HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

Digitalni potpis

CCERT-PUBDOC-2007-02-182

CARNet CERT u suradnji s LS&S

Sigurnosni problemi u računalnim programima i operativnim sustavima područje je na kojem CARNet CERT kontinuirano radi.

Rezultat toga rada ovaj je dokument, koji je nastao suradnjom CARNet CERT-a i LS&S-a, a za koji se nadamo se da će Vam koristiti u poboljšanju sigurnosti Vašeg sustava.

CARNet CERT, www.cert.hr - nacionalno središte za **sigurnost računalnih mreža** i sustava.

***LS&S*, www.lss.hr -** laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Sadržaj

1. Uvod

Digitalni potpisi (eng. *digital signature - DS*) omogućuju utvrđivanje autentičnosti elektroničkog dokumenta, npr. elektroničkog pisma, web stranice ili slikovne datoteke. Dokument je autentičan ako je poznat njegov autor i ako je moguće dokazati da nije neovlašteno izmijenjen. Provjera vjerodostojnosti (eng. *authentication*) potpisanih dokumenata omogućena je korištenjem enkripcije, pri čemu enkripcija predstavlja postupak kodiranja podataka prije slanja kako bi ih samo ovlašteni primatelj mogao dekodirati i razumjeti. Uz to što osigurava autentičnost (identitet pošiljatelja utvrđuje se dešifriranjem sažetka poruke), digitalni potpis osigurava i integritet (provjerom sažetka poruke utvrđuje se je li se poruka mijenjala na putu do primatelja) te neporecivost (pošiljatelj ne može poreći sudjelovanje u transakciji jer jedino on ima pristup do svog privatnog ključa kojim je potpisao poruku).

Korištenjem algoritma digitalnog potpisa potpisnik stvara par ključeva, privatni i javni, ali moguće je i da se za potpisivanje svih poruka koristi isti par. Poruka koju se potpisuje sažima se nekim *hash* algoritmom – stvara se njezin otisak. DS algoritam iz tako dobivenog sažetka poruke i korisnikova privatnog ključa stvara digitalni potpis koji se šalje ili objavljuje zajedno s potpisanom porukom. Osnova sigurnosti digitalnog potpisa je u tajnosti privatnog ključa dok je javni ključ svima dostupan, a omogućuje provjeru autentičnosti poruke.

U nastavku dokumenta opisan je povijesni razvoj digitalnog potpisa, dan je općeniti prikaz načina rada te detaljniji opis triju popularnih DS algoritama. Osim toga, navedene su primjene digitalnih potpisa te je dan pregled mogućih napada s dva primjera. Dokument je zaključen objašnjenjem politike upravljanja digitalnim potpisima i prognozom njihova budućeg razvoja.

2. Povijest digitalnog potpisa

Korištenje Morseove abecede za prijenos poruka telegrafom započinje oko 1860. godine, a već 1869. godine presudom *New Hampshire Supreme Court* suda potpisi preneseni na ovaj način proglašuju se pravomoćnim. Tijekom 1980-ih godina mnoge tvrtke i pojedinci koriste faks uređaje za hitan prijenos papirnatih dokumenata. Iako se kod takvog prijenosa podataka potpis fizički nalazi na papiru, dohvaćanje podataka i njihov prijenos vrši se elektronički. Ovakvi potpisi nazivaju se elektroničkim potpisima. Digitalni potpisi predstavljaju podskupinu elektroničkih potpisa koji koriste različite kriptografske metode. Zbog toga je razvoj digitalnih potpisa usko vezan uz povijest kriptografije.

Razvoj kriptografije s javnim ključem (eng. *public key cryptogtaphy*) započinje 1874. godine opisom jednosmjernih enkripcijskih funkcija u knjizi „*The Principles of Science: A Treatise on Logic and Scientific Method*" autora William Stanley Jevonsa. Ranih 1970-ih godina James H. Ellis, Clifford Cocks i Malcolm Williamson osmišljaju prve algoritme temeljene na asimetričnom ključu, ali ne objavljuju svoja otkrića. Whitfield Diffie i Martin Hellman 1976. godine, pod utjecajem radova Ralpha Merkela na temu distribucije javnih ključeva, objavljuju prvu praktično upotrebljivu metodu razmjene ključeva, koja kasnije postaje poznata pod nazivom *Diffie-Hellman razmjena ključeva* i predstavlja poseban slučaj RSA algoritma.

Spomenuti RSA algoritam prvi puta javno su opisali Ron Rivest, Adi Shamir i Leonard Adleman 1977. godine. Naziv algoritma stvoren je od početnih slova prezimena autora. To je prvi algoritam prikladan za potpisivanje i enkripciju podataka te se smatra sigurnim, pod pretpostavkom korištenja dovoljno dugih ključeva i ažurnih implementacija.

Neal Koblitz i Victor S. Miller 1985. godine predlažu korištenje eliptičkih krivulja nad konačnim poljima u kriptografskim algoritmima s javnim ključam. Na temelju ovakve enkripcije razvijen je ECDSA (eng. *Elliptic Curve DSA*) algoritam, varijanta DSA (eng. *Digital Signature Algorithm*) algoritma, koji pomoću manjeg ključa i s približno jednakim vremenom izvođenja daje sigurniji digitalni potpis jednake veličine.

Sredinom 1990-ih godina započinje standardizacija DS algoritama u Sjedinjenim Američkim Državama: 1994. godine *National Institute of Standards and Technology* institut izdaje standard s oznakom FIBS PUBS 186 (eng. Federal Information Processing Standards Publications), a godinu dana kasnije *American National Standards Institute* institut izdaje ANSI X9.30 standard. Standardizacija na području Europe započinje krajem 1990-ih i početkom 2000-ih godina, na razini Europske Unije i pojedinih zemalja.

3. Principi digitalnog potpisa

Provjeru vjerodostojnosti autora ili podataka moguće je provesti korištenjem:

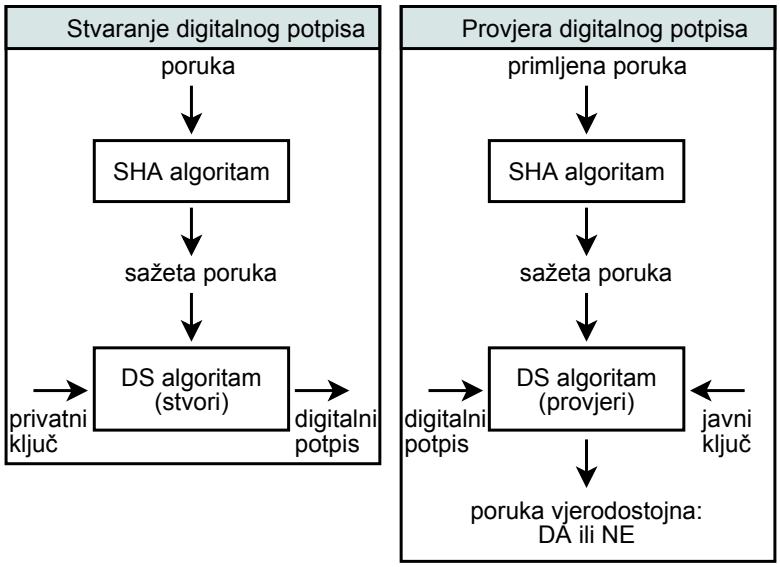
- zaporki – najčešća metoda dokazivanja vjerodostojnosti je pomoću korisničkog imena i uz njega vezane zaporke,
- ispitnog zbroja (eng. *checksum*) - koristi se prvenstveno za provjeru ispravnosti primljenih podataka, ali može poslužiti i za provjeru autentičnosti istih jer neispravan ispitni zbroj ukazuje na neovlaštenu izmjenu podataka,
- CRC provjere (eng. *Cyclic Rendundancy Check*) – konceptualno slično ispitnom zbroju, ali koristi dijeljenje polinoma kako bi se utvrdila ispravnost podataka,
- enkripcije s privatnim ključem,
- enkripcije s javnim ključem i
- digitalnim certifikatima.

3.1. Enkripcija s privatnim ključem

Kod enkripcije s privatnim ključem svako računalo ili korisnik posjeduje tajni ključ pomoću kojega se podaci, prije slanja računalnom mrežom, kriptiraju. Primatelj treba znati pošiljateljev tajni ključ kako bi mogao dekriptirati tako primljene podatke. Zbog toga je prije uspostavljanja komunikacije potrebno znati koja računala (tj. korisnici) će razmjenjivati poruke te na svako računalo instalirati privatne ključeve računala s kojih se očekuju poruke.

3.2. Enkripcija s javnim ključem

Prilikom stvaranja digitalnog potpisa koristi se privatni ključ dok se za njegovu provjeru koristi javni ključ koji odgovara, ali nije jednak, privatnom ključu. Svaki korisnik posjeduje vlastiti privatni i javni ključ. Javni ključevi su javno dostupni i svakom korisniku omogućuju provjeru potpisa. Privatni ključevi dostupni su samo svojim vlasnicima čime je onemogućeno krivotvorenje potpisa.
Podaci koji se obilježavaju digitalnim potpisom skraćeno se nazivaju porukom. U postupku stvaranja digitalnog potpisa za dobivanje sažete inačice poruke (eng. *message digest*) koristi se sigurna jednosmjerna funkcija, tzv. SHA (eng. *Secure Hash Algorithm*) algoritam. To su funkcije koje se matematički vrlo jednostavno izračunavaju, ali im je vrlo teško pronaći inverznu funkciju. Iz tako dobivene sažete inačice poruke DS algoritmom stvara se digitalni potpis. Poruka se, zajedno s pripadnim potpisom, šalje primaocu koji pomoću pošiljateljeva javnog ključa utvrđuje vjerodostojnost poruke i samog digitalnog postupka. U postupku provjere potrebno je koristi SHA algoritam jednak onom korištenom prilikom stvaranja potpisa. Na slici Slika 1 shematski su prikazani opisani postupci stvaranja i provjere digitalnih potpisa.



Slika 1: Shematski prikaz postupka stvaranja i provjere digitalnog potpisa

U opisanom postupku potpisuje se sažeta inačica poruke, a ne cijela poruka, iz sljedećih razloga:

- **Efikasnost**: potpis će biti puno kraći pa će i cjelokupni postupak biti brži jer je u praksi stvaranje sažetka poruke puno brže od stvaranja potpisa.
- **Javna dostupnost dokumenta**: npr. razne diplome, potvrde, dozvole, ugovori i sl., trebaju biti javno dostupni pa se spremaju i prenose bez enkripcije, a priloženi potpis garantira vjerodostojnost pojedinog dokumenta.
- **Integritet**: tekst koji se potpisuje treba biti kraći od duljine privatnog ključa. Kako to poruka koju se potpisuje najčešće nije, potrebno ju je, u slučaju potpisivanja bez sažimanja, razlomiti na dijelove, pojedinačno potpisati svaki dio i poslati. Primatelj tako razlomljene poruke ne bi mogao znati je li koji njezin dio izgubljen ili izbrisan tijekom prijenosa.

3.3. Digitalni certifikati

Digitalni certifikati koriste se kod zahtjevnijih implementacija enkripcije s javnim ključem, npr. kod web poslužitelja. Radi se o certifikatu kojega izdaje jedno ili više ovlaštenih tijela (eng. *Certificate Authority*), a koja predstavljaju dio PKI (eng. *public key infrastructure*) sustava. Spomenuto tijelo djeluje kao posrednik između dva računala ili korisnika, ono potvrđuje njihove identitete i razmjenjuje njihove javne ključeve. Certifikati koriste digitalne potpise za povezivanje javnih ključeva s podacima o identitetu vlasnika, kao što su ime osobe ili organizacije, adresa i sl., i time sprječavaju neovlašten pristup podacima objavljivanjem lažnog javnog ključa.

Mreža povjerenja (eng. *web of trust*) predstavlja alternativu centraliziranim PKI sustavima, a koristi se kod PGP (eng. *Pretty Good Privacy*), GnuPGP i drugih sustava kompatibilnih s OpenPGP standardom. Djeluje tako da korisnici, koristeći vlastite privatne ključeve, potpisuju identifikacijske certifikate drugih korisnika. Spomenuti identifikacijski certifikati mogu sadržavati informacije kao što su javni ključevi i podaci o njihovim vlasnicima. Na primjer, korisnik može prihvatiti vjerodostojnost certifikata ako ga je potpisalo tri ili više korisnika u koje spomenuti korisnik ima djelomično povjerenja, ili jedan potvrđeno vjerodostojan korisnik.

4. DS algoritmi

DS algoritmi općenito se sastoje od tri osnovna koraka: stvaranja javnog i privatnog ključa, stvaranja digitalnog potpisa na temelju sažetka poruke i privatnog ključa te utvrđivanja vjerodostojnosti potpisane poruke korištenjem javnog ključa pošiljatelja. U nastavku teksta opisani su DSA, ECDSA i RSA algoritmi za stvaranje i provjeru digitalnih potpisa.

4.1. DSA

DSA (eng. *Digital Signature Algorithm*) algoritam propisan je DSS (eng. *Digital Signature Standard*) standardom savezne vlade Sjedinjenih Američkih Država i koriste ga sve civilne vladine organizacije te sve ne-vladine tvrtke i organizacije koje surađuju s vladom. Algoritam je 1991. godine patentirao David W. Kravitz, bivši zaposlenik NSA (eng. *National Security Agency*) agencije, a moguće ga je koristiti bez plaćanja naknade.

Algoritam se sastoji od tri koraka:

a) stvaranje ključeva:
   1. odabir 160-bitnog prostog broja q
   2. odabir L-bitnog prostog broja p tako da vrijedi: $p = qz + 1$ za neki cijeli broj z, $512 \leq L \leq 1024$, L djeljivo s 64
   3. odabir h takvog da vrijedi: $1 < h < p - 1$, $g = h^z \bmod p > 1$
   4. stvaranje nasumičnog broja x, tako da vrijedi: $0 < x < q$
   5. izračun $y = g^x \bmod p$
   6. (p, q, g, y) je javni ključ, a x je privatni ključ

b) potpisivanja poruke:
   1. stvaranje nasumičnog broja k, tako da vrijedi: $0 < k < q$
   2. izračun $r = (g^k \bmod p) \bmod q$
   3. izračun $s = (k^{-1}(\text{SHA-1}(m) + xr)) \bmod q$, gdje je SHA-1($m$) tzv. *hash* funkcija primijenjena na poruci m
   4. ponoviti postupak ako je $r = 0$ ili $s = 0$
   5. potpis je (r, s)

c) provjera vjerodostojnosti potpisa:
   1. ako $0 < r < q$ ili $0 < s < q$ nije zadovoljeno potpis se smatra neispravnim
   2. izračun $w = s^{-1} \bmod q$
   3. izračun $u1 = (\text{SHA-1}(m)w) \bmod q$
   4. izračun $u2 = (rw) \bmod q$
   5. izračun $v = ((g^{u1}y^{u2}) \bmod p) \bmod q$
   6. potpis je valjan ako vrijedi $v = r$

Opisani algoritam je varijanta *ElGamal* algoritma kojega je prvi opisao Taher ElGamal 1984. godine.

4.2. ECDSA

ECDSA (eng. *Elliptic Curve DSA*) algoritam je varijanta DSA algoritma koja u radu koristi skupove eliptičnih krivulja. Uz istu razinu zaštite ovaj algoritam rezultira manjim ključevima od DSA algoritma, uz približno jednako vrijeme izvođenja. Eliptička krivulja određena je skupom od šest parametara $T = (p, a, b, G, n, h)$, gdje je p cijeli broj koji određuje konačno polje Φ_p, a dva elementa a,b iz Φ_p određuju eliptičku krivulju $E(\Phi_p)$:

$$E: y^2 \equiv x^3 + ax + b \pmod{p},$$

$G = (x_G, y_G)$ je bazna točka na $E(\Phi_p)$, prosti broj n je red točke G, a cijeli broj h je kofaktor $h = \#E(\Phi_p)/n$.

Prije potpisivanja dokumenta pošiljatelj i primatelj poruke moraju se složiti oko parametara krivulja, a potrebno je i da pošiljatelj posjeduje par ključeva prikladan za kriptiranje korištenjem eliptičnih krivulja: privatni ključ d_A (nasumično odabran cijeli broj iz intervala [1, n - 1]) i javni ključ $Q_A = d_A G$.

Postupak potpisivanja poruke sastoji se od sljedećih koraka:

1. izračun $e = \text{HASH}(m)$, gdje je HASH kriptografska hash funkcija, kao npr. SHA-1
2. nasumični odabir cijelog broja k iz intervala [1, n - 1]
3. izračun $r = x_1 \bmod n$, gdje je $(x_1, y_1) = kG$, ako je $r = 0$ vratiti se na korak 2.
4. izračun $s = k^{-1}(e + d_A r) \bmod n$, ako je $s = 0$ vratiti se na korak 2.
5. potpis je (r, s)

Za provjeru vjerodostojnosti potpisane poruke primatelju je potreban javni ključ potpisnika. Provjera se provodi u sljedećim koracima:

1. ako r i s nisu cijeli brojevi iz skupa [1, n-1] potpis se smatra neispravnim
2. izračun $e = \text{HASH}(m)$, gdje je HASH funkcija jednaka onoj korištenoj prilikom stvaranja potpisa
3. izračun $w = s^{-1} \bmod n$
4. izračun $u_1 = ew \bmod n$ i $u_2 = rw \bmod n$
5. izračun $(x_1, y_1) = u_1 G + u_2 Q_A$
6. potpis je valjan ako vrijedi $x_1 = r \bmod n$

Zbroj skalarnih umnožaka iz 5. koraka provjere vjerodostojnosti potpisa može se brže izračunati pomoću *Strausovog* algoritma (poznatog i kao *Shamirov* trik) nego izravno množenjem i zbrajanjem.

4.3. RSA

RSA (Rivest, Shamir, Adleman) algoritam moguće je koristiti za kriptiranje poruke i za njezino potpisivanje. Pošiljatelj poruku potpisuje pomoću vlastitog privatnog ključa, a kriptira korištenjem javnog ključa primaoca. Nakon primitka poruke primatelj dekriptiranje vrši pomoću vlastitog privatnog ključa, a provjera vjerodostojnosti potpisa provodi se uz korištenje potpisnikovog javnog ključa.

Algoritam se sastoji od sljedećih koraka:

a) stvaranje ključeva:
   1. nasumični odabir dva velika prosta broja p i q
   2. izračun modula $n_1 = pq$
   3. izračun količnika $\Phi(n_1) = (p - 1)(q - 1)$
   4. odabir cijelog broja e_1 takvog da vrijedi $1 < e_1 < \Phi(n_1)$ te da e_1 i $\Phi(n_1)$ nemaju zajedničkih djelitelja osim broja 1
   5. izračun d_1 takvog da zadovoljava relaciju kongruencije $d_1 e_1 \equiv 1 \pmod{\Phi(n_1)}$, odnosno da vrijedi $d_1 e_1 = 1 + k\Phi(n_1)$, za neki cijeli broj k
   6. javni ključ se sastoji od modula n_1 i javnog eksponenta e_1, a privatni ključ se sastoji od modula n_1 i privatnog eksponenta d_1

b) potpisivanje poruke:
   1. izračun *hash* vrijednosti poruke M: $h = \text{HASH}(M)$, gdje je HASH kriptografska *hash* funkcija, kao npr. SHA-1
   2. izračun potpisa $s = h^{(d1 \bmod n1)}$
   3. dodavanje potpisa poruci

c) kriptiranje poruke:
   1. dohvaćanje primateljevog javnog ključa (n_2 i e_2)
   2. iz poruke se M primjenom dogovorenog povratnog protokola (eng. *padding scheme*) dobiva broj m, takav da vrijedi $m < n$
   3. izračun kriptirane poruke $c = m^{e2} \bmod n_2$

d) dekriptiranje poruke:
   1. iz primljene poruke c dobiva se m prema: $m = c^{d2} \bmod n_2$
   2. primjenom inverznog protokola iz koraka 2. postupka kriptiranja iz m se izračunava izvorna poruka M

e) provjera vjerodostojnosti potpisa:
   1. izračun *hash* vrijednosti poruke *M*: h = HASH(M) , gdje je HASH funkcija jednaka onoj korištenoj prilikom stvaranja potpisa
   2. izračun $h_1 = s^{(e1 \bmod n1)}$
   3. potpis je valjan ako vrijedi $h_1 = h$

FDH (eng. *Full Domain Hash*) i RSA-PSS (eng. *Probabilistic Signature Scheme*) algoritmi su varijante RSA algoritma namijenjene isključivo digitalnom potpisivanju poruka.

5. Primjene digitalnih potpisa

5.1. Potpisivanje dokumenata

U nekoliko zemalja digitalni potpis ima status sličan onomu tradicionalnog pisanog potpisa. To znači da digitalno potpisani dokument potpisnika pravno obvezuje, u skladu s uvjetima navedenim u spomenutom dokumentu. Zbog toga se preporuča korištenje različitih parova ključeva za enkripciju i za potpisivanje. Korištenjem para ključeva namijenjenih enkripciji, korisnik može sudjelovati u kriptiranoj komunikaciji (npr. pregovorima o kupnji nekretnine), ali ne potpisuje svaku poruku pravno važećim potpisom. Jednom kada zainteresirane strane postignu dogovor, ugovor se digitalno potpisuje i tek tada su potpisnici pravno vezani potpisanim dokumentom. Tako potpisani ugovor moguće je zatim, zbog dodatne zaštite, slati kriptiranog.

DS algoritmi i protokoli ne pružaju, sami po sebi, informaciju o tome kada je dokument potpisan. Potpisnik može, ali i ne mora, uključiti vremensku oznaku (eng. *time stamp*) unutar digitalnog potpisa ili se u samom dokumentu može spominjati datum i vrijeme potpisivanja. Ovakvo označavanje vremena omogućuje navođenje netočnog, npr. ranijeg datuma ili vremena potpisivanja. Korištenje sigurnih vremenskih oznaka (eng. *trusted time stamp*) sprečava se ovakva zloupotreba digitalnog potpisa. Sigurne vremenske oznake osigurava pouzdana treća strana, tzv. TSA (eng. *TimeStamping Authority*), koja time potvrđuje postojanje određenih podataka prije nekog vremena. Ranjivosti takvog korištenja vremenskih oznaka moguće je umanjiti umetanjem više oznaka različitih TSA organizacija u potpis.

Jedna od osnovnih prednosti korištenja digitalnih potpisa, uz osiguravanje autentičnosti i integriteta dokumenta, je onemogućavanje nepriznavanja dokumenta od strane potpisnika (eng. *non-repudiation*). Ako sporna poruka nije potpisana, njezin ju navodni pošiljatelj uvijek može zanijekati tvrdnjama kako ju je netko drugi napisao i poslao. Takvo što se ne može dogoditi s potpisanim porukama osim u slučaju otkrivanja korisnikovog privatnog ključa, kojeg zbog toga treba čuvati u strogoj tajnosti.

Spremanje privatnog ključa na tzv. pametnoj kartici (eng. *smart card*) jedan je od načina osiguravanja njegove tajnosti. Alternativa je čuvanje privatnog ključa na osobnom računalu korisnika, ali takav pristup ima dva ozbiljna nedostatka:

- korisnik može potpisivati dokumente samo na spomenutom računalu i
- tajnost privatnog ključa ovisi o sigurnosti računala na kojemu je pohranjen.

Karticu je potrebno povezati na računalo i proslijediti joj *hash* vrijednost poruke, ugrađeni procesor zatim iz pohranjenog privatnog ključa i primljenog otiska poruke proračunava potpis koji se potom šalje računalu. Na taj način privatni ključ nikada ne napušta karticu. Većinu kartica potrebno je prije upotrebe aktivirati osobnim identifikacijskim brojem (eng. *Personal Identification Number – PIN*), a građene su tako da onemogućavaju, ili barem otežavaju, neovlašten pristup pohranjenim podacima.

5.2. Slijepi potpisi

Slijepi potpis (eng. *blind signature*) oblik je digitalnog potpisa kod kojega je sadržaj poruke skriven (eng. *blinded*) od potpisnika. Takvim potpisom moguće je provjeriti vjerodostojnost originalne otkrivene (eng. *unblinded*) poruke jednako kao što se to čini običnim digitalnim potpisom. Ovakvi potpisi najčešće se koriste u primjenama gdje je jedna strana autor poruke, a neka druga njezin potpisnik, npr. u kriptografskim sustavima za glasovanje ili kod sigurnih elektroničkih platežnih sustava.

Druga moguća primjena slijepih potpisa je sprječavanje potpisnika da poveže potpisanu skrivenu poruku s kasnije otkrivenom porukom tijekom njezinog eventualnog ocjenjivanja (eng. *unlinkability*). Slijepi potpisi ovako se koriste u primjenama kod kojih je nužna anonimnost pojedinih sudionika.
Slijepo potpisivanje moguće je implementirati pomoću raznih DS algoritama s javnim ključem, npr. RSA ili DSA algoritmom. Poruka se prije potpisivanja skriva, najčešće kombiniranjem s nasumično odabranim ključem (eng. *blinding factor*) i zatim ju se potpisuje nekim od uobičajenih DS algoritama. Vjerodostojnost potpisane skrivene poruke, zajedno s ključem korištenim za njeno skrivanje, moguće je utvrditi pomoću potpisnikovog javnog ključa.

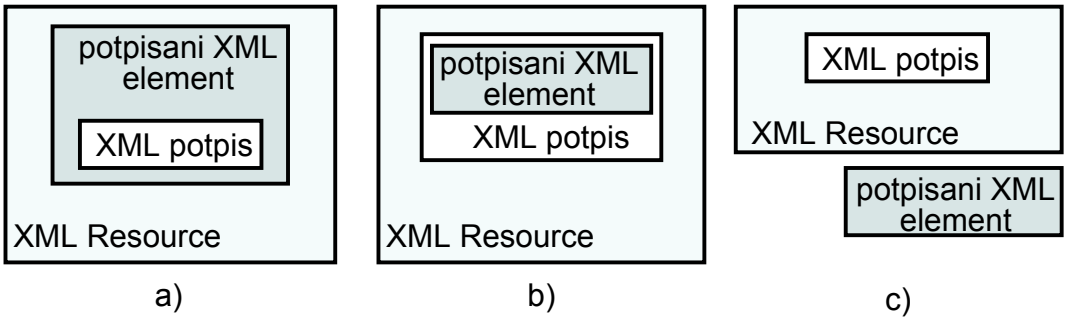
5.3. Digitalni potpisi u web aplikacijama

Različite sadržaje web stranica moguće je potpisati XML digitalnim potpisom koji je reguliran *W3C XML Signature* standardom krovne međunarodne standardizacijske organizacije na području web tehnologija *World Wide Web Consortium*. XML potpisom moguće je potpisati sljedeće tipove podataka:

- XML elemente, skupove XML čvorova (eng. *nodes*) i njihov sadržaj,
- vanjske URI oznake,
- vanjske binarne datoteke i
- binarne podatke ugrađene u XML dokument u obliku znakovnih nizova kodiranih na bazi 64.

Na pojedinoj web stranici moguće je potpisati bilo koji njezin programski dostupan element (dijelove HTML i XML programskog koda te skrivena i vidljiva polja formulara, kao i njihove sadržaje), datoteke prisutne na klijentskom računalu te mrežne resurse koji su dostupni izravno s klijentskog računala ili posredno preko poslužitelja.
Postoje tri tipa XML potpisa:

- omotani (eng. *enveloped*) – potpis je ugrađen u podatke koje potpisuje,
- omotavajući (eng. *enveloping*) – potpisani podaci su ugrađeni u XML potpis i
- odvojeni (eng. *detached*) – XML potpis i potpisani podaci su razdvojeni.



Slika 2: Različiti XML potpisi: a) omotani, b) omotavajući, c) odvojeni

Općenita struktura XML digitalnog potpisa je:

```
<Signature ID?>
   <SignedInfo>
     <CanonicalizationMethod/>
     <SignatureMethod/>
     (<Reference URI? >
       (<Transforms>)?
       <DigestMethod>
       <DigestValue>
     </Reference>)+
   </SignedInfo>
   <SignatureValue>
  (<KeyInfo>)?
  (<Object ID?>)*
 </Signature>
```

gdje „?“ označava jedno ili nijedno pojavljivanje, „+“ označava jedno ili više pojavljivanja, a „*“ označava nula ili više pojavljivanja odgovarajuće oznake. Potpisi su s potpisanim podacima povezani URI oznakama, ako se ti podaci nalaze izvan XML datoteke, ili fragmentnim oznakama (eng. *fragment identifier*), ako su podaci i potpis unutar iste datoteke.

5.4. Zaštita multimedijalnih sadržaja digitalnim potpisom

Zaštita multimedijalnih sadržaja razlikuje se od zaštite ostalih vrsta digitalnih podataka po tome što ih je moguće mijenjati bez narušavanja sadržaja koji se prenosi pa je prilikom provjere autentičnosti potrebno razlikovati obradu od zlonamjernih promjena sadržaja – napada. S druge strane, bilo kakva izmjena nekog elektroničkog pisma ili pravnog dokumenta predstavlja napad i treba biti uočena u postupku provjere vjerodostojnosti.

Multimedijalne sadržaje moguće je štititi ugrađivanjem tzv. vodenih žigova (eng. *watermark*) ili njihovim digitalnim potpisivanjem. Vodeni žig je skup informacija, namijenjenih npr. zaštiti autorskih prava, ugrađen u multimedijalni sadržaj pri čemu sam žig može biti vidljiv ili sakriven od krajnjeg korisnika. Vidljivi žigovi koriste se za ograničavanje uporabe označenog sadržaja dok se skriveni žigovi koriste za utvrđivanje njegova porijekla. Neobrađeni i nesažeti sadržaji naročito su pogodni za zaštitu vodenim žigovima zbog:

- vodeni žig je izravno povezan s podacima na koje se odnosi te ga je moguće jednostavno provjeriti i
- unutar takvih podataka postoji dovoljno prostora za ugradnju žiga bez narušavanja vidljivog sadržaja.

Pojedini vodeni žigovi najčešće su krhki, tako da ne mogu preživjeti obradu sadržaja kojega obilježuju (npr. sažimanje), ali mogu biti i vrlo robusni, što znači da su otporni na obradu sadržaja, pa i na različite zlonamjerne izmjene.

Kako se multimedijalni sadržaji rijetko distribuiraju i koriste bez sažimanja mnogo je prikladnija njihova zaštita digitalnim potpisom. Mnogi standardi za sažimanje podataka, npr. JPEG i MPEG, omogućuju unos korisničkih podataka u poseban odjeljak unutar sažete datoteke, gdje je moguće ugraditi digitalni potpis. Tijekom izmjene sadržaja izvorni korisnički podaci, u ovom slučaju digitalni potpis, najčešće se odbacuju. Čak i ako napadač uspije zadržati potpis unutar izmijenjene datoteke, napad je lako moguće otkriti zbog nepodudaranja *hash* vrijednosti napadnute datoteke i vrijednosti iz potpisa. Pored umetanja u zaglavlje komprimirane datoteke, digitalni potpis moguće je pohraniti u posebnu datoteku koju je onda potrebno distribuirati sa sadržajem na kojega se odnosi.

Postupak digitalnog potpisivanja multimedijalnih sadržaja u načelu se ne razlikuje od potpisivanja ostalih vrsta podataka, a glavna razlika je u informacijama koje se koriste za stvaranje potpisa. Kod tekstualnih dokumenata ili dijelova programskog koda potpisuje se niz bitova tako da promjena barem jednog znaka biva uočena tijekom provjere vjerodostojnosti. Multimedijalni dokumenti se potpisuju tako da se zaštiti njihov sadržaj, vizualne i zvučne informacije koje krajnji korisnik percipira tijekom pregledavanja. Te informacije se ne gube legalnom obradom dokumenta, npr. sažimanjem ili promjenom veličine slike, pa se ne gubi niti vjerodostojnost potpisane datoteke. U slučaju napada na potpisani multimedijalni dokument, najčešće u obliku zamjene pojedinih elemenata zlonamjerno oblikovanim sadržajima, dolazi do izmjene potpisanih informacija koje je moguće otkriti provjerom vjerodostojnosti.

6. Napadi na digitalni potpis

Napadi na digitalni potpis mogu se podijeliti u dvije osnovne skupine:

- napadi uz poznavanje ključa – napadaču je dostupan samo potpisnikov javni ključ,
- napadi uz pristup porukama – napadač ima pristup potpisanim porukama.

Napadi uz pristup porukama mogu se podijeliti prema načinu na koji su poruke dostupne napadaču odabrane:

1. Napad na poznate poruke – napadač ima pristup skupu $m_1, \ldots, m_t$ potpisanih poruka koje nije on odabrao.
2. Napad na generički odabrane poruke – napadač prije pokušaja falsificiranja potpisa odabire skup poruka i daje ih korisniku na potpis. Prilikom odabira poruka napadač nema uvid niti u

jedan vjerodostojan potpis pa je ovo neadaptivan napad, a kako izbor poruka ne ovisi o korisnikovom javnom ključu napad se naziva generičkim – jednak skup poruka koristi se za napade na potpise svih korisnika.

3. Usmjereni napad na odabrane poruke – napadač odabire poruke na temelju korisnikovog javnog ključa, ali bez uvida u vjerodostojan potpis. Ovo je također neadaptivan napad, ali nije generički jer je usmjeren na pojedinog korisnika.
4. Adaptivni napad na odabrane poruke – napadač korisniku na potpis daje poruke odabrane na temelju korisnikova javnog ključa i prethodno pribavljenih potpisa.

Nabrojani tipovi napada poredani su prema rastućoj težini. Najopasniji su adaptivni napadi na odabrane poruke, a to je ujedno i najčešća vrsta napada zbog toga što se korisnici žele pouzdati u korišteni DS sustav toliko da mogu bez straha potpisivati proizvoljne dokumente. Na primjer, javni bilježnici po dužnosti potpisuju gotovo sve dokumente koji im se predoče na ovjeru.

Uspješno provođenje napada na korisnikov DS sustav može rezultirati:

1. potpunim kompromitiranjem DS sustava – napadač je došao u posjed korisnikovog privatnog ključa,
2. univerzalnim krivotvorenjem – napadač je pronašao algoritam funkcionalno ekvivalentan korisnikovom DS algoritmu, koji se npr. temelji na različitom, ali ekvivalentnom privatnom ključu,
3. selektivnim krivotvorenjem – napadač može krivotvoriti potpise samo na prethodno odabranom skupu poruka,
4. egzistencijalnim krivotvorenjem – napadač može krivotvoriti potpis najmanje jedne proizvoljne poruke pa ovakav napad ne predstavlja značajnu prijetnju.

DS sustav je prema tome moguće *potpuno*, *univerzalno*, *selektivno* ili *egzistencijalno* kompromitirati. Sustav za koji je dokazana nemogućnost egzistencijalnog krivotvorenja je sigurniji od sustava kojemu je dokazana samo nemogućnost potpunog kompromitiranja.

6.1. Napad mjerenjem vremena

Stvaranje digitalnog potpisa općenito ima različita trajanja za različite poruke. Razlog tomu su različite optimizacije performansi kako bi se uklonile nepotrebne operacije, grananja i ispitivanja uvjeta, različita stanja RAM priručne memorije, procesorske instrukcije s različitim vremenom izvođenja (npr. množenje i zbrajanje) i drugo. Mjerenjem vremena potrebnog za stvaranje digitalnog potpisa napadač može doći u posjed korisnikovog tajnog ključa. Ovaj napad moguće je primijeniti na Diffie-Hellman, RSA, DSS i druge DS algoritme.

Korištenjem javno dostupnih informacija i informacija do kojih je moguće doći prisluškivanjem napadač mjerenjem vremena potrebnog za potpisivanja većeg broja poruka pomoću jednog tajnog ključa može otkriti njegovu vrijednost, bit po bit. Napad mjerenjem vremena nije primjenjiv ako se za potpisivanje svake poruke koristi drugačiji tajni ključ. Mjerenje vremena potpisivanja može se provesti bilježenjem vremena primitka poruke na napadnutom sustavu i vremena odgovora na spomenutu poruku.

Napad se općenito može opisati kao problem uočavanja signala. „Signal" se sastoji od vremenskih varijacija uzrokovanih bitom privatnog ključa kojega se pokušava otkriti i „šuma" koji se sastoji od netočnosti mjerenja vremena i vremenskih varijacija uzrokovanih ostalim nepoznatim bitovima privatnog ključa. Uz j poruka $y_0, y_1, \ldots, y_{j-1}$ s odgovarajućim vremenskim mjerenjima $T_0, T_1, \ldots, T_{j-1}$, vjerojatnost pogađanja x_b prvih b bitova privatnog ključa proporcionalna je:

$$P(x_b) \propto \prod_{i=0}^{j-1} F\left(T_i - t(y_i, x_b)\right)$$

gdje je $t(y, x_b)$ vrijeme potrebno za prvih b iteracija proračuna digitalnog potpisa pomoću privatnog ključa x_b, a F je pretpostavljena funkcija raspodjele vjerojatnosti $T - t(y, x_b)$ za sve vrijednosti y i za ispravan x_b. Ako je x_{b-1} ispravno određen moguće su dvije vrijednosti bita x_b. Vjerojatnost da je x_b točna vrijednost, a x_b'netočna je:

$$\frac{\prod_{i=0}^{j-1} F\left(T_i - t\left(y_i, x_b\right)\right)}{\prod_{i=0}^{j-1} F\left(T_i - t\left(y_i, x_b\right)\right) + \prod_{i=0}^{j-1} F\left(T_i - t\left(y_i, x_b'\right)\right)}.$$

Najočitiji način sprječavanja napada je maskiranje trajanja postupka potpisivanja tako da ono bude jednako za sve poruke. Izgradnja algoritma koji traje jednako dugo neovisno o ulaznim parametrima i platformi na kojoj se izvodi vrlo je kompliciranо zbog prevoditeljskih optimizacija, stanja RAM priručne memorije, vremena izvođenja instrukcija i drugih čimbenika koji unose nepredvidljive vremenske varijacije. Ako se za produživanje trajanja postupka potpisivanja do postavljene vrijednosti koristi kašnjenje, karakteristike kao što su potrošnja energije ili zauzetost procesorskih resursa od strane pojedinih procesa mogu otkriti stvarno trajanje potpisivanja. Zbog toga je najbolji način sprječavanja napada promjenama tehnika slijepih potpisa maskirati i napadaču nedostupnim učiniti poruku koja ulazi u postupak potpisivanja.

6.2. Napad na algoritam maskiranja poruke RSA digitalnog potpisa

Unutar pojedinih implementacija RSA algoritma, koje koriste algoritam za maskiranje poruke (eng. *padding*) prema RSASSA-PKCS1-v1_5 specifikaciji, otkriven je sigurnosni propust koji omogućuje krivotvorenje. Hash vrijednost poruke *M* se prije potpisivanja maskira kako bi se otežala analiza potpisa, a time i njegovo krivotvorenje:

```
00 01 FF FF … FF 00 || ASN.1  || H(M)
```

gdje je 00 01 FF FF ... FF 00 znakovni niz korišten za maskiranje, ASN.1 je duljina *hash* vrijednosti i druge informacije o korištenoj *hash* funkciji, a H(*M*) je *hash* vrijednost poruke. Nakon primitka potpisane poruke digitalni potpis se dekriptira korištenjem javnog eksponenta (npr. $e = 3$). Time se dobiva maskirana poruka opisane strukture, iz koje se izdvaja H(*M*) i uspoređuje s *hash* vrijednosti poruke na koju se potpis odnosi.
Propust se javlja u postupku izdvajanja H(*M*) iz dekriptiranog potpisa zbog toga što neke implementacije ne provjeravaju jesu li potpisu naknadno dodani podaci. U slučaju korištenja PKCS1-v1_5, kao *hash* vrijednosti se izdvaja sve što se nalazi iza znakovnog niza korištenog za maskiranje i ASN.1 vrijednosti. Za bilo koju poruku *M'* s *hash* vrijednošću H(*M'*) lako je pronaći treći korijen znakovnog niza oblika:

```
00 01 FF FF … FF 00 || ASN.1  || H(M') || smeće
```

gdje je broj ponavljanja niza FF unutar znakovnog niza za maskiranje smanjen, a *smeće* je znakovni niz takav da je cjelokupna izmijenjena maskirana poruka predstavlja treću potenciju nekog broja.
Napad je opisan u slučaju korištenja javnog eksponenta $e = 3$, ali ga je moguće izvesti za bilo koju malenu vrijednost eksponenta *e* za koju je lako pronaći *e*-ti korijen znakovnog niza, kao u primjeru. Napad je moguće spriječiti korištenjem javnog eksponenta većeg iznosa i uvođenjem dodatne provjere u postupku izdvajanja *hash* vrijednosti iz potpisa.

7. Politika upravljanja digitalnim potpisima

Politika upravljanja (eng. *policy*) digitalnim potpisom može se definirati kao imenovani skup pravila koja navode primjenjivost digitalnog potpisa unutar određene zajednice i/ili za određeni skup primjena s uobičajenim sigurnosnim zahtjevima. Politika se kreira za korištenje u točno određenim situacijama i definira prava i obaveze svih zainteresiranih strana.
Politika upravljanja digitalnim potpisima definira se na razini organizacije, pojedinih tvrtki te vladinih i nevladinih agencija i odjela, a sadrži pravila upravljanja i korištenja certifikata s javnim ključevima koji se koriste za ovjeru, autorizaciju, provjeru integriteta i usuglašavanje ključeva. Na primjer, takvi

se certifikati mogu koristiti za utvrđivanje identiteta pošiljaoca elektroničkog pisma, prilikom udaljenog pristupa računalu, za zaštitu integriteta dokumenata ili aplikacija te za utvrđivanje identiteta građana ili drugih pravnih subjekata.
Zbog važnosti politike upravljanja u postupku uspostavljanja povjerenja među korisnicima nužno je tijekom njezinog stvaranja konzultirati sve zainteresirane strane i tijekom implementacije osigurati da korisnici shvaćaju i prihvaćaju prava i obaveze propisane njome.
Politikom upravljanja digitalnim potpisima propisuje se korištenje određene infrastrukture javnih ključeva i normi sigurnosti.

7.1. Infrastruktura javnih ključeva

Infrastruktura javnih ključeva osigurava komponente neophodne za upravljanje (izdavanje, provjeru i opozivanje) javnim ključevima i certifikatima. Digitalni certifikat je elektronički dokument za potvrdu identiteta osobe ili tvrtke, a to se postiže povezivanjem javnog ključa i njegovog vlasnika. FINA je prvi izdavatelj digitalnih certifikata u Hrvatskoj u obliku registra digitalnih certifikata (RDC).
Infrastruktura javnih ključeva osigurava tajnost i integritet potpisanih dokumenata. Nadalje, PKI omogućuje sigurnu autentifikaciju sudionika u komunikaciji, razmjenu dokumenata s mogućnošću kriptiranja, digitalnog potpisivanja i supotpisivanja, te jedinstvenu registraciju javnih ključeva u obliku digitalnih certifikata.
Infrastrukturu javnih ključeva sačinjavaju:

- CA (eng. *Certificate Authority*),
- RA (eng. *Registration Authority*) i
- CR (eng. *Certificate Repository*).

CA izdaje i opoziva certifikate, upravlja njima, čuva ih te jamči njihovu valjanost. CA se nalazi unutar sigurnog okruženja, odvojen od ostalih PKI komponenti. RA obrađuje zahtjeve korisnika za izdavanje certifikata, registrira korisnike i surađuje sa CA tijekom izdavanja certifikata. RA osigurava ispravnu identifikaciju korisnika i neporecivosti digitalnih potpisa. CR pohranjuje javne ključeve, certifikate korisnika i liste opozvanih certifikata (eng. *Certificate Revocation List - CRL*).
Certifikat je kompromitiran ukoliko se izda lažno predstavljenoj osobi. Kako bi se to spriječilo na nivou države osnivaju se TRA (eng. *Trusted Registration Authority*) i TCA (eng. *Trusted Certificate Authority*) tijela, kao institucije od najvišeg povjerenja. U slučaju kompromitiranja tajnog ključa korisniku se izdaje novi par ključeva i certifikat, a stari certifikat dospijeva na listu opozvanih certifikata.

7.2. Norme sigurnosti

Sudionici u razmjeni digitalno potpisanih poruka trebaju koristiti iste norme sigurnosti. Najvažnije propisane norme su:

- **ISO** (eng. *International Organization for Standardization*) / **IEC** (eng. *International Electrotechnical Commision*) - norma 9798 definira:
  - identifikaciju sudionika na računalnoj mreži,
  - tehnike digitalnog potpisa,
  - kriptografske algoritme.
- **NIST** definira algoritme za kriptiranje dokumenta:
  - DES (eng. *Data Encription Standard*),
  - DSA (eng. *Digital Signature Algorithm*).
- **ANSI** (American National Standards Institute) definira norme:
  - x 9.17 - upravljanje ključevima,
  - x 9.30 - kriptografija javnog ključa.
- Struktura certifikata – norma **ISO X.509 Ver 3**:
  - inačica certifikata,
  - serijski broj certifikata,
  - algoritam za izradu digitalnog potpisa,
  - izdavač certifikata,
  - valjanost certifikata (od – do),
  - vlasnik certifikata,

- korišten algoritam za šifriranje javnog ključa (PKCS #1, RSA),
- javni ključ,
- vrsta certifikata (klijent, poslužitelj, osoba, poduzeće),
- javni ključ izdavača certifikata,
- digitalni potpis izdavača certifikata.

- Protokoli za prijenos podataka:
  - **SSL** (eng. *Secure Socket Layer*) – koristi se za identifikaciju klijenta i poslužitelja (razmjena javnih ključeva i certifikata). Osigurava privatnost komunikacije (simetričnim kriptosustavom), identitet sudionika u komunikaciji (asimetričnim kriptosustavom) i jamči pouzdan prijenos podataka (provjera integriteta *hash* funkcijama).
  - **S/MIME** (eng. *Secure Multipurpose Internet Mail Extension*) – koristi se pri prijenosu elektroničke pošte i dodavanje certifikata poruci.
  - **SET** (eng. *Secure Electronic Transaction*) – koristi se pri bankarskim transakcijama i sadrži protokole za naručivanje roba i usluga, autorizaciju plaćanja i provjeru certifikata.

8. Budućnost digitalnog potpisa

Ako se nastave današnji trendovi u korištenju digitalnog potpisa oni će s vremenom potpuno zamijeniti pisane potpise u primjenama od dugoročnih korporativnih ugovora do osobnih pisama i dnevnika. Mnogi računalni stručnjaci slažu se kako će, kroz određeno vrijeme, za pokretanje svih programskih paketa i pristup svim datotekama biti potreban odgovarajući digitalni potpis.

Povećanje učestalosti primjene i važnosti digitalnog potpisa iziskuje standardizaciju algoritama koji ih implementiraju i to na razini država, na razini Interneta te od strane organizacija kao što su IEEE (eng. *Institute of Electrical & Electronics Engineers*) i ANSI (eng. *American National Standards Institute*).

Prednosti DS algoritama koji koriste eliptičke krivulje vjerojatno će rezultirati porastom njihove popularnosti i postupnim istiskivanjem drugih algoritama iz uporabe. Sve veće prihvaćanje digitalnog potpisa kao ravnopravne zamjene pisanog potpisa porasti će i učestalost pokušaja zlouporabe, pa je zbog toga potrebno razvijati sigurnosne mehanizme kojima bi se to spriječilo. Prije svega se to odnosi na korištenje sklopovskih rješenja za čuvanje privatnog ključa.

9. Zaključak

Digitalizacija modernog ureda i sve češća upotreba Interneta u poslovnim primjenama dovodi do potrebe sigurnog i pouzdanog utvrđivanja autentičnosti dokumenata. Digitalni potpisi se zbog toga sve više, po učestalosti uporabe i važnosti dokumenata koje potpisuju, približuju klasičnim pisanim potpisima i samo je pitanje vremena kada će ih učiniti nepotrebnima.
Sve češća upotreba digitalnih potpisa dovodi do veće učestalosti napada – pokušaja krivotvorenja. Zbog toga je potrebno definirati politiku upravljanja digitalnim potpisima koja detaljno propisuje dozvoljene i sigurne načine korištenja digitalnih potpisa, propisuje korištenje infrastrukture digitalnih potpisa te korištenje sigurnosnih normi. Na razini pojedinog korisnika, najbolji način zaštite je očuvanje tajnosti privatnog ključa, korištenjem sklopovske, biometrijske i zaštite privatnim identifikacijskim brojem. Osim toga potrebno je koristiti ažurne implementacije DS algoritama, kod kojih su uklonjeni svi eventualni sigurnosni nedostaci.

10. Reference

[1] Digital Signature Standards, http://csrc.nist.gov/publications/fips/fips186-2/fips186-2-change1.pdf, veljača 2007.
[2] How do digital signatures work?, http://computer.howstuffworks.com/question571.htm, veljača 2007.
[3] William Stallings: Cryptography and Network Security Principles and Practices, 4 izdanje, 2005.
[4] Don Johnson, Alfred Menezes, Scott Vanstone: The Elliptic Curve Digital Signature Algorithm (ECDSA), Certicom Research, 2001
[5] RSA Laboratories: RSA Cryptography Standard, 2002.
[6] Infomosaic Corporation: Digital Signature in a Web Application, DoD PKE Forum Fall 2005, Atlanta
[7] XML-Signature Syntax and Processing, http://www.w3.org/TR/xmldsig-core/, veljača 2007.
[8] Ching-Young Lin, Watermarking and Digital Signature Techniques for Multimedia Authentication and Copyright Protection, doktorska disertacija, Columbia University, 2000.
[9] RSA Laboratories: What is a blind signature scheme?, http://www.rsa.com/rsalabs/node.asp?id=2339, veljača 2007.
[10] Danijel Mokran: Blind Signature, http://os2.zemris.fer.hr/protokoli/2003_mokran/index.html, veljača 2007.
[11] Shafi Goldwasser, Silvio Micali, Ronald L. Rivest: A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks, MIT Laboratory for Computer Science, 1995.
[12] Digital Signature and Confindentiality Certificate Policies, http://www.solutions.gc.ca/pki-icp/guidedocs/cert-policy/cp-pctb_e.asp, veljača 2007.
[13] Paul C. Kocher: Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems, http://cnscenter.future.co.kr/crypto/algorithm/signature.html, veljača 2007.
[14] Digitalni potpisi, http://www.zpr.fer.hr/predmeti/erg/2004/ostroski/index.html, veljača 2007.