ISSN 1392 - 1215 ELEKTRONIKA IR ELEKTROTECHNIKA. 2003. Nr. 4(46)

T 180 TELEKOMUNIKACIJŲ INŽINERIJA

Skaitmeninio „voko“ panaudojimo RSA skaitmeninio parašo algoritme tyrimas

A. Vobolis, P. Nefas
Telekomunikacijų katedra, Kauno technologijos universitetas
Studentų g. 50, LT-3031 Kaunas, Lietuva

Įvadas

Skaitmeniniai „vokai“ buvo išrasti siekiant pasinaudoti viešojo ir slaptojo rakto kriptografijų privalumais. Pagrindiniai privalumai, naudojant viešojo rakto kriptografiją, yra šie: užtikrinamas didesnis saugumas, paprasčiau įdiegti šifravimo algoritmą, nes privačiojo rakto niekada nereikia perduoti ar atskleisti. Viešojo rakto kriptografijoje taip yra garantuojamas skaitmeninio parašo autentiškumas. Slaptojo rakto kriptografijoje slaptasis raktas turi būti perduodamas, nes šifravimas ir dešifravimas yra simetrinės procedūros. Iškyla pavojus, kad perdavimo metu slaptasis raktas gali patekti priešiškam objektui, tokiu atveju šifravimo algoritmas tampa visiškai pažeidžiamas.

Taigi viešojo rakto kriptografijoje kiekvienas vartotojas turi saugoti savo privatųjį raktą. Pagrindinis viešojo rakto kriptografijos trūkumas yra šifravimo greitis. Beveik visi slaptojo rakto šifravimo algoritmai yra daug spartesni už viešojo rakto šifravimo algoritmus [1].

Praktikoje šifravimui naudojami viešojo rakto šifravimo algoritmai, garantuojantys didesnį saugumą kartu su slaptojo rakto šifravimo algoritmais, garantuojančiais didesnį šifravimo greitį. Toks protokolas vadinamas skaitmeniniu „voku“. Skaitmeninis „vokas“ susideda iš šifruoto pranešimo, sukurto naudojant slapto rakto šifravimo algoritmus, ir šifruoto slaptojo rakto, užšifruoto naudojant viešojo rakto šifravimo algoritmus. Vienas iš pagrindinių skaitmeninių „voko“ privalumų yra tai, kad slaptieji raktai gali būti dažnai keičiami. Raktų keitimo procedūros nusako vieną iš pagrindinių šifravimo algoritmo atsparumų, nes priešiškam objektui daug sunkiau dešifruoti sistemą, kurioje slaptieji raktai naudojami tik trumpą laiko intervalą.

Slaptieji raktai yra neilgalaikiai dėl keleto priežasčių, kurių viena iš svarbiausiųjų yra tai, kad, ilgai naudojantis vieninteliu slaptuoju raktu, priešiškas objektas gali analizuoti didelius pranešimų kiekius, šifruotus tuo pačiu raktu, o tai sumažina šifravimo algoritmo atsparumą. Kiekvienais metais sistemos, analizuojančios šifruotus pranešimus, tampa vis spartesnės, atsiranda naujų patobulintų faktorizavimo algoritmų, todėl šifravimo algoritmų atsparumui užtikrinti raktų ilgius ir jų gyvavimo trukmes reikia keisti [2],[5].

Skaitmeninio „voko“ principas gali būti panaudotas RSA skaitmeninio parašo algoritme. Objektas *A* pasirašo pranešimą ir užšifruoja jį objekto *B* viešuoju raktu. Objektas *B,* gavęs pranešimą, iššifruoja jį savo privačiuoju raktu ir patikrina skaitmeninio parašo autentiškumą. Naudojant tokį RSA skaitmeninio parašo algoritmą, gali kilti perblokavimo problema, todėl būtina atsižvelgti į naudojamų modulių išraiškas.

RSA skaitmeninio parašo generavimas

Generuodami RSA skaitmeninio parašo algoritmą objektui *A*, pasirenkame pirminius skaičius p=37 bei q=43 ir apskaičiuojame n=1591 ir ϕ=1512. Pasirenkame atsitiktinį sveikąjį skaičių e =11 iš intervalo $1<e<1512$, tenkinantį sąlygą $gcd(11,1512)=1$. Gaunamas objekto *A* viešas raktas (1591,11).

Skaičiuodami objekto *A* privatųjį raktą, naudojame išplėstinį Euklido algoritmą. Iš dviejų teigiamų sveikųjų skaičių a ir b, kurie tenkina sąlyga $a \geq b$, apskaičiuojamas $d=gcd(a,b)$ ir sveikieji skaičiai x ir y, tenkinantys lygtį $ax + by = d$.

Jeigu $b = 0$, nustatoma $d \leftarrow a$, $x \leftarrow 1$, $y \leftarrow 0$ ir grįžtama (d,x,y). Nustatoma $x_2 \leftarrow 1$, $x_1 \leftarrow 0$, $y_2 \leftarrow 0$, $y_1 \leftarrow 1$, *iki* $b>0$ atliekami žingsniai:

1. $q \leftarrow \lfloor a/b \rfloor$, $r \leftarrow a-qb$, $x \leftarrow x_2-qx_1$, $y \leftarrow y_2-qy_1$,
2. $a \leftarrow b$, $b \leftarrow r$, $x_2 \leftarrow x_1$, $x_1 \leftarrow x$, $y_2 \leftarrow y_1$, $y_1 \leftarrow y$,
3. nustatomi $d \leftarrow a$, $x \leftarrow x_2$, $y \leftarrow y_2$ ir grįžtama (d,x,y).

Pasirinkę e=11, išsprendžiame $ed=11d \equiv 1 \pmod{1512}$.

1 lentelė. Skaičiavimų rezultatai

q	r	x	y	a	b	x_2	x_1	y_2	y_1
-	-	-	-	1512	11	1	0	0	1
137	5	1	-137	11	5	0	1	1	-137
2	1	-2	275	5	1	1	-2	-137	275
5	0	11	-1512	1	0	-2	11	275	-1512

Randamas objekto *A* privatus raktas d = 275.

Generuodami RSA skaitmeninio parašo algoritmą objektui *B*, pasirenkame pirminius skaičius p=83 ir q=97, apskaičiuojame n=8051 ir ϕ=7872. Iš intervalo $1<e<7872$, pasirenkame atsitiktinį sveikąjį skaičių e =5 tenkinantį sąlygą $gcd(5,7872)=1$. Gaunamas objekto *B* viešasis raktas (8051,5).

Pasirinktas e=5. Išsprendžiame ed=5$d \equiv 1 (mod\ 7872)$.

2 lentelė. Skaičiavimų rezultatai

q	*r*	*x*	*y*	*a*	*b*	x_2	x_1	y_2	y_1
-	-	-	-	7872	5	1	0	0	1
1574	2	1	-1574	5	2	0	1	1	-1574
2	1	-2	3149	2	1	1	-2	-1574	3149
2	0	5	-7872	1	0	-2	5	3149	-7872

Apskaičiuojamas objekto B privatusis raktas d=3149.

RSA skaitmeninio parašo algoritmo generavimas skaitmeninio „voko“ principu. 1 variantas

Naudodamas RSA skaitmeninio parašo algoritmą skaitmeninio „voko“ principu objektas A siunčia žinutę objektui B. Palyginę abiejų objektų modulius, matome, kad $n_A < n_B$. Tariame, kad $M = Z_n$ ir kad perteklinė funkcija R: $M \rightarrow Z_n$ yra $R(m)=2^t m+1$ visiems $m \in M$. Pasirenkame t=1. Pranešimui m=14 pasirašyti objektas A apskaičiuoja perteklinę funkciją $\tilde{m} = R(m)$=29 ir skaitmeninį parašą:

$$s = \tilde{m}^{d_A} mod\ n_A = 29^{275} mod\ 1591 = 578. \qquad (1)$$

Skaičiuodami skaitmeninį parašą naudojame kvadratinio dauginimo algoritmą , kur $a \in Z_n$, k- sveikasis skaičius $0 \le k < n$, kurio dvejetainė išraiška yra

$$k = \sum_{i=0}^{t} k_i 2^i. \qquad (2)$$

Atliekami žingsniai :

1. $b \leftarrow 1$, jei k=0, grįžtama *(b)*, $A \leftarrow a$, jei k_0=1, tai $b \leftarrow a$;
2. nuo i=1 iki t: $A \leftarrow A^2 mod\ n$, jei k_i=1, tai $b \leftarrow A \cdot b mod\ n$;
3. grįžtama (b).

3 lentelė. Skaičiavimų rezultatai

i	0	1	2	3	4
k	1	1	0	0	1
A	29	841	877	676	359
b	29	524	524	524	378

i	5	6	7	8
k	0	0	0	1
A	10	100	454	877
b	378	378	378	578

Pasirašęs pranešimą objektas A jį užšifruoja objekto B viešuoju raktu ir išsiunčia:

$$c = s^{e_B} mod\ n_B = 578^5 mod\ 8051 = 2662. \qquad (3)$$

4 **lentelė.** Skaičiavimų rezultatai

i	0	1	2
k	1	0	1
A	578	3993	3069
b	578	578	2662

Objektas B, gavęs šifruotą pranešimą, pirma jį iššifruoja privačiuoju raktu ir gauna skaitmeninį parašą:

$$s = c^{d_B} mod\ n_B = 2662^{3149} mod 8051 = 578. \qquad (4)$$

5 lentelė. Skaičiavimų rezultatai

i	0	1	2	3	4	5
k	1	0	1	1	0	0
A	2662	1364	715	4012	2195	3527
b	2662	2662	3294	3837	3837	3837

i	6	7	8	9	10	11
k	1	0	0	0	1	1
A	934	2848	3747	7116	4717	5176
b	1063	1063	1063	1063	6449	578

Pasinaudojęs objekto A viešuoju raktu, objektas B gauna pranešimą $\tilde{m}$:

$$\tilde{m} = s^{e_A} mod\ n_A = 578^{11} mod\ 1591 = 29. \qquad (5)$$

6 lentelė. Skaičiavimų rezultatai

i	0	1	2	3
k	1	1	0	1
A	578	1565	676	359
b	578	882	882	29

Primantysis objektas B autentifikuoja skaitmeninį parašą, kadangi $\tilde{m}$ atitinka perteklinę funkciją $\tilde{m} \in M_R$ ir apskaičiuoja $m = R^{-1}(\tilde{m})$=14.

RSA skaitmeninio parašo algoritmo generavimas skaitmeninio „voko“ principu. 2 variantas

Siųstoji ir gautoji žinutės sutampa, tačiau naudojant RSA skaitmeninio parašo algoritmą skaitmeninio „voko” principu, kai objektas B siunčia pranešimą objektui A, kur objektų moduliai $n_A < n_B$, gaunamas neadekvatus rezultatas.

Tariame, kad $M = Z_n$ ir kad perteklinė funkcija R: $M \rightarrow Z_n$ yra $R(m)=2^t m+1$ visiems $m \in M$. Pasirenkame t=1. Pranešimui m=14 pasirašyti objektas B apskaičiuoja perteklinę funkciją $\tilde{m} = R(m)$=29 ir skaitmeninį parašą:

$$s = \tilde{m}^{d_A} mod\ n_A = 29^{3149} mod\ 8051 = 6967. \qquad (6)$$

7 lentelė. Skaičiavimų rezultatai

i	0	1	2	3	4	5
k	1	0	1	1	0	0
A	29	841	6844	7669	1006	5661
b	29	29	5252	6486	6486	6486

i	6	7	8	9	10	11
k	1	0	0	0	1	1
A	3941	1102	6754	7601	1225	3139
b	7452	7452	7452	7452	6917	6967

Pasirašęs pranešimą objektas B jį užšifruoja objekto A viešuoju raktu ir išsiunčia:

$$c = s^{e_B} mod\ n_B = 6967^{11} mod\ 1591 = 1248. \qquad (7)$$

8 lentelė. Skaičiavimų rezultatai

i	0	1	2	3
k	1	1	0	1
A	6967	861	1506	861
b	6967	517	517	1248

Objektas *A,* gavęs šifruotą pranešimą, pirma jį iššifruoja privačiuoju raktu ir gauna skaitmeninį parašą:

$$s=c^{d_B} \bmod n_B=1248^{275} \bmod 1591=603. \qquad (8)$$

9 lentelė. Skaičiavimų rezultatai

i	0	1	2	3
k	1	1	0	0
A	1248	1506	861	1506
b	1248	517	517	517

i	4	5	6	7	8
k	1	0	0	0	1
A	861	1506	861	1506	861
b	1248	1248	1248	1248	603

Pasinaudojęs objekto *B* viešuoju raktu, objektas *A* gauna pranešimą $\tilde{m}$:

$$\tilde{m}=s^{e_A} mod\ n_A=603^5 mod\ 8051=6221. \qquad (9)$$

10 lentelė. Skaičiavimų rezultatai

i	0	1	2
k	1	0	1
A	603	1314	3682
b	603	603	6221

Priimantysis objektas A atmeta skaitmeninį parašą, kadangi $\tilde{m}$ neatitinka perteklinės funkcijos $\tilde{m} \in M_R$.

Susiduriama su perblokavimo problema, kuri atsiranda todėl, kad skaitmeninis parašas *s* yra didesnis už modulį n_B.

Perblokavimo problemos sprendimai

Pergrupavimas. RSA skaitmeninio parašo algoritmas bus pakankamai efektyvus, jei skaitmeninis parašas generuojamas bus pradžioje pasirenkant mažesnį modulį, t.y. jeigu $n_A > n_B$, tada objektas *A* pradžioje turi šifruoti pranešimą naudodamas objekto *B* viešąjį raktą, o paskui pasirašyti turimą pranešimą naudodamas privatųjį raktą. Tačiau čia atsiranda algoritmo atsparumo trūkumas, nes kai objektas *A* pirma šifruoja pranešimą, o tik paskui jį pasirašo, priešiškas objektas gali pašalinti parašą ir pakeisti jį savo parašu. Nors priešiškas objektas ir nežinos, kas yra užšifruota, tačiau toks algoritmas jau bus neatsparus. Pergrupavimas nėra pakankamai efektyvus problemos sprendimas.

Dviejų modulių panaudojimas

Kiekvienas objektas yra sugeneravęs atskirus šifravimo ir pasirašymo modulius. Jei kiekvieno objekto skaitmeninio parašo modulis yra mažesnis už galimus šifravimo modulius, niekada neiškils perblokavimo problema. Tai galima pasiekti nustatant, kad šifravimo modulis būtų $(t+1)$ bitų, o skaitmeninio parašo modulis būtų t bitų.

Modulio formos nustatymas

Objektas pasirenka tokius pirminius skaičius *p* ir *q*, kad modulis n turi specialią formą. Didžiausios eilės bitas yra vienetas, o *k* eilės bitai yra nuliai. Objekto *A* *t* bitų modulis *n* tokios formos gali būti randamas apskaičiuojant $2^{t-1} \le n < 2^{t-1}+2^{t-k-1}$. Pasirenkamas $\lceil t/2 \rceil$ bitų ilgio pirminis skaičius *p*, ieškomas pirminis skaičius *q* intervale tarp $\lceil 2^{t-1}/p \rceil$ ir $\lfloor (2^{t-1}+2^{t-k-1})/p \rfloor$. Pasirenkant šį metodą perblokavimo problemos visiškai neišvengiama, tačiau jos tikimybė sumažinama iki nykstamai mažo dydžio.

Tarkim, n_A yra modulis pranešimui *m* $s=m^{dA} \bmod n_A$. Skaitmeninis parašas *s* turi vienetą vienoje iš $k+1$ bitų pozicijų, išskyrus aukčiausią, taigi *s* yra mažesnis už n_A ir turi nulį aukščiausioje bitų pozicijoje, taigi yra mažesnis už bet kurį modulį tokios pačios bitų eilės. Tikimybė, kad s neturės nė vieno vieneto pozicijose $k+1$, išskyrus pirmąją, yra mažesnė nei $(1/2)^k$, kuri yra nykstamai maža, pasirenkant $k>100$. Sukuriame $t=14$ bitų ilgio modulį tokį, kad aukščiausias bitas būtų vienetas, o kiti $k=3$ bitai nuliai. Pasirenkame 7 bitų pirminį skaičių $p=107$. Turime pasirinkti pirminį skaičių *q* intervale tarp $\lceil 2^{t-1}/p \rceil = \lceil 2^{13}/107 \rceil = 77$ ir $\lfloor (2^{t-1}+2^{t-k-1})/p \rfloor = \lfloor (2^{13}+2^9)/107 \rfloor = 82$. Pasirenkame $q=79$; tada $n=10000100000101$ [4].

Naudojant RSA skaitmeninio parašo algoritmą "voko" principu dėl skirtingo modulių dydžio gali kilti perblokavimo problema, šią problemą galima spręsti įvairiais sprendimo būdais vienas iš jų yra pergrupavimas, kuris nėra pakankamai efektyvus dėl galimybės priešiškam objektui pakeisti skaitmeninį parašą. Kitas sprendimo būdas yra dviejų modulių šifravimui ir pasirašymui panaudojimas, kur objektai susitaria dėl modulių dydžių. Pats efektyviausias problemos sprendimas yra modulio formos nustatymas, nors panaudojus šį metodą atsiranda tikimybė dėl perblokavimo, tačiau ji yra nykstamai maža, o metodas lengviausiai praktiškai realizuojamas.

Literatūra

1. **Burnett Steve, Paine Stephen.** The RSA Security's Official Guide to Cryptography. McGraw-Hill Professional, 2001. - P. 70 –81.
2. **Jalote Pankaj, Mel H.X., Baker doris M..** Cryptography Decrypted. Addison Wesley Longman, Inc., 2000 - P. 89 – 100.
3. **Menezes A. J., van Oorschot P. C., and Vanstone S. A.** Handbook of Applied Cryptography, CRC Press, Boca Raton, 1996 – P. 308-319.
4. **Pomerance C.** Cryptography and Computational Number. - Theory American Mathematical Society, Providence. - Vol. 421990. - P. 128-138.

Pateikta spaudai 2003 02 28

A. Vobolis, P. Nefas. Skaitmeninio „voko“ panaudojimo RSA skaitmeninio parašo algoritme tyrimas // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2003. – Nr. 4(46). – P. 57-60.

Skaitmeniniai „vokai“ buvo išrasti siekiant pasinaudoti viešojo ir slaptojo rakto kriptografijų privalumais. Praktikoje šifravimui naudojami viešojo rakto šifravimo algoritmai, garantuojantys didesnį saugumą kartu su slaptojo rakto šifravimo algoritmais, garantuojančiais didesnę šifravimo spartą. Skaitmeninio „voko“ principas gali būti panaudotas RSA skaitmeninio parašo algoritme. Objektas *A* pasirašo pranešimą ir užšifruoja jį objekto *B* viešu raktu. Objektas *B,* gavęs pranešimą, iššifruoja jį savo privačiuoju raktu ir patikrina skaitmeninio parašo autentiškumą. Naudojant RSA skaitmeninio parašo algoritmą „voko” principu, dėl skirtingo modulių dydžio gali kilti perblokavimo problema, galimi šios problemos sprendimo būdai yra pergrupavimas, dviejų modulių panaudojimas, modulio formos nustatymas. Bibl.5 (lietuvių kalba; santraukos lietuvių, anglų, rusų k.).

A. Vobolis, P. Nefas. Investigation of RSA Digital Signature Scheme Working in Digital Envelope Algorithm // Electronics and Electrical Engineering. – Kaunas: Technologija, 2003. – No. 4(46). – P. 57-60.

Digital envelopes were invented to obtain the advantage of secret and public key cryptographies. In practice public-key algorithms which increase security are used together with secret key algorithms which increase encryption speed in such called “digital envelope” principle. The algorithm of digital envelope successfully can be used in RSA digital signature scheme. Entity A signs the message and encrypts it with entities B public key. Entity B receives the message and decrypts it with its private key and verifies authenticity of digital signature. Using RSA digital signature scheme working in digital envelope algorithm in case of different modules sizes re-blocking problem can occur. To solve re-blocking problem we can use such solutions as reordering, using two modules or prescribing the modules form. Bibl.5 (in Lithuanian, summaries in Lithuanian, English, Russian).

А. Воболис, П. Нефас. Исследование RSA алгоритма шифрования с точки цифрового „конверта” // Электроника и электротехника. – Каунас: Технология 2003. – № 4(46). - С. 57-60.

Цифровые “конверты” были изобретены для того, чтобы использовать преимущества криптографий открытых и скрытых ключей. Обычно в практике для шифрования используются алгоритмы шифрования открытых ключей, которые обеспечивают наибольшую сохранность информаций в месте с алгоритмами шифрования скрытых ключей, которые обеспечивают скорость шифрования. Все цифровые принципы “конверта” могут полностью быть использованы в RSA цифровой подписи.
Объект А подписывает сообщения и шифрирует эго открытым ключом объекта В. Когда объект В получает сообщение, он дешифрирует своим собственным ключом и проверяет сообщения на аутентификацию. Так используя RSA алгоритм шифрования может возникнуть проблема блокировки. Возможны несколько вариантов разрешения этой проблемы: перегруппирование, использование двух модулей и определение формы модуля. Библ.5 (на литовском языке; рефераты на литовском, английском и русском яз.).