대리 서명방식에 관한 연구 (I)
- 제1부 : 보증부분 위임에 의한 대리 서명방식 -

김승주, 김준태, 정윤정○, 원동호
성균관대학교 정보공학과

A Study on the Proxy Signatures (I)
- Part 1 : The Proxy Signatures for Partial Delegation with Warrant -

Seungjoo Kim, Joontae Kim, Yoonjung Chung, Dongho Won
Dept. of Information Engineering, Sung-Kyun-Kwan University

요 약

Mambo, Usuda, Okamoto 등에 의하여 최초로 제안된 대리 서명방식은 원서명자가 지정한 대리 서명자가 원서명자를 대신하여 서명하는 것을 허용한다. 본 논문에서는 기존의 대리 서명방식에서 발생하는 대리 서명자의 유효 위임 서명기간과 위임을 한 원서명자의 탈법을 방지할 수 있는 새로운 보증 부분 위임에 의한 대리 서명방식을 제안하였다.

제안한 방식은 유효기간의 정보가 포함된 보증서를 해쉬 함수를 이용하여 대리 서명자에게 위임 정보로 제공하고 다시 대리 서명자는 원서명자로부터 받은 위임 정보에 자신의 정보를 부가하여 대리 서명 생성 정보로 사용토록 하였다.

1. 서 론

암호 시스템의 기능은 정보의 보호기능과 인증기능으로 나눌 수 있다. 정보보호기능은 정보가 노출된다고 해도 키를 알지 못하는 한 정보의 정확한 의미를 파악하지 못하게 하여 정보를 보호하는 것이고, 인증기능은 정보의 전달상태 또는 통신시의 송·수신자간의 상대방 확인기능을 갖춰 분쟁을 해결할 수 있는 기능을 제공하는 기능이다. 인증기능은 자신이 보낸 정보가 전송도중 변경되지 않고 상대방에게 정확하게 전달되었는가를 확신하는 메시지 인증기능과 정보의 생성, 보관, 처리 등의 행위에 관여한 사용자가 맞는가를 확인하는 사용자 인증기능으로 구분할 수 있다. 일상생활에서 우리가 사용하는 서명이나 인감과 같은 효과를 전자적으로 수행하는 디지틀 서명(digital signatures)은 이러한 사용자 인증과 메시지 인증기능을 모두 만족하여야 한다.

일반적으로 서명이나 인감은 개인의 필요성에 의하여 언제든지 발급될 수 있고, 이 서명 또는 인감을 수신한 사람 역시 수신된 서명이나 인감의 정당성을 쉽게 확인할 수 있으며, 서명의 생성자나 인감의 소유자 이외에는 이 서명이나 인감을 발급할 수 없어야 한다. 따라서 디지틀 서명에서도 서명자만이 서명을 생성할 수 있는 유일성, 위조가 불가능한 위조 불가능성, 서명의 진위를 쉽게 확인할 수 있는 진위의 확인의 용이성, 자신의 서명을 위조된 것이라고 거부하는 것이 불가능한 거부의 불가능성 등의 요구 사항을 만족하여야 한다.

이러한 디지틀 서명은 사용목적에 따라 기능이 다른 특수한 서명들이 제안되었다. 본 논문

에서는 M. Mambo와 E. Okamoto가 제안한 부분 위임(partial delegation)에 의한 대리 서명 방식(proxy signatures)[16]에서 위임기간의 불확실성에 따른 대리 서명의 유효기간과 위임을 한 원서명자(original signer)가 대리 서명자(proxy signer)를 가장한 대리 서명을 수행하였을 때에 발생할 수 있는 문제점을 해결할 수 있는 방식을 제안하였다.

대리 서명방식은 원서명자가 자신의 부재중에 자신을 대신해서 서명을 할 수 있는 대리 서명자를 지정하여 대신 서명토록 하는 서명방식이다. 따라서 대리 서명자로 지정 받은 자가 서명한 서명으로부터 원서명자의 대리서명기간등의 지정한 내용을 서명 검증자가 확인할 수 있어야 한다. 또한 대리 서명자임을 인정한다는 내용 등을 확인할 수 있어야 한다.

제안한 방식에서는 부분 위임방식에 유효기간 등의 정보가 포함된 보증서(warrant)를 해쉬 함수(hash function)를 이용하여 대리 서명정보에 포함시킴으로써, 대리 서명방식에 이 사실을 확인시킬 수 있는 방법을 구성하였다.

2. 대리 서명방식

2.1 대리 서명방식의 개념

대리 서명방식(proxy signatures)이란 대리 서명자(proxy signer)로 하여금 원서명자(original signer)를 대신하여 서명을 할 수 있는 서명시스템을 말한다. 대리 서명방식의 조건은 두 가지로 나눌 수 있다. 첫째, 원서명자로부터 지정 받은 사람만 대리 서명을 생성할 수 있어야 하며, 대리 서명자로 지정 받지 못한 제삼자는 대리 서명을 생성할 수 없어야 한다. 또한 대리 서명을 검증하는 사람은 대리 서명으로부터 원서명자가 대리 서명자에게 대리 서명을 위임한 사실을 확인할 수 있어야 한다.

이와 같은 대리 서명방식은 다음과 같은 상황에서 유용하게 사용될 수 있다. 어떤 회사의 간부가 컴퓨터 네트워크가 없는 장소에 출장중, 본인 앞으로 도착한 전자문서에 대한 응답이 필요한 경우 사전에 다른 사람이 그 문서를 받아 답신을 할 수 있도록 해야 할 때 문서의 서명이 문제가 된다. 이때 사전에 대리 서명을 할 수 있도록 조치할 수 있다.

지금까지 제안된 대리 서명방식의 종류는 완전 위임(full delegation), 부분 위임(partial delegation) 그리고 보증 위임(delegation with warrant) 등 세 가지 방식이 있다.

1) 완전 위임방식

원서명자가 자신의 비밀 서명정보 X_A를 대리 서명자에게 알려주어 대리 서명자로 하여금 대리 서명하게 하는 방법으로 서명결과는 원서명자의 서명과 동일하게 된다. 이 경우는 원서명자의 비밀 서명정보 X_A의 관리에 문제점이 생기게 되고, 대리 서명자의 절대적인 신뢰를 전제로 하지 않고는 사용이 불가능하다. 즉 대리 서명자의 부당한 X_A의 사용과 노출이 염려되는 방법으로 매우 제한적인 방식이다.

2) 부분 위임방식

부분 위임방식에서는 원서명자가 자신의 비밀 서명정보 X_A로부터 새로운 비밀 서명정보 σ를 생성하여 비밀리에 대리 서명자에게 전달한다. 그러면 대리 서명자는 원서명자로부터 비밀리에 제공받은 σ를 자신의 대리 서명의 비밀 서명정보로 사용하게 된다. 물론 대리 서명을 생성하는 비밀 서명정보 σ로부터 원서명자의 비밀 서명정보 X_A를 구할 수 없어야 한다. 즉 원서명자는 완전 위임방식에서 발생하는 자신의 비밀 서명정보 X_A의 노출 위험성에 대해 안전하다.[14][15][16]

부분 위임방식은 다음 두 가지로 나눌 수 있다.

① 대리인 비보호형 대리 서명방식 (proxy-unprotected proxy signatures)

대리 서명자는 원서명자를 대신해서 대리 서명을 생성할 수 있으나, 대리 서명자 이외에 원서명자 또한 정당한 대리 서명자를 가장하여 대리 서명을 생성할 수 있다. 그러나, 대리 서명자로 지정 받지 않은 제삼자는 대리 서명을 생성할 수 있다.

② 대리인 보호형 대리 서명방식 (proxy-protected proxy signatures)

정당한 대리 서명자만이 대리 서명이 가능하다. 따라서, 제삼자뿐만 아니라 원서명자 또한 정당한 대리 서명자를 가장하여 대리 서명을 생성할 수 없다.

3) 보증 위임방식

보증 위임방식은 원서명자가 문서로 대리 서명자임을 밝히는 방법으로 두 가지 방식이 있다.[12][13]

① 보증서 기반 대리 서명방식 (delegation proxy)

원서명자가 지정한 사람을 대리 서명자로 선언하는 서류에 일반적인 디지틀 서명을 통하여 서명한 후, 그 서명된 보증서(warrant)를 대리 서명자에게 전달한 다음 대리 서명자로 하여금 그 사실을 전제로 대리 서명을 시행토록 하는 방법이다.

② 소지자 기반 대리 서명방식 (bearer proxy)

원서명자는 대리 서명자가 사용할 새로운 비밀 서명정보와 그에 대한 공개 검증정보를 생성하고, 생성된 공개 검증정보에 대한 서명을 하여 대리 서명자에게 준다. 이때 생성된 비밀키는 대리 서명자에게 비밀리에 전달된다.

대리 서명방식은 앞에서 설명한 세 방식을 이용하여 서명을 한다. 각자의 특징을 보면 완전 위임방식의 경우 원서명자 대리 서명자에게 직접 자신의 비밀 서명정보 X_A를 전달해 주므로해서 원서명자의 비밀 서명정보가 일반인에게 노출될 염려가 있을 뿐만 아니라 대리 서명자 선정에도 제한적이다. 즉, 대리 서명자로 선정된 사람의 X_A에 대한 비밀유지가 문제가 된다.

또한 보증 위임방식의 경우 보증서를 검증하는 추가적인 계산량과 전송량의 증가가 초래되므로 실용성 측면의 문제가 있으나, 보증서가 있어 대리 서명자의 통제기능이 가능하다.

부분 위임방식은 완전 위임방식에서와 같은 원서명자의 비밀 정보 누설을 방지할 수 있으며, 또한 보증서를 검증하는 과정이 별도로 요구되는 것이 아니라 대리 서명 검증시에 함께 검증 가능함으로 보증 위임방식에 비해 효율적이다. 그러나 보증서가 없어 대리 서명자로의 유효기간 등이 문제가 될 수 있다.

따라서 본 논문에서는 부분 위임방식의 장점과 보증 위임방식의 장점만을 취한 보증 부분 위임방식(partial delegation with warrant)을 제안한다. 부분 위임의 경우에는 대리 서명자가 대리 서명을 할 수 있는 기간을 명시할 수 없기 때문에 대리인을 철회하고자 하는 경우에 별도의 대리 서명 철회 과정(proxy revocation protocol)이 요구되나, 제안된 방식에서는 대리 서명자가 사용할 비밀 서명정보에 대리 서명을 할 수 있는 기간을 명시할 수 있으므로 이러한 과정이 필요 없다. 또한, 보증 위임에서와 같이 보증서를 검증하는 과정이 별도로 요구되는 것이 아니라 대리 서명 검증시에 보증서도 함께 검증 가능함으로 부분 위임과 같은 효율성을 갖는다.

2.2 Mambo의 부분 위임에 의한 대리 서명방식

M. Mambo와 E. Okamoto는 이산대수 문제를 이용하여 부분 위임에 의한 대리 서명방식을 제안하였다. 이 방식은 앞에서 설명한 바와 같이 원서명자의 비밀 서명정보 X_A를 포함시킨 새로운 대리 서명자의 비밀 서명정보를 생성시켜 비밀리에 대리 서명자에게 전달한다. 이 방식의 순서는 다음과 같다.

[초기화]

- p : 큰 소수 $p > 2^{512}$
- g : 원시 원소 $g \in Z_p$
- X_A : 원서명자의 비밀 서명정보 $X_A \in Z_p$
- Y_A : 원서명자의 공개 검증정보 $Y_A = g^{X_A} \pmod p$
- Sign(·) : 일반적인 디지틀 서명방식

[프로토콜]

단계 1) (대리 서명용 키 생성) 원서명자는 다음을 계산한다.

$$k \in_R Z_p$$
$$K = g^k \pmod p$$
$$\sigma = X_A + kK \pmod{p-1}$$

단계 2) (대리 서명용 키의 분배) 원서명자는 자신이 계산한 대리 서명자의 비밀 서명정보 σ를 비밀리에 K와 같이 대리 서명자에게 전달한다.

단계 3) (대리 서명용 키의 검증) 대리 서명자로 지정 받은 사람은 원서명자가 생성한 대리 서명을 위한 비밀 서명정보 σ의 정당성을 원서명자의 공개 검증정보 Y_A를 이용하여 확인한다.

$$g^{\sigma} \stackrel{?}{=} Y_A K^K \pmod p$$

단계 4) (대리 서명자에 의한 서명) 대리 서명자는 원서명자가 생성하여 비밀리에 제공한 비밀 서명정보 σ를 이용하여 일반적인 디지틀 서명방식을 이용하여 문서 m의 대리 서명을 생성한다.

$$(m, \mathrm{Sign}_{\sigma}(m), K)$$

단계 5) (대리 서명의 검증) 대리 서명자의 새로운 공개 검증 정보는 $V = Y_A K^K \pmod p$가 된다. 대리 서명을 검증하려는 사람은 V를 계산하는 과정에 Y_A가 포함되어 있으므로 원서명자의 위임 사실을 인지하게 되며 대리 서명의 검증은 일반적인 디지틀 서명의 검증 순서를 따르게 된다.

3. 보증 부분 위임에 의한 대리 서명방식의 제안

본 절에서는 대리 서명자에게 제공하는 대리 서명용 키의 공개키와 대리 서명용 키의 유효기간을 포함하는 메시지에 원서명자가 서명함으로서 만들어진 보증서를 사용하여 부분 위임을 실현시키는 방법을 제안하고자 한다. 제안되는 방식을 대리인 비보호 방식과 대리인 보호

방식으로 나뉘어 설명하고자 한다.

- 보증 부분 위임방식

보증 부분 위임이란 원서명자가 대리 서명용 비밀정보 σ를 자신의 비밀 서명정보 X_A와 유효기간과 대리 서명자와의 관계 등이 언급된 보증서 m_w를 이용하여 생성하는 경우를 말한다. 이때 원서명자의 비밀 서명정보 X_A는 σ와 m_w로부터 계산 불가능하여야 한다. 보증 부분 위임의 형태는 다음과 같은 두 가지 형태로 분류된다.

① 대리인 비보호형 대리 서명방식 (proxy-unprotected proxy signatures)

대리 서명자는 원서명자를 대신해서 대리 서명을 생성할 수 있으나, 대리 서명자 이외에 원서명자 또한 정당한 대리 서명자를 가장하여 대리 서명을 생성할 수 있다. 그러나, 대리 서명자로 지정 받지 않은 제삼자는 대리 서명을 생성할 수 있다.

② 대리인 보호형 대리 서명방식 (proxy-protected proxy signatures)

정당한 대리 서명자만이 대리 서명이 가능하다. 따라서, 제삼자뿐만 아니라 원서명자 또한 정당한 대리 서명자를 가장하여 대리 서명을 생성할 수 없다.

제안하는 보증 부분 위임에 의한 대리 서명방식은 근본적으로 Mambo의 대리 서명방식과 기능은 유사하나, 보증서 기능을 추가하기 위해 공개 해쉬함수를 이용하였다. 대리 서명과 관련된 유효정보나 대리 서명자로 지정된 사실 등을 m_w으로 표시하여 $e = h(m_w, K)$를 계산한다.

한편 일반적으로 대리 서명자는 원서명자가 항상 정직하다는 전제로 대리 서명을 시행하게 된다. 그러나, 예를 들어, 원서명자가 대리 서명을 만들어 대리 서명자를 곤란하게 만들 경우가 있다. 이러한 경우를 방지하기 위해 대리 서명자 자신도 자신을 보호할 수 있는 기능이 있어야 한다. 앞에서 설명한 바와 같이 대리 서명자가 자신을 보호하기 위한 방식을 대리인 보호형 대리 서명방식, 그렇지 못한 경우를 대리인 비보호형 대리 서명방식이라 한다. 이 두 방식을 구별해서 본 논문에서는 제안을 하였다.

[초기화]

Mambo의 초기화 설정과정과 동일하다.

[프로토콜 I (대리인 비보호형 대리 서명방식)]

1. (대리 서명용 키 생성) 원서명자는 대리 서명자에게 다음과 같은 서명용 키 σ를 생성한다. 이때, m_w에는 원서명자의 ID, 대리 서명자의 ID, 위임 기간 등이 명시된다. 따라서, σ는 m_w와 K에 대한 원서명자의 서명이다 (단, $h(\cdot)$는 안전한 해쉬 함수이다). 이제 (m_w, σ, K)를 대리 서명자에게 비밀리에 전달한다.

$$k \in Z_{p-1} - \{0\},\ K = g^k \pmod p$$
$$e = h(m_w, K),\ \sigma = X_A + k \pmod{p-1}$$

2. (대리 서명용 키의 검증) 대리 서명자는 자신이 받은 (m_w, σ, K)로부터 이 m_w, K에 대한 원서명자의 서명임을 확인한다.

$$e = h(m_w, K),\ g^{\sigma} \equiv Y_A^{e} K \pmod p$$

3. (대리 서명자에 의한 서명) 대리 서명자는 비밀키 σ를 사용하여 임의의 메시지 m에 대한

ElGamal 형태의 서명 $Sign_\sigma(m)$을 생성하고 (m, $Sign_\sigma(m)$, m_w, K)를 서명 수신자에게 전달한다.

4. (대리 서명의 검증) 서명 수신자는 먼저 $e = h(m_w, K)$와 $V' = Y_A^e K \pmod p$를 계산한다. $V' = g^\sigma \pmod p$이므로 V'을 사용하여 서명 $Sign_\sigma(m)$을 검증할 수 있다. 이때 검증 방법은 ElGamal 형태의 서명방식에 의한다. 관계식 $V' = Y_A^e K \pmod p$는 원서명자가 V'에 대응되는 σ를 만들어서 대리인에게 주었음을 의미한다. 왜냐하면 대리인은 이러한 관계식을 만족시키는 σ를 만들 수 없기 때문이다. 따라서, 서명 $Sign_\sigma(m)$이 원서명자를 대신한 대리 서명임을 확인할 수 있다.

논문[14]에서 Usuda는 보증서 m_w를 사용하지 않고 관계식 $g^\sigma = Y_A^{h(K)} K \pmod p$에 의하여 부분 위임에 의한 대리 서명을 제안하였다. 따라서, 이 경우에는 대리인 위임 기간이 지날 경우 원서명자는 대리인 철회를 위한 별도의 철회 과정을 수행하여야 한다.

대리인 비보호형에서는 원서명자가 대리인을 가장하여 대리 서명을 할 수 있다. 따라서, 제3자는 원서명자가 서명한 것을 대리 서명자가 서명한 것으로 오인할 수 있다. 그러므로 대리 서명자를 보호하기 위한 방안이 요구된다. 대리인 보호형 대리 서명방식에서는 이러한 문제점을 해결할 수 있다. 다음의 프로토콜에서 X_p는 대리 서명자의 비밀키이고 $Y_p = g^{X_p} \pmod p$는 대리 서명자의 공개키이다.

[프로토콜 II (대리인 보호형 대리 서명방식)]

지금까지 설명한 대리 서명방식은 원서명자의 신뢰성을 전제로 하는 방식이다. 그러나 원서명자의 정직하지 못한 행동 즉, 원서명자가 대리 서명자를 가장한 서명은 대리 서명자를 곤란하게 만들 수 있다.

1. (원서명자에 의한 대리 서명용 키 생성) 대리인 비보호형 프로토콜과 동일하다.

$$k \in Z_{p-1} - \{0\},\ K = g^k \pmod p$$
$$e = h(m_w, K),\ \sigma = X_A + k \pmod{p-1}$$

2. (키의 검증) 대리인 비보호형 프로토콜과 동일하다.

$$e = h(m_w, K),\ g^\sigma = Y_A^e K \pmod p$$

3. (대리 서명키의 변환) 대리 서명자는 원서명자가 서명용 키를 알 수 없도록 다음과 같이 변환하여 대리 서명용 키 σ'를 생성한다.

$$\sigma' = \sigma + X_p\, h(m_w, K) \pmod{p-1}$$

4. (대리 서명자에 의한 서명) 대리 서명자는 비밀키 σ'를 사용하여 임의의 메시지 m에 대한 ElGamal 형태의 서명 $Sign_{\sigma'}(m)$을 생성하고 (m, $Sign_{\sigma'}(m)$, m_w, K)를 서명 수신자에게 전달한다.

5. (대리 서명의 검증) 서명 수신자는 먼저 $e = h(m_w, K)$와 $V' = (Y_A Y_p)^e K \pmod p$를 계산한다. $V' = g^\sigma \pmod p$이므로 V'을 사용하여 서명 $Sign_{\sigma'}(m)$을 검증할 수 있다. 이때 검증 방법은 ElGamal 형태의 서명방식에 의한다. 관계식 $V' = (Y_A Y_p)^e K \pmod p$에서는 원서명자의 공개키 Y_A 뿐 아니라 대리 서명자의 공개키 Y_p로 사용되므로 이러한 관계식을 만족하는 V'에 대응되는 σ'을 원서명자 또는 대리 서명자 혼자서는 만들 수 없음을 의미하며, 이러한 관계식을 만족시키는 σ'은 둘의 협조로 만들었음을 확인할 수 있다. 따

라서, 서명 $Sign_{\sigma}(m)$이 원서명자를 대리하여 대리인이 대리 서명한 것임을 확인할 수 있다.

4. 결 론

본 논문에서는 기존의 대리 서명방식에서 필요로 하는 보증서 기능을 포함시킨 보증 부분 위임에 의한 대리 서명방식을 제안하였다. 제안한 보증서 기능을 갖는 부분 위임 방식의 대리 서명은 기존의 보증 위임에 의한 대리 서명방식보다 계산량이 적으며 보통의 부분 위임에 의한 대리 서명방식보다 구조적으로 우수하다.

제안된 보증 부분 위임에 의한 대리 서명방식은 보증서 기능을 갖고 있어 위임을 한 원서명자의 위임기간 등의 유효정보를 설정할 수 있어 응용범위가 클 것으로 사료된다.

참 고 문 헌

[1] W. Diffie and M. Hellman, "New directions in cryptography," IEEE. Trans. on Information Theory IT-22, pp.644-654, 1976.

[2] R.L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystem," communication of ACM vol 21, no.2, pp.120-126, 1978.

[3] T. ElGamal, "A public key cryptosystem and signature scheme based in discrete logarithm," IEEE. Trans. on Information Theory vol.31 no.4, pp.469-472, 1995.

[4] Schnorr, C., "Efficient signature generation by smart cards," Journal of Cryptology, vol. 4, no.3, pp.161-174, 1991.

[5] NIST FIPS PUB XX. "Digital Signature Standard," National Institute of Standards and Technology," U.S Department of commerce. Draft. 1 Feb. 1993.

[6] T. Okamoto, "Provably secure and practical identification schemes and corresponding signature schemes," Proc. Crypto'92, pp.31-53, 1993.

[7] Kaisa Nyberg and Rainer A. Rueppel, "Message Recovery for signature schemes based on the discrete logarithm problem," Eurocrypt'94, pp.175-190, 1994.

[8] D. Chaum and H. Antwerpen, "Undeniable signature," Crypto'89, pp.212-216.

[9] S.J. Park, K.H. Lee and D.H. Won "An entrusted undeniable signature" JW-ISC'95, pp.2.1-2.7, 1995.

[10] S.J. Kim, S.J. Park, and D.H. Won, "Nominative signature," Proc. of ICEIC'95, International Conference on Electronics, Informations and Communications, pp.II-68~II-71, 1995.

[11] S.J. Kim, S.J. Park, and D.H. Won "Zero-knowledge nominative signature," Pragocrypt'96, International Conference on the Theory and Applications of Cryptology, pp.38-392, 1996.

[12] V. Varadharajan, P. Allen, and S. Black, "An analysis of the proxy problem in distributed systems," IEEE. Computer Society Symposium on Research in Security and Privacy, pp.255-275, 1991.

[13] B.C. Neuman, "Proxy-based authentication and accounting for distributed system" Proc. of 13th International Conference on Distributed Computing System. pp.283-291, 1993

[14] K. Usuda, M. Mambo, T. Uyematsu, and E. Okamoto, "Proposal of an automatic signature scheme using compiler, " IEICE Trans. Fundamentals, vol.E79-A, no.1, pp.94-101, 1996.

[15] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signature for delegating signing operation", Proc. of third ACM conference on computer and communication security, pp.48-57, 1996.

[16] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signature : Delegation of the power to sign message," IEICE Trans. Fundamentals, vo.E79-A, no.9, 1996.

[17] S.J. Kim, S.J. Park and D.H. Won, "Proxy signatures, revisited," Proc. of ICICS'97, International Conference on Information and Communications Security, Springer, Lecture Notes in Computer Science, LNCS, 1997.

[18] B.S. Kaliski, "A response to DSS," Nov. 1991.