韓國 OA學會 論文誌
第4卷 第2號, 1999. 6.

99-4-2-1-5

안전성에 근거를 둔 디지털서명 성능분석에 관한 연구

이 지 영*

A Study on Performance Analysis of Digital Signature Based on the Security

Jie-Young Lee*

요 약

본 논문은 디지털서명을 위한 암호화 기법의 분석과 이산대수 문제와 소인수 분해와 같이 계산 복잡도의 어려움에 안전성의 근거를 둔 암호방식을 비교한다. 특히 계산량에 의한 성분분석과 데이터 크기 비교 및 처리속도를 시뮬레이션에 의해 비교, 검토하였다.

Abstract

In this paper we will look at its cryptographic analysis for digital signature and compare it with other complexity measures such as discrete logarithm problem and factorization problem which are based on the security.

The paper especially tries to computational complexity so that it can compare and checks the performance analysis, comparison of data size and processing speed through the simulation me

* 세명대학교 컴퓨터과학과 부교수
논문접수 : 1999. 5.20. 심사완료 : 1999. 6.19.

Ⅰ. 서론

컴퓨터의 발전으로 정보화시대가 급속히 도래함에 따라서 컴퓨터 통신이 매우 빠르게 발전하고 있다. 일반적으로 컴퓨터 네트워크는 통신이 비보호 채널을 통하여 이루어지기 때문에 전송 도중에 제3자로부터 정보가 변경되지 않았음을 확인(Validation) 하는 것과 또한 수신자는 그 정보를 정당한 송신자가 송신했다는 것을 증명할 수 있는 인증(Authentication)이 필요하다.

디지털 서명은 이 인증과 확인의 기능을 동시에 할 수 있도록 해 주는 것으로써 송신되는 메시지에 덧붙여지던지 혹은 메시지의 일부분으로 포함되어 메시지와 같이 수신측으로 전송된다.

분 논문에서는 DSS 디지털 서명방식[1]의 알고리즘과 Schnorr 디지털 서명방식[2][4]과 ESIGN 디지털 서명방식[3]을 이론적인 계산량에 의한 성능분석과 시뮬레이션에 의한 성능평가를 한다.

Ⅱ. 디지털 서명

공개키 암호방식의 개념에서 출발한 디지털 서명은 전자송금, 전자문서, 컴퓨터 프로그램 등에 대한 서명 알고리즘으로 계산된 2진 비트열이다.

공개키 암호방식은 정보보호기능을 가지고 있으면 비밀키와 공개키를 만들어내는 과정, 비밀키를 사용하는 과정, 또한 공개키를 사용하는 과정을 수행한다.

디지털 서명은 통신간의 발신처 확인과 데이터 내용의 변경이 되지 않았는지를 알 수 있는 무결성보호 및 서명자를 인증하는 기능 등을 기지고 있다.

디지털 서명은 안전성 유지를 위해 큰 수의 곱셈이나 누승 등의 연산을 요구한다. 만약 메시지의 양이 클 경우에 디지털 서명으로 인한 계산량이 급격히 많아져 overhead가 커지는 문제점을 가지고 있다. 이러한 문제점 때문에 해쉬함수를 이용하여 원래의 메시지를 압축한 뒤, 압축된 메시지에 대한 서명만을 계산, 전송하는 방식을 많이 사용하고 있다.

1991년 8월 미국의 NIST (National Institute of Standard and Technology)는 표준 디지털 서명(Digital Signature Standard, DSS)안을 제안하였다.[1]

Ⅲ. 디지털 서명 방식 비교

1. DSS 디지털 서명 방식

1991년 8월 미국의 NIST는 DSA(Digital Signature Algorithm)를 발표한 후, 이를 미국 내 디지털 서명 표준안으로 제안하였는데 이 방식이 DSS(Digital Signature Standard)이다. 이 DSS는 Schnorr 방식의 변형된 형태로 제안되었으며 이 방식의 안전성은 이산대수 문제의 어려움에 근거한다.

이 DSS 방식의 알고리즘을 사전준비단계, 서명생성단계, 서명검증단계로 나누면 다음과 같다.

사전 준비 과정 ;

① 소수 법(prime modulus) p, $2511 < p < 2512$

② 소수 법 q, $q \mid p-1$ 이고 $2159 < q < 2160$

③ $g = h(p-1)/q \bmod p$ $(g > 1,\ 1 < h < p-1)$ 즉, g의 위수가 q이다.

④ 해쉬함수 H, 출력은 160 비트이다.

⑤ 공개키 $Y = gx \bmod p$ 이고, $0 < x < q$ 인 난수 x는 비밀키이다.

서명 생성 과정 ;

① 랜덤변수 k, $0 < k < q$를 발생한다.

② R = (gk mod p) mod q를 계산한다.
③ H(M)를 구한다. (M은 서명할 메시지)
④ k-1k mod q = 1인 k-1을 계산한다.
⑤ S = k-1(H(M) + xR) mod q를 구한다.
⑥ 서명 (R, S)를 M에 추가한 (M, R, S)를 수신자에게 보낸다.

서명 검증 과정;
① W= S-1 mod q를 계산한다.
② H(M)를 구한다.
③ A = H(M)W mod q를 계산한다.
④ B = RW mod q를 계산한다.
⑤ V = (gAYB mod p) mod q를 계산한다.
⑥ V = R를 확인, 만족되면 서명이 적합하다.
-⑥의 증명 ;
V = (gAYB mod p) mod q
= (gH(M)WgxRW mod p) mod q
= (gW(H(M)+xE)) mod q
= (gk mod p) mod q
= R

서명 생성 과정시 k를 알게 되면 x를 쉽게 구할 수 있으므로 k의 비밀도 잘 유지되어야 한다.

2. Schnorr 디지털 서명 방식

이산대수 문제 해결의 어려움을 안전성의 근거로 삼고 있는 방식은 스마트 카드 구현에 적절한 개인 식별 방식과 함께 디지털 서명 방식을 1989년 Schnorr가 제안하였다.[4]

이 Schnorr 디지털 서명 방식을 3단계로 나누면 다음과 같다.

사전 준비 과정 ;
① 소수 법 p와 q
(q | p-1, q ≧ 2140 , p ≧ 2512)
② 위수가 q인 α즉,
αq ≡ 1(mod p)
(1 〈 α 〈 p-1)
③ 72 비트 출력을 가진 해쉬 함수 h
④ 비밀키 x : 0 〈 x 〈 q
⑤ 공개키 y : y = α-x mod p

서명 생성 과정 ;
① 0 〈 k 〈q인 랜덤변수 k를 생성한다.
② r ≡ αk mod p
③ e = h(r, M)
④ s ≡ k + xe (mod q)
⑤ (M, e, s)를 수신자에게 전송한다.

서명 검증 과정 ;
① v ≡ αsYe mod p
② e = h(v, M) 이면, 서명이 유효하다.

위의 서명 생성 과정 ④에서 k를 알면 곧 x를 계산할 수 있어 서명이 도용될 수 있다. 또한 동일한 r을 두 번 사용하는 것을 알면 DSS나 ElGamal 서명 방식과 같이 비밀키 x를 계산할 수 있다. 서명 생성 과정

중 ①, ②는 서명할 메시지에 독립적이어서 전 처리가 가능하다.

A		B
공개키 : Y = g^{X_A} mod p 비밀키 : XA, 0 〈 XA 〈 q	p, q, H, g 공개키[YA] [YB]	
① 랜덤변수 k, 0 〈 k 〈 q ② R = (gk mod p) mod q ③ H(M) ④ k-1k mod q = 1인 k-1 ⑤ S=k-1(h(M)+xR) mod q	⑥(M,R,S) ——〉	
		① W = S-1 mod q ② H(M) ③ A = H(M)W mod q ④ B = RW mod q ⑤ V = (gAYB mod p) mod q ⑥ V = R 확인, 만족되면 서명 이 유효

그림 3.1. DSS 디지털 서명 방식

3. ESIGN 디지털 서명 방식

1991년 후지오까, 오까모도, 미야구찌는 소인수분

A		B
① xA: 0 < xA <q, 비밀키 ② yA: yA ≡ α^{-X_A} mod p	p, q, α, h 공개키 (yA) (yB)	
① k : 0 < k < q ② r ≡ αk mod p ③ e = h(r, M) ④ s ≡ k + xe (mod q)	⑤(M, e, s) ——————>	
		① v ≡ αsYAe mod p ② e = h(y, M) 이면 서명이 유효

그림 3.2. Schnorr 디지털 서명 방식

A		B
① 비밀키 : 소수 PA, QA ② 공개키 : NA = PA2QA	해쉬함수:H(M) ∈ZN 매개변수K:정수(K≧4) 공개키 : (NA) (NB)	
① X: 0≦X≦PQ-1 ② 서명 S를 계산 $W = \lceil \frac{H(M)-(X^k \bmod N)}{PQ} \rceil$ $Y = \frac{W}{(KX^{k-1})} \bmod P$ $S = X + YP_AQ_A$	③ (M, S) ——————>	
		① H(M) ≦ SK mod NA $< H(M) + 2^{\lceil 2\lceil N_A\rceil /3\rceil}$ 이면 서명 유효

그림 3.3. ESIGN 디지털 서명 방식

해 문제의 어려움에 안전성의 근거를 두고 있는 ESIGN 디지털 서명 방식을 제안하고 구현하였다.[3]

이 방식을 3단계로 나누어보면 다음과 같다.

표기 ;

⌈M⌉ : M 보다 크거나 같은 최소 의 정수

Zn : 0에서 n-1까지의 정수의집합

M : 메시지

사전 준비 과정 ;

① 비밀키 : 큰 소수 P, Q

② 공개키 : N = P2Q

③ 해쉬함수 : 어떤 양의 정수 M에 대해 H(M)∈Zn인 일방향 해쉬함수

④ 매개변수 K : 정수 (K≧4)

해쉬함수 H와 K는 시스템에 고정

서명 생성 과정 ;

① X를 랜덤하게 생성한다.

(0 ≦ X ≦ PQ-1)

② 서명 S를 계산한다.

$$W = \lceil \frac{H(M)-(X^k \bmod N)}{PQ} \rceil$$

$$Y = \frac{W}{(KX^{k-1})} \bmod P$$

$$S = X + YPQ$$

③ (M, S)를 수신자에게 전송한다.

서명 검증 과정 ;

① H(M) ≦ SK mod N

< H(M) + 2⌈ 2| N| /3⌉

표 4.1. 오까모도의 수행시간 비교
(8비트, CPU, 256 RAM, 8K ROM에서의 구현시간)

방식 \ 시간	서명 생성 시간 (sec)	서명 검증 시간 (sec)	서명 전체 시간 (sec)
DSS	17	28	45
Schnorr	15	16	31
ESIGN	0.4	0.3	0.7

표 4.2. 곱셈 연산수 비교

방식 \ 연산수	서명시 연산수	검증시 연산수	전체 연산수
DSS	241	281	522
Schnorr	211	228	439
ESIGN	11	5	16

(M, S)가 위의 부등식을 만족하면 서명은 유효하다.

K가 2인 경우 깨졌으나, K가 3이상인 경우는 깨어지지 않고 있으며, 이것을 공격하는 것은 N을 소인수 분해하는 문제만큼 어렵다고 추측된다. 또한 스마트 카드에 구현을 위해 매개변수 K의 값을 계산이 용이한 2의 멱승(즉 4, 8, 16 등) 형태를 선택했다.

Ⅳ. 성능 분석 및 평가

오까모도의 수행시간을 비교한 후 DSS 디지털 서명 방식의 알고리즘과 Schnorr의 디지털 서명 방식의 ESIGN 디지털 서명 방식을 곱셈 연산수 비교, 계산량 비교, 사전 계산에 의한 서명 생성 비교, 데이터 크기 비교, 사전 처리 없는 처리속도 비교, 사전 처리에 의한 처리 속도의 비교는 다음과 같다.

표 4.3. 계산량 비교

방 식	서명 생성	서명 검증
DSA	240M(512)+I(160) < 241M(512)	280M(512)+I(160) < 281M(512)
Schnorr	210M(512)	242M(512)
ESIGN	2M(576)+ I(192) < 5M(512)	3M(576) < 4M(512)

표 4.4. 사전 계산에 의한 서명 생성 비교

방식	사전계산	서명생성
DSA	241M(512)	almost 0
Schnorr	210M(512)	almost 0
ESIGN	5M(512)	almost 0

표 4.5. 데이터 크기 비교

방식	공개키 (bits)	비밀키 (bits)	서명크기 (bits)
DSA	1696	160	320
Schnorr	1676	140	268
ESIGN	587	384	576

표 4.6. 사전 처리 없는 처리 속도 비교(8bit CPU)

방 식	서명 생성 (sec)	서명 검증 (sec)
DSA	17	20
Schnorr	15	17
ESIGN	0.4	0.3

표 4.7. 사전 처리에 의한 처리속도 비교(8bit CPU)

방식	사전 계산 (sec)	서명 생성 (sec)	서명 검증 (sec)
DSA	17	0**	20
Schnorr	15	0**	17
ESIGN	0.4	0**	0.3

** 50 msec 이하

표 4.8. 처리 속도와 데이터 크기에 대한 총 비교

방식	안전성 증명가능?	사전계산(sec)	서명생성(sec)	서명검증(sec)	공개키 (bit)	비밀키 (bit)	서명 (bit)
DSA	No	17	0**	20	1696	160	320
Schnorr	No	15	0**	17	1676	140	268
ESIGN	No	0.4	0**	0.3	558	383	576

** 50 msec 이하

표 4.3과 표 4.4는 계산량에 의한 성능 분석을 한 것이며 표4.6과 표 4.7은 시뮬페이션에 의한 평가이며, 표 4.8은 처리속도와 데이터 크기에 대한 총 비교를 나타낸다.

이상에서 알 수 있듯이 DSS 방식은 이산대수 문제의 해결의 어려움에 안전성의 근거를 두고 있으며 Schnorr 방식 또한 같은 문제의 어려움에 안전성의 근거를 두고 있으나 ESIGN 방식은 RSA방법과 같이 소인수 분해의 문제 해결의 어려움에 안전성의 근거를 두고 있다.

또한 512비트 이상의 큰 수를 멱승하기 위하여 다정도 연산을 이용하여 소프트웨어를 구현하였다.

DSS 방식을 검토해 보면 160 비트의 q는 안전도면에서 불충분하며, 개인의 비밀키가 K에 의존하는 점과 확인 권한자가 고의적으로 안전도가 약한 P를 선택할 수 있다는 점이 있고 또한 서명 검증 과정의 비효율성도 단점이다.

Schnorr 방식은 서명 생성 과정 중 ①, ②는 서명할 메시지에 독립적이어서 전처리가 가능하다. 즉 프로세서의 idle-time에 서명 생성 시간의 대부분을 차지하는 모듈러 승산의 전 처리가 가능하여 서명 생성 시간을 줄일 수 있다.

또한 ESIGN 서명 생성 과정은 서명할 메시지에 독립적인 계산이 포함되어 있다. 그러므로 서명할 메시지를 받기 전에 과정들을 수행할 수 있다.

V. 결론

본 논문에서는 NIST가 고려한 디지털 서명 표준화의 사항을 중심으로 디지털 서명 알고리즘을 이산 대수 문제 해결의 어려움에 안전성의 근거를 두고 있는 방법과 소인수 분해 문제의 어려움에 안전성의 근거를 두고 있는 DSS 방식과 Schnorr 및 ESIGN 방식등을 계산량에 의한 성능분석과 데이터 크기 비교 및 처리속도를 시뮬레이션에 의해 비교, 검토하였다.

ESIGN 서명 생성 과정은 서명할 메시지에 독립적인 계산이 포함되어 있어며 Scnorr 방식은 서명 생성 과정 중 서명할 메시지에 독립적이어서 전 처리가 가능하지만 DSS 방식은 서명 검증 과정에서 비효율적이어서 향후 이의 보완이 필요할 것이다.

References

[1] Specification for a digital Signature standard. NIST. FIPS XX, Draft, August 1991.

〔2〕 Schnorr, "Method for identifying subscribers and for generating and verifying electronic signatures in a data exchange system", US patent 4, 995, 082, Feb. 1991.
〔3〕 A. Fujioka, T. Okamoto, and S. Miyaguchi, "ESIGN : An Efficient Digital Signature Implementation for Smart Cards," Proceedings of EUROCRYPT'91, LNCS 547, Springer-Verlag, pp.446-457, 1991.
〔4〕 C.P.Schnorr, "Efficient Identification and Signatures for Smart Card", Crypto'89, pp.239-252, 1989.
〔5〕 T.ElGamal, "A Public key Cryptosystem and a Signature Scheme Based on Discrete Logarithm", IEEE Tran. Inform. Theory 31, pp.469-472, 1985.
〔6〕 한국 전자 통신 연구소, "현대 암호학", 1991.
〔7〕 D.Knuth, The Art of Computer Programming,Vol2:Seminumerical Algorithms, Addison-Wesley, Reading, Mass, 2nd, 1981.

저자소개

이지영
한국 OA학회 논문지 제3권 제2호 참조
현재 세명대학교 컴퓨터 과학과 부교수