

NAK-P-2008-02

## 전자기록물 전자서명 인증서 장기검증 기술규격

Technical Standard of Long-Term Validation Service  
for the Digital Signature Certificate of Digital Archives



2008년 11월 4일 제정

- 제 정 자 : 행정안전부 국가기록원장
- 제 정 일 : 2008년 11월 4일(행정안전부 고시 제 2008-43호)
- 심의부회 : 국가기록관리위원회, 표준전문위원회
- 원안작성 :
  - 국가기록원 기록정보화과 김동명(공업연구사)
- 검토·관리 :
  - 국가기록원 표준협력과 김형국(학예연구관), 김재평(공업연구사)
- 자 문 :
  - 충남대학교 전기정보통신공학부 이규철(교수)

(1) 이 표준에 대한 의견 또는 질문은 아래 전화로 연락하거나 홈페이지를 이용하여 주십시오.

표준열람 : 행정안전부 국가기록원(<http://www.archives.go.kr>)

행정안전부 국가기록원 기록정책부 표준협력과(042-481-6248, 6265)

기록정보서비스부 기록정보화과(042-481-8970)

(2) 이 표준에 대한 저작권은 국가기록원에 있으며, 이 문서의 전체 또는 일부에 대하여 상업적 이익을 목적으로 하는 무단 복제 및 배포를 금지합니다.

Copyright© National Archives of Korea(2008). All Rights Reserved.

## 목 차

머리말 .....	ii
<b>1 적용범위 .....</b>	<b>1</b>
<b>2 인용표준 .....</b>	<b>1</b>
<b>3 용어정의 .....</b>	<b>1</b>
<b>4 전자서명 인증서 장기검증 개요 .....</b>	<b>4</b>
4.1 전자서명 인증서 장기검증 정의 .....	4
4.2 전자서명 인증서 장기검증 목적 .....	4
4.3 전자서명 인증서 장기검증 필요성 .....	5
4.4 전자서명 인증서 장기검증 방법 .....	8
4.5 전자서명 인증서 장기검증 구현 .....	12
<b>5 전자서명 인증서 장기검증 규격 .....</b>	<b>15</b>
5.1 전자서명 인증서 장기검증데이터 .....	15
5.2 전자서명 인증서 장기검증데이터 처리 세부내용 .....	22

## 머리말

이 표준은 기록물관리기관이 보존 중인 전자기록물의 진본성과 무결성을 확인하기 위한 방법의 하나로, 행정전자서명 인증서를 장기적으로 검증하는 처리방식과 데이터 규격을 정의한 것으로서 표준전문위원회 및 국가기록관리위원회의 심의를 거쳐 제정한 공공표준이다.

이 표준의 법률적 근거는 다음과 같다.

- 공공기록물관리에 관한 법률 제20조
- 공공기록물관리에 관한 법률 시행령 제32조 제4항, 제35조 제2항, 제36조 제2항, 제40조 제3항, 제44조 제2항, 제46조 제2항 및 제5항

이 표준은 국가기록원에 의해 유지 및 관리되며, 관련 법령의 개정, 기술의 발전, 관계기관의 요청 등으로 인해 개정이 필요할 경우에는 필요성 및 타당성 검토를 거쳐 개정안을 마련하고 전문가 검토 및 의견수렴 절차를 거쳐 개정을 추진한다.

이 표준은 저작권법에서 보호대상이 되는 저작물이다.

# 전자기록물 전자서명 인증서 장기검증 기술규격

## 1 적용범위

이 표준은 전자기록물 전자서명 인증서의 유효성을 장기적으로 검증하는 기술에 필요한 처리절차와 데이터 구조를 정의하고 있으며, 기록물관리기관이 기록관리시스템 또는 영구기록관리시스템으로 관리하는 전자기록물 전자서명의 인증서를 장기적으로 검증하는 경우에 적용한다.

## 2 인용표준

- RFC 3029, 인증서 검증 및 공증 서비스를 위한 프로토콜 정의
- RFC 3126, 전자서명 장기검증기술에 대한 프로세스 정의

## 3 용어정의

이 표준의 목적을 위해 다음의 용어와 정의를 적용한다.

### 3.1 전자기록물 전자서명 인증서 장기검증시스템

전자기록물 전자서명의 유효성 검증을 위해 과거의 행정전자서명 인증폐기 목록을 지속적으로 보존하며, 장기검증 이용 시스템으로부터 전자기록물의 전자서명에 사용된 인증서와 그 서명시점정보를 받아서 해당시점에 유효했던 인증서임을 검증할 수 있는 서비스를 제공하는 시스템. 약칭하여 장기검증시스템이라고 한다.

### 3.2 인증서 주체

전자서명 행위의 주체로서 책임과 권한이 부여되는 사람 또는 시스템. 혹은 해당 전자서명을 확인하기 위하여 인증기관이 발급하는 인증서에 포함되는 여러 정보항목 중에서 인증서의 소유자를 뜻하는 항목 이름

### 3.3 인증기관 (CA, Certificate Authority)

전자서명이 해당 주체의 것임을 누구나 확인할 수 있도록 인증서 주체와 전자서명검증키(공개키) 등 필요정보를 묶어 배포하는 인증서를 발급하며, 발급한 인증서의 유효성을 검증할 수 있도록 인증서에 인증기관 명의의 전자서명을 부여하고 인증서 유효기간을 설정하며, 그 유효기간중이라도 여러가지 이유로 더 이상 유효하지 않은 인증서들의 목록을 매일 정리하여 인증서폐기목록(CRL)으로 게시하는 등의 서비스를 제공하는 기관

### 3.4 시점확인(TSA, TimeStamping Authority) 시스템

행위의 시점을 기만하거나 부인하지 못하도록 공인된 현재시점 정보(타임스탬프)를 제공하는 기관 또는 시스템. 이 타임스탬프도 전자서명과 같은 무결성이 있기 때문에 전자문서의 생성시점이나 전자서명시점, 장기검증데이터 유효성 연장시점 등을 확인하는 서비스 등에 사용 가능하다.

### 3.5 타임스탬프 토큰(TST, TimeStamp Token)

TSA로부터 발급받은 오브젝트에 대한 시점확인 정보. 약칭 타임스탬프라고 한다.

### 3.6 감사기록

시스템의 주요 변경 사항을 전자서명하여 기록함으로써 변경에 대한 무결성을 확인하도록 하는 기록

### 3.7 ES (Electronic Signature Formats)

장기검증 근거데이터의 포맷으로 전자서명(Signed Data) 구조체

### 3.8 ES-A (ES with the additional validation data for the ES-C and ES-X forming)

ES-X 데이터 또는 ES-C 데이터 포맷의 확장형 구조체. 이 기술규격에서는 장기검증데이터의 기본정보구조인 ES-X 데이터를 계속 유효하도록 타임스탬프 토큰을 주기적으로 추가하여 확장 보존하는 정보구조로 사용한다.

### 3.9 ES-C (ES with Complete validation data)

ES-T 데이터 포맷에 인증서 경로 레퍼런스를 추가한 구조체

### 3.10 ES-T (ES with Time-Stamp)

ES 데이터 포맷에 TimeStamping 을 추가한 구조체

### 3.11 ES-X (ES with the additional validation data forming the eXtended Validation Data)

ES-C 데이터 포맷에 인증서 레퍼런스를 추가한 확장형 구조체. 이 기술규격에서는 매일 수집된 인증서폐기목록에 장기검증시스템의 전자서명을 적용하고 그 시점에 대하여 발급받은 시점확인 타임스탬프를 추가적용하여 구성하는 장기검증데이터 기본정보구조로 사용한다.

### 3.12 장기검증 데이터

ES, ES-T, ES-C, ES-X, ES-A 등의 형식을 가지는 데이터에 대한 통칭으로 장기검증을 위한 근거 정보 데이터

### 3.13 OID (Object Identifier)

객체 구별자

### 3.14 ASN.1 (Abstract Syntax Notation One)

추상구문기법, 데이터의 구성을 표현하는 문법, ITU-T를 통해서 표준화 됨

### 3.15 CRL 배포지점 (CRL Distribution Point)

인증서 내부에 포함된 정보로 인증서를 발급한 인증기관이 해당 인증서의 CRL 정보를 게시하는 위치를 알려준다. 주요 프로토콜은 LDAP 또는 HTTP를 이용한다.

### 3.16 해쉬함수

다양한 길이의 입력을 고정된 짧은 길이의 출력으로 변환하는 함수.

### 3.17 X.509

X.509 인증서는 IETF의 PKI 인증서와 X.509 v3 인증서 표준의 CRI 프로파일을 가리키며, RFC 3280에 정의되어 있음

### 3.18 HSM (Hardware Security Module)

증명서 발급 시스템에서는 공개키 기반 구조의 환경에서 운영상의 신뢰성을 확보하기 위하여 주요 인증시스템 키를 안전하게 보관하기 위한 하드웨어

### 3.19 WORM 스토리지 ("Write Once Read Many" Storage)

장기적인 원본 데이터 보관을 목적으로 하는 아카이빙 스토리지

### 3.20 인증서폐기목록(CRL, Certificate Revocation List)

현재 유효기간이 남아 있는 인증서 중 여러 가지 사유(개인키 분실, 퇴직, 보직 변경 등)로 인해 사용할 수 없는 인증서들의 목록을 발급 인증기관이 서명하여 배포하는 목록

## 4 전자서명 인증서 장기검증 개요

### 4.1 전자서명 인증서 장기검증 정의

전자서명 인증서 장기검증이란 전자기록물의 진본성 및 무결성을 입증하기 위해 전자기록물에 적용된 전자서명 인증서의 유효성을 장기적으로 검증하는 방법이다.

### 4.2 전자서명 인증서 장기검증 목적

위·변조 및 훼손이 쉬운 전자기록물에 대하여 진본성, 무결성 등을 시스템적으로 보장하기 위해 전자서명이 적용되지만, 오랜 시간이 지난 후 그 유효성 검증에 제한을 받기 때문에 전자기록물을 장기적으로 보관·관리하는 경우에는 해당 전자서명의 효력을 지속적으로 확인할 수 있는 방법이 필요하다.

전자서명 인증서 장기검증은 전자기록물의 전자서명 인증서에 대해 장기적으로 유효함을 검증할 수 있는 방법을 제공함으로써 전자서명이 적용된 전자기록물에 대해 장기적으로 진본성, 무결성 등을 확인할 수 있게 해주며, 이를 위해서 타임스탬프 토큰과 인증서 검증정보를 활용하여 전자서명에 사용한 인증서가 유효한 상태에서 전자서명이 생성되었는지를 검증한다.

### 4.3 전자서명 인증서 장기검증 필요성

#### 4.3.1 전자서명을 통한 기록물의 무결성 검증

장기보관 전자기록물은 장기보존포맷 생성 시 전자서명을 포함하도록 하고 있다. 전자기록물 장기보존포맷에 포함된 전자서명은 전자기록물 무결성 확인을 위해 사용한다. 이를 위해 장기보존포맷 생성 시 대상 전자기록물에 대해 전자서명을 생성·보관하며, 무결성 확인이 필요한 시점에 보관한 전자서명 검증을 통해 해당 전자기록물 무결성을 확인하는 것이다. 전자기록물 무결성 확인을 위한 전자기록물 전자서명은 전자서명 생성 및 검증과정을 거치게 된다. 전자서명 생성 및 검증과정에서는 전자기록물 원본 데이터, 전자서명 시 생성된 전자기록물 전자서명 데이터 및 전자서명에 사용된 인증서를 이용하여 처리하게 된다.

먼저, 전자서명 생성은 그림 1과 같이 전자서명을 하고자 하는 대상 데이터에 대해 전자서명생성키(개인키)를 이용하여 전자서명을 생성하게 된다. 이러한 전자서명 생성 과정을 간략히 표현하면 다음과 같다. 이렇게 전자서명을 수행한 후에는 해당 전자기록물뿐만 아니라 해당 전자기록물에 대한 전자서명, 검증에 필요한 전자서명검증키(공개키)가 포함된 인증서를 함께 보관하도록 되어 있다.

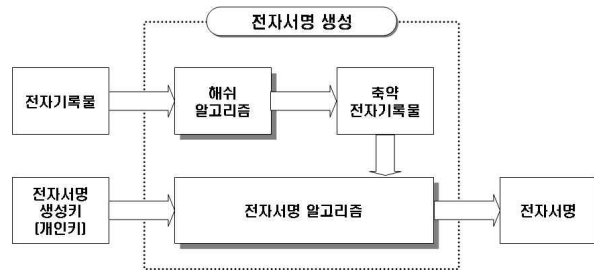


그림 1 - 전자서명 생성 과정

그림 2는 전자서명 검증 과정으로 전자서명 과정의 역순의 개념으로 처리된다. 전자기록물과 전자서명 그리고 전자서명검증키(공개키)를 넣어 전자서명

의 유효성 여부를 확인하는 과정으로 구성되며, 다음과 같이 간략히 표현할 수 있다. 전자기록물 전자서명에 대한 유/무효 여부가 확인되면 해당 전자기록물은 전자서명이 생성된 시점부터 서명을 확인한 시점까지 위·변조되지 않은 상태에서 보관되고 있었음을 신뢰할 수 있게 된다.

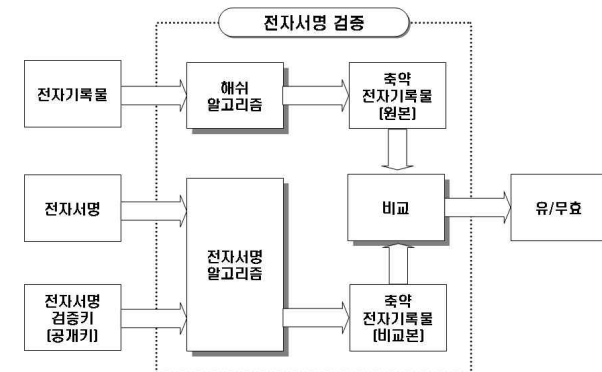


그림 2 - 전자서명 검증 과정

#### 4.3.2 기존의 단기적 전자서명 인증서 유효성 검증 방법

전자기록물 무결성 확인을 위한 전자서명 검증 시, 해당 전자기록물 전자서명 값의 확인뿐만 아니라 전자서명에 사용한 전자서명검증키(공개키)가 전자서명 시점에 유효했는지도 확인해야만 한다. 이를 위해 전자서명 생성·검증 과정에서 전자서명 인증서를 사용함으로써 전자서명검증키(공개키)의 유효성을 확인할 수 있게 된다.

그림 3의 전자서명 인증서에는 인증서 일련번호, 인증서 유효기간, 인증서 주체(소유자) 및 전자서명검증키(공개키) 등의 내용을 포함하고 있으며, 인증서 발급 인증기관에서 해당 내용에 대한 전자서명을 한 뒤 발급한다.

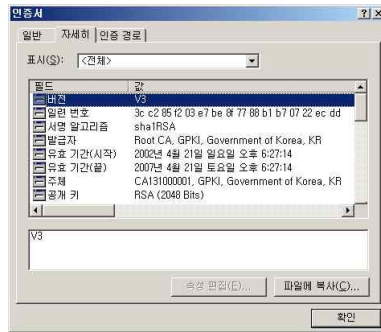


그림 3 - 전자서명 인증서

전자서명검증키(공개키) 유효성 확인을 위한 인증서 유효성 확인은 두 가지 사항을 확인하는 과정이다. 첫 번째는 전자서명 인증서를 신뢰할 수 있는 인증기관에서 발급했는지에 관한 것이고, 두 번째는 해당 인증기관을 통해 전자서명 검증에 사용하는 인증서가 현재 유효한지에 대해 확인을 하는 것이다.

인증기관의 인증서 생성 시스템을 통해 발급된 전자서명 인증서는 디렉토리 시스템에 게시된 인증서폐기목록(CRL, Certificate Revocation List) 확인을 통해 해당 인증서 유효성을 확인하게 된다. 인증서폐기목록이란 인증기관이 발급한 인증서 중에서 인증서의 유효기간이 남아있는 인증서 중 현재 유효하지 않은 인증서 목록을 포함하고 있다. 예를 들어 인증서 주체가 퇴직, 보직변경 등의 사유 발생시에 더 이상 전자서명생성키(개인키) 및 전자서명검증키(공개키)를 사용하지 못하도록 인증서를 폐기하는 것이다. 그림 4는 다음은 전자서명 사용자 인증서를 검증하는 절차를 간략히 표현한 것이다.



그림 4 - 전자서명 사용자 인증서 검증 절차

전자서명 시 사용한 인증서에 대한 검증은 인증서를 발급한 인증기관의 발급

경로인 인증경로에 대한 역순으로 모든 발급 인증기관의 인증서 전자서명을 검증하는 방법을 이용한다. 인증경로의 역순으로 검증한 후 가장 최상위에 있는 기관의 인증서가 신뢰할 수 있는 지에 대한 확인을 하고, 마지막으로 전자서명에 사용한 사용자 인증서가 현재 유효한지에 대해 인증서폐기목록(CRL)을 통해 확인함으로써 인증서 유효성을 확인하게 된다.

### 4.3.3 인증서 및 인증서폐기목록(CRL)의 유효기간

전자기록물 전자서명에 사용하는 인증서에는 유효기간이 있다. 민간용 인증서는 보통 1년이 유효기간이며, 행정전자서명용 인증서는 유효기간이 2년 3개월이다. 이처럼 전자서명 인증서의 유한성으로 인해 전자서명 당시에는 유효한 인증서로 전자서명한 전자기록물이었더라도, 인증서의 유효기간이 경과하였을 경우에는 해당 인증서의 유효성을 검증하지 못하기 때문에 당시의 전자서명을 검증하지 못하는 상황이 발생하고 전자기록물의 무결성 확인이 불가능해진다.

또한, 인증서 유효성 검증은 인증서 유효기간 확인과 더불어 인증서 발급 기관인 인증기관에서 배포하는 인증서폐기목록(CRL)을 통해서도 확인하여야 한다. 이 때 인증서폐기목록에 폐기된 인증서 상태 정보를 새롭게 반영하기 위해 주기적으로 갱신하고 있기 때문에 이전의 인증서폐기목록은 새로운 인증서폐기목록이 만들어졌을 때에는 더 이상 유효하지 않게 되어 있다. 즉, 인증서폐기목록에 인증서 상태 정보가 적절히 반영되어 인증서 유효성 검증에 사용되어야 하기 때문에 인증서폐기목록도 전자서명 인증서와 마찬가지로 유효기간을 갖게 되는 것이다.

결론적으로 인증서 및 인증서폐기목록의 유효기간은 전자서명 안전성 보장, 인증서 상태정보의 적절한 반영 등 관리상의 이유로 인해 제한할 수밖에 없으므로 장기보존 전자기록물에 대한 무결성 확인을 위해서 전자서명을 사용할 경우에는 이러한 제한사항을 고려하여 전자기록물 전자서명이 장기적으로 검증 가능하게 하여야 한다.

## 4.4 전자서명 인증서 장기검증 방법

### 4.4.1 전자서명 인증서 장기검증 적용 개념

장기보존 전자기록물 전자서명의 경우 전자서명 값 및 해당 전자서명 인증서에 대한 검증이 전자기록물을 장기적으로 보존하는 동안에 항상 가능하여야 한다. 특히, 전자기록물의 진본성, 무결성은 전자기록물 전자서명을 통해 보장되기 때문에 전자서명의 완벽한 검증을 수행하여야만 해당 전자기록물의 무결성을 확인할 수 있다.

만약, 전자기록물 전자서명 값에 대한 검증만을 수행하고 전자서명 인증서 검증을 수행하지 않을 경우에는 전자기록물의 무결성을 보장할 수 없다. 전자서명 인증서를 확인하지 못한 상황에서는 전자기록물 전자서명에 사용한 전자서명 키 및 서명 생성 시각 등을 임의로 조작하여 사용할 수 있기 때문에 해당 전자기록물 전자서명을 신뢰할 수 없게 된다.

완전한 서명 검증을 위해서는 전자서명 값과, 이를 수행한 인증서 정보 그리고 전자서명 수행 시점의 인증서 상태 정보가 필요하며, 이 중 시간이 경과할 경우 신뢰가 불가능해지는 인증서 상태 정보인 인증서폐기목록(CRL)을 장기적으로 관리할 수 있어야 한다. 전자기록물에 대한 전자서명 결과로 생성되는 장기보존포맷은 서명대상이 되는 전자문서와 전자서명값 그리고 전자서명 인증서 정보를 포함하고 있으며, 전자기록물 전자서명 인증서 장기검증시스템은 전자서명에 사용된 인증서의 서명 당시 유효성 정보인 인증서폐기목록(CRL)을 장기적으로 유효성이 유지될 수 있도록 관리하는 시스템이다. 이를 통해 장기보존포맷으로 관리되는 전자기록물에 대한 장기적인 유효성 검증을 가능하도록 한다.

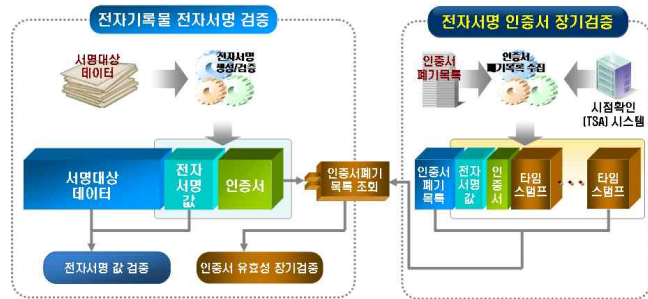


그림 5 - 전자서명 인증서 장기검증

#### 4.4.2 전자서명 인증서 장기검증시스템의 특징

##### 4.4.2.1 표준적용

장기검증을 위해 단순히 인증서폐기목록(CRL)에 대한 관리를 하는 시스템이 아니라 수집된 인증서폐기목록(CRL)에 대해 국제 표준 (RFC 3126)을 준용하여 적용되었다.

##### 4.4.2.2 독립적인 관리

전자서명을 수행하는 측이나 전자서명된 전자기록물을 관리하는 측에서는 별도의 관리가 필요하지 않다. 따라서 전자서명을 수행하거나 관리하는 시스템이 별도의 부하를 받지 않게 되어 장기검증을 적용하더라도 시스템의 확장이 불필요하다.

##### 4.4.2.3 계량적인 관리

전자서명이 수행된 전자기록물을 재가공하는 것이 아니라 별도의 장기검증 정보를 생성하여 관리하며, 전자기록물의 수와 관계없이 인증서폐기목록(CRL)의 양에 비례하며 그 양은 행정전자서명 적용할 경우 년 간 약 14GB의 정도가 소요된다. (CRL 크기 X CRL 개수 X 일 : 50KB X 780여개 X 365일)

##### 4.4.2.4 확대적용 가능

전자서명 인증서 장기검증시스템은 특정 시스템을 목적으로 개발된 시스템이 아니라 범용적인 목적으로 개발된 시스템이다. 따라서 전자서명이 된 데이터의 유효성을 확인하기 위한 타 서비스의 경우에도 서비스의 큰 변경 없이 바로 적용이 가능하다. 단, 전자서명 수행 시 전자서명과 함께 전자서명이 수행된 시점을 확인하기 위해 타임스탬프가 포함되어 있어야 한다.

#### 4.4.3 전자서명 인증서 장기검증데이터의 생성

전자서명 인증서의 유효성 검증 시 사용되는 인증서폐기목록(CRL)의 보관을

통해서 전자서명이 수행된 시점의 인증서 상태에 대한 확인을 수행할 수 있다. 하지만 단순히 보관하는 것만으로는 유효성 확인이 불가능하다. 이유는 인증서폐기목록(CRL) 자체 또한 일종의 전자서명문이며, 이러한 전자서명에 사용된 인증기관(CA) 인증서의 유효성에 대한 문제가 재귀적으로 발생할 수 있기 때문이다. 따라서 유효기간이 1~2일에 불과한 인증서폐기목록에 대해 장기적으로 유효성을 확인할 수 있는 방법이 필요하다.

이러한 전자서명 데이터의 유효성을 장기적으로 늘릴 수 있는 방법 중에서 전자서명 인증서 장기검증시스템은 RFC 3126(전자서명 장기검증기술에 대한 프로세스 정의)을 적용하였다. 즉, 인증서폐기목록(CRL)을 매일 매일 수집하여 전자서명한 후 RFC 3126에 의한 방법으로 보존하여 추후 인증서 검증 시의 검증정보로 제공을 하는 것이다.

이러한 방법은 전자기록물에 대한 장기검증을 직접 적용하는 방법에 비해 처리 방법이 간략하고 전자기록물의 수나 양에 관계없이 계획적인 장기검증 방법을 제공하기 때문에 대량 전자서명 데이터의 장기검증에 효과적인 방법을 제시할 수 있다.

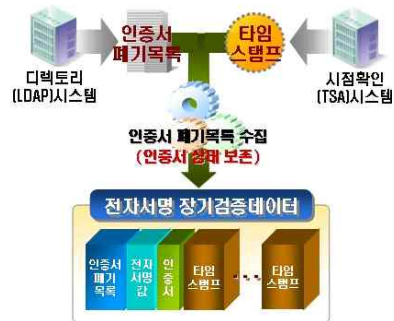


그림 6 - 장기검증데이터의 생성 방법

그림 6은 인증서폐기목록(CRL)을 획득하고 이에 대해 RFC 3126의 방법에 따라 시점확인시스템을 통해 타임스탬프를 추가해 가면서 인증서 장기검증이 가능한 장기검증데이터를 생성하는 방법을 도식적으로 표현한 것이다.



그림 7 - 전자서명 인증서 장기검증 데이터 구성요소

그림 7은 전자서명 인증서 장기검증데이터의 구성요소를 도식화한 것이다. 그림 7에서와 같이 타임스탬프를 수행한 시점확인시스템의 유효기간이 도래할 때마다 다시 타임스탬프를 발급받아 장기검증데이터의 유효성을 올려나간다.

이와 같은 전자서명 인증서 장기검증시스템을 적용하면, 장기검증을 하고자 하는 측에서는 전자기록물의 전자서명을 검증한 후 인증서의 유효기간이 경과하였을 경우 검증하고자 하는 인증서와 전자서명이 수행된 시점을 전자서명 인증서 장기검증시스템에 제시하게 된다. 전자서명 인증서 장기검증시스템에서는 인증서에 포함된 인증서폐기목록(CRL)의 게시 위치를 확인한 후 요청한 전자서명 수행 일자의 전자서명 인증서 장기검증데이터를 전달해 주고, 검증하고자 하는 측에서 이 데이터의 유효성을 확인한 후, 신뢰할 수 있을 경우 포함된 인증서폐기목록(CRL)을 이용함으로써 전자서명 인증서 장기검증을 완료하게 된다.

#### 4.5 전자서명 인증서 장기검증 구현

##### 4.5.1 주기적인 장기검증데이터 생성 및 갱신

RFC 3126에 의해 적용된 인증서 장기검증을 위한 장기검증데이터는 기본적으로 두 가지 정보로 이루어진다. 첫 번째는 최초 인증서폐기목록 수집 후 수행되는 ES-X 데이터 정보이며, 두 번째는 ES-X 데이터에 이용된 타임스탬프의 유효기간이 도래하였을 때 갱신을 수행하여 생성되는 ES-A 데이터이다. 이후 다시 타임스탬프의 유효기간이 도래하였을 경우에는 ES-A 데이터



생성이 지속적으로 이루어진다. 현재 시점확인시스템의 인증서 유효기간이 3년이므로 ES-A의 생성은 3년에 1회 정도 이루어진다고 볼 수 있다.

**4.5.2 장기검증데이터 기본정보 생성 (ES-X의 생성)**

그림 8에서 보는 것처럼 ES-X는 장기검증데이터 생성을 위한 기본정보로 RFC 3126의 적용을 위한 기본정보 생성이라고 볼 수 있다. 즉, 인증서폐기목록(CRL)을 수집한 후, 이를 RFC 3126에서 지정한 데이터의 형식으로 만드는 작업이다. 이 때 생성된 ES-X는 인증서폐기목록(CRL)과 함께 시점확인시스템으로부터 발급받은 타임스탬프 정보를 포함한다. 이 장기검증데이터는 최초 받은 타임스탬프 토큰의 유효기간까지 유효하다.



그림 8 - 장기검증데이터 생성을 위한 기본정보

**4.5.3 장기검증데이터 보존정보 생성 (ES-A의 생성)**

시점확인시스템의 인증서 교체 또는 인증서 유효기간 도래 등으로 ES-X의 유효성을 더 이상 확인할 수 없게 될 경우 아직 유효기간이 남아있는 상태에서 ES-A라는 정보를 생성해야 지속적인 장기검증데이터의 유효성을 유지할 수 있다.

ES-A는 최초 생성된 ES-X가 조만간 만료될 경우 또는 조만간 만료될 ES-A를 갱신하는 경우에 생성되며, 이는 새로운 시점확인시스템으로부터 타임스탬프 토큰을 발급받아 정보를 추가하는 것으로 수행된다. 이후 다시 타임스탬프의 유효기간이 만료되기 전에 다시 ES-A를 수행하는 방법으로 장기검증데이터의 유효성을 유지한다.



그림 9 - 장기검증데이터 보존정보 생성

**4.5.4 장기검증데이터 요청 및 검증**

전자기록물의 유효성 검증을 수행할 경우, 인증서의 유효기간이 남아있는 경우에는 기존의 전자서명 유효성 검증 방법인 통합검증서버를 이용하여 유효성을 검증하게 된다. 그러나 유효기간이 종료하였을 경우에는 장기검증시스템으로부터 유효성을 검증할 수 있는 장기검증용 근거정보를 요청하여야 한다. 이 때 요청하는 정보는 인증서폐기목록(CRL)을 확인할 수 있는 전자서명용 인증서와 전자서명이 수행된 날짜를 제공하여야 한다.

전자서명 인증서 장기검증시스템은 인증서에 포함된 인증서폐기목록(CRL)의 위치를 확인한 후 해당하는 일자의 장기검증데이터를 요청하는 측에 전달한다. 장기검증데이터를 전달받은 측은 장기검증데이터의 유효성을 확인한 후 유효할 경우 포함된 인증서폐기목록(CRL)의 내용을 확인하여 전자서명의 유효성 여부를 확인할 수 있게 된다.



그림 10 - 장기검증데이터 요청 및 검증

## 5 전자서명 인증서 장기검증 규격

### 5.1 전자서명 인증서 장기검증데이터

전자서명 인증서 장기검증데이터는 다음의 형식으로 처리, 저장 또는 송수신된다. 각 형식별로 장기검증데이터가 사용되는 단계는 “5.2 전자서명 인증서 장기검증데이터 처리 세부내역”을 참조하도록 한다.

- ES (Electronic Signature) 형식
- ES-T (ES with Time-Stamp) 형식
- ES-C (ES with Complete validation data) 형식
- ES-X (ES with eXtended validation data) 형식
- ES-A (ES with Archive) 형식

#### 5.1.1 ES(Electronic Signature) 형식

##### 5.1.1.1 데이터 구조

ES 형식은 사용자에게 의해 생성되며 RFC 2630의 SignedData 형식을 기반으로 한다. 위와 같이 SignedData.encapContentInfo 필드에 사용자가 서명하고 자 하는 CRL이 삽입되며, SignedData.SignerInfo.signedAttrs 필드에는 다양한 속성들이 포함될 수 있다. ES 형식만으로는 전자서명의 효력을 사용자 인증서 만료 후까지 연장시킬 수 없으므로 부가적인 정보(타임스탬프 토큰, 인증서 검증정보)를 ES 형식 안에 삽입해야 하며, 이러한 부가 정보는 사용자, 검증자, TTP 등에 의해 SignedData.SignerInfo.unsignedAttrs 필드에 저장되게 된다.



그림 11 - ES 데이터구조

#### 5.1.1.2 ASN.1 표기

[ES 형식에 사용되는 RFC 2630의 SignedData]

```

SignedData ::= SEQUENCE {
    version CMSVersion,
    digestAlgorithms DigestAlgorithmIdentifiers,
    encapContentInfo EncapsulatedContentInfo,
    certificates [0] IMPLICIT CertificateSet OPTIONAL,
    crls [1] IMPLICIT CertificateRevocationLists OPTIONAL,
    signerInfos SignerInfos }
  
```

```

SignerInfo ::= SEQUENCE {
    version CMSVersion,
    sid SignerIdentifier,
    digestAlgorithm DigestAlgorithmIdentifier,
    signedAttrs [0] IMPLICIT SignedAttributes OPTIONAL,
    signatureAlgorithm SignatureAlgorithmIdentifier,
    signature SignatureValue,
    unsignedAttrs [1] IMPLICIT UnsignedAttributes OPTIONAL }
  
```

#### 5.1.2 ES-T (ES with Time-Stamp) 형식

##### 5.1.2.1 데이터 구조

ES-T 형식은 사용자가 생성한 ES 형식에서 타임스탬프 토큰을 추가한 형식으로 생성된 타임스탬프 토큰은 SignedData.SignerInfo.unsignedAttrs 필드에 저장된다. 타임스탬프 토큰의 속성 OID 및 ASN.1 구조는 그림 12의 ASN.1 표기와 같으며, 타임스탬프 토큰의 해쉬값은 SignedData.SignerInfo.signature 필드의 값이다.



그림 12 - ES-T 데이터구조

5.1.2.2 ASN.1 표기

[타임스탬프 토큰의 속성 OID 및 ASN.1 표기]

```
id-aa-signatureTimeStampToken OBJECT IDENTIFIER ::= {
    iso(1)
    member-body(2)
    us(840)
    rsadsi(113549)
    pkcs(1)
    pkcs-9(9)
    smime(16)
    id-aa(2) 14}
```

```
SignatureTimeStampToken ::= TimeStampToken
```

5.1.3 ES-C (ES with Complete validation data) 형식

5.1.3.1 데이터 구조

ES-C 형식은 ES-T 형식에서 사용자 인증서 검증정보의 레퍼런스를 추가한 형식으로 획득된 인증서 검증정보 레퍼런스는 인증서 경로와 인증서 상태정보 레퍼런스로 나누어지며 각각 SignedData.SignerInfo.unsignedAttrs 필드에 저장된다.

인증서 경로 및 인증서 상태정보 레퍼런스의 속성 OID 및 ASN.1 구조는 그림 13의 ASN.1 표기와 같다. 인증서 경로 레퍼런스의 경우 OtherCertID.issuerSerial 필드는 반드시 이용되어야 하며, OtherCertID.OtherCertHash 필드는 레퍼런스되

는 모든 인증서의 해쉬값과 일치해야 한다. 인증서 상태정보 레퍼런스의 경우 CRL, OSCP 등이 이용될 수 있다.



그림 13 - ES-C 데이터구조

5.1.3.2 ASN.1 표기

[인증서 경로 레퍼런스의 속성 OID 및 ASN.1 표기]

```
id-aa-ets-certificateRefs OBJECT IDENTIFIER ::= {
    iso(1)
    member-body(2)
    us(840)
    rsadsi(113549)
    pkcs(1)
    pkcs-9(9)
    smime(16)
    id-aa(2) 21}
```

```
CompleteCertificateRefs ::= SEQUENCE OF OTHERCertID
```

```
OtherCertID ::= SEQUENCE {
    otherCertHash          OtherHash,
    issuerSerial           IssuerSerial OPTIONAL}
```

[인증서 상태정보 레퍼런스의 속성 OID 및 ASN.1 표기]

```
id-aa-ets-revocationRefs OBJECT IDENTIFIER ::= {
    iso(1)
    member-body(2)}
```

```
us(840)
rsadsi(113549)
pkcs(1)
pkcs-9(9)
smime(16)
id-aa(2) 22}
```

CompleteRevocationRefs ::= SEQUENCE OF CrIOcspRef

```
CrIOcspRef ::= SEQUENCE {
    crlids          [0] CRLListID      OPTIONAL,
    ocspids         [1] OcspListID     OPTIONAL,
    otherRev        [2] OtherRevRefs   OPTIONAL }
```

### 5.1.4 ES-X (ES with eXtended validation data) 형식

#### 5.1.4.1 데이터 구조

ES-X 형식은 전자서명 검증 시 이용되는 인증서 검증정보의 레퍼런스를 추가한 형태로 RFC 3126의 ES-X 1 (Type 1 X-Time-Stamp) 형식과 같다. 따라서 본 기술규격에서의 모든 ES-X의 명칭은 Type 1 X-Time-Stamp의 약어를 칭한다.

ES-X 형식은 ES-C 형식에 부가적인 타임스탬프 토큰을 추가한 형식으로 획득된 타임스탬프 토큰은 SignedData.SignerInfo.unsignedAttrs 필드에 저장된다.

ES-X 형식의 타임스탬프 토큰의 속성 OID 및 ASN.1 구조는 ASN.1 표기와 같으며, 다음 필드들의 연결된 값의 해쉬값이 타임스탬프 대상이 된다.

- SignedData.SignerInfo.signature 필드
- SignedData.SignerInfo.unsignedAttrs.SignatureTimeStampToken 필드
- SignedData.SignerInfo.unsignedAttrs.CompleteCertificateRefs 필드
- SignedData.SignerInfo.unsignedAttrs.CompleteRevocationRefs 필드

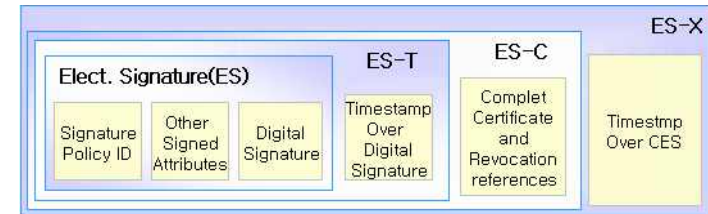


그림 14 - ES-X 데이터구조

#### 5.1.4.2 ASN.1 표기

[ES-X 형식의 타임스탬프 토큰의 속성 OID 및 ASN.1 표기]

id-aa-ets-escTimeStamp OBJECT IDENTIFIER ::= {

```
iso(1)
member-body(2)
us(840)
rsadsi(113549)
pkcs(1)
pkcs-9(9)
smime(16)
id-aa(2) 25}
```

ESCTimeStampToken ::= TimeStampToken

#### 5.1.5 ES-A (ES with Archive) 형식

##### 5.1.5.1 데이터 구조

ES-A 형식은 이전의 타임스탬프용 인증서가 만료된 경우에 사용한다. 또한, ES-A 형식은 여러개의 중첩된 타임스탬프를 포함하도록 지원하는 형식이다.

ES-A 형식은 ES-C 또는 ES-X 형식에 부가적인 타임스탬프 토큰을 추가한 형식으로 획득된 타임스탬프 토큰은 SignedData.SignerInfo.unsignedAttrs 필드에 저장된다.

ES-A 형식의 새로운 타임스탬프 토큰의 속성 OID 및 ASN.1 구조는 ASN.1 표기와 같으며, 다음 필드들의 연결된 값의 해쉬값이 타임스탬프 대상이 된다.

pkcs-9(9)  
smime(16)  
id-aa(2) 27}

ArchiveTimeStampToken ::= TimeStampToken

- SignedData.encapContentInfo.eContent 필드
- SignedData.SignerInfo.signedAttrs 필드
- SignedData.SignerInfo.signature 필드
- SignedData.SignerInfo.unsignedAttrs.SignatureTimeStampToken 필드
- SignedData.SignerInfo.unsignedAttrs.CompleteCertificateRefs 필드
- SignedData.SignerInfo.unsignedAttrs.CertificateValues 필드
- SignedData.SignerInfo.unsignedAttrs.CompleteRevocationRefs 필드
- SignedData.SignerInfo.unsignedAttrs.RevocationValues 필드
- SignedData.SignerInfo.unsignedAttrs.ESCTimeStampToken 필드
- SignedData.SignerInfo.unsignedAttrs.TimestampedCertsCRLs 필드
- SignedData.SignerInfo.unsignedAttrs.ArchiveTimeStampToken 필드들

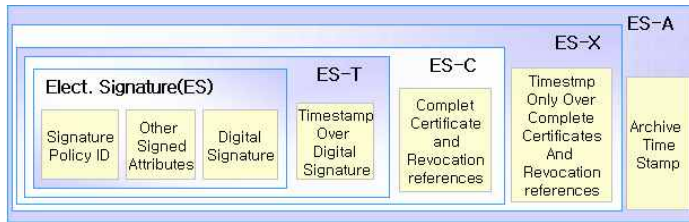


그림 15 - ES-A 데이터구조

5.1.5.2 ASN.1 표기

[ES-A 형식의 타임스탬프 토큰의 속성 OID 및 ASN.1 표기]

```
id-aa-ets-archiveTimestamp OBJECT IDENTIFIER ::= {
    iso(1)
    member-body(2)
    us(840)
    rsadsi(113549)
    pkcs(1)
```

5.2 전자서명 인증서 장기검증데이터 처리 세부내용

5.2.1 주기적인 장기검증데이터 생성

5.2.1.1 처리 개요

검증데이터의 요청 시 올바른 장기검증데이터를 제공할 수 있도록 장기검증 시스템에 등록된 인증서들에 대해 주기적으로 장기검증데이터를 생성 및 갱신하는 등의 관리작업을 수행하는 기능이다.

장기검증이 가능한 전자서명을 생성할 때 본 표준은 RFC 3126에 준하는 검증데이터(ES-X, ES-A)를 전자서명에 사용된 인증서의 해당 시점의 CRL 정보로부터 생성하여, 검증데이터를 요청 시 요청에 제시된 시점의 검증데이터를 반환한다. 이 기능은 아래의 절차에 의해 처리되며 절차의 세부 기능에 대한 설명은 본 기술규격에 따른다.

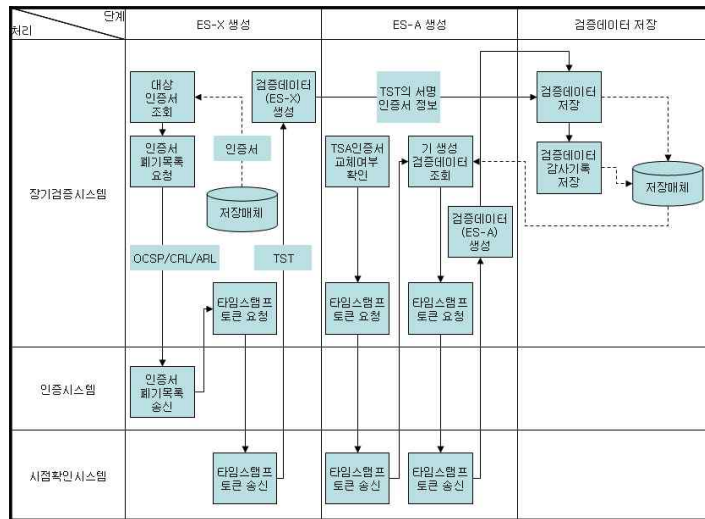
5.2.1.2 메시지 규격

전자서명 인증서 장기검증데이터 검증에서는 ES, ES-T, ES-C, ES-X, ES-A 형식의 장기검증데이터를 이용한다. 각 형식에 대한 설명은 전자서명 인증서 장기검증 규격 "5.1 전자서명 인증서 장기검증데이터"를 참조한다

**5.2.1.3 세부 처리 절차**

장기검증시스템은 기 등록된 인증서들의 CRL 정보를 취득하여 CRL의 해쉬 값에 대해 주기적으로 전자서명 인증서 장기검증데이터를 생성한다. 이 때 시점확인시스템의 인증서가 교체되었는지 또는 인증경로상의 인증기관 인증서가 폐기되었는지 등의 정보를 취득하며 이 정보를 바탕으로 ES-X만 만들 것인지 ES-A를 생성할 것인지 또는 ES-A를 추가할 것인지를 결정한다.

주기적인 전자서명 인증서 장기검증데이터의 생성을 위한 세부 처리 절차는 **그림 16**과 같다.



**그림 16 - 장기검증데이터 생성을 위한 세부절차**

**5.2.1.3.1 ES-X 생성**

- 1) 주기적으로 ES-X 생성을 요청한다.
- 2) 등록된 인증서들에 대한 인증서를 조회한다.
  - 등록된 인증서로부터 인증서폐기목록 배포지점을 추출하여 그 정보를 바탕으로 3의 절차를 진행시킬 수 있다.
  - 등록된 인증서폐기목록의 배포지점으로부터 3의 절차를 진행시킬 수 있다.

- 인증서폐기목록의 배포지점에 대한 정보 취득은 위의 두 가지 방법 중 하나 이상의 방법을 반드시 지원하여야한다.
- 3) 인증서들의 인증서폐기목록을 취득한다.
  - 4) 시점확인시스템의 인증서가 교체되었는지 파악한다.
  - 5) 인증서폐기목록에 대해 전자서명을 수행하여 ES를 생성한다.
  - 6) GPKI 라이브러리를 이용하여 ES를 입력으로 하는 타임스탬프토큰을 요청한다.
  - 7) ES-T를 생성한다.
  - 8) ES-C를 생성한다.
  - 9) GPKI 라이브러리를 이용하여 ES-C를 입력으로 하는 타임스탬프토큰을 요청한다.
  - 10) ES-X를 생성 및 저장한다.
  - 11) ES-X 생성에 대한 장기검증데이터 감사기록을 저장한다.

**5.2.1.3.2 ES-A 생성**

ES-X는 ES-C 형식의 장기검증데이터로부터 ESCTimeStampToken을 추가시킴으로서 장기검증데이터의 검증을 처리할 수 있도록 고안되었지만 ES-C 형식을 생성한 시점에서 사용된 알고리즘, 키, 다른 암호학적 데이터가 취약해질 경우, 암호 함수가 깨질 경우, 이전의 타임스탬프용 인증서가 만료된 경우에는 동작이 불가능하며 이를 보완하기 위해 ES-A 형식을 사용한다.

ES-A 생성에 대한 처리 절차는 아래와 같다.

- 1) ES-A 생성을 요청한다.
- 2) 기 생성된 장기검증데이터를 획득한다.
  - 기 생성된 장기검증데이터는 ES-X이거나 ES-A일 것이다.
  - 기 생성된 장기검증데이터가 ES-X일 경우 ES-A로 변경된다.
  - 기 생성된 장기검증데이터가 ES-A일 경우 TST를 하나 더 추가한다.
- 3) GPKI 라이브러리를 이용하여 기 생성된 장기검증데이터를 입력으로 하는 타임스탬프토큰을 요청한다.
- 4) ES-A를 생성 및 저장한다.
- 5) ES-A 생성에 대한 장기검증데이터 감사기록을 저장한다.

**- 시점확인시스템 인증서 교체**

시점확인시스템은 시점확인을 위한 신뢰성 있는 시간을 이용하여 특정시점에 특정 데이터가 존재했음을 증명해주는 기능을 제공하며 이를 증명하기 위해 전자서명 검증 및 시점확인시스템 인증서의 검증이 필요하다. 만약 시점확인시스템의 인증서가 키의 유출 및 보안상의 이유 또는 유효기간의 초과 또는 다른 이유로 폐기 또는 갱신, 재발급되었을 경우 해당 인증서를 교체하여야 할 것이다. 장기검증시스템은 시점확인시스템 인증서의 위치가 사전에 등록된 IP, PORT 정보를 이용하여 시점확인시스템 인증서가 교체되었는지 파악할 수 있는 기능을 제공하여야 한다.

시점확인시스템 인증서가 교체되면 해당 시점확인시스템 인증서로 발급된 토큰을 사용한 장기검증데이터에 대해 ES-A를 생성해야 한다.

**5.2.2 장기검증데이터 요청**

**5.2.2.1 처리 개요**

전자서명 인증서 장기검증데이터 요청은 전자서명에 대한 인증서 장기검증을 처리하기 위해 장기검증데이터를 획득하는 과정에서 사용된다. 이 과정에서는 전자서명이 생성된 시점정보와 전자서명용 인증서 정보가 사용되며, 그 결과로 인증서의 유효성 검증에 필요한 전자서명 인증서 장기검증데이터가 이용가능하게 된다.

장기검증데이터 요청에 대한 결과 장기검증데이터는 ES-X 형식이거나 ES-A 형식이 되는데, 각 형식에 대한 설명은 각각 전자서명 인증서 장기검증 규격 “5.1 전자서명 인증서 장기검증데이터”를 참조한다

**5.2.2.2 메시지 규격**

전자서명 인증서 장기검증데이터 요청에서는 장기검증데이터를 요청하는 데에 필요한 요청메시지와 요청메시지의 처리 결과를 응답하기 위한 응답메시지가 사용되며, 그 규격은 다음과 같다.

**5.2.2.2.1 요청 메시지**

요청 메시지 규격은 다음과 같다.

메시지 필드 명		type(크기)			
PlainMessage (1..1)	Header (1..1)	ProtocolType (1..1)	byte (1Byte)		
		ProtocolVersion (1..1)	byte (1Byte)		
		BodyLength (1..1)	int (4Byte)		
	Body (1..1)	RoamingMessage	RoamingMessageLength (1..1)	int (4Byte)	
			OpCode (1..1)	byte (1Byte)	
			ResultCode (1..1)	byte (1Byte)	
			ClientPLength (1..1)	byte (1Byte)	
			ClientIP (1..1)	string or binary	
			ClientPort (1..1)	byte (2Byte)	
			OrgCodeLength (1..1)	byte (1Byte)	
			OrgCode (1..1)	string or binary	
			ContentsList (1..1)	ContentsCount (1..1)	int (4Byte)
				VerifyDateLength (1..1)	int (4Byte)
				VerifyDate (1..1)	string
CRLDPLength (1..1)	int (4Byte)				
	CRLDP (1..1)	string			

요청 메시지 규격에서 사용하는 메시지 필드에 대한 세부내용은 다음과 같다.

필드명	설명	생성규칙	예
ProtocolType	프로토콜 형식	프로토콜 형식에 대한 정의는 PLAIN (0x14), ALERT (0x15), HANDSHAKE (0x16), APPLICATION_DATA (0x17) 가 있으며, 이중 반드시 0x14를 사용하여 PlainMessage를 표현함.	0x14
ProtocolVersion	프로토콜 버전	현 버전에서는 1	1
BodyLength	Body 데이터 크기	Body 데이터 크기로 Byte 단위	
RoamingMessageLength	RoamingMessage 데이터 크기	RoamingMessage 데이터 크기로 Byte 단위	
OpCode	서비스 Code로 opCode 테이블 참조	사용할 서비스로 특정 기능에 대한 코드로 원본을 이용한 전자서명인 SignedData 생성할 경우 0x01 Byte 값으로 표현, opCode 테이블 참조	0x01
ResultCode	작업에 대한 처리 결과	성공(0x00), 실패(0x01)로 구분	0x00
ClientPLength	ClientIP 데이터의 크기	int 형 값으로 ClientIP 데이터의	

		크기를 지정	
ClientIP	클라이언트의 IP 주소	String 형식으로 IP 주소를 표현	"1.1.1.1"
ClientPort	클라이언트의 소켓 포트	short 형식으로 클라이언트의 소켓 포트 정보	
OrgCodeLength	OrgCode 데이터의 크기	int 형 값으로 OrgCode 데이터의 크기를 지정	
OrgCode	기관코드	String 형식으로 기관코드 정보	
ContentsCount	ContentsList에 담긴 필드의 개수	ContentsList의 각 항목은 크기, 값 쌍으로 이루어 데이터의 처음 4Byte에 int 형 값으로 크기, 값 쌍의 개수를 표현	
VerifyDateLength	VerifyDate 데이터의 크기	int 형 값으로 VerifyDate 데이터의 크기를 지정	
VerifyDate	검증 일자	String 형식으로 검증일자를 표현	"yyyymmdd"
CRLDPLength	CRLDP 데이터의 크기	int 형 값으로 CRLDP 데이터의 크기를 지정	
CRLDP	인증서의 CRL 배포지점 URI	String 형식으로 인증서의 CRL 배포지점 URI를 표현	

5.2.2.2.2 응답 메시지

응답 메시지 규격은 다음과 같다.

메시지 필드 명			type(크기)		
PlainMessage (1..1)	Header (1..1)	ProtocolType (1..1)	byte (1Byte)		
		ProtocolVersion (1..1)	byte (1Byte)		
		BodyLength (1..1)	int (4Byte)		
	Body (1..1)	RoamingMessage	RoamingMessageLength (1..1)	int (4Byte)	
			OpCode (1..1)	byte (1Byte)	
			ResultCode (1..1)	byte (1Byte)	
			ClientPLength (1..1)	byte (1Byte)	
			ClientIP (1..1)	string or binary	
			ClientPort (1..1)	byte (2Byte)	
			OrgCodeLength (1..1)	byte (1Byte)	
			OrgCode (1..1)	string or binary	
			ContentsList (1..1)	ContentsCount (1..1)	int (4Byte)
				VerifyDataLength (1..1)	int (4Byte)
				VerifyData (1..1)	binary
PolicyDataLength (1..1)	int (4Byte)				
		PolicyData (1..1)	binary		

응답 메시지 규격에서 사용하는 메시지 필드에 대한 세부내용은 다음과 같다.

필드명	설명	생성규칙	예
ProtocolType	프로토콜 형식	프로토콜 형식에 대한 정의는 PLAIN (0x14), ALERT (0x15), HANDSHAKE (0x16), APPLICATION_DATA (0x17)가 있으며, 이중 반드시 0x14를 사용하여 PlainMessage를 표현함.	0x14
ProtocolVersion	프로토콜 버전	현 버전에서는 1	1
BodyLength	Body 데이터 크기	Body 데이터 크기로 Byte 단위	
RoamingMessageLength	RoamingMessage 데이터 크기	RoamingMessage 데이터 크기로 Byte 단위	
OpCode	서비스 Code로 opCode 테이블 참조	사용할 서비스로 특정 기능에 대한 코드로 원본을 이용한 전자서명인 SignedData 생성할 경우 0x01 Byte 값으로 표현, opCode	0x01



필드명	설명	생성규칙	예
		테이블 참조	
ResultCode	작업에 대한 처리 결과	성공(0x00), 실패(0x01)로 구분	0x00
ClientIPLength	ClientIP 데이터의 크기	int 형 값으로 ClientIP 데이터의 크기를 지정	
ClientIP	클라이언트의 IP 주소	String 형식으로 IP 주소를 표현	"1.1.1.1"
ClientPort	클라이언트의 소켓 포트	short 형식으로 클라이언트의 소켓 포트 정보	
OrgCodeLength	OrgCode 데이터의 크기	int 형 값으로 OrgCode 데이터의 크기를 지정	
OrgCode	기관코드	String 형식으로 기관코드 정보	
ContentsCount	ContentsList에 담긴 필드의 개수	ContentsList의 각 항목은 크기, 값 쌍으로 이루어 데이터의 처음 4Byte에 int 형 값으로 크기, 값 쌍의 개수를 표현	
VerifyDataLength	VerifyData 데이터의 크기	int 형 값으로 VerifyData 데이터의 크기를 지정	
VerifyData	검증 데이터	ResultCode가 성공일 경우 Binary 형식으로 검증 데이터를 표현하며, 실패일 경우 int 형 값으로 해당 에러코드를 표현	
PolicyDataLength	PolicyData 데이터의 크기	int 형 값으로 PolicyData 데이터의 크기를 지정	
PolicyData	정책 데이터	ResultCode가 성공일 경우 Binary 형식으로 정책 데이터를 표현하며, 실패일 경우 String 형식으로 해당 에러메시지를 표현	

### 5.2.2.3 세부 처리 절차

장기검증데이터 요청은 장기검증 이용자에 의해 이루어지며, 장기검증시스템에 의해 처리된다. 개념적으로 요청에 대한 메시지와 응답에 대한 메시지로 구분되며, 실제 처리 절차의 개념과 상세 절차는 그림 17과 같다.

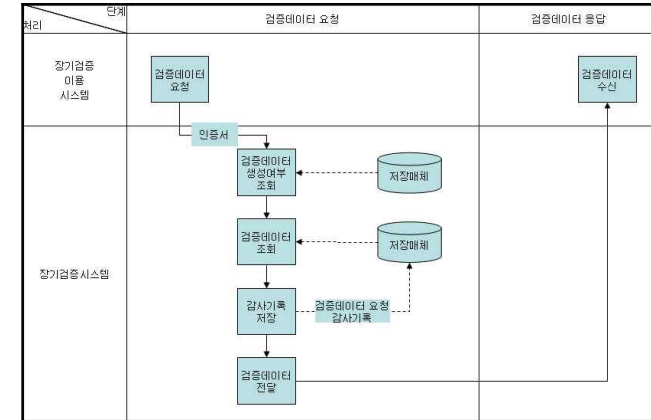


그림 17 - 검증데이터 요청 절차

- 1) 검증하고자 하는 인증서와, 검증 요청일을 장기검증시스템에 전송하여 장기검증데이터를 요청한다.
- 2) 장기검증시스템은 DB에서 장기검증데이터 생성 여부를 조회한다.
  - 생성이 되어있지 않을 경우 생성이 가능한지 확인하고 가능할 경우 장기검증데이터를 생성한다.
- 3) 장기검증시스템은 WORM에서 장기검증데이터를 조회한다.
- 4) 장기검증시스템은 검증데이터 요청에 대한 감사기록을 DB에 저장한다. 장기검증시스템으로부터 해당 인증서의 CRL 배포지점과 검증 요청일에 해당하는 CRL 관련 장기검증데이터를 수신한다.

처리 과정에서 요구되는 세부 처리 과정은 아래의 세부 기능의 명세를 참조한다.

### 5.2.3 장기검증데이터 검증

#### 5.2.3.1 처리 개요

전자서명 인증서 장기검증데이터 검증에서는 인증서의 유효성을 검증하는 절차를 처리한다. 이를 위해 전자서명 인증서 장기검증데이터 요청을 통해 장기검증데이터를 획득하고, 이렇게 획득한 장기검증데이터 활용 및 검증을 통해 인증서 유효성 확인이 가능해진다.

전자서명 인증서 장기검증데이터 처리 결과에 따라 아래의 2단계로 구분한다.

- 유효(valid) 상태 : 인증서가 검증을 통과했고 인증서 검증정책에 부합하다는 것을 의미한다. 인증서 검증정책은 인증서정책의 한 부분으로 인증서를 검증 시에 인증서를 생성하는 서명자와 검증자의 기술적인 요구사항을 정의한 것이다.
- 무효(Invalid) 상태 : 인증서 형식이 올바르지 않거나 인증서의 유효성 검증이 실패하는 경우이다.

#### 5.2.3.2 메시지 규격

전자서명 인증서 장기검증데이터 검증에서는 ES-X 또는 ES-A 형식의 장기검증데이터를 이용한다. 각 형식에 대한 설명은 각각 전자서명 인증서 장기검증규격 “5.1 전자서명 인증서 장기검증데이터”를 참조한다.

#### 5.2.3.3 세부 처리 절차

장기검증데이터 검증은 특정 인증서가 특정 시점에 유효했음을 증명함을 목적으로 하며 이를 위해 본 규격이 지정한 절차를 이행함으로써 전자서명용 인증서의 유효성 확인 내용의 적절성을 보장한다.

#### 5.2.3.3.1 장기검증데이터의 검증을 위한 주요 처리 절차

장기검증데이터 검증 시 아래의 과정을 준수하여 모든 검증이 성공일 경우 특정 시점에서 전자서명용 인증서의 유효성을 보장할 수 있다. 그렇지 않은 경우 유효성을 보장할 수 없으며 해당 인증서를 이용하여 전자서명한 데이터에 대한 유효성 역시 실패로 간주한다.

전자서명 인증서 장기검증데이터 검증의 세부 처리 절차는 아래와 같다.

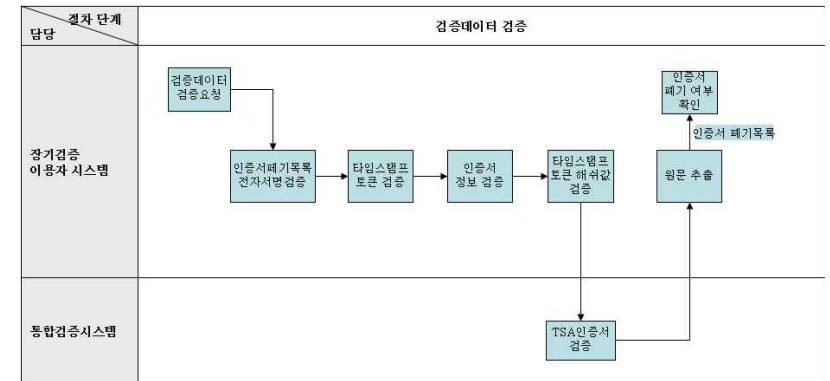


그림 18 - 장기검증데이터 검증의 세부처리 절차

- 1) 장기검증데이터 검증을 요청한다.
- 2) 장기검증데이터를 해석(파싱)하고, 장기검증데이터의 구문 무결성 및 값 무결성을 검증한다.
  - ① 인증서폐기목록의 전자서명의 서명값을 검증한다.
  - ② 타임스탬프토큰을 검증한다.
  - ③ 인증서 정보를 검증한다.
  - ④ 타임스탬프토큰의 해쉬값을 비교 검증한다.
  - ⑤ 시점확인시스템 인증서를 검증한다.
  - ⑥ 전자서명에 사용된 원문인 CRL 정보를 추출하여 검증한다.
- 3) 인증서 폐기 여부를 확인한다.

위의 모든 항목에 대해 검증결과가 유효하고, 인증서가 폐기된 상태가 아닐

경우 인증서에 대한 유효성을 보장한다. 다만 장기검증데이터의 검증은 전자서명 메시지의 검증 결과가 아닌 전자서명에 사용된 인증서에 대한 검증을 확인하는 과정이므로 유의한다. 각 전자서명 인증서 장기검증데이터 검증의 세부 처리 절차에 대한 상세설명은 아래와 같다.

**- 장기검증데이터 구문 검증**

검증데이터의 구조가 RFC 3126에 근거하여 적절한지, 연관된 값의 논리적인 구성에 문제가 없는지를 검증하여 검증데이터에서 검증에 필요한 핵심정보(CRL)를 추출가능함을 검증한다.

**- ES-X 형식의 검증 절차**

ES-X 형식의 검증 절차는 아래와 같이 ES-C 형식(①~③)의 검증절차에 추가적으로 ES-C 형식에서 사용된 인증서와 폐지정보에 대한 검증 절차가 추가된다.

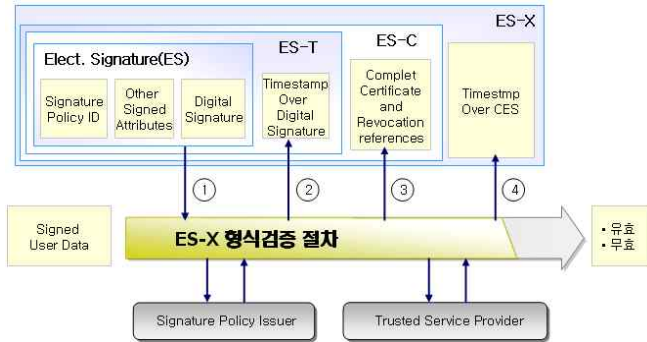


그림 19 - ES-X 형식 검증 절차

- 1) 장기검증시스템으로부터 장기검증데이터를 받는다.
- 2) ES 형식에서 인증서폐기목록의 전자서명 값을 체크하고 타임스탬프에 대한 검증을 수행한다.
- 3) 동일한 서명정책에 따라서 신뢰기관이 제공하는 부가적인 정보(예: CRL, 인증서)를 이용하여 인증서폐기목록 검증을 수행한다.
  - 모든 필요한 인증서와 폐지상태 정보를 얻는다.
  - ES 형식에 대하여 위에서 얻은 정보를 이용하여 모든 검증조건을 체크

한다.

- 사용된 인증서와 폐지에 관한 참조들을 기록한다.

4) CRL 등 인증 정보의 전자서명을 검증한다.

5) 사용자에게 검증 결과를 통보한다.

**- ES-A 형식의 검증 절차**

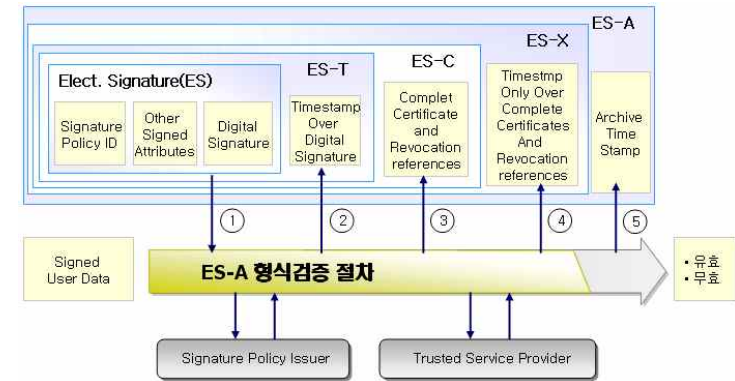


그림 20 - ES-A 검증 절차

- 1) 장기검증시스템으로부터 장기검증데이터를 받는다.
- 2) ES 형식에서 인증서폐기목록의 전자서명 값을 체크하고 타임스탬프에 대한 검증을 수행한다.
- 3) 동일한 서명정책에 따라서 신뢰기관이 제공하는 부가적인 정보(예: CRL, 인증서)를 이용하여 인증서폐기목록 검증을 수행한다.
  - 모든 필요한 인증서와 폐지상태 정보를 얻는다.
  - ES 형식에 대하여 위에서 얻은 정보를 이용하여 모든 검증조건을 체크한다.
  - 사용된 인증서와 폐지에 관한 참조들을 기록한다.
- 4) CRL 등 인증 정보의 전자서명을 검증한다.
- 5) ES-A 형식이 지정하는 해쉬값을 체크하고 타임스탬프에 대한 검증을 수행한다.
- 6) 사용자에게 검증 결과를 통보한다.

**- 인증서폐기목록 검증**

인증서폐기목록의 구조가 RFC 3280에 근거하여 적절한지, 연관된 값의 논리적인 구성에 문제가 없는지를 검증하여 인증서폐기목록의 전자서명의 검증을 통해 신뢰할 수 있는 폐기목록인지를 검증하여 검증에 필요한 핵심정보를 추출가능한가를 검증한다.

**- 타임스탬프토큰 검증**

GPKI의 시점확인 서비스를 이용하여 생성된 타임스탬프토큰의 유효성을 검증한다.

**- 인증서 정보 검증**

인증서의 데이터가 X.509 규격에 적절한가를 검증하며 유효기간에 대한 검증을 제외하며 RFC 3280에 정의된 인증경로에 대한 검증에 대해서는 장기검증 데이터에 포함되었거나 혹은 저장된 상위 인증서목록에 대한 개별 검증을 수행하도록 한다.