유연성을 가진 새로운 멀티-사인크립션 프로토콜
(A New and Flexible Multi-signcryption Protocol)

서 승 현 † 이 상 호 ††
(Seung-Hyun Seo) (Sang-Ho Lee)

요 약 멀티-사인크립션 프로토콜은 사인크립션의 확장개념으로 다수의 서명자들이 각 메시지들에 대해서 함께 사인크립션 과정을 수행하는 프로토콜을 말한다. 이것은 기밀성 기능과 인증 기능을 제공함으로써 소프트웨어나 기타 여러 문서들이 인터넷상에서 전전하게 배포되고, 송·수신될 수 있도록 하는 유용한 암호학적 프로토콜이다. 본 논문에서는 기존의 멀티-사인크립션 프로토콜들의 취약점들을 분석하고, 기존 연구들의 효율성과 취약점들을 개선하여 새로운 멀티-사인크립션 프로토콜을 제안한다. 제안하는 프로토콜은 메시지 유연성과 순서의 유연성, 메시지 검증성과 순서의 검증성, 메시지 기밀성, 메시지 위조불가능성, 부인방지성, 강건성을 효율적으로 제공한다. 따라서 제안하는 프로토콜은 인터넷상에서, 다수 서명자들의 메시지들을 악의적인 사용자로부터 보호하기에 적합하다.

키워드 : 사인크립션, 멀티-사인크립션, 다중 서명

Abstract Multi-signcryption scheme is an extension of signcryption scheme for multi-signers performing together the signcryption operation on messages, and it provides useful cryptographic functions such as confidentiality and authenticity for the sound circulation of messages through the Internet. In this paper, we show the weaknesses of the previous multi-signcryption schemes. And then we propose a new multi-signcryption scheme that improves the weaknesses and the efficiency of the previous schemes. Our scheme efficiently provides message flexibility, order flexibility, message verifiability, order verifiability, message confidentiality, message unforgeability, non-repudiation and robustness. Therefore, it is suitable for protecting messages and multi-signers from malicious attacks in the Internet.

Key words : Signcryption, multi-signcryption, multi-signature

1. 서 론

인터넷의 계속적인 발전으로 데이타나 소프트웨어 프로그램, 개인 문서 등이 인터넷을 통해서 배포되고 있으며, 이메일 시스템을 통해서 설문 조사 문서, 회사 문서, 리포트, 프로그램 등 많은 정보들이 오고 간다. 이 과정에서 사용자는 인터넷을 통해 관련 문서나 프로그램을 첨부해서 전송하기도 하고, 수신자는 첨부된 문서의 내용을 수정하거나 개선하여 다른 사람에게 혹은 원래 문서를 전송한 사용자에게 다시 전송하기도 한다. 또한 회사 내에서 사원들의 개인적인 의견수렴 및 개인의 선호도 조사를 하거나, 회사문서의 검토 및 수정 등을 할 때, 편리하게 이메일 시스템을 포함한 인터넷 통신 시스템을 사용하고 있다.

그러나, 최근에 악의적인 사용자들이 컴퓨터 바이러스 프로그램들을 비롯한 악성 코드들을 인터넷상에 유포하거나, 일반 메시지나 프로그램에 악성 코드를 첨부하여 이메일을 통해 배포함으로써 사용자들에게 치명적인 손해를 입히는 일들이 많이 발생하고 있다. 또한 개인의 사적인 정보나 회사 기밀 정보 등이 인터넷상에서 그대로 유포되는 경우도 있어서, 이를 악용하는 사용자들로 인하여 개인이나 회사가 큰 피해를 보고 있다.

따라서, 메시지 작성자 및 수정해서 배포한 사람이 누구인지를 인증하고, 인증되지 않은 메시지로부터 사용자가 피해를 보는 일을 방지하며, 사적인 정보나, 기밀성을 요구하는 메시지는 외부에 노출되지 않도록 보호해야 한다. 이를 위해서, 사용자 인증과 메시지 기밀성을 동시에 제공하면서 다수의 사용자들이 효율적으로 사용할 수 있는 프로토콜 개발이 필요하다.

멀티-사인크립션(multi-signcryption) 프로토콜[1,2]은

† 학생회원 : 이화여자대학교 컴퓨터학과
seosh@ewhain.net
†† 종신회원 : 이화여자대학교 컴퓨터학과 교수
shlee@ewha.ac.kr
논문접수 : 2004년 6월 30일
심사완료 : 2005년 3월 4일

1997년 Yuliang Zheng이 제안한 사인크립션(sign-cryption) 프로토콜[3,4]의 확장개념으로, 다수의 서명자들이 각 메시지들에 대해서 함께 사인크립션 과정을 수행하는 프로토콜을 의미하며, 다중 서명(multi-signature)[5]과 암호화(encryption) 기법을 함께 사용함으로써 사용자 인증과 메시지 기밀성을 효율적으로 제공한다. 따라서, 멀티-사인크립션 프로토콜은 인터넷상에서 악의적인 코드 전송으로부터 사용자를 보호하고, 메시지가 사용자들에게 건전하게 배포될 수 있도록 안전한 송·수신 환경을 제공하는데 유용한 프로토콜이다.

최근에, Mitomi와 Miyaji[1,6]가 메시지 유연성을 가진 멀티 사인크립션 프로토콜을 제안하였으나, 이 프로토콜은 메시지 기밀성(message confidentiality)을 제공하지 않아, 기밀성이 요구되는 메시지의 송·수신에는 적합하지 않은 문제점이 있다. 이를 개선하여 Pang과 Catania, Tan[2]이 메시지 기밀성을 제공하는 멀티-사인크립션 프로토콜을 제안하였으나, 이 프로토콜의 경우 메시지 서명자의 순서가 사전에 고정됨으로써, 순서의 유연성(order flexibility)을 제공하지 못하고, 멀티-사인크립션을 수행할 서명자들이 원본 메시지 작성자의 서명을 검증하지 못하는 문제점이 있다.

본 논문에서는 기존의 멀티-사인크립션 프로토콜들의 취약성을 분석하고, 이를 개선하여 유연성을 가진 새로운 멀티-사인크립션 프로토콜을 제안하였다. 제안한 프로토콜에서는 원본 메시지 작성자가 메시지에 서명을 하여 멀티-사인크립션을 수행할 서명자들에게 전송하고, 멀티-사인크립션을 수행하는 서명자들은 각각 원본 메시지를 수정하여 원본 메시지 작성자만이 수정된 사항을 알 수 있도록 멀티-사인크립션을 수행함으로써, 메시지의 기밀성(message confidentiality)을 제공한다. 또한 인증되지 않은 메시지는 복호화될 수 없게 함으로써 메시지 강건성(robustness)도 만족시키며, 누구나 언제든지 멀티-사인크립션에 참여할 수 있도록 사전에 서명자 및 서명자의 순서를 고정시키지 않았다. 그 밖에, 메시지 유연성(message flexibility)과 메시지 검증성(message verifiability), 메시지 위조불가능성(message unforgeability), 부인 방지성(non-repudiation)등을 효율적으로 제공한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 사용할 용어들을 정의하고, 멀티-사인크립션 프로토콜을 안전하게 설계하기 위해서 고려해야 할 보안 특성과 요구조건을 기술한다. 3장에서는 제안한 프로토콜과 비교대상이 되는 기존의 멀티-사인크립션 프로토콜들을 소개하고, 4장에서는 본 논문에서 제안한 프로토콜을 설명한다. 5장에서는 제안한 프로토콜의 안전성 및 특성을 분석하고, 기존의 멀티-사인크립션 프로토콜들과 제안한 프로토콜의 효율성을 비교 분석한다. 마지막으로 6장에서는 결론을 맺는다.

2. 개 요

이 장에서는 본 논문에서 사용할 용어들을 정의하고, 안전하고 유연성 있는 멀티-사인크립션 프로토콜을 설계하기 위해서 만족시켜야 하는 요구사항들을 기술한다.

2.1 용어 정의

[기호 정의]

- I_0 : 원본 메시지를 작성하고 배포한 서명자이며, I_j $(1 \le j \le n)$들이 수행한 멀티-사인크립션 결과를 검증하는 검증자
- I_j : I_0의 원본 메시지를 수정한 후, 수정한 메시지에 대해서 멀티-사인크립션을 수행하는 j번째 서명자 $(1 \le j \le n)$
- ID_i : 서명자 I_i $(0 \le i \le n)$의 식별정보
- m_i : 서명자 I_i $(0 \le i \le n)$가 생성한 메시지
- $E_K(\,)$: 키 K를 사용한 대칭키 암호화 알고리즘 (symmetric key encryption algorithm)
- x_i : 서명자 I_i의 비밀키(private key) $x_i \in Z_q^*$
- y_i : 서명자 I_i의 공개키(public key) $y_i = g^{x_i} \pmod p$
- g : 곱셈 군(multiplicative group) Z_p^*의 생성자 (generator), p의 원시 근(primitive root)
- p, q : 강한 소수(strong prime), $p = 2*q+1$
- $H(\cdot), h_1(\cdot), h_2(\cdot)$: 강한 일방향 해쉬 함수(strong one-way hash function)

2.2 요구사항

유연성을 가진 멀티-사인크립션 프로토콜을 안전하게 설계하기 위해서 고려해야 할 보안 특성과 요구조건 [1,6]은 다음과 같다. 본 논문에서 제안하는 프로토콜은 이러한 요구조건들을 만족시키도록 설계되었다.

(1) 메시지의 유연성(message flexibility)
: 멀티-사인크립션이 수행될 메시지가 사전에 고정되지 않는다.

(2) 순서의 유연성(order flexibility)
: 서명자의 순서가 미리 정해지지 않는다.

(3) 메시지와 순서의 검증성(message and order verifiability)
: 원본 메시지를 생성한 사람과 원본 메시지를 수정한 사람이 서로 구분되며, 메시지가 수정된 순서도 알 수 있다.

(4) 메시지 기밀성(message confidentiality)
: 악의적인 공격자나 제 3자가 멀티-사인크립션이 수행된 메시지의 내용을 알아내는 것은 계산적으

로 불가능하다.

(5) 메시지 위조불가능성(message unforgeability)
: 악의적인 공격자나 제 3자가 서명자를 가장하여, 멀티-사인크립션이 수행된 메시지를 위조해내는 것은 계산적으로 불가능하다.

(6) 부인 방지성(non-repudiation)
: 서명자가 메시지에 멀티-사인크립션을 수행한 후, 자신이 멀티-사인크립션 메시지를 생성했다는 사실을 부인할 수 없다.

(7) 강건성(robustness)
: 멀티-사인크립션이 수행된 메시지를 받은 후, 검증에 실패하면 멀티-사인크립션 메시지는 복호화 될 수 없다. 따라서 이 특성을 만족하면, 인증되지 않은 악의적인 메시지로부터 수신자를 보호할 수 있다.

3. 기존 연구

이 장에서는 기존에 발표된 멀티-사인크립션 프로토콜들을 소개하고, 그들의 취약점을 기술한다.

3.1 Mitomi-Miyaji 프로토콜

Mitomi와 Miyaji[1,6]는 처음으로, 메시지 유연성을 가진 다중 서명 프로토콜을 제안하였고, 제안한 서명 프로토콜에 암호화 기능을 결합시킴으로서, 멀티-사인크립션 프로토콜을 제안하였다. Mitomi-Miyaji 프로토콜 수행과정은 그림 1에서 요약되어 있다. 첫 번째 서명자 I_0이 원본 메시지 m_0을 작성한 후, 멀티-사인크립션을 수행하고, 수행결과(ID_0, s_0, C_0)를 다음 서명자 I_j에게 보낸다. I_j는 멀티-사인크립트 메시지를 검증하고 복호화한 후, m_0을 수정하여 m_j를 작성하고, m_j에 멀티-사인크립션을 수행하여 그 결과(ID_j, s_j, C_j)를 다음 서명자에게 전송한다. 마지막으로 서명자 I_n이 멀티-사인크립션을 수행한 뒤 (ID_0, s_0, C_0), ..., (ID_n, s_n, r_n, C_n)을 검증자에게 전송한다.

그러나, Mitomi-Miyaji 프로토콜에서는 멀티-사인크립션을 수행하는데 사용한 세션키를 누구든지 계산할 수 있는 문제점이 있다. 즉, 세션키 $K_j = h_2(r_j - g^{s_j^{-1}} \cdot y_j^{r_j \cdot s_j^{-1}})$가 서명자 I_j로부터 전송받은 값들(s_j, r_j)과 I_j의 공개키 y_j로 계산되기 때문에, 멀티-사인크립션이 수행된 메시지를 받는 사람은 누구나 멀티-언사이크립션(multi-unsigncryption) 과정에서 필요한 세션키를 계산할 수 있다.

따라서 전송되는 멀티-사인크립션 메시지를 도청한 공격자나 외부의 악의적인 사용자들도 쉽게 멀티 사인크립션이 수행된 결과를 검증하고 복호화할 수 있으므로, 메시지 기밀성을 제공하지 못한다. 그러므로 서명자의 사적인 정보가 원본 메시지에 첨가 되거나 수정된 사항에 기밀성이 요구 될 경우, Mitomi-Miyaji 프로토콜을 사용하는 것은 적합하지 않다.

3.2 Pang-Catania-Tan 프로토콜

Pang과 Catania, Tan[2]은 메시지 기밀성을 만족시키도록 Mitomi-Miyaji 프로토콜을 개선하여 멀티-사인크립션 프로토콜을 제안하였다. Pang-Catania-Tan 프로토콜은 누구나 멀티-사인크립션이 수행된 메시지를 검증하고 복호화할 수 있었던 Mitomi-Miyaji 프로토콜과 달리, 원본 메시지를 배포한 서명자 I_0만이 멀티-사인크립션이 수행된 메시지를 검증할 수 있고 복호화할 수 있게 함으로써 메시지 기밀성을 제공하며, I_0가 서명자인 동시에 검증자가 된다.

Pang-Catania-Tan 프로토콜은 다음과 같이 수행되며, 수행과정은 그림 2에서 요약되었다. I_0가 멀티-사인크립션을 수행할 서명자들을 미리 정하고 그 서명자들의 순서를 결정한 후, 원본 메시지 m_0에 서명을 하여 메시지와 서명값(ID_0, s_0, r_0)을 첫 번째 서명자 I_1에게 전송한다. 이후, I_1는 m_0을 수정하여 m_1을 작성하고

Multi-Signcryption of m_j $(0 \le j \le n)$				Multi-Unsigncryption
I_0	...	I_n		I_j
Chooses $k_0 \in_R Z_q$ **Computes:** $R_0 = g^{k_0} \bmod p$, $K_1 = h_2(h_1(m_0 \| ID_0))$, $r_0 = R_0 + h_1(m_0 \| ID_0) \bmod q$, $s_0 = (x_0 r_0 + 1) \cdot k_0^{-1} \bmod q$, $C_0 = E_{K_0}(m_0 \| ID_0)$	$\xrightarrow{(ID_0, s_0, C_0)}$	**Chooses** $k_n \in_R Z_q$ **Performs Multi-Unsigncryption** **Computes:** $R_n = g^{k_n} \bmod p$, $K_n = h_2(r_{n-1} \cdot h_1(m_n \| ID_n))$, $r_n = R_n + h_1(m_n \| ID_n) \cdot r_{n-1} \bmod q$, $s_n = (x_n r_n + 1) \cdot k_n^{-1} \bmod q$, $C_n = E_{K_n}(m_n \| ID_n)$	$\xrightarrow{(ID_1, s_1, C_1), \ldots, (ID_n, r_n, s_n, C_n)}$	**For** $j = n, \ldots, 3, 2, 1$ **Computes:** $R_j' = g^{s_j^{-1}} \cdot y_j^{r_j \cdot s_j^{-1}} \bmod p$, $T_j = r_j - R_j' \bmod q$, $K_j = h_2(T_j)$ **Decrypts:** m_j, ID_j **Recovers:** $r_{j-1} = T_j \cdot (h_1(m_j \| ID_j))^{-1} \bmod q$

그림 1 Mitomi-Miyaji 프로토콜

Set-up	Multi-Signcryption of m_j $(1 \le j \le n)$			Multi-Unsigncryption
I_0		I_j	...	I_0
Chooses $k_0 \in_R Z_q^*$ Computes: $R_0 = y_0^{k_0} \bmod p$, $r_0 = (h_1(m_0 \| ID_0))^{-1} \cdot R_0 \bmod q$, $s_0 = (x_0 r_0 + y_1) \cdot k_0^{-1} \bmod q$	$\xrightarrow{(ID_0, s_0, r_0)}$	Chooses $k_j \in_R Z_q^*$ Computes: $R_j = y_0^{k_j} \bmod p$, $r_j = (h_1(m_j \| ID_j) \cdot r_{j-1})^{-1} \cdot R_j \bmod q$, $s_j = (x_j r_j + y_{j+1}) \cdot k_j^{-1} \bmod q$, $K_j = h_2(r_{j-1} \cdot h_1(m_j \| ID_j))$, $C_j = E_{K_j}(m_j \| ID_j)$	$\xrightarrow{(ID_1, s_1, C_1), \ldots, (ID_n, r_n, s_n, C_n)}$	For $j = n, \ldots, 3, 2, 1$ Computes: $R_j' = y_0^{s_j^{-1} \cdot y_{j+1}} \cdot y_j^{x_0 \cdot r_j \cdot s_j^{-1}} \bmod q$, $T_j = r_j^{-1} \cdot R_j' \bmod q$, $K_j = h_2(T_j)$ Decrypts: m_j, ID_j Recovers: $r_{j-1} = T_j \cdot (h_1(m_j \| ID_j))^{-1} \bmod q$

그림 2 Pang-Catania-Tan 프로토콜

멀티-사인크립션을 수행해서 결과 값 (ID_1, s_1, C_1)를 그 다음 서명자에게 전송한다. 최종적으로 마지막 서명자 I_n이 m_0을 수정하여 m_n을 작성한 후, 그것에 멀티-사인크립션을 수행하고 결과 메시지와 이전 서명자로부터 받았던 멀티-사인크립션 메시지들을 모두 I_0에게 보낸다.

Pang-Catania-Tan 프로토콜은 I_0의 비밀키 x_0를 멀티-사인크립션 복호화에 사용함으로써, I_0만이 메시지를 복호화할 수 있게 하여 메시지 기밀성을 제공하였다. 그러나 I_0가 사전에 멀티-사인크립션을 수행할 서명자들을 지정하고, 그들의 순서도 정해 놓기 때문에 순서의 유연성을 제공하지 못한다. 따라서 자신의 순서가 돌아올 때까지 서명자는 서명을 하지 못하고 기다리고 있어야 하며, 멀티-사인크립션이 수행되는 동안에는 새로운 서명자가 추가될 수 없는 불편이 있다. 또한 I_0가 메시지 m_0에 대한 서명 (s_0, r_0)을 생성할 때, 계산하는 $R_0 = g^{x_0 \cdot k_0}$값은 I_0의 비밀키 x_0와 I_0가 선택한 비밀 랜덤 값 k_0를 사용해서 계산하기 때문에, 다른 서명자들 I_j가 이 서명값 (s_0, r_0)을 메시지 m_0과 함께 받는다고 할지라도, I_0가 정말로 m_0를 작성해서 서명했는지에 대한 검증이 이루어지지 못한다. 따라서 원본 서명자가 작성한 서명에 대해서 부인 방지성을 만족하지 못한다.

4. 제안하는 멀티-사인크립션 프로토콜

이 장에서는 기존 프로토콜들의 취약성을 해결하고, 효율성을 개선하여 유연성을 가진 새로운 멀티-사인크립션 프로토콜을 제안한다. 본 논문에서 제안하는 프로토콜은 Pang-Catania-Tan 프로토콜과 같이 원본 메시지를 작성하여 배포한 서명자 I_0만이 멀티-사인크립션이 수행된 결과 메시지를 검증하고 복호화할 수 있게 함으로써 Mitomi-Miyaji 프로토콜과 달리, 메시지 기밀성을 제공하였다. 또한 순서의 유연성을 부가함으로써 Pang-Catania-Tan 프로토콜과 다르게, 사전에 서명자들이 고정되고 순서가 정해지는 것에 따른 불편을 해소하였으며 효율성도 개선시켰다. 제안하는 프로토콜은 Set-up 단계, Multi-Signcryption 단계, Multi-Unsigncryption 단계로 구성되며 수행과정은 다음과 같고 그림 3에서 요약하였다.

[Set-up 단계]

이 단계에서는 서명자 I_0가 원본 메시지 m_0을 작성하고 m_0에 서명을 하여 멀티-사인크립션을 수행할 서명자에게 m_0와 서명 값을 보낸다.

서명자 I_0는 원본 메시지 m_0를 작성하고, m_0에 대한 서명 값을 계산하기 위해서 임의의 난수 $k_0 \in_R Z_q^*$를 선택하여 $R_0 = g^{k_0} (\bmod\ p)$, $r_0 = H(m_0 \| ID_0 \| R_0) (\bmod\ q)$, $s_0 = (x_0 + r_0) \cdot k_0^{-1} (\bmod\ q)$를 계산한다. 다음 서명자 I_j을 선택하여, ID_0와 함께 m_0와 서명 값 (r_0, s_0)를 전송하고 자신의 서명 값을 공개한다. 제안하는 프로토콜은 순서의 유연성을 제공하기 때문에 서명자는 자유롭게 다음 서명자를 선택할 수 있으나 편의상 서명자 I_j $(1 \le j \le n)$의 다음 서명자를 I_{j+1}로 한다.

[Multi-Signcryption 단계]

이 단계에서 서명자 I_j $(1 \le j \le n)$들은 I_0의 서명을 검증하고, 원본 메시지 m_0을 수정하여 수정한 메시지 m_j에 멀티-사인크립션을 수행한다.

(1) I_j은 I_0로부터 메시지 m_0와 서명값 (ID_0, r_0, s_0)을 받은 후, I_0의 서명을 다음과 같이 검증한다. $R_0' = (y_0 \cdot g^{r_0})^{s_0^{-1}} = g^{(x_0 + r_0) \cdot (x_0 + r_0)^{-1} \cdot k_0} (mod\ p)$을 계산하고, $H(m_0 \| ID_0 \| R_0')$값과 r_0가 같은지를 확인한다.

(2) I_j는 m_0을 수정하여 m_j을 작성한다. m_j에 대한 멀티-사인크립션을 수행하기 위해서, I_j는 임의의 난수 $k_j \in_R Z_q^*$을 선택한 후, I_0의 공개키와 k_j을 이

용해 세션키 $K_j = y_0^{k_j} (= g^{x_0 k_j}) (mod\ p)$을 계산하고, I_0의 서명 값인 r_0을 사용하여 서명 값 $r_j = H(m_j \| ID_j \| K_j \cdot r_0 (mod\ q)$와 $s_j = (x_j + r_j) \cdot k_j^{-1} (mod\ q)$을 계산한다. 또한 세션키 K_j로 (ID_0, ID_j, m_j)을 암호화하여 $C_j = E_{K_j}(ID_0 \| ID_j \| m_j)$를 생성한다. 여기서, r_j와 s_j은 서명자 I_j을 인증하기 위함이고, C_j는 m_0을 수정해서 생성한 m_j의 기밀성을 제공하기 위함이다. I_j은 다음 서명자 I_{j+1}을 선택하여 멀티-사인크립트 메시지 (ID_j, r_j, s_j, C_j)와 I_0의 메시지 m_0, 서명값(ID_0, r_0, s_0)을 전송한다.

(3) 서명자 I_{j+1}는 멀티-사인크립션 메시지를 I_j로부터 받고서 m_0에 대한 서명검증 후, m_0을 수정하여 m_{j+1}을 작성한다. m_{j+1}에 대한 멀티-사인크립션을 수행하기 위해서, I_{j+1}는 임의의 난수 $k_{j+1} \in {}_R Z_q^*$을 선택한 후, I_0의 공개키와 k_{j+1}을 이용해 세션키 을 계산하고, I_j의 서명 값인 r_j을 사용하여 서명 값 $r_{j+1} = H(m_{j+1} \| ID_{j+1} \|$ 와 $s_{j+1} = (x_{j+1} + r_{j+1}) \cdot k_{j+1}^{-1} (\mathrm{mod}\ q)$을 계산한다. 또한 세션키 K_{j+1}으로 $(ID_j, ID_{j+1}, m_{j+1})$을 암호화하여 $C_{j+1} = E_{K_{j+1}}(ID_j \| ID_{j+1} \| m_{j+1})$를 생성한다.

만약 I_{j+1}이 마지막 서명자라면 즉 $I_{j+1} = I_n$이면, I_n은 멀티 사인크립션이 수행된 메시지들 (ID_1, s_1, C_1), ..., $(ID_{n-1}, s_{n-1}, C_{n-1})$, (ID_n, r_n, s_n, C_n)을 원본 메시지의 작성자이면서 검증자인 I_0에게 전송한다. I_{j+1}이 마지막 서명자가 아니라면, 다음 서명자 I_{j+2}를 선택하여 멀티-사인크립션 메시지 $(ID_1, s_1, C_1), \ldots, (ID_j, s_j, C_j), (ID_{j+1}, r_{j+1}, s_{j+1}, C_{j+1})$와 I_0의 메시지 m_0, 서명값 (ID_0, r_0, s_0)을 전송한다. 여기서 서명값 $r_j (1 \le j \le n)$들은 체인형태로 서로 연결되어 있기 때문에, 마지막 서명자 I_n이 생성한 r_n값만 알면, 향후 멀티-사인크립션 검증과정에서 이전 서명자들의 $r_j (1 \le j \le n-1)$ 값은 검증자가 계산해낼 수 있다. 따라서 전송되는 통신 메시지 양을 줄이기 위해서, 이전의 서명자들이 생성한 r_j 값은 전송하지 않고, 마지막 서명자 I_n이 생성한 r_n만을 다른 멀티-사인크립션 메시지들과 함께 보낸다.

[Multi-Unsigncryption 단계]

이 단계에서 원본 메시지 서명자 I_0는 $I_j (1 \le j \le n)$들이 작성한 멀티-사인크립션 메시지들을 검증하고, m_0을 수정하여 작성한 $m_j (1 \le j \le n)$들을 복호화한다.

(1) I_0는 마지막 서명자 I_n로부터 멀티-사인크립션 메시지들 (ID_1, s_1, C_1), ... ,$(ID_{n-1}, s_{n-1}, C_{n-1})$, (ID_n, r_n, s_n, C_n)을 받은 후 다음과 같이 검증하고 복호화한다.

(2) $j = n, \ldots, 3, 2, 1$동안, I_0는 자신의 비밀키 x_0값과, I_j의 공개키 y_j값, 서명값(r_j, s_j)을 이용해서 다음과 같이 세션키 K_j'을 계산한 후 C_j를 복호화하고, r_{j-1} 값을 복원해낸다.

(i) $K_j' = (y_j \cdot g^{r_j})^{s_j^{-1} \cdot x_0} = g^{(x_j + r_j) \cdot (x_j + r_j)^{-1} \cdot k_j \cdot x_0} = g^{k_j \cdot x_0}$ $(mod\ p)$을 계산한다.

세션키 K_j' 값이 올바로 계산되어 K_j값과 같다면, C_j를 복호화해서 메시지 m_j와 서명자 I_j의 식별정보 ID_j, 서명자 I_j 이전에 서명한 서명자 I_{j-1}의 식별정보 ID_{j-1}를 얻을 수 있다.

(ii) $r_{j-1} = H(m_j \| ID_j \| K_j')^{-1} \cdot r_j (mod\ q)$을 계산한다.

서명 값의 일부인 r_{j-1}이 복원되면 $j = j-1$로 두고, 단계 (i)로 가서 $j = 0$이 될 때까지 단계 (i), (ii)를 수행한다. 복원된 r_{j-1}을 사용해 단계 (i)에서 세션키를 계산하고 C_{j-1}를 올바로 복호화하면, 복원된 r_{j-1}이 올바른 값임을 확인할 수 있다.

Set-up	Multi-Signcryption of m_j $(1 \le j \le n)$			Multi-Unsigncryption
I_0		I_j	...	I_0
Chooses $k_0 \in_R Z_q^*$ **Computes:** $R_0 = g^{k_0} \bmod p$, $r_0 = H(m_0 \| ID_0 \| R_0) \bmod q$ $s_0 = (x_0 + r_0) \cdot k_0^{-1} \bmod q$	(ID_0, r_0, s_0, m_0) ⟶	**Chooses** $k_j \in_R Z_q^*$ **Computes:** $K_j = y_0^{k_j} \bmod p$, $r_j = H(m_j \| ID_j \| K_j) \cdot r_{j-1} \bmod q$ $s_j = (x_j + r_j) \cdot k_j^{-1} \bmod q$ $C_j = E_{K_j}(m_j \| ID_j \| ID_{j-1})$	(ID_1, s_1, C_1), ..., $(ID_{n-1}, s_{n-1}, C_{n-1})$, (ID_n, r_n, s_n, C_n) ⟶	**For** $j = n, \ldots, 3, 2, 1$ **Computes:** $K_j = (g^{y_j} \cdot y_j)^{x_0 \cdot s_j^{-1}} \bmod p$ $= g^{k_j \cdot x_0}$ **Decrypts:** m_j, ID_j **Recovers:** $r_{j-1} = H(m_j \| ID_j \| K_j)^{-1} \cdot r_j \bmod q$

그림 3 제안하는 프로토콜

5. 제안하는 멀티-사인크립션 프로토콜 분석

이 장에서는 2.2절에서 기술한 요구 조건에 따라 제안하는 프로토콜의 안전성과 특성을 분석하고, 기존의 유연성을 가진 멀티-사인크립션 프로토콜들과 효율성을 비교 분석한다. 표 1에서는 변형된 Zheng 프로토콜, Mitomi-Miyaji 프로토콜, Pang- Catania-Tan 프로토콜과 제안하는 프로토콜과의 특성 및 안전성을 비교 분석한 결과를 보여주고, 표 2에서는 이들 프로토콜간의 효율성을 비교 분석한 결과를 보여준다.

여기서 변형된 Zheng의 프로토콜이란 Zheng의 사인크립션 프로토콜을 단순한 체인형식으로 만든 것으로, 원본 메시지 m_0를 작성한 서명자 I_0가 SDSS(Shortening method to Digital Signature Standard)[3]를 이용해서 m_0에 대한 서명을 생성하여 I_j에게 전송하고, 서명자 I_j는 사인크립션을 수행하여 생성한 사인크립션 메시지를 이전 서명자 I_{j-1}로부터 받은 I_{j-1}의 사인크립션 메시지와 함께 다음 서명자 I_{j+1}에게 전송하는 형태의 프로토콜을 의미한다. 따라서 변형된 Zheng의 사인크립션 프로토콜의 경우, 서명자 순서의 유연성은 제공하지만 메시지 서명자의 순서를 검증할 수 있는 정보를 포함하고 있지 않기 때문에 서명자의 순서 검증을 못한다.

5.1 안전성 및 특성 분석

(1) 메시지의 유연성 : 제안하는 프로토콜에서는 각 서명자가 원본 메시지를 각각 수정하여 멀티-사인크립션을 수행하기 때문에, 멀티-사인크립션이 수행될 메시지가 사전에 고정되지 않으므로, 위의 특성을 만족한다.

(2) 순서의 유연성 : 제안하는 프로토콜에서는 서명자의 순서가 사전에 정해져 있지 않고, 각 서명자마다 임의로 다음 서명자를 선택할 수 있으며, 원한다면 언제든지 멀티-사인크립션 과정에 참여할 수 있기 때문에 순서의 유연성을 만족한다.

(3) 메시지와 순서의 검증성 : I_0는 마지막 서명자인 I_n으로부터 멀티-사인크립션 메시지들을 받을 때, 서명자의 ID_n와 함께 받으므로 마지막 서명자의 신원을 확인할 수 있으며, 서명자의 공개키와 자신의 비밀키, 서명정보를 이용해서 멀티-사인크립션이 수행된 메시지들 $C_n, C_{n-1}, \ldots, C_1$을 복호화해서 각 서명자가 수정한 메시지 순서를 확인해 볼 수 있다. 또한 $C_n, C_{n-1}, \ldots, C_1$는 각각 이전 서명자의 식별 정보도 포함하고 있기 때문에, 서명자의 순서가 검증되고 동시에 메시지가 수정된 순서도 알 수 있다.

(4) 메시지 기밀성 : 악의적인 공격자가 서명자들 사이에서 통신되는 모든 멀티-사인크립션 메시지들을 도청한다면, (ID_1, r_1, s_1, C_1), ..., $(ID_{n-1}, r_{n-1}, s_{n-1}, C_{n-1})$, (ID_n, r_n, s_n, C_n)을 알아낼 수 있다. 그러나 이산대수 문제의 어려움[7]에 근거하여 원본 메시지 작성자이면서 검증자인 I_0의 비밀키 x_0을 알지 못하기 때문에, 암호화된 메시지 $C_n, C_{n-1}, \ldots, C_1$를 복호화해 낼 수 없으므로, 원본 메시지 m_0의 수정된 사항들 $m_1, m_2, \ldots, m_n$을 알 수 없다. 따라서 메시지 기밀성이 보장된다.

(5) 메시지 위조불가능성 : 악의적인 공격자가 서명자 I_j를 가장하여, I_j가 생성한 멀티-사인크립션 메시지 (ID_j, r_j, s_j, C_j)를 위조하려고 시도한다고 가정하자.

공격자는 원본 메시지 m_0을 임의로 수정해 m_j''을 만들고, 임의의 난수 $k_j'' \in {}_RZ_q^*$을 선택하여 $K_j'' = y_0^{k_j''} (= g^{x_0 k_j''}) \pmod p$를 계산해낼 수 있다.

또한 도청한 서명값 r_{j-1} 값을 이용해서, $r_j'' = H(m_j'' \| ID_j \| K_j'') \cdot r_{j-1} \pmod q$ 서명값의 일부를 위조해 만들어 낼 수 있다. 그러나 I_j의 비밀키인 x_j값을 알 수 없으므로, $s_j'' = (x_j + r_j'') \cdot (k_j'')^{-1} \pmod q$값을 계산해낼 수 없다. 따라서 공격자는 서명자 I_j가 생성한 멀티-사인크립션 메시지를 위조해낼 수 없다.

(6) 부인 방지성 : 멀티-사인크립션이 수행된 각 메시지들은 모두 서명자 $I_j\ (1 \le j \le n)$의 비밀키 x_j값을 서명 부분에 포함하고 있기 때문에, 서명자 I_j이외의 다른 사람이 I_j가 생성한 멀티-사인크립션 메시지를 생성해 낼 수 없다. 따라서 I_j가 멀티-사인크립션을 수행하고

표 1 특성 및 안전성 비교 분석

	Zheng	Mitomi-Miyaji	Pang-Catania-Tan	제안 프로토콜
메시지의 유연성	O	O	O	O
순서의 유연성	O	O	X	O
메시지의 검증성	O	O	O	O
순서의 검증성	X	O	O	O
메시지 기밀성	O	X	O	O
메시지 위조불가능성	O	O	O	O
부인 방지성	O	O	△	O
강건성	O	O	O	O

난 후에, 자신이 멀티-사인크립션 메시지를 생성한 사실을 부인 할 수 없다.

(7) 강건성 : 검증자가 멀티-사인크립션된 메시지를 받은 후, 서명부분 검증이면서 세션키 계산과정인 $K_j=(y_j \cdot g^{r_j})^{s_j^{-1} \cdot x_0} (mod\, p)$계산에 실패하면, 올바른 세션키가 계산되지 못했으므로, 암호화 메시지 C_j는 복호화되지 않는다. 따라서 인증되지 않은 암호화 메시지는 복호화될 수 없기 때문에, 임의의 악의적인 메시지로부터 수신자를 보호할 수 있다.

5.2 효율성 분석

이 장에서는 제안한 프로토콜의 효율성을 계산 비용과 통신 오버헤드 측면에서 고찰하고, 기존의 멀티-사인크립션 프로토콜들과 비교 분석한다. 계산 비용은 모듈라 곱셈 연산과 모듈라 지수승 연산 수로 측정하고, 통신 오버헤드 비용은 전송되는 메시지 크기로 측정하였다.

계산상 편의를 위해서 멀티-사인크립션의 서명자 수는 n으로 정하고, 메시지 크기는 $|M|$ bits로 표시하였다. 두 소수 p, q는 각각 1024 bits, 160 bits로 가정하고, 암호학적 해쉬 함수결과의 크기는 160 bits로 가정하였다.

통신 오버헤드 측면에서 보면 제안하는 프로토콜을 포함하여 기존의 멀티-사인크립션 프로토콜의 경우, i번째 서명자 I_i가 i+1 번째 서명자 I_{i+1}에게 전송하는 메시지의 크기는 모두 $i \cdot |M| + (i+1) \cdot |q| = i \cdot (|M|+160) + 160$ 이고, 마지막 서명자가 검증자에게 전송하는 메시지 크기는 $n \cdot |M| + (n+1) \cdot |q| = n \cdot (|M|+160) + 160$이다. 반면에 변형된 Zheng 프로토콜의 경우, i번째 서명자 I_i가 i+1 번째 서명자 I_{i+1}에게 전송하는 메시지의 크기는 $i \cdot |M| + i \cdot |q| + i \cdot |H(.)| = i \cdot (|M|+320)$ 이고, 마지막 서명자가 검증자에게 전송하는 메시지 크기는 $n \cdot |M| + n \cdot |q| + n \cdot |H(.)| = n \cdot (|M|+320)$이다. 따라서 제안하는 프로토콜을 포함하여 메시지 유연성을 가진 멀티 사인크립션 프로토콜의 경우, 변형된 Zheng의 프로토콜에 비해서 통신 오버헤드가 50%로 감소되며, 멀티-사인크립션을 수행하는 서명자의 수가 많을수록 통신 오버헤드는 크게 감소됨을 알 수 있다.

계산 비용을 측정할 때, 원본 메시지 m_0에 대한 서명 생성 및 검증부분과 m_0를 수정한 메시지 m_j들에 대한 멀티-사인크립션 생성 및 검증부분으로 나누어서 측정하였다. 멀티-사인크립션 수행의 경우, 서명자가 $I_j(1 \le j \le n)$이고 검증자가 I_0이며, 원본메시지에 대한 서명 수행의 경우, 서명자가 I_0이고 검증자가 $I_j(1 \le j \le n)$이다. 표 2에서 기존 프로토콜과 제안하는 프로토콜간의 계산량 비교 결과를 볼 수 있다.

멀티-사인크립션 수행부분에서 실행 시간의 대부분을 차지하는 모듈러 지수승 계산 횟수의 경우, 변형된 Zheng의 프로토콜, Mitomi-Miyaji 프로토콜, 제안하는 프로토콜 모두 서명자는 1회, 검증자는 $2n$회 이지만, Pang-Catania-Tan 프로토콜은 검증자 계산량이 $2n+1$회로 위의 프로토콜들보다 많다. 또한 160-bit 모듈라 곱셈 계산횟수 경우, 변형된 Zheng의 프로토콜은 서명자 1회, 검증자 n회로 횟수가 가장 적었고, 제안하는 프로토콜은 서명자 2회, 검증자 $2n$회로 변형된 Zheng의 프로토콜보다는 계산 횟수가 늘었지만, 다른 유연성을 가진 멀티-사인크립션 프로토콜들에 비해서는 적었다.

서명 수행부분에서는 Pang-Catania-Tan 프로토콜의 경우 원본 메시지에 대한 서명검증이 불가능하므로 계산량 비교에서 제외하고 나머지 프로토콜들에 대해서만 비교한다. 모듈라 지수승 계산횟수의 경우, 변형된 Zheng의 프로토콜과 제안하는 프로토콜은 모두 서명자 1회 검증자 1회인 반면, Mitomi-Miyaji 프로토콜은 모든 서명자들이 받았던 이전 메시지를 다 검증해야만 원본 메시지에 대한 서명을 검증할 수 있기 때문에 계산량이 많으며, 서명자 I_0는 1회 검증자 $I_j(1 \le j \le n)$는 $2j$회이다. 또한 160-bit 모듈라 곱셈 계산의 횟수의 경우, 변형된 Zheng의 프로토콜과 제안하는 프로토콜은 모두

표 2 계산량 비교 분석

		Zheng		Mitomi-Miyaji		Pang-Catania-Tan		제안 프로토콜	
		I_j	I_0	I_j	I_0	I_j	I_0	I_j	I_0
m_0에 대한 서명 수행부분 서명자: I_0 검증자: I_j	160-bit 곱셈	1	1	$2j$	2	계산 못함	3	1	1
	1024-bit 곱셈	1	0	j	0		0	1	0
	1024-bit 지수승	1	1	$2j$	1		1	1	1
m_j에 대한 멀티-사인크립션 수행부분 서명자: I_j 검증자: I_0	160-bit 곱셈	1	n	3	$2n$	4	$4n+2$	2	$2n$
	1024-bit 곱셈	0	n	0	n	0	n	0	n
	1024-bit 지수승	1	$2n$	1	$2n$	1	$2n+1$	1	$2n$

서명자 1회 검증자 1회인 반면, Mitomi-Miyaji 프로토콜은 서명자 I_0는 2회 검증자 $I_j(1 \le j \le n)$는 $2j$회이다.

따라서 효율성 측면에서 보면, 제안하는 프로토콜은 기존의 멀티-사인크립션 프로토콜보다는 계산적으로 효율적이고, 변형된 Zheng의 프로토콜과 비교 했을 때, 제안하는 프로토콜은 160-bit 모듈라 곱셈 계산 횟수는 다소 늘어나지만, 통신 오버헤드부분은 상당히 감소시켰다. 또한 기존의 프로토콜들은 기본 요구사항 중에서 만족 시키지 못한 요구사항들이 있었던 것에 비해서, 제안하는 프로토콜은 기본 요구사항을 모두 만족하기 때문에, 이점을 고려해 보았을 때 기존의 프로토콜들에 비해서 제안하는 프로토콜이 효과적이라고 할 수 있다.

6. 결 론

멀티-사인크립션 프로토콜은 여러 사용자들이 인터넷을 통해서 안전하게 메시지를 송·수신할 수 있도록 사용자 인증과 메시지 기밀성을 효율적으로 제공해주는 프로토콜이다.

본 논문에서는 기존의 프로토콜들을 개선하여, 유연성을 가진 새로운 멀티-사인크립션 프로토콜을 제안하였다. 제안한 프로토콜은 멀티-사인크립션 프로토콜이 만족해야 하는 기본적인 보안 요구사항 및 특성을 모두 만족할 수 있도록 설계되었고, 기존의 프로토콜들 보다 개선된 효율성을 가진다.

따라서 본 프로토콜은 메시지 작성 및 수정자의 인증을 제공함으로써, 건전한 메시지의 유포 및 송·수신이 가능하게 하였고, 메시지 기밀성을 제공함으로써 안전한 메시지 송·수신을 가능하게 하기 때문에 이메일을 비롯한 인터넷 통신 시스템에서 유용하게 사용될 수 있을 것으로 기대된다.

참 고 문 헌

[1] S. Mitomi and A. Miyaji, "A General Model of Multisignature Schemes with Message Flexibility, Order Flexibility, and Order Verifiability," IEICE Transaction on Fundamentals,Vol. E84-A, No. 10, pages 2488-2499, 2001.

[2] X. Pang, B. Catania, and K.-L. Tan, "Securing Your Data in Agent-Based P2P Systems," In Proceedings of 8th International Conference on Database Systems for Advanced Applications (DASFAA '03), pages 55-62, 2003.

[3] Y. Zheng, "Signcryption and Its Applications in Efficient Public Key Solutions," In Proceedings of 1997 Information Security Workshop (ISW'97), LNCS 1397, pages 291-312, Springer-Verlag, 1997.

[4] Y. Zheng, "Digital Signcryption or How to Achieve Cost (Signature & Encryption)<< Cost (Signature) + Cost (Encryption)," Advances in Cryptology-Crypto'97, LNCS 1294, pages 165-179, Springer-Verlag, 1997.

[5] M. Burmester, Yvo Desmedt, H. Doi, M. Mambo, E. Okamoto, M. Tada, and Y. Yoshifuji, "A Structed ElGamal-Type Multisignature Scheme," In Proceedings of PKC 2000, pages 466-482, Springer-Verlag, 2000.

[6] S. Mitomi and A. Miyaji, "A Multisignature Scheme with Message Flexibility, Order Flexibility, and Order Verifiability," In Proceedings of ACISP 2000, LNCS 1841, pages 298-312, Springer-Verlag, 2000.

[7] A. J. Menezes, P. C. Oorschot and S. A. Vanstone, Handbook of Applied Cryptography, CRC, 1997.

서 승 현

2000년 이화여자대학교 수학과 학사. 2002년 이화여자대학교 과학기술대학원 컴퓨터학과 석사. 2002년~현재 이화여자대학교 과학기술대학원 컴퓨터학과 박사과정 관심분야는 정보보호, 암호프로토콜 설계, 홈 네트워크 보안

이 상 호

1979년 서울대학교 계산통계학과 학사 1981년 한국과학기술원 전산학과 석사 1987년 한국과학기술원 전산학과 박사 1983년~현재 이화여자대학교 컴퓨터학과 교수. 관심분야는 알고리즘 설계, 정보보호, 바이오인포매틱스