Signcryption을 이용한 부정자 추적 프로토콜

신성한*, 박지환*, 허영**
부경대학교 전자계산학과* 한국전기연구소 영상응용그룹**

Traitor Traceable Protocols using Modified Signcryption

Seong-Han Shin*, Ji-Hwan Park*, Young Huh**
Dept. of Computer Science, PuKyong National University*
Applied Imaging Research Group, Korea Electrotechnology Research Institute**

요 약

본 논문에서는 Y. Watanabe et al.에 의해 제안된 서명부가문서를 이용한 부정자 추적기법을 분석하여 PKI(Public Key Infrastructure)기반에서 구현 가능하도록 프로토콜을 재구성하고, 그 안전성을 분석한다. 그리고 변형된 Signcryption을 이용한 프로토콜을 제안하여 기존의 유사 프로토콜과 비교·분석하여 제안된 프로토콜이 더욱 효율적임을 보인다.

1. 서론

최근 저작권 보호의 문제를 해결하기 위해서 디지털 워터마킹이나 핑거프린팅 그리고 부정자 추적기법들이 연구되고 있다. 여기에서는 전자 상거래의 필수 요소인 컨텐츠 배포 시스템에 부정자 추적기법을 적용하고자 한다[1].

우선 컨텐츠 배포시스템이 갖추어야 하는 요구사항을 대략 정리하면 다음과 같다.

(1) 수신자의 비밀정보가 필요로 할 경우 컨텐츠 제공자는 수신자의 도움 없이도 재 배포된 서명부가문서를 통해 세션키를 도출할 수 있어야 한다.

(2) 컨텐츠 제공자는 컨텐츠를 부정 유출한 부정자를 특정하여 단독으로 제3자에게 증명할 수 있어야 한다.

(3) 컨텐츠 제공자가 고의로 분쟁을 일으켜 수신자의 키(거래 당시의 키)를 도출하였더라도 수신자의 키를 재 사용할 수 없어야 한다.

(4) 중재자의 역할은 단지 분쟁 조정만 한다.

(5) 능동적인 공격자에 대해 안전해야 한다.

(6) 부정이 발생하였을 시에는 제3자도 검증 가능하여야 한다.

이상과 같은 요구사항에 대하여 WZI방식[1]에서는 (1)~(4)까지의 조건을 만족하면서 서명부가문서의 특성을 이용하여 부정자를 추적하고 있다. 그러나, 마지막 2가지 조건과 프로토콜 자체에서 다음과 같은 문제점을 가지게 된다.

(1) 컨텐츠 제공자가 컨텐츠를 불법 유통시킨 수신자(부정자)를 고발하는 과정에서 수신자의 서명부가문서로부터 세션키를 구하여 거래 당시의 키를 도출하게 된다. 그러나, 컨텐츠 제공자만이 검출할 수 있는 것이 아니라, 제3자(adaptive attacker등)에 의해 모든 프로토콜 과정이 도청된 경우에도 키는 도출된다. 따라서 컨텐츠 제공자가 자신이 서명자라는 것을 제3자에게 확신시킬 수 없게 된다. 또한 세션키와 거래 당시의 키는 수신자와의 연관성을 가지지 못하므로 부정자를 특정하는 것은 단지 서명의 검증에 지나지 않는다.

(2) 프로토콜의 마지막 과정에서 컨텐츠 제공자는 수신자가 원하는 컨텐츠를 제공하게 되는데, 이 때 악의의 컨텐츠 제공자가 임의의 다른 컨텐츠를 제공한다면 수신자는 제3자(여기에서는 중재자)에게 자신의 세션키를 알리지 않고 컨텐츠 제공자에 대한 부정을 증명하기 위해 ZKIP(Zero-Knowledge Interactive Proofs)을 수행해야 한다[2]. 하지만, WZI방식에서 사용한 Signcryption은 부인봉쇄기능를 수행하는 과정에서 기밀성이 유지되지 못하므로 이에 대한 보완이 요구된다[3].

(3) 프로토콜 자체의 문제점으로 수신자가 컨텐츠 제공자에게 보내는 서명부가문서에 Signcryption에서 사용하게 될 공유키가 포함되어 있다. 만약, 공격자가 모든 프로토콜을 도청(eavesdropping)하고 있는 경우에 어떤 공격자도 컨텐츠 제공자와 마찬가지로 공유키로 모든 컨텐츠를 복호하고 검증할 수 있다. 즉, 수신자의 세션키 없이도 Unsigncryption 과정을 수행할 수 있다.

본 논문에서는 첫 번째 문제점을 해결하기 위하여 거래 당시의 키와 수신자의 정보를 서로 연관시킴으로써 수신자 자신이 컨텐츠가 불법 유출되는 것을 막도록 하는 Self-enforcement 기능을 가지게 된다. 또한 세션키와 수신자의 ID(certificate의 ID)가 서로 연관되므로 컨텐츠 제공자만이 아니라 제3자에 의해서도 부정을 저지른 수신자를 추적할 수 있게 된다. 두 번째 문제점은 변형된 Signcryption[4]을 이용하여 Non-repudiation 기능을 제공하고자 한다. 즉, Direct Verifiable Signcryption을 사용함으로써 제3자에게 부정을 증명할 경우 상호 영지식 증명과정을 거치지 않고 바로 증명할 수 있다. 마지막 문제점을 해결하기 위하여 변형된 Signcryption을 이용한 AKA (Authenticated Key Agreement) 프로토콜을 수행한다. 제안된 프로토콜에서는 Signcryption에서 사용될 공유키를 직접 전송하지 않고 암시적 키 합의(Implicit Key Agreement)를 사용한다.

본 논문의 구성은 다음과 같다. 2장에서는 제안된 프로토콜에 사용될 Parameter와 구성요소를 정의하고, 3장에서는 기존의 프로토콜보다 동등 이상의 성능을 가지는 2가지 프로토콜을 제안한다. 4장에서는 제안된 프로토콜의 안전성과 효율성을 기존의 프로토콜과 비교·분석하고 마지막 장에서는 결론 및 향후과제를 제시한다.

2. Preliminary

2.1 Parameter

p와 q는 큰 소수로서 $q \mid p-1$인 동시에 $|q| = 160$ 비트이고, g는 Z_p^* 상에서 위수가 q인 원시근으로 한다. $H(\cdot)$는 임의의 길이의 메시지를 Z_q^*에 사상하는 이상적인 일방향성 해쉬함수를 나타내고 $KH_K(m)$는 K를 키로 하는 Keyed Hash Function이다. 또, 서명자를 Alice, 서명부가문서의 수신자를 Bob, 중재자를 Chris라 한다. Alice의 비밀키를 $x_a \in Z_q^*$, 대응하는 공개키를 y_a라 하고 마찬가지로 Bob의 비밀키, 공개키를 각각 x_b, y_b 로 나타낸다. 또한, 여기에서 사용되는 세션키 x, y는 OTP(One Time Pad)와 같은 역할을 한다. Alice, Bob의 일반적인 검증 가능한 서명방식을 각각 $Sig_A(\cdot)$, $Sig_B(\cdot)$로 표기한다. 그리고 CA로부터 발급받은 인증서를 $Cert_A$, $Cert_B$로 한다.

2.2 구성요소

이 논문에서는 Schnorr Signature Scheme를 이용하여 이산대수의 지식을 증명한다[5]. 또한, Off-line상에서 Digital Coin이 이중 사용된 경우에 피발행자의 익명성을 박탈하는 수단으로써 이산대수를 이용한 방법을 사용한다[8,9]. 마지막으로 컨텐츠를 제공하는 과정에서는 변형된 Signcryption[4]을 사용한다.

3. 프로토콜

3.1 제안 프로토콜I

3.1.1 Step 1

Alice는 Signcryption에서 사용하게 될 세션키 x를 생성하고 $X \equiv g^x \bmod p$를 계산한다. X에 대한 서명부가문서를 인증서와 함께 Bob에게 전송한다.

$$x \in_R Z_q^* \qquad X \equiv g^x \bmod p \tag{1}$$

3.1.2 Step 2

Bob은 Alice의 인증서 유효기간을 확인하고 $Sig_A(X)$를 검증한다. Bob은 이 프로토콜에 사용하게 될 세션키 y를 선택하고, $Y \equiv g^y \bmod p$를 계산한다. 여기에서 사용되는 y'는 사용자의 ID정보를 포함한다. U와 v은 Bob이 Y의 이산대수 $Y = \log_g y$ 를 알고 있다는 것의 증명이다.

$$y, u \in_R Z_q^* \qquad y' = H(y \| ID_B)$$

$$Y \equiv g^y \bmod p \tag{2}$$

$$U \equiv g^u \bmod p \quad v \equiv u - y\,H(U) \bmod q \tag{3}$$

$$res = (Y,\ y',\ U,\ v) \tag{4}$$

3.1.3 Step 3

Alice는 마찬가지로 Bob의 인증서 유효기간을 확인하고 서명을 검증한 후, 아래의 검사식을 Check한다.

- Bob이 Y의 이산대수 지식 y의 증명

$$U \stackrel{?}{=} g^{v} Y^{H(U)} \bmod p \quad (5)$$

검사를 통과한 후에, Alice는 아래와 같이 컨텐츠 m에 관해 변형된 Signcryption과정[4]을 거친다.

$$K = H(Y^{x} \bmod p) \quad (6)$$
$$c = E_K(m) \quad (7)$$
$$r = KH_X(m) \quad (8)$$
$$s \equiv x/(r+x_a) \bmod q \quad (9)$$

Alice는 Bob에 대해서 (c, r, s)를 전송한다.

Bob은 (c, r, s, y)를 Alice의 m에 대한 서명부가문서로 한다.

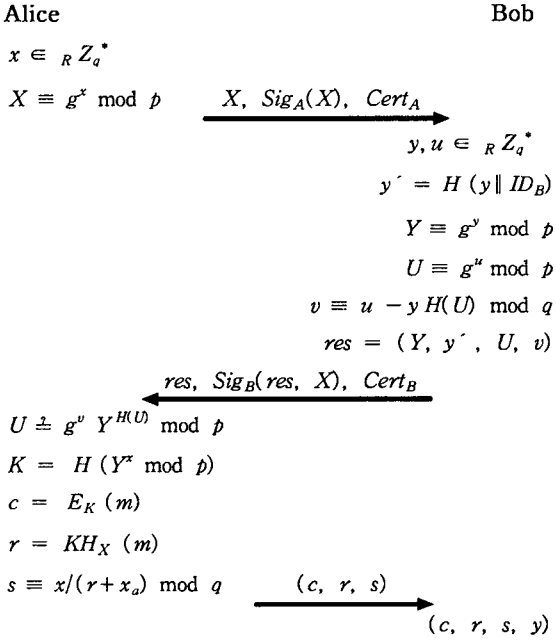

그림1. 제안 프로토콜I

3.1.4 서명검증

Bob은 서명부가문서 (c, r, s, y)를 검증하고 컨텐츠를 복호한다. 식 (13)이 성립하면 Alice가 보내온 서명은 정당하다.

$$X \stackrel{?}{=} (y_a\, g^{r})^{s} \bmod p \quad (10)$$
$$K = H(X^{y} \bmod p) \quad (11)$$
$$m = D_K(c) \quad (12)$$
$$r \stackrel{?}{=} KH_X(m) \quad (13)$$

3.2 제안 프로토콜II

여기에서는 AKA 프로토콜를 적용하여 동일한 성능을 가지는 프로토콜을 제안한다.

3.2.1 Step 1

Alice의 서명생성은 3.1.1과 동일하다.

3.2.2 Step 2

Bob은 Alice의 인증서 유효기간을 확인하고 서명을 검증한다. 검증 후, Bob은 암시적 키 합의를 위해 AKA 프로토콜을 수행한다. 서명의 검증에 사용될 키 k를 구하고, 서명과 인증서 (r_B, s_B), $Cert_B$를 Alice에게 보낸다.

$$y \in_R Z_q^* \qquad Y \equiv g^{y} \bmod p$$
$$k = H(y_a{}^{y} \bmod p) \quad (14)$$
$$r_B = KH_Y(k, X) \quad (15)$$
$$s_B \equiv y/(r_B+x_b) \bmod q \quad (16)$$

3.2.3 Step 3

Alice는 Bob의 인증서 유효기간을 확인하고 (r_B, s_B)를 가지고 Unsigncryption 과정을 거친다. 식 (19)가 성립하면 Bob이 보내온 서명은 정당하다.

$$Y \equiv (y_b\, g^{r_B})^{s_B} \bmod p \quad (17)$$
$$k = H(Y^{x_a} \bmod p) \quad (18)$$
$$r_B \stackrel{?}{=} KH_Y(k, X) \quad (19)$$

모든 검사를 통과한 후에, Alice는 아래와 같이 키 K를 생성하고, Signcryption 과정을 거친다.

$$K = H(Y^{x} \bmod p) \quad (20)$$
$$c_A = E_K(m) \quad (21)$$
$$r_A = KH_X(m) \quad (22)$$
$$s_A \equiv x/(r_A+x_a) \bmod q \quad (23)$$

Alice는 Bob에게 (c_A, r_A, s_A)를 전송한다.

Bob은 (c_A, r_A, s_A, y)를 Alice의 m에 대한 서명부가문서로 한다.

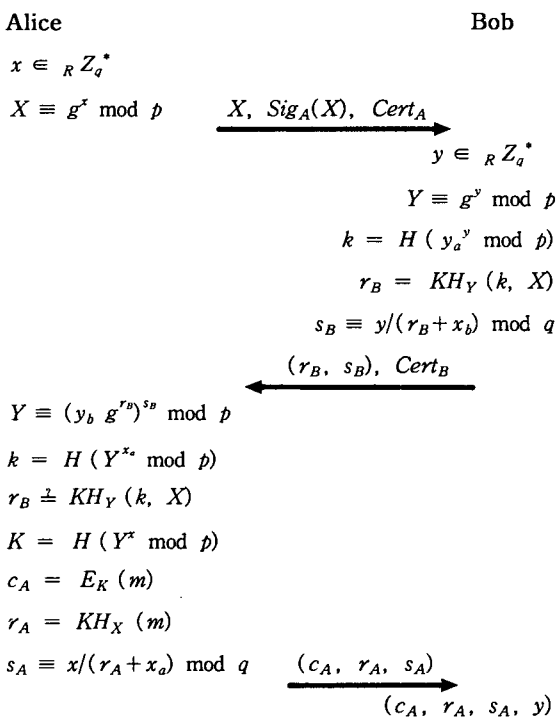

그림3. 제안 프로토콜II

3.2.4 서명검증

Bob은 Alice가 보내온 $(c_A,\ r_A,\ s_A,\ y)$를 검증하고 식 (26)로부터 컨텐츠 m을 복호한다. 마지막 식이 성립하면 Alice가 보내온 서명은 정당하다.

$$X \stackrel{?}{=} (y_a\, g^{r_A})^{s_A} \bmod p \quad (24)$$

$$K = H(X^y \bmod p) \quad (25)$$

$$m = D_K(c_A) \quad (26)$$

$$r_A \stackrel{?}{=} KH_X(m) \quad (27)$$

4. 안전성 분석과 비교

4.1 프로토콜I의 안전성

4.1.1 프로토콜 자체의 안전성

Step1에서 Bob에게 전달되는 정보는 이산대수문제의 안전성에, Step2에서의 정보도 이산대수문제, 해쉬함수의 안전성에 근간을 두고 있다. 특히 Schnorr Identification Scheme을 Signature 형태로 변경하여 사용할 때, Challenge 값에 해쉬를 취함으로써 Random Oracle Model에서 영지식이 가능하다는 증명이 나와 있다[8]. Step3에서도 기존의 Signcryption이 가지는 취약점 즉, Non-repudiation을 보완한 Direct Verifiable Signcryption[4]을 사용함으로써 Unforgeability와 Confidentiality도 동시에 만족한다. 그리고 Signcryption에서 사용되는 공유키 K는 Alice와 Bob사이에 직접 전송되지 않고 Diffie-Hellman Key Agreement를 따른다. 따라서, 각 Step에서 서명 부가문서는 모두 검증 가능하고 Alice와 Bob 그리고 제3자에 대해서 프로토콜의 안전성이 보장된다.

4.1.2 Alice의 부정에 대한 안전성

Alice가 Step3에서 고의로 Bob에게 보내야 할 컨텐츠를 임의의 다른 컨텐츠 m'로 대치한다면 Bob은 $(m',\ r,\ s)$를 Chris(혹은 제3자)에게 보냄으로써 자신의 세션키 y를 밝히지 않고, 간단하게 Alice의 부정을 증명할 수 있다[4].

$$X \equiv (y_a\, g^r)^s \bmod p \quad (28)$$

$$r \stackrel{?}{=} KH_X(m') \quad (29)$$

위의 과정을 만족하는 경우, Chris는 Alice가 부정을 행했음을 인정할 수 있다. 만약 Bob이 정당한 컨텐츠에 대해서 위와 같은 과정을 거치게 되면 컨텐츠 정보 m이 Chris에게 전달된다. 이러한 경우에는 Step3에서 Alice가 Signcryption하는 과정을 다음과 같이 바꿈으로 해서 정당한 컨텐츠가 제3자의 검증과정에서 노출되는 것을 막을 수 있다[9].

$$c = E_K(m) \quad (30)$$

$$r = KH_X(c) \quad (31)$$

$$s \equiv x/(r + x_a) \bmod q \quad (32)$$

Bob은 Chris에게 $(c,\ r,\ s)$를 보낸다. 제3자의 검증과정은 다음과 같이 컨텐츠의 내용을 모르고 검증을 수행한다. 마지막 식이 정당하다면 이것이 Alice의 서명임을 증명할 수 있다.

$$X \equiv (y_a\, g^r)^s \bmod p \quad (33)$$

$$r \stackrel{?}{=} KH_X(c) \quad (34)$$

Alice의 또 다른 부정 행위로 다음을 생각해 볼 수 있다. Bob이 부정을 행하여 세션키 y가 Alice에게 알려질 경우 Alice는 다른 컨텐츠 m'에 대해 검증 가능한 서명을 생성할 수 있다. 이렇게 Alice가 Bob을 모함하기 위해 부정을 행한 경우, Bob은 Alice의 세션키 x와 비밀키 x_a를 밝힘으로써 Alice의 부정을 막을 수 있다. Bob은 이미 가지고 있는 $(c,\ r,\ s)$와 또 다른 컨텐츠 m'에 대한 Signcrypted 메시지

(c', r', s')로 아래의 연립 일차합동식을 구할 수 있다. 두 일차합동식을 풀면 Alice의 세션키와 비밀키를 구할 수 있다(Digital coin의 이중사용 방지기법 [6,7]). 마지막으로 Bob은 Alice의 세션키로 식 (37)을 확인할 수 있다.

$$s \equiv x/(r+x_a) \bmod q \quad (35)$$

$$s' \equiv x/(r'+x_a) \bmod q \quad (36)$$

$$X \triangleq g^x \bmod p \quad (37)$$

4.1.3 Bob의 부정에 대한 안전성

Alice는 후에 서명부가문서 (c, r, s, y)를 발견한 경우, 이 서명부가문서가 Bob에 의해 불법 유통되었다라는 것을 임의의 제3자에게 간단하게 증명할 수 있다. 결국 Alice는 서명 중의 y를 이용하여 Bob이 컨텐츠를 불법 복사하여 유포하였음을 Chris에게 증명한다.

1. Alice는 보관하고 있던 Bob의 서명부가문서 res, $Sig_B(res, X)$와 Signcryption에서 사용된 키 K, 인증서 $Cert_B$ 그리고 발견된 Signcrypted 메시지 (c, r, s, y)를 Chris에게 보낸다.

2. Chris는 우선 인증서의 공개키 정보로 Bob의 서명을 검증하고, Signcrypted 메시지로부터 키 K를 확인한다. 또한, Bob이 이산대수 지식 y를 알고 있음을 확인하고, 세션키 y와 인증서에 포함된 ID로 y'가 성립함을 보임으로써 Bob의 부정을 증명한다.

$$K \triangleq H((y_a g^r)^{sy} \bmod p) \quad (38)$$

$$U \triangleq g^v Y^{H(U)} \bmod p \quad (39)$$

$$Y \triangleq g^y \bmod p \quad (40)$$

$$y' \triangleq H(y \| ID_B) \quad (41)$$

위와 같은 검증을 통과한 경우 Chris는 Bob이 부정자라고 판정할 수 있다. 기존의 논문에서는 Alice로 한정하여 이러한 과정이 성립하도록 하였지만(Bob의 세션키 도출은 Alice만 가능하다), 제3자(예를 들어 adaptive attacker등)에게도 가능하게 함으로써 Bob이 불법적으로 컨텐츠를 유통시키는 것을 억제할 수 있다(Self-Enforcing property). 여기에서 Alice는 자신이 정당한 컨텐츠 제공자라는 것은 아래의 식 (42)로 증명한다.

$$K \triangleq H(Y^x \bmod p) \quad (42)$$

4.1.4 Chris의 부정에 대한 안전성

Alice와 Bob의 부정에 대한 검증을 수행하는 경우, 모든 세션키가 OTP와 같은 성질을 가지므로 Chris의 부정 행위는 의미가 없어지게 된다. 만약, 4.1.2절에서 Alice의 부정 행위가 증명된 경우 Alice는 반드시 자신의 인증서 $Cert_A$를 갱신해야 한다. 만약 갱신하지 않는다면 제3자(adaptive attacker)는 식 (9)로부터 세션키를 도출할 수 있으므로 Alice의 모든 컨텐츠를 복호할 수 있게 된다. 마찬가지로 Bob의 부정 행위가 증명된 경우에도 인증서를 갱신해야 한다.

4.2 프로토콜II의 안전성

4.2.1 프로토콜 자체의 안전성

4.1.1에서와 동일하게 증명할 수 있다. Step2에서의 k값은 오직 Alice만이 구할 수 있으므로 수신자 지정 검증방식을 취하고 있다.

4.2.2 Alice의 부정에 대한 안전성

Alice의 부정 행위에 대한 안전성은 4.1.2와 유사하다. 우선, Alice가 Step3에서 고의로 Bob에게 보내야 할 컨텐츠를 임의의 다른 것으로 대치한다면 Bob은 (m', r_A, s_A)를 Chris에게 보냄으로써 간단하게 Alice의 부정을 증명할 수 있다.

$$X \equiv (y_a g^{r_A})^{s_A} \bmod p \quad (43)$$

$$r_A \triangleq KH_X(m') \quad (44)$$

또한 Bob은 제3자의 검증과정에서 식 (30~34)와 동일하게 컨텐츠 정보 m 대신에 암호문을 사용하여 검증을 수행할 수 있다.

그리고 Bob의 세션키가 Alice에게 알려질 경우, Alice는 또 다른 메시지에 대하여 서명부가문서를 만들 수 있다. 이때, Bob은 식 (35~37)을 사용하여 동일하게 Alice의 세션키와 비밀키를 구함으로써 Alice의 부정을 증명할 수 있다.

4.2.3 Bob의 부정에 대한 안전성

Alice는 후에 서명부가문서 (c_A, r_A, s_A, y)를 발견한 경우, 4.1.3과 동일하게 수행한다.

1. Alice는 프로토콜 과정에서 보존하여 둔 r_B, s_B와 $Cert_B$, 공유키 K 그리고 발견된 Signcrypted 메시지를 Chris에게 보낸다.

2. Chris는 Signcrypted 메시지로부터 키 K를 확인한다. 또한, Bob의 세션키 y를 확인하고 식 (47)이 성립함을 보임으로써 Bob의 부정을 증명한다.

$$K \triangleq H((y_a\, g^{r'})^{s_A\, y} \bmod p) \quad (45)$$

$$Y \triangleq g^{y} \bmod p \quad (46)$$

$$k \triangleq H(y_a^{\,y} \bmod p) \quad (47)$$

위의 검증을 통과한 경우 Chris는 Bob이 부정자라고 판정할 수 있다.

4.2.4 Chris의 부정에 대한 안전성

4.1.3과 동일하다.

4.3 효율의 비교

여기에서는 기존의 방식과 제안된 프로토콜I, II를 비교한다. 표1에서 알 수 있듯이 제안된 프로토콜I, II의 계산량과 통신 오버헤드는 기존의 방식보다 효율적이다. 또한, 통신횟수나 논쟁에 참여하는 Entity수, 논쟁 후 서명자의 부정 그리고 결탁 가능성은 기존의 프로토콜과 동일한 성능을 가진다. 그리고 기존의 프로토콜에 비해 제안된 프로토콜들은 공개키에 의한 Direct Verifiability가 가능하다는 장점이 있다.

표1. 각 프로토콜의 비교

	기존방식[1]	제안 프로토콜I	제안 프로토콜II
통신횟수	3	3	3
계산량	20	12	13
통신량	$9\|p\|+5\|q\|+\|H\|$	$5\|p\|+2\|q\|+2\|H\|$	$2\|p\|+2\|q\|+2\|H\|$
논쟁에 참여하는 개체	2개	2개	2개
논쟁 후 서명자의 부정	불가능	불가능	불가능
결탁	불가능	불가능	불가능
공개키에 의한 검증	불가능	가능	가능

5. 결론

본 논문에서는 기존의 서명부가문서를 이용한 부정자 추적 프로토콜을 보다 상세히 분석함으로써 PKI기반에 적용 가능한 효율적인 프로토콜로 개선하였다. 또한 변형된 Signcryption을 이용하여 제안된 방법은 기존 프로토콜의 특성을 모두 포함한 간결하면서 실제 적용이 가능한 실용적인 프로토콜이다. 추후의 연구과제로 타원곡선 암호시스템에의 적용과 무선 인터넷상에서 적용 가능한 프로토콜의 개발을 고려하고 있다.

[참고문헌]

[1] Y. Watanabe, Y. Zheng, H. Imai, "Traitor Traceable Signature Scheme", In Proc. of ISIT2000(International Symposium on Information Theory), pp.463, 2000

[2] Y. Zheng, "Shortened Digital Signature, Signcryption and Compact and Unforgeable Key Agreement Schemes", P1363 Submissions to the Study Group for Future Public-Key Cryptography Standards

[3] H. Petersen, M. Michels, "Cryptanalysis and Improvement of Signcryption Schemes (Revised Version)", IEE Computer and Digital Techniques, 1998

[4] F. Bao, R. H. Deng, "A Signcryption Scheme with Signature Directly Verifiable by Public Key", PKC'98, Springer-Verlag, LNCS 1431, pp.55-59, 1998

[5] C. P. Schnorr, "Efficient Identification and Signature for Smart Cards", Eurocrypt'89, Springer-Verlag, LNCS 435, pp.688-689, 1989

[6] W. Mao, "Blind Certification of Public Keys and Efficiently Revocable Electronic Cash: Secure against Capable Attackers", In HPL-96-134, 1996

[7] S. Brands, "Untraceable Off-line Cash in Wallets with Observers", In Proc. of Crypto'93, pp.302-318, 1993

[8] Pointcheval, D., Stern J., "Security Proofs for Signature Schemes", Eurocrypt'96, Springer-Verlag, LNCS 1070, pp.239-251, 1996

[9] C. Gamage, J. Leiwo, Y. Zheng, "Encrypted Message Authentication by Firewalls", PKC'99, Springer-Verlag, LNCS 1560, pp.69-81, 1999