do dogovora o ključu z napadalcem Oskarjem. Zato poskrbita, da se, medtem ko se dogovarjata o ključu, tudi identificirata, tj. overita ključ. Takemu protokolu rečemo **overjena uskladitev ključev** in zanj potrebujemo praštevilo p in generator α v $\mathbb{Z}_p^*$, ki sta javno znana. Anita in Bojan potrebujeta vsak svoj par ključev (a_A, b_A) in (a_B, b_B), kjer sta a_A in a_B zasebni vrednosti, ki ju skrbno varujeta, b_A in b_B pa javni vrednosti, ki sta overjeni s certifikatoma $C(A)$ in $C(B)$, ter omogočata njuno

(pravilno) identifikacijo. Tudi CA mora imeti zasebni in javni kljuc. Varnost prvega je osrednjega pomena, sicer ne bi mogli ločiti prave in ponarejene certifikate, tako da bi se slej ko prej zamajalo zaupanje v celoten sistem. Drugi ključ pa nam (kot uporabniku) sme biti podtaknjen, sicer bi bili pripravljeni sprejeti certifikat lažne CA (in bi nas lahko na ta način vsaka stara ženica zlahka prepričala, da je naša babica).

Anita in Bojan izpeljeta protokol na naslednji način:

1. *Anita izbere* $a_A \in_R \{0, \ldots, p-2\}$, *izračuna* $\beta_A = \alpha^{a_A} \bmod p$ *in pošlje Bojanu.*
2. *Bojan izbere* $a_B \in_R \{0, \ldots, p-2\}$, *izračuna* $\beta_B = \alpha^{a_B} \bmod p$, $K = (\beta_A)^{a_B} \bmod p$ in podpis $y_B = \mathrm{sig}_B(\beta_B, \beta_A)$ *ter pošlje potrdilo* $(C(B), \beta_B, y_B)$ *Aniti.*
3. *Tudi Anita izračuna* $K = (\beta_B)^{a_A} \bmod p$ *ter preveri podpis* y_B *z uporabo* ver_B *in potrdilo* $C(B)$ *z uporabo* $\mathrm{ver}_{\mathrm{CA}}$. *Nato izračuna podpis* $y_A = \mathrm{sig}_A(\beta_A, \beta_B)$ *in pošlje potrdilo* $(C(A), y_A)$ *Bojanu.*
4. *Bojan preveri podpis* y_A *z uporabo* ver_A *in potrdilo* $C(A)$ *z uporabo* $\mathrm{ver}_{\mathrm{CA}}$.

V prikazu vidimo, da si uporabnika Anita in Bojan izmenjata naslednje informacije:

$$\text{Anita} \quad \xrightarrow{\qquad\qquad \beta_A \qquad\qquad} \quad \xleftarrow{C(B),\ \beta_B,\ \mathrm{sig}_B(\beta_B, \beta_A)} \quad \xrightarrow{\qquad C(A),\ \mathrm{sig}_A(\beta_A, \beta_B) \qquad} \quad \text{Bojan}$$

Kaj lahko tukaj naredi napadalec Oskar? Najprej prestreže javno vrednost β_A in jo zamenja s svojo $\beta_{A'} = \alpha^{a'_A}$. Potem prejme β_B, $\mathrm{sig}_B(\beta_B, \beta_{A'})$ od Bojana. Zdaj bi rad zamenjal β_B z $\beta_{B'} = \alpha^{a'_B}$, vendar to pomeni, da mora zamenjati tudi $\mathrm{sig}_B(\beta_B, \beta_{A'})$ s $\mathrm{sig}_B(\beta_{B'}, \beta_A)$.

Ker pa ne pozna Bojanovega zasebnega ključa, ki ga potrebuje za algoritem sig_B, tega ne more izračunati. Prav tako ni sposoben zamenjati $\mathrm{sig}_A(\beta_A, \beta_{B'})$ s podpisom $\mathrm{sig}_A(\beta_{A'}, \beta_B)$, ker ne pozna Anitinega zasebnega ključa, ki ga potrebuje za algoritem sig_A. Očitno je protokol varen pred napadom srednjega moža.

6. ZAKLJUČEK

Kakšne so torej prednosti asimetričnih sistemov pred simetričnimi? Videli smo, da moramo pri simetričnem sistemu za vsako komunikacijo najprej vzpostaviti povezavo s centrom, ki nam izda sejni ključ. Na ta način so torej onemogočene neprikljućene (off-line) transakcije. Zaradi visokih cen telekomunikacij, predvsem v Evropi, obstaja močna želja ravno po takšnih transakcijah. Pogosta so tudi plačila manjših zneskov, kjer se strošek transakcije ne razlikuje bistveno od vrednosti transakcije.

... nadaljevanje zgodbe o volku in sedmih kozličkih:

... toda najmlajši kozliček je zopet nezaupljivo zajokal: "*Najprej nam pokaži svoj certifikat, da bomo zares prepričani, da si naša draga mama!*" A volk ni imel certifikata in je moral zopet oditi.

Ker ni vedel, kaj je to certifikat, se je zbegan pritožil prijatelju volku, ki je vedel nekaj malega o internetu in ta mu je svetoval certifikatno agencijo. Vendar mu tam zvijače in grožnje niso pomagale: nihče mu ni verjel, da je koza s sedmimi kozlički in tako ni bil nihče pripravljen izdati certifikata. Ko se je že mislil vdati, mu je spet neki drugi volk povedal za certifikatno agencijo, kjer ne sprašujejo ali si volk ali koza. Volk je poskočil od sreče in je še pred koncem dneva s pravim certifikatom zopet odšel k sedmim kozličkom ter rekel: "*Odprite otročički, vaša ljuba mamica je prišla domov in vsakemu nekaj prinesla iz gozda.*" Nato je položil s svojimi belimi tacami na okno še certifikat.

Kozlički so se posvetovali, nato pa je najmlajši odgovoril: "*Žal ne smemo odpreti vrat. Ti nisi naša mama! Ne zaupamo certifikatni agenciji, ki je izdala Tvoj certifikat!*"

Ko je volk zaslišal te besede, se je razbesnel in pojedel svoj nevreden certifikat. Toda le-ta se mu je zataknil v grlu, da ni mogel dihati in se je mrtev zgrudil. Tedaj so kozlički odprli vrata, zaplesali okoli mrtvega volka ter zapeli "*Volk je mrtev! Volk je mrtev!*"