Biometriával ötvözött digitális aláírás

JEGES ERNŐ, HORNÁK ZOLTÁN

BME, Méréstechnika és Információs Rendszerek Tanszék, SEARCH Laboratórium

jeges@mit.bme.hu

hornak@mit.bme.hu

Kulcsszavak: biometria, nyilvános kulcsú kriptográfia, hibajavító kódolás, csatornakódolás

Az elektronikus ügyintézések biztonsága megköveteli, hogy biztonságos elektronikus aláírási technikák álljanak rendelkezésünkre, amelyek erős kriptográfiai módszerek révén biztosítják, hogy a dokumentum aláírója azonosítható, az aláírás ténye letagadhatatlan, valamint az aláírt dokumentum tartalma sértetlen legyen. A bemutatott biometriával ötvözött digitális aláírás technológia alapvetően az aláíró fél azonosításának, és az aláírás letagadhatatlanságának megerősítésére koncentrál.

A jelenlegi elektronikus aláírási rendszerekben a leggyengébb, nem erősíthető láncszemet a valódi személy és az őt azonosító titkos kulcs közötti kapcsolat képezi. A titkos kulcsot tartalmazó eszköz eltulajdonítható, míg a kulcshoz való hozzáférés a mai megoldásokban csupán jelszavas megoldással védhető, ami nem bizonyító erejű.

Az általunk javasolt módszer alapötlete az, hogy az aláíró felet azonosító titkos kulcsot olyan kódolt formában tároljuk, hogy csak a kulcs tulajdonosának ujjnyomatából kiolvasható adat segítségével legyen visszaállítható, és csak így lehessen aláírást készíteni vele. Ilyen módon a kódolt titkos kulcs, vagyis a kártya eltulajdonítása révén sokkal nehezebb visszaélni, hiszen az aláírás elkészítéséhez szükséges még – a jelenlegi PIN kód mellett – a tulajdonos ujjnyomata is. Fontos megjegyezni, hogy ez a módosítás nem befolyásolja a nyilvános kulcs használatát, így a tanúsítvány és az aláírás visszaellenőrzésének folyamata teljesen kompatíbilis a jelenlegi PKI ajánlásokkal és a meglévő alkalmazásokkal.

1. A digitális aláírás

Dokumentumok hitelességének igazolására, illetve vitás esetekben az eredetiség eldöntésére egy rendkívül egyszerű, de mégis kellően hatásos módszer a hagyományos, kézírással készített aláírás. A kézírást pontosan utánozni nehéz, a hamisított aláírást szakértők nagy bizonyossággal képesek felismerni, illetve megkülönböztetni a valódi hiteles aláírástól. Mivel tehát az írás egyértelműen az adott személyre jellemző, a *hamisíthatatlanságon* túl az aláírás egy igen fontos tulajdonsága a *letagadhatatlanság*. Ha valaki valamit kézírásával leír, akkor nem tudja annak tényét letagadni, mert a kézírását nehezen tudja úgy megváltoztatni, hogy az azonosságot szakértők ne tudnák megállapítani.

A hagyományos kézírással történő aláírás analógiájára született meg a digitális aláírás az elektronikus dokumentumok hitelességének az igazolására. A jelenlegi elektronikus aláírások alapja egy titkos kulcs (az elektronikus aláírásról szóló törvény szóhasználatával élve „aláírás létrehozó adat”), amely használatával – feltételezve, hogy az csak a jogosult személy *birtokában* lehet – a háttérben alkalmazott kriptográfia garantálja, hogy az illető nevében más digitális aláírást nem képes készíteni. Ezen rendszerek leggyengébb láncszeme pontosan ez a birtokviszony, tehát a titkos kulcs eltulajdoníthatóságának a kérdése. A vonatkozó törvény úgy rendelkezik, hogy akár a jogos felhasználó, akár más írt alá a kulcs felhasználásával, a kötelezettségvállalás következményeit a kulcs tulajdonosának kell teljesítenie, vagyis nem a valós aláírót, hanem a felelősséget vizsgálja, ami lényeges különbség.

Az aláíró személy és a nyilvános kulcs egymáshoz rendelésének megoldásai közül egyértelműen a *nyilvános kulcsú infrastruktúrát* (PKI), vagy az általa menedzselt elektronikus igazolványok rendszerét támogatják a törvényi szabályozások [1]. A PKI rendszerében egy mindenki által megbízhatónak elfogadott harmadik fél *(Certificate Authority, CA)* elektronikus dokumentumba foglalja az adott személy nevét és más azonosító jellemzőit, illetve a nyilvános kulcsát, majd ezeket együttesen a tanúsító intézet a saját titkos kulcsával aláírja, ezzel biztosítva, hogy észrevétlenül nem történhet változtatás a rögzített adatokban.

Eddig a titkos kulcs birtoklását általában chipkártyával és a hozzá tartozó titkos PIN kóddal oldották meg. Az ennél jóval erősebb biometrikus módszereket szinte kizárólag a PIN kódot kiváltó megoldásként alkalmazták, ami azonban nem gátolja meg, hogy a titkos kulcs illetéktelen személy birtokába kerüljön.

Cikkünkben egy olyan, a hagyományos nyilvános kulcsú infrastruktúrára épülő módszert mutatunk be, ahol a biometrikus azonosítás – esetünkben ujjnyomat alapú azonosítás – olyan módon és olyan mértékben épül be az elektronikus aláírás folyamatába, hogy a titkos kulcs az esetlegesen alkalmazott chipkártya eltulajdonításával és feltörésével sem szerezhető meg, mert a kulcs gyakorlatilag a jogosult személy ujjnyomatában van eltárolva, illetve annak segítségével van kódolva.

2. Biometria bevonása az aláírási folyamatba

Számos módszer ismert a biometria területén belül az ujjnyomat alapú megoldástól a hang-, illetve arcfelismerésen keresztül a retina és írisz vizsgálatáig. Az egyes módszerek jelentős eltérést mutatnak abból a szempontból, hogy mekkora bizonyossággal, milyen tévedési arányokkal képesek felismerni személyeket, illetve mennyire könnyű őket becsapni (ujjnyomatról készült szilikon replikával, egy hangfelvétel viszszajátszásával vagy éppen egy fénykép felmutatásával). Ennek tükrében az olcsó és egyszerű, illetve a drága de megbízhatóbb megoldások között lehet válogatni; biometrikus digitális aláírás megvalósításához legcélszerűbbnek az ujjnyomat alapú megoldás mutatkozott, amely egyrészt optimális az ár-teljesítmény viszony tekintetében, másrészt az ujj leolvasóra helyezése emberi szempontból is jól kifejezi az aláírás folyamatát és szándékát.

Az ujjnyomatok azonosítására számos alapmódszert dolgoztak ki az elmúlt évtizedek, sőt évszázadok során, amióta rájöttek, hogy nyomozati és bizonyítási eljárásokban is sikerrel alkalmazhatják az ujjnyomat egyediségét. Az ujjnyomatok gépi kezelése során ezen daktiloszkópiai módszerek közül a minutia alapú azonosítási eljárás vált szinte egyedülállóvá.

1. ábra Példa ujjnyomat képek

A fodorszálak végét, különféle elágazásaik illetve összefutásaik helyét nevezzük minutia-pontoknak. Ezen pontok elhelyezkedése rendkívül jellemző egy adott ujjra. A legtöbb azonosítási módszer kizárólag ezen pontok relatív elhelyezkedéséből dönti el két ujj azonosságát vagy különbözőségét.

A minutia pontok esetében a következő jellemzők vizsgálhatóak:

– *Elhelyezkedés:* a pont síkbeli koordinátája, mely megadható egy képen belül abszolút vagy egy alapponthoz viszonyított relatív értékkel.
– *Irányultság:* minden ponthoz rendelhető egy irányszög, amit az érintett fodorszál vagy fodorszálak iránya határoz meg.
– *Görbület:* a barázda irányultság megváltozásának mértéke.

Az ujjnyomat-képet az ujjnyomat-olvasó eszköz segítségével tapogathatjuk le, amely általában élőujj- és ujjnyomat-replika detektálást is tartalmaz, tehát meg tudja állapítani, hogy az olvasott ujj ténylegesen élő ujjról származik, vagy csak egy, az adott ujj redőit utánzó, úgynevezett replika került az olvasóra.

A minutia alapú ujjnyomat azonosítási módszer alapja tehát a fodorszálak meghatározása, majd azok alapján a minutia pontok helyzetének a vizsgálata. A módszerek által meghatározott pontokat általában különféle szűrőknek vetik alá, kihasználva, hogy a valódi minutia pontok – akárcsak a valódi fodorszálak – jellemzői bizonyos szabályokat követnek. A fedésbe hozható pontok vizsgálatával megállapíthatjuk két ember ujjnyomatának azonosságát, ezáltal azonosítva az ujjnyomatokhoz tartozó embereket.

A biometriával ötvözött, titkos kulcsot tárolni képes rendszerek zöme, mint már említettük, egyszerűen biometrikus módszerekkel védi a hozzáférést a letárolt titkos kulcshoz. Ezzel szemben mi azt a célt tűztük magunk elé, hogy a titkos kulcsot egyáltalán nem tároljuk, hanem azt a kulcspár generálást követően kódoljuk, majd töröljük. A későbbiekben, ha a titkos (aláíró) kulcsra van szükség, a kódolt információból a titkos kulcs visszaállítása csak az ujjnyomatkép ismeretében lehetséges.

A digitális aláíráshoz szükséges titkos kulcs tárolásának megvalósításához a minutia pontok fent felsorolt jellemzői azért fontosak, mert minden egyes független jellemző felhasználható a titkos kulcs bitjeinek a kódolására az ujjnyomat-képben.

3. A titkos kulcs eltárolása az ujjnyomatban

Elektronikus aláírás készítéséhez valamilyen kriptográfiai szempontból erős rejtjelkulcsra, bináris adatra van szükség. Elviekben ez a bitsorozat magából az ujjnyomat-képből is származtatható lenne, de mivel azonos ujjnyomatokból minden esetben bitről bitre azonos bináris adatot kell kapnunk, az olvasó bizonytalansága és az eltérő olvasási környezetből fakadó különbségek miatt ez így nehezen megvalósítható módszernek bizonyult.

A megvalósított módszerünk lényege ennek megfelelően az, hogy a regisztráció során a kinyert minutia pontok alapján egy próba pontsorozatot *(challenge minutia vektort)* tárolunk el, amelybe egyrészt a valós pontokon túl álpontokat is belekeverünk, másrészt a pontok irányultságát megváltoztatjuk – ezáltal kódolva azt az információt, ami végső soron a titkos kulcs visszaállíthatóságát biztosítja. A visszaállítási bizonytalanságok természetesen ilyen módon is megmaradnak, ezeket azonban már képesek vagyunk kezelni hibajavító kódolás alkalmazásával.

A regisztráció során a titkos kulcs generálásához felhasznált (eredeti) bitsorozatot ezek után aláíráskor úgy rekonstruáljuk, hogy a fenti challenge minutia vektort az aktuálisan levett ujjnyomat-mintából kapott minutia pontokkal „ütköztetjük”. Ezen művelet eredményeképpen, a megfelelő hibajavítás után visszakapjuk az eredeti bitsorozatot, amely segítségével képesek vagyunk a tanúsítványban szereplő publikus kulcshoz tartozó titkos kulcs újragenerálására.

A fentiekből már látható, hogy a módszer két fő folyamatból áll: a regisztrációból és az aláírásból. Az első során áll elő a publikus kulcsot tartalmazó tanúsítvány, illetve a challenge minutia vektor; az aláírási folyamat során a challenge minutia vektor és a levett minutia pontok alapján újra előáll a titkos kulcs, amely segítségével megtörténik az aláírás.

A regisztráció lépései a következők (a vastag betűk a folyamat során előálló, elmentett eredményeket jelölik):

- Megfelelő hosszú véletlen bináris vektor előállítása.
- A bináris vektor bővítése megfelelő számú paritásbittel a hibajavításhoz.
- Az így előállt teljes bitsorozat, illetve a regisztrált ujjnyomat-mintából nyert minutia pontok alapján a **challenge minutia vektor** előállítása, elmentése. A kódolás menetét alább mutatjuk be.
- A teljes bitsorozat alapján RSA kulcspár generálása, a titkos kulcs törlése.
- **Tanúsítvány** készítése a publikus kulcs alapján, kapcsolódva valamely létező nyilvános kulcsú infrastruktúrához.

Az aláírás során a feladat a titkos kulcs rekonstruálása. Ehhez az aktuálisan levett ujjnyomat mintán kívül a rendelkezésünkre áll a challenge minutia vektor, illetve a létrejött titkos kulcs ellenőrzéséhez a neki megfelelő, tanúsítványba foglalt publikus kulcs.

- A teljes bitsorozat visszaállítása a challenge minutia vektor és az aktuális ujjnyomat minutia pontjai alapján. Ez a regisztrációkor használt kódolás inverz műveleteként is felfogható, tehát mint egy dekódolás.
- A teljes visszaállított bitsorozat hibajavítása.
- A teljes bitsorozat alapján az RSA kulcspár generálása.
- A generált publikus kulcs és a tanúsítványba foglalt publikus kulcs összehasonlítása. Egyezés esetén a titkos kulcs elfogadása, aláírás; ellenkező esetben hibajelzés.

A továbbiakban ismertetjük a fenti rendszer megvalósítása közben felmerült problémákat, illetve az ezen problémákra általunk adott megoldásokat.

Kódolás és dekódolás

A kódolás folyamata nem más, mint a challenge minutia vektor előállítása. Ennek során a teljes bitsorozat bitjeinek megfelelően generáljuk a challenge vektor pontjait. Esetünkben a bitsorozatot ötösével kezeljük, amiből nyilvánvaló, hogy a hibajavító kódokkal bővített teljes bitsorozat hosszának öttel oszthatónak kell lennie. Az egyes bit-ötösöket a következő séma szerint kódoljuk:

0	1	2	3	4
Ál / valós	Irányultság módosítása			

2. ábra Az egy minutia-pontnak megfelelő ötbites futam

Mint az az ábrán látható, a teljes bitsorozatot ötös futamokra bontjuk, és ezen részek alapján generáljuk sorra a challenge minutia vektor pontjait.

Ha egy ilyen futam első (0.) bitjének értéke 1, valós minutia pontot választunk a regisztráltak közül, ellenkező esetben véletlenszerű ál-minutia pontot generálunk a pont alatt található fodorszálak irányultságával. A további 4 bit értékétől függően az adott ál- vagy valós pont szögét módosítjuk *dFi* szöggel a következő táblázat szerint:

Kód bitek	dFi (fok)	Kód bitek	dFi (fok)
0000	0,00	1100	90,00
0001	11,25	1101	101,25
0011	22,50	1111	112,50
0010	33,75	1110	123,75
0110	45,00	1010	135,00
0111	56,25	1011	146,25
0101	67,50	1001	157,50
0100	78,75	1000	168,75

3. ábra A 180 fokos szögkódolás

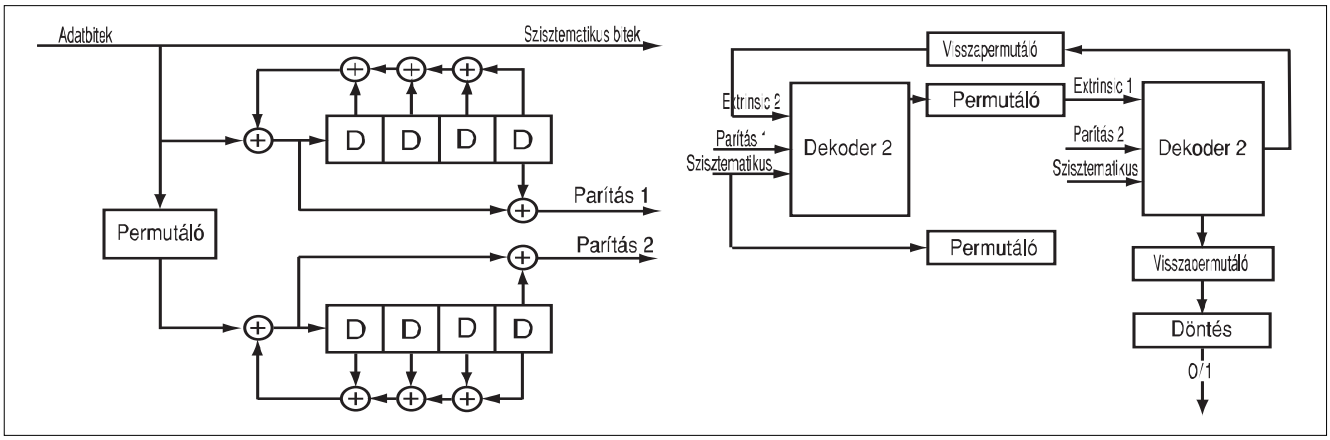
A fenti táblázatból kiolvashatjuk, hogy amennyiben az 1-4 bitek értéke *1101*, a vektorban szereplő minutiapont eredeti szögét 101.25 fokkal (9 x 11.25°) kell módosítani (modulo 180). Ez tulajdonképpen megfelel a 11.25 fokos lépésközben felírt szögmódosítás értékek Gray-kódolásának, amely tulajdonsága az, hogy a szomszédos értékek Hamming távolsága 1. Ez esetünkben azért fontos, mert ez a kódolás a szögmegállapítás bizonytalanságából adódó bithibákat a minimálisra csökkenti, ráadásul a következőkben ismertetett hibajavító kódoláshoz illeszkedve dekódoláskor a határhelyzetben talált szögek esetében a megfelelő bitekben törléses hibákat jelezhetünk, ezzel is növelve a hibajavító képességet.

Hatékony hibajavítás

Az általában hibajavításra használt csatornakódolókkal ellentétben esetünkben a kódolás és a dekódolás ideje kevésbé volt kritikus paraméter, a hibajavító módszer kiválasztását inkább az a tény határozta meg, hogy az ujjnyomat általunk választott kódolása jelentős bizonytalanságot hordozott. Ennek megfelelően – a szokásos fogalmakkal élve – egy igen alacsony jel/zaj viszonnyal rendelkező csatornával álltunk szemben.

Számos alternatíva megvizsgálása után nyilvánvalóvá vált, hogy például az űrtávközlésben használt, hatékony hibajavító paraméterekkel rendelkező, és jól skálázható Turbó-kódolás az, amely igényeinket kielégítheti. A Turbó-kódoló alapötlete az, hogy két (vagy akár több), párhuzamosan kapcsolt rekurzív konvolúciós kódolót használunk [3].

Az első az eredeti szisztematikus bitekből, a második pedig azok permutációjából állít elő paritásbiteket. Az ilyen módon párhuzamosan képzett paritásbiteket a szisztematikus bitekhez fűzzük; így, amennyiben az egyenként előállt paritásbitek száma megegyezik a szisztematikus bitek számával, egy 1/3 jelsebességű kódot kapunk (a kódszó hoszsza a szisztematikus bitek



4. ábra Két konvolúciós kódolót tartalmazó Turbó-kódoló (balra) és dekódoló (jobbra)

száma plusz a két kódoló által előállított paritás bitek száma). További ötlet, amely a Turbó-kódolást igen jól skálázhatóvá teszi az, hogy nem viszünk át a csatornán minden paritásbitet, hanem közülük bizonyos minta szerint törlünk. Így tetszőleges jelsebességet érhetünk el, természetesen valamelyest veszítve a hibajavítási képességen.

A konvolúciós kódok dekódolásához hasonlóan dekódoláskor itt is a kapott bitek alapján becsüljük az egyes kódolási lépésekben a konvolúciós kódoló állapotát, ezáltal az eredeti szisztematikus bitek értékét is, felhasználva a csatorna kimenetén vett értékeket és az előző dekódoló által szolgáltatott információt. Egy tipikus, két konvolúciós kódolót tartalmazó Turbó-kódoló és az ennek megfelelő dekódoló elrendezés a *4. ábrán* látható.

Az előzőekben bemutatott kódolásnak megfelelően az egyes bitek átvitelének modellezésére létrehoztuk a nem-szimmetrikus bináris törléses csatorna-modellt (NBEC). Ez a modell az egyes bitek átvitelekor értelmezi a törléses hibát, így különböző hibaparaméterrel írhatjuk le mind az egyszerű, mind a törléses hiba valószínűségét különböző bemeneti bit-értékek esetében (innen az *aszimmetrikus* elnevezés). Az NBEC csatornát tehát négy hibaparaméterrel (p_{0X}, p_{01}, p_{1X} és p_{10}) írhatjuk le.

Mivel a kódolás bit-ötösöket rendel egy-egy minutiaponthoz, és ezt az öt bitet különböző szabályok szerint kódoljuk, az egyes bitek (0-4) esetében más-más hibaparaméter-négyest definiálhatunk. Ilyen módon kapjuk meg a tényleges alkalmazott $NBEC_5$ csatornamodellt, amelyet 5, egymástól független NBEC csatornaként képzelhetünk el *(5. ábra)*.

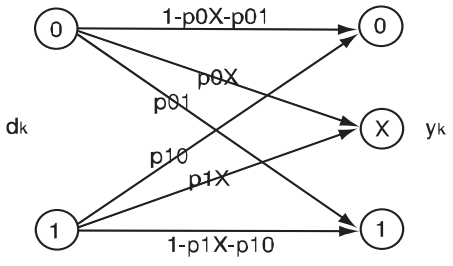
Mint már említettük, a paritásbitekkel kiegészített teljes bitsorozat hosszának oszthatónak kell lennie ötel. Az átvitelre nem kerülő paritásbitek adott minta szerinti törlése miatt 8-al is osztható hosszú bitsorozatot kell választanunk. A különböző lehetséges paritásbit hosszak áttekintése után a 120+120 bites kódszó mellett döntöttünk, ami azt jelenti, hogy a létrejött paritásbitek felét töröljük, ezáltal egy 1/2 jelsebességet kódolást kapunk. Ilyen módon a 120 darab szisztematikus véletlenbit kriptográfiai értelemben erős kulcsot biztosít, miközben a challenge vektor 240/5=48 darab minutia-pontot tartalmaz.

Az ötös futamokon belül a 0. bit kódolását figyelembe véve tehát átlagosan 24 valós minutiára lesz szükségünk, hiszen a véletlen bitek átlagosan fele nulla, ami megfelel az egy ujjnyomatban fellelhető minutiapontok eloszlásának, hiszen a 600 ujjat tartalmazó adatbázisunkban az átlagos minutia-pont szám 40-re adódott.

Determinisztikus kulcspár generálás

Ahhoz, hogy a korrektül visszaállított bitsorozatból minden esetben ugyanazt a kulcspárt kapjuk, módosítanunk kellett az OpenSSL véletlenszám-generátorát, amelyre a kulcsgenerálás támaszkodik. Így az általunk

5. ábra A nem szimmetrikus bináris törléses csatorna (NBEC), és az NBEC5 statisztikailag meghatározott hibaparaméterei



	0	1	2	3	4
p_{00}	0,53	0,46	0,45	0,42	0,36
p_{0X}	0,00	0,02	0,02	0,03	0,05
p_{01}	0,01	0,02	0,02	0,04	0,09
p_{10}	0,13	0,02	0,02	0,04	0,08
p_{1X}	0,03	0,02	0,02	0,03	0,05
p_{11}	0,31	0,45	0,46	0,42	0,38

előállított bitsorozatot adagolva a generátor magjának, az OpenSSL minden esetben azonos véletlenszámsorozatot, ezáltal azonos kulcspárt generál.

4. Összefoglaló

Mint a biometrikus rendszerek esetében általában, a biometrikus digitális aláírás esetében is a téves visszautasítások *(FRR – False Rejection Rate)* és a téves elfogadások *(FAR – False Acceptance Rate)* száma az alapvető hibaparaméter. Esetünkben az első paraméter elsősorban a képvételi hibák, a mintavétel bizonytalanságából adódik, a másodikat pedig elsősorban az erős hibajavítás növelheti. Egy biometrikus rendszer hangolásánál a két érték általában egymás ellen hat, így a projektünk során is meg kellett találnunk a paraméterek azon halmazát, ahol a két hibaparaméter a megfelelő értéket mutatja.

Az általunk választott paraméterek mellett, tesztjeink során az FAR 10^{-6}-nál kisebb értékre adódott, míg az FRR értéke 15% körüli volt. Mindkét érték a biometrikus rendszerek esetében elfogadható határon belül van, eme utóbbi azonban vélhetően tovább csökkenthető különböző szűrők alkalmazásával, illetve a mintán fellépő nem-lineáris torzulások megfelelő kezelésével.

Összefoglalásként elmondhatjuk, hogy a biometrikus digitális aláírás megvalósítható, és több szempontból is erősebb védelmet nyújt, mint a mára már hagyományossá váló pusztán chipkártyán alapuló titkos kulcs tárolási módszerek. A kidolgozott eljárás rendkívüli erénye, hogy mind a tanúsítvány formátuma és tartalma, mind az aláírás ellenőrzés folyamata teljesen megegyezik a mai megoldásokkal, teljes mértékben kompatibilis azokkal.

A bemutatott eljárás a digitális aláírás használatának széleskörű elterjedése által gerjesztett növekvő felhasználói igények mellett komoly sikerre számíthat.

Köszönetnyilvánítás

A kutatási projektet az Infokommunikációs Technológiák és Alkamazások támogatta (IKTA-00160/2002).

Irodalom

[1] 2001. évi XXXV. törvény az elektronikus aláírásról; www.ihm.hu/miniszterium/jogszabalyok/ealairas.pdf
[2] The OpenSSL Project; OpenSSL: The Open Source toolkit for SSL/TLS; www.openssl.org/
[3] Guangchong Zhu, Performance Evaluation of Turbo Codes. Queen's University, Kingston, Ontario, Canada, 1998. http://markov.mast.queensu.ca/Papers/zhu_proj98.ps