식별번호를 이용한 본인확인 기술규격

Subscriber Identification Based on Virtual ID

v1.21

2009년 9월

한국인터넷진흥원
Korea Internet & Security Agency

목 차

1. 개 요

본 기술규격은 인터넷 상에서의 공인인증서를 이용한 본인확인 시 주민등록번호 및 사업자 등록번호를 이용하는 경우에 해당 정보의 노출 없이 안전하게 관련 정보를 생성하고 검증하기 위한 절차를 명시한다.

2. 규격의 구성 및 범위

본 규격에서는 식별번호와 관련된 정보를 공인인증서에 넣어서 사용하는데 필요한 문법 및 절차에 대해서 크게 3부분으로 나누어 기술한다.

첫 번째로 6장에서는 가입자가 공인인증서를 발급 받는 과정에서 공인인증기관이 식별번호 정보를 공인인증서에 주입하는 전체적인 절차를 기술하고 있다.

두 번째로 7장에서는 식별번호 관련 정보를 생성하고 처리하기 위한 속성을 정의하고 해당 필드들에 대한 설명을 기술하고 있다.

세 번째로 8장에서는 식별번호 관련 정보를 인증서에 주입할 때 어느 곳에 어떤 형태로 주입될 것인가에 대한 사항과 공인인증서 요청시 암호화된 식별번호를 어디에 담을 것인가에 대하여 기술하고 있다.

3. 관련 표준

3.1 국외 참고권고 및 표준

[X509]	ITU-T Recommendation X.509 (1997) | ISO/IEC 9594-8:1998, *Information technology - Open Systems INterconnection - The Directory: Authentication Frame work*
[RFC2459]	IETF RFC 2459 (1999), *Internet X.509 Public Key Infrastructure: Certificate and CRL Profile*
[RFC3280]	IETF RFC 3280 (2002), Internet X.509 Public Key Infrastructure: Certificate and CRL Profile

[RFC2510] IETF RFC 2510 (1999), *Internet X.509 Public Key Infrastructure: Certificate Management Protocols*

[RFC2511] IETF RFC 2511 (1999), *Internet X.509 Certificate Request* Message Format

[PKCS#5v1.5] RSA Laboratories PKCS#5 v1.5 (1993), Password-Based Cryptography Standard

[PKCS#5v2.0] RSA Laboratories PKCS#5 v2.0 (1999), Password-Based Cryptography Standard

[PKCS#8] RSA Laboratories PKCS#8 v1.2 (1993), Private Key Information Syntax Standard

[PKCS#10] RSA Laboratories PKCS#10 v1.7 (2000), *Certification Request Syntax Standard*

[PKCS#11] RSA Laboratories PKCS#11 v2.11 (2001), *Cryptographic Token Interface Standard Revision 1*

3.2 국내 표준

[TTA-120029] TTAS.KO-12.0029, *식별번호를 이용한 본인확인 기술, 2005*

[KCAC.TS.CERTPROF] KISA, KCAC.TS.CERTPROF, v1.70, *전자서명 인증서 프로파일 규격,* 2009

[KCAC.TS.UI] KISA, KCAC.TS.UI v1.80, *공인인증기관간 상호연동을 위한 사용자 인터페이스 기술규격,* 2009

3.3 기타

해당사항 없음

4. 정의

4.1 전자서명법 용어 정의

본 규격에서 사용된 다음의 용어들은 전자서명법 및 동법 시행령, 공인인증

기관의 시설 및 장비 등에 관한 규정(행정안전부 고시)에 정의되어 있다.

가) 전자서명키, 전자서명생성키, 전자서명검증키
나) 인증서
다) 공인인증서
라) 공인인증기관
마) 전자서명인증관리체계
바) 가입자
사) 이용자
아) 가입자 소프트웨어
자) 키분배용 인증서
차) 키분배용개인키

본 규격에서는 다음의 용어들에 대하여 추가적으로 정의한다.

가) 식별번호 : 전자서명법 시행규칙 제13조의2에 의한 주민등록번호, 사업자등록번호 및 고유번호

5. 약어 및 기호

본 규격에서 사용하는 다음의 약어들은 [TTA-120012]의 정의를 준용한다.

가) CA
나) OID
다) DN

추가적으로 본 규격에서 사용하는 다음의 기호들에 대해 의미를 정의한다.

1) h( ) : 해쉬 함수
2) R : 비트열 난수
3) E( ) : 암호화 함수
4) IDN : 식별번호
5) pk : 전자서명검증키

6) sk : 전자서명생성키
7) VID : 가상식별번호
8) EVID : 암호화된 가상식별번호

6. 식별번호 주입 절차

가입자의 식별번호 관련 정보를 공인인증기관이 주입하기 위해서는 가입자가 신원확인 절차를 거친 후에 공인인증기관에게 전자서명검증키에 대한 공인인증서를 요청하는 단계에 이루어진다.

가입자는 전자서명검증키에 대한 인증 요청을 수행하기 위해서 가입자의 전자서명검증키, DN 및 식별번호 관련 정보와 함께 가입자의 전자서명생성키로 전자서명하여 공인인증기관에게 전송한다. 공인인증기관은 이에 대한 전자서명 검증 및 식별번호 관련 정보 검증을 통하여 전자서명키 검증 및 식별번호의 정당성을 확인하고 공인인증서 발행 시점에 식별번호 관련 정보를 가입자의 공인인증서에 주입한다.

6.1 공인인증기관 인증서 획득

가입자가 공인인증기관에게 전달하는 데이터의 기밀성 확보를 위하여 암호화를 수행하며 이를 위하여 공인인증기관의 키분배용 인증서를 미리 획득하여야 한다. 공인인증서를 획득하는 방법은 본 규격에서 다루지 않는다.

6.2 가입자 전자서명키 생성

가입자가 사용할 전자서명키를 생성하며 생성된 전자서명키 중에서 전자서명검증키 pk는 공인인증서의 형태로 저장되고 전자서명생성키 sk는 암호화되어 별도의 저장장치에 안전하게 저장된다.

6.3 난수 생성

본 규격에서는 적어도 160비트 이상의 안전한 임의의 난수를 생성하여 사용해야 한다.

난수는 가상 식별번호를 생성하는데 사용되며 가상 식별번호와 함께 공인인증기관에 전달되어 공인인증기관이 식별번호를 검증하는데 사용된다.

6.4 난수 저장

생성된 난수는 안전한 저장매체에 표준화된 방법으로 저장한다. 자세한 저장형식 및 방법에 대해서는 8장 8.1에서 기술한다.

6.5 가상 식별정보 생성

실제로 공인인증서에 포함될 식별번호 관련 정보를 다음과 같이 계산한다.

VID = h(h(IDN, R))

여기서 사용되는 해쉬 함수는 모두 동일한 해쉬함수를 사용한다.

6.6 검증 정보 생성 및 암호화

VID 정보만으로는 공인인증기관이 VID 정보의 진위를 파악할 수 없으므로 가입자는 공인인증기관에게 정보제공시 VID 생성에 중요한 데이터인 R값을 함께 전송한다.

R값은 가입자를 식별할 수 있는 결정적인 값이므로 이에 대한 기밀성 유지가 중요하기 때문에 VID 정보와 R정보를 함께 암호화하여 인증 요청 메시지에 담아 공인인증기관에게 전송한다.

EVID = E(VID, R)

여기서 암호화에 사용되는 알고리즘 및 공개키는 공인인증기관의 키분배용 인증서에서 추출하여 사용한다.

6.7 공인인증서 요청 메시지 생성

공인인증서 요청 메시지에는 다음의 정보가 포함되도록 생성한다.

1) 가입자 DN
2) 가입자 전자서명검증키
3) EVID

인증 요청 메시지에 EVID를 포함하기 위해서는 7.2에서 표현하는 속성에 따라 구현하여 저장한다.

6.8 공인인증서 요청 메시지 전송

생성된 전자서명검증키의 공인인증서 요청 메시지는 별도의 온라인 또는 오프라인 방식으로 공인인증기관에게 전달된다. 온라인의 경우, 공인인증기관에게 전달하는 프로토콜은 본 표준에서 다루지 않는다.

6.9 공인인증서 요청 메시지 처리

가입자로부터 전달된 공인인증서 요청 메시지를 수신한 공인인증기관은 해당 인증요청서에 대하여 가입자의 전자서명검증키와 합치되는 전자서명생성키를 소유하고 있는지의 여부를 확인한다.

6.10 메시지 복호화 및 VID 검증

인증 요청 메시지로부터 EVID 정보를 추출하여 공인인증기관의 키분배용개인키로 복호화하여 VID 정보와 R정보를 추출한다. 공인인증기관은 다음을 계산하여 계산된 결과와 추출한 VID가 동일한 지의 여부를 확인한다.

$$VID' = h(h(IDN, R))$$

여기서 가입자 식별번호 IDN은 공인인증기관이 미리 소유하고 있어야 하며 일반적으로 가입자 신원확인 단계에서 공인인증기관이 소유하게 된다.[v1.11, 2002. 9. 2. 개정]

6.11 공인인증서 내에 VID 주입

공인인증서의 확장필드 중 subjectAltName 부분에 8.3에서 기술하는 형식에 따라 구성하여 주입한다.

6.12 공인인증서 전송

공인인증기관은 공인인증서를 생성하여 가입자에게 온라인 또는 오프라인으로 전송한다.

7. 식별번호 관련 정보의 ASN.1 표현

7.1 가상 식별번호 속성

```
id-VID OBJECT IDENTIFIER ::= { id-kisa-identifyData 1 }
VID ::= SEQUENCE {
        hashAlg                 HashAlgorithm,
        virtualID       [0]     OCTET STRING }

HashAlgorithm ::= AlgorithmIdentifier

HashContent ::= SEQUENCE {
        idn             PrintableString,
        randomNum       BIT STRING
}
```

VID 구조에서 사용되는 구성요소는 다음과 같다.

- hashAlg은 VID를 생성하는데 사용된 해쉬 알고리즘 및 파라미터를 나타내고 있으며 알고리즘 파라미터는 해당 알고리즘에 따른다.
- virtualID은 6장 6.5에서 기술한 방식에 따라 계산된 값으로 DER 코딩된 HashContent값을 2회 해쉬하여 계산한다.

HashContent 구조에서 사용되는 구성요소는 다음과 같다.

- idn는 가입자 식별번호로서 일반적으로 주민등록번호 및 사업자 등록번호 등이 될 수 있다. 가입자 식별번호는 '-' 등과 같은 구분자를 제거한 상태의 숫자열로만 구성되며 PrintableString으로 표현한다.
- randomNum는 160비트 이상의 길이를 가지는 안전한 임의의 난수이다.

7.2 암호화된 식별번호 속성

```
id-EncryptedVID OBJECT IDENTIFIER ::= { id-kisa-identifyData 2 }
EncryptedVID ::= SEQUENCE {
        version         [0]     INTEGER DEFAULT 0,
        vidHashAlg      [1]     VIDHashAlgorithm OPTIONAL,
        vidEncAlg       [2]     VIDEncryptionAlgorithm,
        certID          [3]     IssuerAndSerialNumber,
        encryptedVID    [4]     OCTET STRING }

VIDHashAlgorithm ::= AlgorithmIdentifier

VIDEncryptionAlgorithm ::= AlgorithmIdentifier

IssuerAndSerialNumber ::= SEQUENCE {
        issuer          Name,
        serialNumber    CertificateSerialNumber }

EncryptContent ::= SEQUENCE {
        vid             VID,
        randomNum       BIT STRING }
```

EncryptedVID 구조에서 사용되는 구성요소는 다음과 같다.

- version은 본 표준에 대한 버전 정보로서 본 규격을 준용하는 경우에는 v1(0)값을 사용한다.
- vidHashAlg는 VID를 생성하는데 사용된 해쉬 알고리즘 및 파라미터를 나타내며 알고리즘 파라미터는 해당 알고리즘에 따른다.
- vidEncAlg는 VID를 암호화하는데 사용된 비대칭 암호알고리즘 및 파라미터를 나타내며 이는 공인인증기관의 인증서에 포함되어 있는 알고리즘과 동일한 알고리즘이어야 한다.

- certID는 VID를 암호화하는데 사용된 공인인증기관의 인증서 식별자로 공인인증기관 인증서의 발급자와 공인인증기관 인증서의 일련번호로 구성된다.
- encryptedVID는 DER 인코딩된 EncryptContent 값을 공인인증기관의 공개키로 암호화한 결과를 나타낸다.

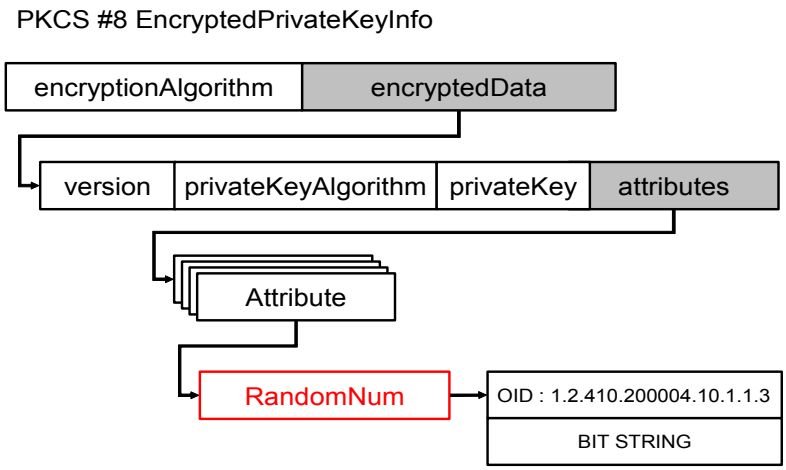
8. 식별번호 관련 정보 주입 위치

8.1 난수 정보 주입 위치

본 규격에서는 난수 정보를 전자서명생성키와 함께 [PKCS#5v2.0] 및 [PKCS#8] 표준을 준용하여 안전하게 저장하는 것을 기본으로 한다.[v1.11 2002. 9. 2 개정]

다음은 [PKCS#5v2.0] 및 [PKCS#8] 형식에 난수 정보를 포함시키는 구조도를 나타낸 것이다.

예제 1) PKCS#5 및 PKCS#8 내에 난수 정보 주입



[v1.11, 2002. 9. 2 개정]

단, 스마트카드에 난수 정보를 저장하는 경우는 [KCAC.TS.UI] 부록 3. 스마트카드 파일 구성도 및 메모리 맵을 준용하여야 한다. 이 때 식별번호 생성용 난수 정보 및 서명용개인키의 저장 방식은 다음과 같다.

- 스마트카드가 메모리 맵 F306을 지원하지 않는 경우 전자서명생성키는 [PKCS#5v1.5]를 적용하여 메모리 맵 F303에 저장하고 식별번호 생성용 난수 정보는 [TTA.120029]를 적용하여 F305에 저장한다.
- 스마트카드가 메모리 맵 F306을 지원하는 경우 식별번호 생성용 난수 정보는 전자서명생성키와 함께 [PKCS#5v2.0] 및 [PKCS#8] 표준을 준용하여 F303에 저장한다.

또한 PKCS#11 호환 암호토큰을 저장매체로 사용하는 경우 난수의 저장을 위해서는 다음과 같은 데이터 오브젝트를 정의해야 한다.

속성	데이터 타입	권고값
CKA_CLASS	CK_OBJECT_CLASS	CKO_DATA
CKA_TOKEN	CK_BBOOL	FALSE(default)
CKA_PRIVATE	CK_BBOOL	TRUE
CKA_MODIFIABLE	CK_BBOOL	FALSE
CKA_LABEL	Local string	Random number for VID
CKA_APPLICATION	Local string	Licensed PKI Application
CKA_VALUE	Byte array	R (난수)

CKA_CLASS는 오브젝트의 유형을 나타내는 것으로 반드시 CKO_DATA값을 사용해야 한다. CKA_TOKEN은 토큰 오브젝트 또는 세션 오브젝트를 나타내는 것이며 CKA_PRIVATE은 암호토큰에 사용자 인증을 거쳐야만 오브젝트에 접근 가능함을 표현하는 것으로 본 규격에서는 TRUE값을 가진다.

다음으로 난수 정보는 처음 생성된 후 수정되지 말아야 하므로 CKA_MODIFIABLE의 값은 FALSE이다. 저장된 난수 정보에 접근하기 위한 식별자로 CKA_LABEL이 사용되며 CKA_APPLICATION은 난수 정보를 필요로 하는 가입자 소프트웨어명 등을 나타낸다. 마지막으로 실제 난수 정보를 포함하는 속성은 CKA_VALUE이다.

8.2 공인인증서 요청 시 암호화된 식별번호 주입 위치

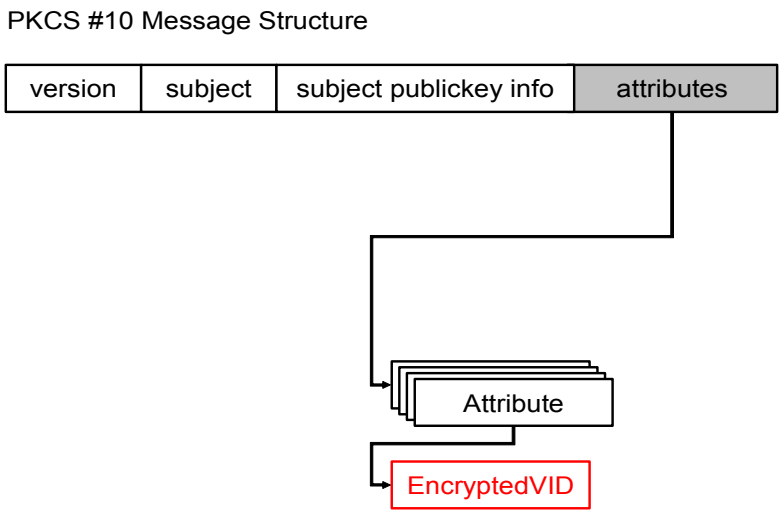
가입자는 공인인증서 요청 메시지에 식별번호 관련 정보를 포함하여 공인인증기관에게 전송한다.

공인인증서 요청 메시지에 식별번호 관련 정보를 포함하기 위해서는 7.2에서 정의하고 있는 암호화된 식별정보 id-encryptedVID 속성을 이용한다.

공인인증서 요청 메시지에는 해당 속성을 포함할 수 있는 구조를 이미 가지고 있어야 하며 추가되는 속성을 지원할 수 있는 유연성을 가지고 있어야 한다.

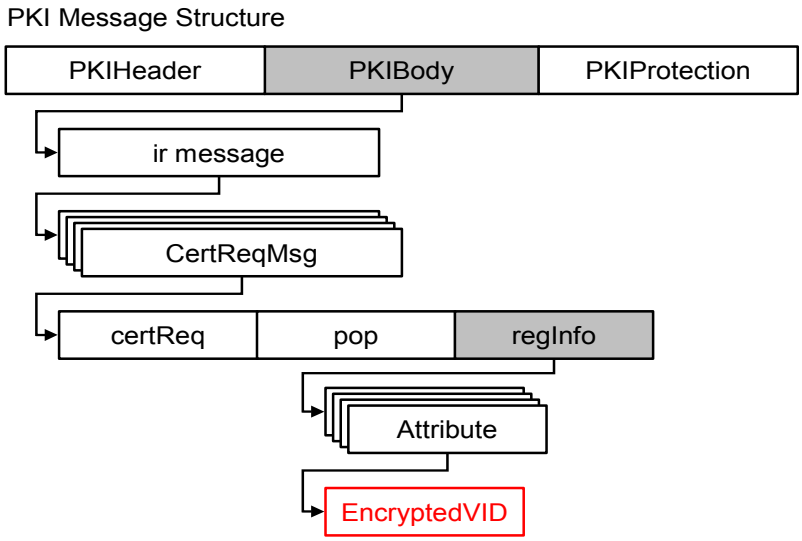
본 규격에서는 공인인증서 요청 메시지에 id-encryptedVID 속성을 포함시키는 것에 대한 예제를 다음에서 보인다.

예제 2) PKCS #10 내에 식별번호 관련 정보 주입



PKCS#10은 기본적으로 가입자 DN 및 가입자 전자서명검증키 정보 등으로 구성되지만, 추가적인 정보 입력을 위하여 attributes 구성요소를 포함하고 있다. attributes 구성요소는 OID와 구체적인 구조를 가지는 모든 속성들을 포함시킬 수 있으므로 7.2에서 정의한 EncryptedVID에 대한 OID id-encryptedVID를 이용하여 해당 내용을 포함시킬 수 있다.

예제 3) RFC 2510/RFC 2511 내에 식별번호 관련 정보 주입



RFC 2510은 기본적으로 공인인증서 발급요청, 갱신, 폐지 및 키복구 등과 같은 공인인증서 관리에 관련된 프로토콜을 정의하고 있으며 이를 위하여 공개키 기반구조에서 사용되는 메시지 형식을 정의한다.

가입자는 공인인증서 요청을 위하여 ir 메시지를 이용하여 해당 메시지는 RFC 2511에서 정의하는 메시지 구조를 가지고 있으며 이중에서 regInfo 부분에서 가입자에 대한 부가 정보를 입력할 수 있다.

regInfo 구성요소는 하나 이상의 Attribute 형태의 값들을 포함할 수 있으며 EncryptedVID 값을 OID와 함께 포함시킬 수 있다.

8.3 공인인증서 내 식별번호 주입 위치

공인인증서 내에서 식별번호를 주입하기 위해서는 식별번호 정보를 위하여 소유자 대체명칭 확장필드에 id-kisa-identifyData 명칭 형식을 이용하여 저장한다.

```
id-kisa-identifyData OBJECT IDENTIFIER  ::=  { id-attribute 1 }
IdentifyData ::= SEQUENCE {
        realName        UTF8String,
        userInfo        SEQUENCE SIZE (1..MAX) OF
                        AttributeTypeAndValue OPTIONAL }
```

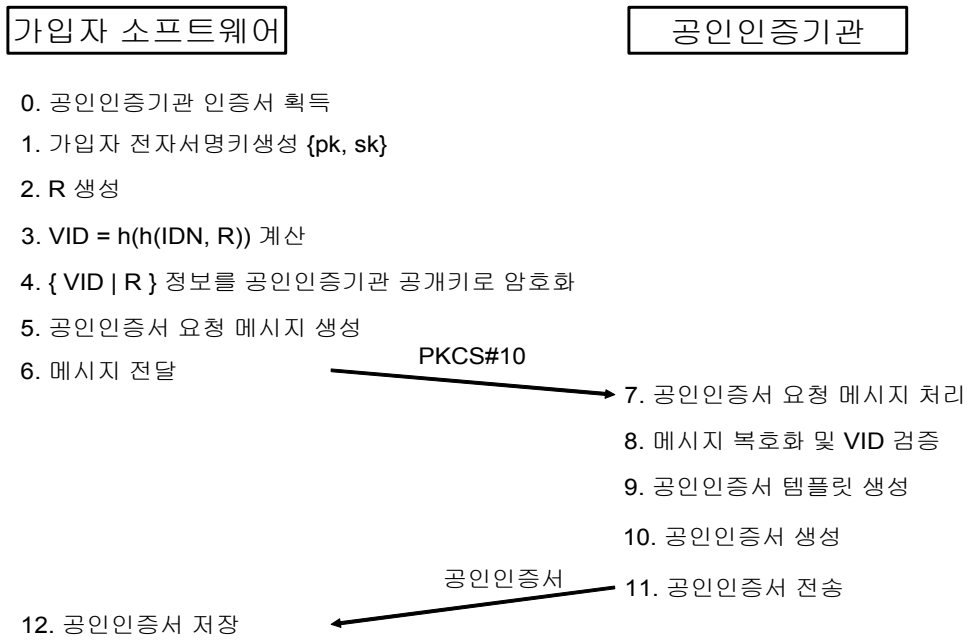
[v1.11, 2002. 9. 2 개정]

realName 필드에는 공인인증서 소유자에 대한 실명을 한글로 표기하며 UTF8String으로 인코딩하여 저장한다.

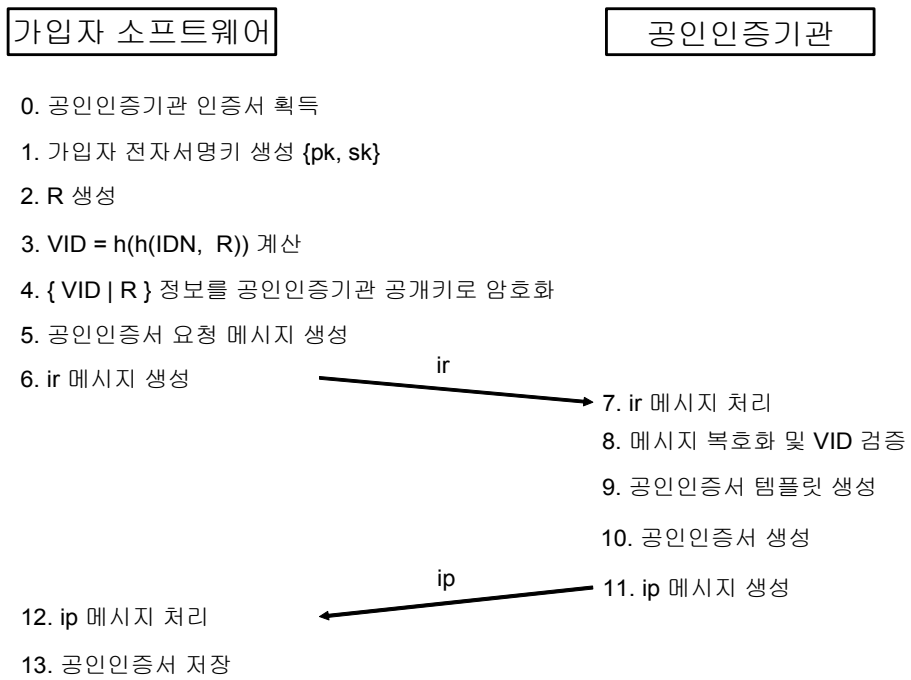
userInfo 필드에는 공인인증서 소유자의 추가 식별정보를 저장할 수 있으며 식별번호관련 정보 id-VID를 여기에 저장한다.

부록 A. 식별번호 관련 정보 주입 절차

1. PKCS #10을 이용한 식별번호 관련 정보 주입 절차



2. RFC 2510을 이용한 식별번호 관련 정보 주입 절차



부록 B. 식별번호 검증 예제

식별번호를 이용하여 사용자 접근통제 및 사용자 인증을 수행하는 응용의 다양한 환경에 따라 서로 다른 검증절차가 수행될 수 있다.

1. 응용사이트에서 식별번호를 필요로 하는 경우

다음의 예는 사용자가 식별번호와 난수를 제시하는 경우, 응용사이트에서의 검증 절차이다.

단계	사용자		응용사이트
1	IDN, R 전송	→	
2	공인인증서 전송	→	
3			공인인증서에서 VID 추출
4			VID′ = h(h(IDN, R)) 계산
5			VID와 VID′ 동일 여부 확인

1 : 사용자는 식별번호 IDN과 R을 응용사이트에게 전달하며 이 때 전달되는 IDN과 R은 안전한 방법으로 전달되어야 한다.
2 : 단계 1에서 전달한 IDN 정보가 포함되어 있는 공인인증서를 응용사이트에게 전달한다.
3 : 단계 2에서 전달된 공인인증서로부터 VID 값과 해쉬 알고리즘을 추출한다.
4 : 단계 1에서 전달된 IDN 정보와 R 정보를 이용하여 단계 3에서 추출한 해쉬 알고리즘으로 VID′ 값을 계산한다.
5 : 단계 4에서 계산된 VID′ 값이 단계 3에서 추출한 VID 값과 동일한 지의 여부를 확인한다.

단계 1, 2에서 전송되는 데이터의 순서는 관계가 없으며 단지 IDN과 R정보는 제 3자에게 유출되지 않도록 안전하게 전달되어야 한다. 해당 정보를 안전하게 전달하는 방법에 대해서는 본 규격에서 다루지 않는다.

2. 응용사이트에서 이미 식별번호를 알고 있는 경우

다음의 예는 응용사이트가 이미 사용자의 식별번호를 알고 있는 경우 사용자는 난수만을 제시하고 응용사이트에서 이를 검증 절차이다.

단계	사용자		응용사이트
1	R 전송	→	
2	공인인증서 전송	→	
3			공인인증서에서 VID 추출 및 IDN 획득
4			VID′ = h(h(IDN, R)) 계산
5			VID와 VID′ 동일 여부 확인

1 : 사용자는 R을 응용사이트에게 전달하며 이 때 전달되는 R은 안전한 방법으로 전달되어야 한다.

2 : 단계 1에서 전달한 R정보와 관련 있는 IDN 정보가 포함되어 있는 공인인증서를 응용사이트에게 전달한다.

3 : 단계 2에서 전달된 공인인증서로부터 VID 값과 해쉬 알고리즘을 추출하고 사용자와 부합되는 IDN 값을 가져온다.

4 : 단계 1에서 R 정보를 이용하여 단계 3에서 추출한 해쉬 알고리즘으로 VID′ 값을 계산한다.

5 : 단계 4에서 계산된 VID′ 값이 단계 3에서 추출한 VID 값과 동일한 지의 여부를 확인한다.

단계 1, 2에서 전송되는 데이터의 순서는 관계가 없으며 단지 R정보는 제 3자에게 유출되지 않도록 안전하게 전달되어야 한다. 해당 정보를 안전하게 전달하는 방법에 대해서는 본 규격에서 다루지 않는다.

3. 식별번호를 응용사이트로부터 보호하고자 하는 경우

다음의 예는 사용자가 응용사이트에게 식별번호 제공을 원하지 않을 경우의 응용사이트에서의 검증 절차이다.

단계	사용자		응용사이트
1	h(IDN, R) 전송	→	
2	공인인증서 전송	→	
3			공인인증서에서 VID 추출
4			VID′ = h(h(IDN, R)) 계산
5			VID와 VID′ 동일 여부 확인

1 : 사용자는 식별번호 IDN과 R의 해쉬값 h(IDN, R)만을 응용사이트에게 전달하며 이 때 전달되는 h(IDN, R)는 안전한 방법으로 전달되어야 한다.

2 : 단계 1에서 전달한 h(IDN, R)정보가 포함되어 있는 공인인증서를 응용사이트에게 전달한다.

3 : 단계 2에서 전달된 공인인증서로부터 VID 값과 해쉬 알고리즘을 추출한다.

4 : 단계 1에서 전달된 h(IDN, R)정보를 이용하여 단계 3에서 추출한 해쉬 알고리즘으로 VID′ 값을 계산한다.

5 : 단계 4에서 계산된 VID′ 값이 단계 3에서 추출한 VID 값과 동일한 지의 여부를 확인한다.

단계 1, 2에서 전송되는 데이터의 순서는 관계가 없으며 단지 h(IDN, R)정보는 제3자에게 유출되지 않도록 안전하게 전달되어야 한다. 해당 정보를 안전하게 전달하는 방법에 대해서는 본 규격에서 다루지 않는다.

부록 C. ASN.1 표기

```
IDNumber88 { iso(1) member-body(2) korea(410) kisa(200004) npki(10) attributes(1)
                identifyData(1) }

DEFINITIONS EXPLICIT TAGS ::=BEGIN

-- EXPORTS All;

    IMPORTS

      AlgorithmIdentifier, Name, CertificateSerialNumber, Attribute, AttributeTypeAndValue
          FROM PKIX1Explicit88 {iso(1) identified-organization(3) dod(6) internet(1)
          security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-explicit-88(1)};

-- KISA specific OIDs
id-KISA  OBJECT IDENTIFIER  ::=
        { iso(1) member-body(2) korea(410) kisa(200004)}

-- KISA arcs
id-npki OBJECT IDENTIFIER  ::=  { id-KISA 10 }
id-attribute OBJECT IDENTIFIER  ::=  { id-npki 1 }
id-kisa-identifyData OBJECT IDENTIFIER  ::=  { id-attribute 1 }
id-VID OBJECT IDENTIFIER ::= { id-kisa-identifyData 1 }
id-EncryptedVID OBJECT IDENTIFIER ::= { id-kisa-identifyData 2 }
id-randomNum OBJECT IDENTIFIER ::= { id-kisa-identifyData 3 }


-- Virtual ID

VID ::= SEQUENCE {
        hashAlg                 HashAlgorithm,
        virtualID       [0]     OCTET STRING }

HashAlgorithm ::= AlgorithmIdentifier

HashContent ::= SEQUENCE {
        idn             PrintableString,
        randomNum       BIT STRING }
```

```
-- Encrypted VID

EncryptedVID ::= SEQUENCE {
        version         [0]     INTEGER DEFAULT 0,
        vidHashAlg      [1]     VIDHashAlgorithm OPTIONAL,
        vidEncAlg       [2]     VIDEncryptionAlgorithm,
        certID          [3]     IssuerAndSerialNumber,
        encryptedVID    [4]     OCTET STRING }

VIDHashAlgorithm ::= AlgorithmIdentifier

VIDEncryptionAlgorithm ::= AlgorithmIdentifier

IssuerAndSerialNumber ::= SEQUENCE {
        issuer Name,
        serialNumber CertificateSerialNumber }

EncryptContent ::= SEQUENCE {
        vid             VID,
        randomNum       BIT STRING }


-- OtherName in SAN, Refer to [TTA-120012]

IdentifyData ::= SEQUENCE {
        realName        UTF8String,
        userInfo        SEQUENCE SIZE (1..MAX) OF AttributeTypeAndValue
                        OPTIONAL }
END
```

부록 D. 규격 연혁

버전	제・개정일	제・개정내역
v1.00	2001년 9월	・“식별번호를 이용한 신원확인 기술규격“명으로 제정
v1.10	2002년 6월	・“공인인증서내 신원확인 기술규격” 명칭으로 개정 ・전체적인 문서 형식 및 구성을 TTA 단체표준 문서양식에 맞게 개정 ・3장 관련 표준 및 규격 표기 정리 ・4장 개정된 전자서명법을 준용하는 용어로 대체 ・5장 기호 R 추가 ・6장 전자서명생성키 해쉬값 h(sk)를 난수 R로 대체 ・7장 7.1 가상 식별번호 속성 ASN.1 형식 수정 ・7장 7.1 VID OID 수정 ・7장 7.2 EVID OID 수정 ・8장 8.1 난수 정보 저장 위치 추가 및 randomNum OID 추가 ・부록 A. 식별번호 관련정보 주입절차 수정 ・부록 B. 식별번호 검증 예제 2, 3 추가 ・부록 C. ASN.1 표기 수정 ・부록 D. 규격 연혁 추가 ・“기술규격 작성 공헌자“ 추가
v1.11	2002년 9월	・“식별번호를 이용한 본인확인 기술규격” 명칭으로 개정 ・1장, 6장 6.10 “신원확인”을 “식별번호”로 수정 ・8장 8.1 “식별번호 관련”을 “난수”로 수정 ・8장 8.1 스마트카드에 난수 정보 저장과 관련하여 문구 해석에 오해의 소지가 있어 이를 명확히 하였음. 향후 스마트카드 기술규격이 개정될 때 난수 정보를 [PKCS#5v2.0] 및 [PKCS#8]을 적용할 수 있도록 개정이 필요함 ・8장 8.3 id-kisa-identifyData OID를 “id-npki"에서 ”id-attribute"로 수정
v1.20	2008년 10월	・관련 국내 표준 및 규격 갱신 내용 반영 ・스마트카드 메모리 맵에서 식별번호 생성용 난수의 저장 위치가 변경됨에 따른 개정
v1.21	2009년 9월	・공인전자서명인증체계 기술규격 개정에 따라 본문 내용 중 관련 기술규격 참조 변경 사항 개정

규격 작성 공헌자

본 규격의 제·개정을 위해 아래와 같이 많은 분들이 공헌을 하였습니다.

구분	성명	소속사
과제 제안		한국인터넷진흥원
규격 제출		한국인터넷진흥원 비씨큐어 한국정보인증 한국증권전산
규격 검토	이재일	한국인터넷진흥원
	김승주	한국인터넷진흥원
	백종현	한국인터넷진흥원
	이석래	한국인터넷진흥원
	박상환	한국인터넷진흥원
	이원철	한국인터넷진흥원
	박윤식	한국인터넷진흥원
	박순태	한국인터넷진흥원
	이용	한국인터넷진흥원
	조지용	한국인터넷진흥원
	박종욱	한국인터넷진흥원
	박정환	한국인터넷진흥원
	서정훈	한국인터넷진흥원
	정찬주	한국인터넷진흥원
	지석진	한국인터넷진흥원
	심문보	한국인터넷진흥원
	백종현	한국인터넷진흥원
	황보성	한국인터넷진흥원
	이향진	한국인터넷진흥원
	권성호	한국인터넷진흥원
	박상준	비씨큐어
	정권성	비씨큐어
	이동석	한국정보인증

	박광춘	한국정보인증
	김재중	한국정보인증
	김연숙	한국정보인증
	김근옥	한국정보인증
	하광필	코스콤
	배오열	코스콤
	김성덕	코스콤
	장석한	코스콤
	이성진	코스콤
	이성국	코스콤
	김호술	금융결제원
	김동식	금융결제원
	박철우	금융결제원
	오중효	금융결제원
	최영준	금융결제원
	반형식	한국전산원
	유주현	한국전산원
	김정	한국전자인증
	윤오영	한국전자인증
	김형욱	한국전자인증
	이성철	한국무역정보통신
	국상진	한국무역정보통신
	백주성	소프트포럼
	권용찬	이니텍
	박선우	한국정보인증
	이규승	코스콤
	이한욱	금융결제원
	정순구	한국전자인증
규격안 편집	박상환	한국인터넷진흥원
	이원철	한국인터넷진흥원