NAK-P-2008-06

전자서명 장기검증 통합연계 API 기술규격

The API Specification of Integrated Interface for Long-Term Validation of Digital Signature

2008년 12월 23일 제정

○ 제 정 자 : 행정안전부 국가기록원장
○ 제 정 일 : 2008년 12월 23일(행정안전부 고시 제 2008-52호)
○ 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

이 표준은 행정안전부 국가기록원의 '전자기록물 전자서명 장기검증관리체계'에서 제공하는 기능을 각급 기록관리시스템에서 이용하기 위해 필요한 통합연계 API를 정의한 기술규격으로, 표준전문위원회의 전문심의 및 국가기록관리위원회의 심의를 거쳐 제정한 공공표준이다.

이 표준의 법률적 근거는 다음과 같다.

- 공공기록물 관리에 관한 법률 제20조
- 공공기록물 관리에 관한 법률 시행령 제32조 4항, 제35조 2항, 제36조 2항, 제40조 3항, 제44조 2항, 제46조 2항 및 5항
- 전자서명법 제26조의2, 제26조의3
- 전자정부법 제20조
- 전자정부법 시행령 제16조, 제33조의5

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 보호 대상이 되는 저작물이다.

전자서명 장기검증 통합연계 API 기술규격

1 적용범위

이 표준은 각급 기록관리시스템 및 영구기록관리시스템이 국가기록원에서 제공하는 전자서명 장기검증관리체계(장기검증시스템, 진본확인시스템, 통합전자서명관리시스템) 및 행정전자서명인증체계와 연계하고자 하는 경우에 적용한다.

2 인용표준

이 표준은 다음의 표준을 참조하여 관련 조항을 구성하였다.

- PKCS#7 Cryptographic Message Syntax(RFC 2315)
- Cryptographic Message Syntax(RFC 2630)
- Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List(CRL) Profile(RFC 3280)
- 전자서명 장기검증기술에 대한 프로세스 정의(RFC 3126)
- 인증서 검증 및 공증 서비스를 위한 프로토콜 정의(RFC 3029)
- 전자문서 증명서 포맷 및 운용절차 기술규격(한국전자거래진흥원)

3 용어정의

3.1 용어

3.1.1 장기검증관리체계

국가기록원에서 시행하는 전자기록물에 대한 전자서명 및 검증 시 이용되는 시스템. 전자서명장기검증시스템, 진본확인시스템, 통합전자서명시스템으로 구성되어 있다.

3.1.2 장기검증통합연계 API
장기검증관리체계와 연계하는 각급 기록관리시스템 등에서 전자기록물 전자서명 장기검증 등의 기능을 적용할 때 사용하는 API함수

3.1.3 전자서명
문서나 메시지를 보낸 사람의 신원이 진짜임을 증명하기 위해 사용되는 서명. 이것은 또한 전달된 메시지나 문서의 원래 내용이 변조되지 않았다는 것을 보증하기 위해 사용될 수도 있다. 전자서명을 사용함으로써 얻어질 수 있는 부가적인 이득은, 전자서명이 쉽게 전송될 수 있고, 쉽게 부인할 수 없으며, 다른 사람이 흉내 낼 수 없고, 타임스탬프가 자동으로 유지될 수 있다는 점 등이다.

3.1.4 개인키
암호/복호를 위해 비밀 메시지를 교환하는 당사자만이 알고 있는 키.

3.1.5 인증서
디지털 인증서는 웹상에서 비즈니스 또는 기타의 거래를 수행할 때, 사용자의 자격을 확립하는 일종의 "전자 신용카드". 이것은 인증기관으로부터 발급되며, 수령인이 그 인증서의 진위여부를 확인할 수 있도록 소유자의 이름, 일련번호, 유효기간, 인증서 소유자의 공개키 사본 (메시지나 전자서명의 암호화 및 복원에 사용됨), 그리고 인증서 발급기관의 전자서명 등이 포함된다. 일부 디지털 인증서는 X.509 표준을 따른다. 디지털 인증서는 인증된 사용자들이 다른 사용자들의 공개키를 볼 수 있도록 등록장소에 보관될 수 있다.

3.1.6 해쉬
해쉬는 하나의 문자열을 원래의 것을 상징하는 더 짧은 길이의 값이나 키로 변환하는 절차

3.1.7 타임스탬프 서버
시점확인을 요청한 전자문서에 대하여 당해 전자문서가 인증기관에 제시된 특정시점을 확인하여 알려주는 기관

3.1.8 타임스탬프 토큰
타임스탬프 서버로부터 발급받은 시각 검증 정보

3.1.9 감사기록
장기검증 데이터를 생성하는 기록

3.1.10 장기검증데이터
전자기록물 전자서명을 검증하는데 필요한 인증서 유효성 검증데이터

3.1.11 전자서명문
인증서를 통해 전자서명을 완료한 후의 데이터 포맷. 보통 전자서명의 결과값을 말한다.

3.1.12 진본확인서
장기검증관리체계의 진본확인체계에서 발급되는 증명서. 전자기록물의 장기보존패키지 내에 삽입되어 전자기록물에 대한 진본확인을 증명한다.

3.2 약어

3.2.1 NAK
National Archives of Korea(국가기록원)

3.2.2 LTVS
Long Term Verification System(장기검증 시스템)

3.2.3 TSA
TimeStamping Autority

3.2.4 TST
TimeStamp Token

3.2.5 GPKI
Government Public Key Infrastructure (행정전자서명 인증관리체계)

4 장기검증 통합연계API

4.1 장기검증 통합연계API 정의

각급 기록관리시스템 등에서 전자기록물 전자서명, 전자서명 장기검증, 진본확인 및 통합전자서명 처리를 위해 행정전자서명인증관리센터 통합검증서버 또는 장기검증관리체계 등과 연동할 경우 해당 기능을 효율적으로 이용할 수 있도록 제공하는 API를 말한다.

4.2 장기검증 통합연계API 목적

각급 기록관리시스템 등이 장기검증관리체계와 상호 연동이 가능하도록 제공 기능 목록, 특정 처리를 요청하고 응답하는 메시지 규격 등을 정의하여 제공해야 한다. 그러나 다양한 시스템 지원 및 적용의 용이성 측면에서 관련 API를 통합하여 특정 기능을 사용할 수 있도록 체계적인 사용방안을 제공하는 것이 바람직하기 때문에 장기검증 통합연계API가 요구된다.

4.3 장기검증 통합연계API 제공 기능범위

장기검증 통합연계API는 각급 기록관리시스템 등의 시스템 개발 지원 및 운용을 고려하여 **그림 1**과 같이 GPKI, 장기검증시스템, 진본확인시스템 및 통합전자서명관리시스템과 연계할 수 있는 전자서명, 전자서명 장기검증 등의 API 기능을 제공한다.

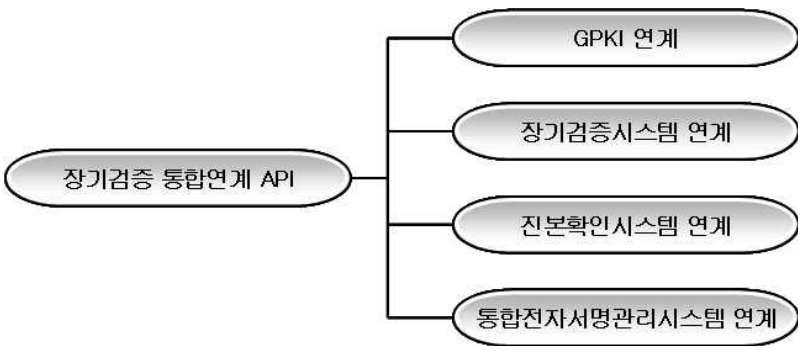

그림 1 - 장기검증통합연계API 연계 기능범위

4.4 장기검증통합연계API 연계 방법

각급 기록관리시스템이 장기검증관리체계의 장기검증시스템, 통합전자서명관리시스템, 진본확인시스템을 이용할 경우를 고려하여 그에 맞는 연계방법을 제공해야 한다. 이를 위해 장기검증통합연계API는 장기검증관리체계 각 시스템이 제공하는 기능에 근거하여 장기검증통합연계API를 구성하고 제공한다.

또한, 각급 기록관리시스템이 GPKI를 이용하는 업무에 있어서 필요한 연계 기능을 추가적으로 제공한다.
장기검증통합연계API를 사용함으로써 각급 기록관리시스템 등에서 전자서명, 전자서명 장기검증 등 각 기능을 처리할 경우 표준 절차를 구성할 수 있도록 보장할 수 있게 된다. 그리고 장기검증통합연계API를 통한 처리결과로 실패 등의 내용이 발생 한 경우에는 각급 기록관리시스템이 해당 처리결과에 따라 적절히 처리방안을 정하게 한다.
통합연계API의 이용자와 통합연계API 모듈의 개념도는 **그림 2**와 같다.

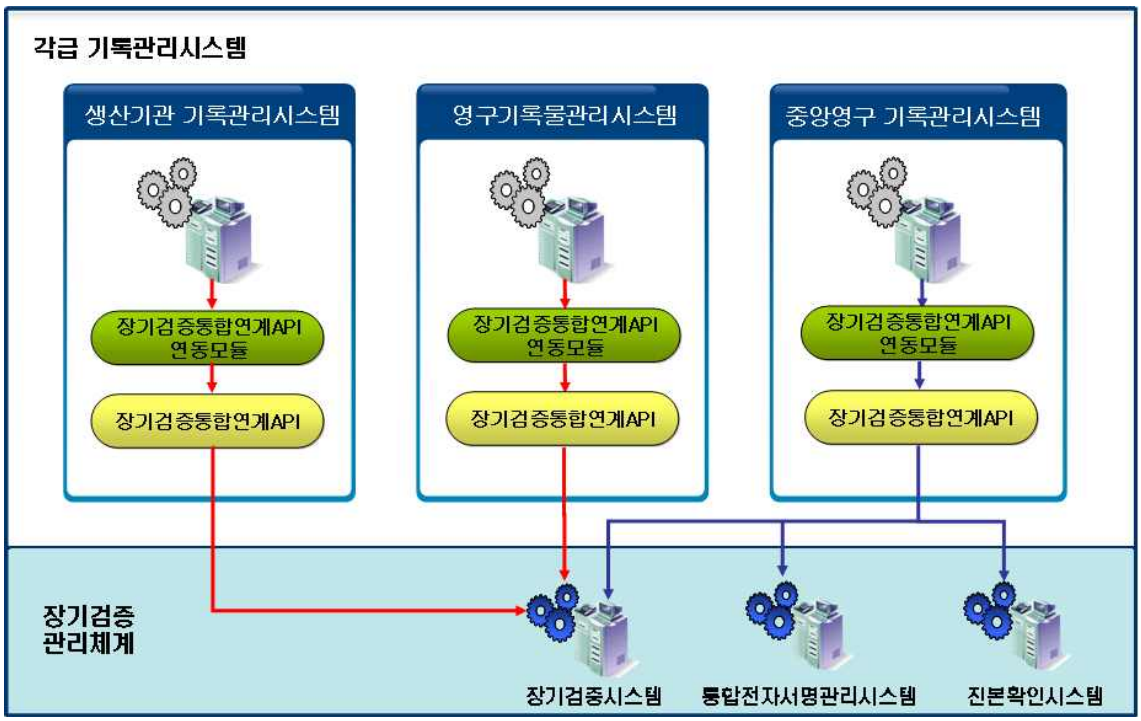

그림 2 - 장기검증통합연계API 연계 모듈

5 장기검증통합연계API 세부 내용

5.1 장기검증통합연계API 기능

각급 기록관리시스템과의 원활한 연계가 가능하도록 제공하는 장기검증통합연계API 기능은 다음과 같다.

- GPKI 연계
- 장기검증시스템 연계
- 통합전자서명관리시스템 연계
- 진본확인시스템 연계

5.1.1 GPKI 연계

행정전자서명에서 제공되고 있는 기능을 사용할 수 있도록 장기검증통합연계API에 의해 제공되는 기능은 아래와 같다.
- 시점확인
- 인증서 검증
- 인증서 관리
- 개인키 관리
- Base64 인코딩

5.1.2 장기검증시스템 연계

전자서명 장기검증시스템과의 연계를 위해 제공되는 기능은 아래와 같다.

- 전자서명 장기검증 요청
- 전자서명 장기검증

5.1.3 통합전자서명관리시스템 연계

통합전자서명관리시스템과의 연계를 위해 제공되는 처리기능은 아래와 같다.

- 전자서명 생성

- 전자서명 검증

5.1.4 진본확인시스템 연계

진본확인시스템 연계를 위해 제공되는 처리기능은 아래와 같다.

- 진본확인서 발급
- 진본확인서 검증

5.2 장기검증통합연계API 데이터 구조

장기검증통합연계API 사용을 위해 별도의 데이터 구조를 사용하지는 않는다. 그러나 장기검증통합연계API를 통해 장기검증시스템, 통합전자서명관리시스템 및 진본확인시스템에서 정의한 데이터 구조를 처리하게 된다.

5.3 장기검증통합연계API 처리 세부내용

5.3.1 GPKI API 연계

5.3.1.1 시점확인 모듈

5.3.1.1.1 NAK_G_TSP_MakeTimeStampToken

시점확인 토큰을 생성한다.

```
int NAK_G_TSP_MakeTimeStampToken(
        void * pVoid,
        char * pcInTsaIP,
        int nInTsaPort,
        unsigned char * pucInHashData,
        int nInHashDataLen,
        unsigned char ** ppucOutTSACert,
        int * pnOutTSACertLen,
        unsigned char ** ppucOutToken,
        int * pnOutToeknLen,
        int nHashAlg
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcInTsaIP [in] TSA 서버의 IP
nInTsaPort [in] TSA 서버의 Port
pucInHashData [in] 시점확인을 요청할 전자 문서의 해쉬값
nInHashDataLen [in] 시점확인을 요청할 전자 문서의 해쉬값 길이
ppucOutTSACert [out] TSA 서버의 인증서 데이터
pnOutTSACertLen [out] TSA 서버의 인증서 데이터 길이
ppucOutToken [out] 타임 스탬프 토큰 데이터
pnOutToeknLen [out] 타임 스탬프 토큰 데이터 길이
nHashAlg [in] 시점확인을 요청할 전자 문서 메시지 다이제스트 알고리즘

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

시점확인 토큰을 생성 하는 API로써 시점확인서버IP, 시점확인서버Port, 시점확인을 요청할 전자문서, Hash 알고리즘 등을 Input으로 요구한다. 획득한 TimeStamp 토큰은 추후, NAK_G_TSP_VerifyTimeStampToken 함수를 이용해 검증 할 수 있으며, NAK_G_TSP_ParseTimeStampToken 함수를 이용해 내용을 확인할 수 있다. 획득한 TSA서버 인증서는 검증할 것을 권장한다.

· 요구사항

nHashAlg Type은 헤더파일에 다음과 같이 정의되어 있다.

```
#define HASH_ALG_SHA1   0x01   /* SHA1 */
#define HASH_ALG_MD5   0x02   /* MD5 */
#define HASH_ALG_HAS160   0x03   /* HAS160 */
#define HASH_ALG_SHA256   0x04   /* SHA256 */
```

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_TSP_VerifyTimeStampToken
NAK_G_TSP_ParseTimeStampToken

NAK_G_TSP_GetTSTGenTime

5.3.1.1.2 NAK_G_TSP_VerifyTimeStampToken

시점확인 토큰을 검증한다.

```
int NAK_G_TSP_VerifyTimeStampToken (
        void * pVoid,
        unsigned char * pucInHashData,
        int nInHashDataLen,
        unsigned char * pucInToken,
        int nInTokenLen
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInHashData [in] 시점확인을 요청했던 전자 문서의 해쉬값
nInHashDataLen [in] 시점확인을 요청했던 전자 문서의 해쉬값 길이
pucInToken [in] 시점확인 토큰
nInTokenLen [in] 시점확인 토큰 길이

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

시점확인 토큰을 검증 하는 API로써 시점확인을 요청했던 전자문서, 시점확인 토큰 등을 Input으로 요구한다.

· **요구사항**

해당사항 없음.

· **참고**

NAK_G_TSP_MakeTimeStampToken
NAK_G_TSP_ParseTimeStampToken
NAK_G_TSP_GetTSTGenTime

5.3.1.1.3 NAK_G_TSP_ParseTimeStampToken

시점확인 토큰의 내용을 확인한다.

```
int NAK_G_TSP_ParseTimeStampToken(
        void * pVoid,
        unsigned char * pucInToken,
        int nInTokenLen,
        unsigned char ** ppucOutCN,
        int * pnOutCNLen,
        unsigned char ** ppucOutDN,
        int * pnOutDNLen,
        unsigned char ** ppucOutPolicy,
        int * pnOutPolicyLen,
        unsigned char ** ppucOutHashAlg,
        int * pnOutHashAlgLen,
        unsigned char ** ppucOutHashValue,
        int * pnOutHashValueLen,
        unsigned char ** ppucOutSerialNum,
        int * pnOutSerialNumLen,
        unsigned char ** ppucOutGenTime,
        int * pnOutGenTimeLen,
        unsigned char ** ppucOutNonce,
        int * pnOutNonceLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInToken [in] 시점확인 토큰
nInTokenLen [in] 시점확인 토큰 길이
ppucOutCN [out] 서버 인증서의 DN의 CN값
pnOutCNLen [out] 서버 인증서의 DN의 CN값 길이
ppucOutDN [out] 서버 인증서의 DN값
pnOutDNLen [out] 서버 인증서의 DN값 길이
ppucOutPolicy [out] 서버가 사용한 시점확인 토큰 생성 정책 식별자
pnOutPolicyLen [out] 서버가 사용한 시점확인 토큰 생성 정책 식별자 길이
ppucOutHashAlg [out] 시점확인을 요청한 데이터의 메시지 다이제스트를 생성하는데 사용된 해쉬 알고리즘
pnOutHashAlgLen [out] 시점확인을 요청한 데이터의 메시지 다이제스트를 생성하는데 사용된 해쉬 알고리즘 길이
ppucOutHashValue [out] 시점확인을 요청한 데이터의 메시지 다이제스트
pnOutHashValueLen [out] 시점확인을 요청한 데이터의 메시지 다이제스트 길이

ppucOutSerialNum [out] 시점확인 토큰의 일련번호
pnOutSerialNumLen [out] 시점확인 토큰의 일련번호 길이
ppucOutGenTime [out] 시점확인 토큰의 생성 시간
pnOutGenTimeLen [out] 시점확인 토큰의 생성 시간 길이
ppucOutNonce [out] 응답 메시지의 재사용 방지를 위한 랜덤값
pnOutNonceLen [out] 응답 메시지의 재사용 방지를 위한 랜덤값 길이

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

시점확인 토큰의 내용을 확인하는 API로써 시점확인 토큰 등을 Input으로 요구한다.

· **요구사항**

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· **참고**

NAK_G_TSP_MakeTimeStampToken
NAK_G_TSP_VerifyTimeStampToken
NAK_G_TSP_GetTSTGenTime

5.3.1.1.4 NAK_G_TSP_GetTSTGenTime

인증서 검증을 수행한다.

```
int NAK_G_TSP_GetTSTGenTime (
        void * pVoid,
        unsigned char * pucInToken,
        int nInTokenLen,
        unsigned char ** ppucOutGenTime,
        int * pnOutGenTimeLen
)
```

· **파라미터(Parameters)**

void [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInToken [in] 시점확인 토큰
nInTokenLen [in] 시점확인 토큰 길이
ppucOutGenTime [out] 시점확인 토큰의 생성 시간
pnOutGenTimeLen [out] 시점확인 토큰의 생성 시간 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

시점확인 토큰의 생성시간은 "2007-01-26T16:02:09-09:00"와 같은 형식으로 출력된다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_TSP_MakeTimeStampToken
NAK_G_TSP_VerifyTimeStampToken
NAK_G_TSP_ParseTimeStampToken

5.3.1.2 인증서 검증 모듈

5.3.1.2.1 NAK_G_CER_VerifyCert

인증서 검증을 수행한다.

```
int NAK_G_CER_VerifyCert(
        void * pVoid,
        char * pcInConfigFilePath,
        char * pcInCertPolicies,
        int nCertType,
        unsigned char * pucInTrustedCert,
        int nInTrustedCertLen,
        unsigned char * pucInCert,
```

```
        int nInCertLen,
        unsigned char * pucInMyCert = NULL,
        int nInMyCertLen = 0,
        unsigned char * pucInMyPriKey = NULL,
        int nInMyPriKeyLen = 0
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcInConfigFilePath [in] 환경 파일 위치
pcInCertPolicies [in] 허용하는 인증서 정책 목록
nCertType [in] 검증할 인증서 종류
pucInTrustedCert [in] 신뢰하는 최상위 인증기관 인증서
nInTrustedCertLen [in] 신뢰하는 최상위 인증기관 인증서 길이
pucInCert [in] 검증할 인증서
nInCertLen [in] 검증할 인증서 길이
pucInMyCert [in] 검증 요청자의 인증서
nInMyCertLen [in] 검증 요청자의 인증서 길이
pucInMyPriKey [in] 검증 요청자의 인증서 개인키
nInMyPriKeyLen [in] 검증 요청자의 인증서 개인키 길이

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

인증서 검증 절차
① 인증서 경로 획득
② 인증서 경로 구성
③ 최상위 인증기관 인증서 신뢰성 확인
④ 인증서 경로 검증
⑤ 인증서 경로의 폐지 여부 확인

· **요구사항**

해당사항 없음.

· **참고**

NAK_G_CER_GetSerialNum
NAK_G_CER_GetIssuerName
NAK_G_CER_GetValidity
NAK_G_CER_GetSubjectName
NAK_G_CER_GetPubKeyAlg
NAK_G_CER_GetCRLDP

5.3.1.3 인증서 관리 모듈

5.3.1.3.1 NAK_G_CER_GetSerialNum

인증서의 일련번호를 획득한다.

```
int NAK_G_CER_GetSerialNum (
        void * pVoid,
        unsigned char * pucInCertificate,
        int nInCertificateLen,
        unsigned char ** ppucOutBuffer,
        int * pnOutBufferLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInCertificate [in] 인증서 데이터
nInCertificateLen [in] 인증서 데이터의 길이
ppucOutBuffer [out] 인증서의 식별번호
pnOutBufferLen [out] 인증서의 식별번호 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

일련번호는 인증기관에서 발급하는 인증서에 부여하는 해당 인증기관에서 유일한 숫자값이다.
일련번호는 Hex값으로 출력된다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_CER_VerifyCert
NAK_G_CER_GetIssuerName
NAK_G_CER_GetValidity
NAK_G_CER_GetSubjectName
NAK_G_CER_GetPubKeyAlg
NAK_G_CER_GetCRLDP

5.3.1.3.2 NAK_G_CER_GetIssuerName

인증서의 발급자 이름을 확인한다.

```
int NAK_G_CER_GetIssuerName (
        void * pVoid,
        unsigned char * pucInCertificate,
        int nInCertificateLen,
        unsigned char ** ppucOutBuffer,
        int * pnOutBufferLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInCertificate [in] 인증서 데이터
nInCertificateLen [in] 인증서 데이터의 길이
ppucOutBuffer [out] 인증서 발급자 정보
pnOutBufferLen [out] 인증서 발급자 정보 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

인증서의 발급자 이름을 확인한다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_CER_VerifyCert
NAK_G_CER_GetSerialNum
NAK_G_CER_GetValidity
NAK_G_CER_GetSubjectName
NAK_G_CER_GetPubKeyAlg
NAK_G_CER_GetCRLDP

5.3.1.3.3 NAK_G_CER_GetValidity

인증서의 유효기간을 확인한다.

```
int NAK_G_CER_GetValidity (
        void * pVoid,
        unsigned char * pucInCertificate,
        int nInCertificateLen,
        unsigned char ** ppucOutBuffer,
        int * pnOutBufferLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInCertificate [in] 인증서 데이터
nInCertificateLen [in] 인증서 데이터의 길이
ppucOutBuffer [out] 인증서의 유효기간
pnOutBufferLen [out] 인증서의 유효기간 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조

int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

유효기간은 “2003-06-12 14:51:41 ~ 2003-09-12 14:51:41” 와 같은 형식으로 리턴된다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_CER_VerifyCert
NAK_G_CER_GetSerialNum
NAK_G_CER_GetIssuerName
NAK_G_CER_GetSubjectName
NAK_G_CER_GetPubKeyAlg
NAK_G_CER_GetCRLDP

5.3.1.3.4 NAK_G_CER_GetSubjectName

인증서의 소유자 이름을 확인한다.

```
int NAK_G_CER_GetSubjectName (
        void * pVoid,
        unsigned char * pucInCertificate,
        int nInCertificateLen,
        unsigned char ** ppucOutBuffer,
        int * pnOutBufferLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInCertificate [in] 인증서 데이터
nInCertificateLen [in] 인증서 데이터의 길이
ppucOutBuffer [out] 인증서의 소유자
pnOutBufferLen [out] 인증서의 소유자 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

인증서의 소유자 이름을 확인한다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_CER_VerifyCert
NAK_G_CER_GetSerialNum
NAK_G_CER_GetIssuerName
NAK_G_CER_GetValidity
NAK_G_CER_GetPubKeyAlg
NAK_G_CER_GetCRLDP

5.3.1.3.5 NAK_G_CER_GetPubKeyAlg

인증서의 공개키 알고리즘을 획득한다.

```
int NAK_G_CER_GetPubKeyAlg (
        void * pVoid,
        unsigned char * pucInCertificate,
        int nInCertificateLen,
        unsigned char ** ppucOutBuffer,
        int * pnOutBufferLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInCertificate [in] 인증서 데이터
nInCertificateLen [in] 인증서 데이터의 길이
ppucOutBuffer [out] 공개키 알고리즘
pnOutBufferLen [out] 공개키 알고리즘 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

알고리즘은 RSA인 경우는 “rsaEncryption”, KCDSA인 경우는 “kcdsa1WithSHA1”, ECC 인 경우는 “ecPublicKey” 와 같이 리턴된다.
알 수 없는 알고리즘인 경우에는 알고리즘의 OID값이 출력된다. (예제 : “1 2 840 113549 1 1 1”)

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_CER_VerifyCert
NAK_G_CER_GetSerialNum
NAK_G_CER_GetIssuerName
NAK_G_CER_GetValidity
NAK_G_CER_GetSubjectName
NAK_G_CER_GetCRLDP

5.3.1.3.6 NAK_G_CER_GetCRLDP

인증서의 CRL 배포지점을 확인한다.

```
int NAK_G_CER_GetCRLDP (
        void * pVoid,
        unsigned char * pucInCertificate,
        int nInCertificateLen,
        unsigned char ** ppucOutBuffer,
        int * pnOutBufferLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInCertificate [in] 인증서 데이터
nInCertificateLen [in] 인증서 데이터의 길이

ppucOutBuffer [out] 인증서 배포지점 정보
pnOutBufferLen [out] 인증서 배포지점 정보 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

인증서의 CRL 배포지점을 확인한다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_CER_VerifyCert
NAK_G_CER_GetSerialNum
NAK_G_CER_GetIssuerName
NAK_G_CER_GetValidity
NAK_G_CER_GetSubjectName
NAK_G_CER_GetPubKeyAlg

5.3.1.4 개인키 관리 모듈

5.3.1.4.1 NAK_G_PRI_Decrypt

암호화된 개인키를 비밀번호로 복호화 한다.

```
int NAK_G_PRI_Decrypt (
        void * pVoid,
        char * pcInPasswd,
        unsigned char * pucInEncPriKey,
        int nInEncPriKeyLen,
        unsigned char ** ppucOutPurePriKey,
        int * pnOutPurePriKeyLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcInPasswd [in] 개인키의 비밀번호
pucInEncPriKey [in] 암호화된 개인키
nInEncPriKeyLen [in] 암호화된 개인키 길이
ppucOutPurePriKey [out] 복호화된 개인키
pnOutPurePriKeyLen [out] 복호화된 개인키 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

암호화된 개인키를 비밀번호로 복호화 한다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

해당사항 없음

5.3.1.5 BASE64 모듈

5.3.1.5.1 NAK_G_B64_Encode

BASE64 인코딩을 수행한다.

```
int NAK_G_B64_Encode(
        void * pVoid,
        unsigned char * pucInData,
        int nInDataLen,
        unsigned char ** ppucOutEncData,
        int * pnOutEncDataLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInData [in] BASE64 인코딩할 데이터
nInDataLen [in] BASE64 인코딩할 데이터 길이
ppucOutEncData [out] 인코딩 된 데이터
pnOutEncDataLen [out] 인코딩 된 데이터 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

데이터를 BASE64 인코딩 즉, 바이너리 데이터를 아스키 텍스트로 변환한다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_B64_Decode

5.3.1.5.2 NAK_G_B64_Decode

BASE64 인코딩을 수행한다.

```
int NAK_G_B64_Decode (
        void * pVoid,
        unsigned char * pucInEncData,
        int nInEncDataLen,
        unsigned char ** ppucOutData,
        int * pnOutDataLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucInEncData [in] BASE64 디코딩할 데이터
nInEncDataLen [in] BASE64 디코딩할 데이터 길이
ppucOutData [out] 디코딩 된 데이터
pnOutDataLen [out] 디코딩 된 데이터 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

데이터를 BASE64 디코딩 즉, 아스키 텍스트를 바이너리 데이터로 변환한다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_G_B64_Encode

5.3.2 장기검증체계 연계

5.3.2.1 장기검증 수행

5.3.2.1.1 NAK_L_API_SetCVAServerInfo

장기검증 서버의 정보를 입력한다.

```
int NAK_L_API_SetCVAServerInfo(
        void * pVoid,
        char * pcCVAServerIP,
        int nCVAServerPort,
        char * pcSysCode
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcCVAServerIP [in] 장기검증 서버의 IP
nCVAServerPort [in] 장기검증 서버의 Port
pcSysCode [in] 기관코드

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

전자서명 장기검증 서버의 정보를 입력한다.

· **요구사항**

해당사항 없음

· **참고**

NAK_L_API_LongTermVerifyCert

5.3.2.1.2 NAK_L_API_LongTermVerifyCert

전자 서명 장기검증 서버로부터 받은 검증데이터를 검증한다. 또한 인증서 검증을 수행한다.

```
int NAK_L_API_LongTermVerifyCert (
        void * pVoid,
        char * pcInConfigFilePath,
        char * pcInSignedDate,
        unsigned char * pucInCert,
        int nInCertLen,
        unsigned char * pucInMyCert,
        int nInMyCertLen
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcInConfigFilePath [in] 통합검증서버 정보를 포함하고 있는 환경파일

(gpkiapi.conf) 위치
pcInSignedDate [in] 전자서명을 생성한 날짜(장기검증을 받고자 하는 날짜)
pucInCert [in] 검증하고자 하는 인증서
nInCertLen [in] 검증하고자 하는 인증서 길이
pucInMyCert [in] 검증 요청자의 인증서
nInMyCertLen [in] 검증 요청자의 인증서 길이

· 리턴값(Return Values)

성공: 0
실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

인증서를 이용하여 전자 서명 장기검증을 수행한다.

· 요구사항

해당사항 없음

· 참고

NAK_L_API_SetCVAServerInfo

5.3.3 진본확인시스템 연계

5.3.3.1 증명서 발급 및 검증

5.3.3.1.1 NAK_L_API_SetDCAServerInfo

진본확인 서버의 정보를 입력한다.

```
int NAK_L_API_SetDCAServerInfo(
        void * pVoid,
        char * pcDCAServerIP,
        int nDCAServerPort,
        char * pcSysCode
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcCVAServerIP [in] 진본확인 서버의 IP
nCVAServerPort [in] 진본확인 서버의 Port
pcSysCode [in] 기관코드

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

진본확인 서버의 정보를 입력한다.

· 요구사항

해당사항 없음

· 참고

NAK_D_API_RequestARCCertInfo
NAK_D_API_VerifyARCCertInfo
NAK_D_API_ParseARCCertInfo
NAK_D_API_ReleaseARCCertInfo

5.3.3.1.2 NAK_D_API_RequestARCCertInfo

증명서 정보를 요청한다.

```
int NAK_D_API_RequestARCCertInfo(
        void * pVoid,
        char * pcPolicyOID,
        IN_TARGET_ORG_AND_ISSUED stOrgAndIssued,
        unsigned char ** ppucOutARCCertInfo,
        int * pnOutARCCertInfoLen,
        int * pnOutDocHashAlg
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수

pcPolicyOID [in] 증명서 정책 OID
stOrgAndIssued [in] 증명서 원문증명 정보
ppucOutARCCertInfo [out] 증명서 정보
pnOutARCCertInfoLen [out] 증명서 길이
pnOutDocHashAlg [out] 전자문서 해쉬 알고리즘

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

해당사항 없음

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_D_API_SetDCAServerInfo
NAK_D_API_VerifyARCCertInfo
NAK_D_API_ParseARCCertInfo
NAK_D_API_ReleaseARCCertInfo

5.3.3.1.3 NAK_D_API_VerifyARCCertInfo

증명서를 검증한다.

```
int NAK_D_API_VerifyARCCertInfo(
        void * pVoid,
        char * pcCertToolkitConfPath,
        unsigned char * pucARCCertInfo,
        int nARCCertInfoLen,
        unsigned char * pucRootCert,
        int nRootCertLen,
        ARCCERT_VERIFY_ITEM stACVeriftItem
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcCertToolkitConfPath [in] 증명서 환경설정파일 위치
pucARCCertInfo [in] 증명서 정보
nARCCertInfoLen [in] 증명서 길이
pucRootCert [in] 최상위 인증서
nRootCertLen [in] 최상위 인증서 길이
stACVeriftItem [in] 증명서 검증 항목 정보

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

해당사항 없음

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_D_API_SetDCAServerInfo
NAK_D_API_RequestARCCertInfo
NAK_D_API_ParseARCCertInfo
NAK_D_API_ReleaseARCCertInfo

5.3.3.1.4 NAK_D_API_ParseARCCertInfo

증명서를 파싱하여 그 결과를 얻는다.

```
int NAK_D_API_ParseARCCertInfo(
        void * pVoid,
        unsigned char * pucARCCertInfo,
        int nARCCertInfoLen,
```

```
        ARCCERT_INFO * pstARCCertInfo
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pucARCCertInfo [in] 증명서 정보
nARCCertInfoLen [in] 증명서 길이
pstARCCertInfo [out] 증명서 정보

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

해당사항 없음

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_D_API_SetDCAServerInfo
NAK_D_API_RequestARCCertInfo
NAK_D_API_VerifyARCCertInfo
NAK_D_API_ReleaseARCCertInfo

5.3.3.1.5 NAK_D_API_ReleaseARCCertInfo

증명서를 파싱한 결과의 메모리를 해제한다.

```
int NAK_D_API_ReleaseARCCertInfo(
        void * pVoid,
        ARCCERT_INFO * pstARCCertInfo
```

```
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pstARCCertInfo [in] 증명서 정보

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

해당사항 없음

· 요구사항

해당사항 없음

· 참고

NAK_D_API_SetDCAServerInfo
NAK_D_API_RequestARCCertInfo
NAK_D_API_VerifyARCCertInfo
NAK_D_API_ParseARCCertInfo

5.3.4 통합전자서명관리시스템 연계

5.3.4.1 전자서명 요청

5.3.4.1.1 NAK_S_API_SetISMServerInfo

통합전자서명 서버의 정보를 입력한다.

```
int NAK_S_API_SetISMServerInfo(
        void * pVoid,
        char * pcISMServerIP,
        int nISMServerPort,
```

```
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcISMServerIP [in] 통합전자서명 서버의 IP
nISMServerPort [in] 통합전자서명 서버의 Port

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

통합전자서명 서버의 정보를 입력한다.

· 요구사항

해당사항 없음

· 참고

NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_VerifySignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.1.2 NAK_S_API_SignTobe

원본에 대한 Hash를 생성하여 통합전자서명서버에 서명 데이터 생성을 요청한다.

```
int NAK_S_API_SignTobe(
        void * pVoid,
        char * pcID,
        char * pcPass,
        unsigned char * pucInTobe,
        int nInTobeLen,
        unsigned char ** ppucOutSignedData,
        int * pnOutSignedDataLen
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pucInTobe [in] 원본 데이터
nInTobeLen [in] 원본데이터의 길이
ppucOutSignedData [out] 전자서명 데이터
pnOutSignedDataLen [out] 전자서명 데이터 길이

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

전자서명을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 통합서명 서버에 등록이 되어 있어야 한다.

· **요구사항**

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· **참고**

NAK_S_API_SetISMServerInfo
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe

NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_VerifySignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.1.3 NAK_S_API_SignHashBig

원본 데이터가 대용량일 경우 원본에 대한 Hash를 생성하여 서버에 서명 데이터 생성을 요청한다.

```
int NAK_S_API_SignHashBig(
        void * pVoid,
        char * pcID,
        char * pcPass,
        char * pcInTobePath,
        unsigned char ** ppucOutHashSignedData,
        int * pnOutHashSignedDataLen,
        int nHashAlgo
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pcInTobePath [in] 원본 데이터의 경로
ppucOutHashSignedData [out] 원본의 Hash에 대한 전자서명 데이터
pnOutHashSignedDataLen [out] 원본의 Hash에 대한 전자서명 데이터 길이
nHashAlgo [in] 해쉬 알고리즘

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

전자서명을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 서버에 등록이 되어 있어야 한다.

· 요구사항

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.
#define ISM_HASH_ALG_SHA1 0x05 /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03 /* MD5 */
#define ISM_HASH_ALG_SHA256 0x09 /* SHA256 */

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_VerifySignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.1.4 NAK_S_API_SignatureTobe

원본에 대해 서버에 서명 값 생성을 요청한다.

```
int NAK_S_API_SignatureTobe(
        void * pVoid,
        char * pcID,
        char * pcPass,
        unsigned char * pucInTobe,
        int nInTobeLen,
        unsigned char ** ppucOutSignature,
        int * pnOutSignatureLen,
        unsigned char ** ppucOutMakeSignatureDate,
        int * pnOutMakeSignatureDateLen,
        int nHashAlgo
```

```
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pucInTobe [in] 원본 데이터
nInTobeLen [in] 원본 데이터의 길이
ppucOutSignature [out] 원본에 대한 서명값
pnOutSignatureLen [out] 원본에 대한 서명값 길이
ppucOutMakeSignatureDate [out] 서명값을 생성한 일시
pnOutMakeSignatureDateLen [out] 서명값을 생성한 일시 길이
nHashAlgo [in] 해쉬 알고리즘

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

전자서명을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 서버에 등록이 되어 있어야 한다.

· 요구사항

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.

```
#define ISM_HASH_ALG_SHA1 0x05  /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03  /* MD5 */
#define ISM_HASH_ALG_SHA256  0x09  /* SHA256 */
```

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig

NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_VerifySignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.1.5 NAK_S_API_SignatureHashBig

원본 데이터가 대용량일 경우 원본에 대한 Hash를 생성하여 서버에 서명 값 생성을 요청한다.

```
int NAK_S_API_SignatureHashBig(
        void * pVoid,
        char * pcID,
        char * pcPass,
        char* pcTobePath,
        unsigned char ** ppucOutSignature,
        int * pnOutSignatureLen,
        unsigned char ** ppucOutMakeSignatureDate,
        int * pnOutMakeSignatureDateLen,
        int nHashAlgo
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pcTobePath [in] 원본 데이터 경로
ppucOutSignature [out] 원본에 대한 서명값
pnOutSignatureLen [out] 원본에 대한 서명값 길이
ppucOutMakeSignatureDate [out] 서명값을 생성한 일시
pnOutMakeSignatureDateLen [out] 서명값을 생성한 일시 길이
nHashAlgo [in] 해쉬 알고리즘

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조

int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

전자서명을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 서버에 등록이 되어 있어야 한다.

· 요구사항

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.
#define ISM_HASH_ALG_SHA1 0x05 /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03 /* MD5 */
#define ISM_HASH_ALG_SHA256 0x09 /* SHA256 */

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe NAK_S_API_VerifySignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.1.6 NAK_S_API_SignatureLocal

원본에 대한 Hash를 생성하여 서명 값 생성을 요청한다.

```
int NAK_S_API_SignatureLocal(
        void * pVoid,
        char* pcTobePath,
        char* pcCertPath,
        char* pcKeyPath,
        char* pcKeyPwd,
        unsigned char ** ppucOutSignature,
        int * pnOutSignatureLen,
```

```
        unsigned char ** ppucOutMakeSignatureDate,
        int * pnOutMakeSignatureDateLen,
        int nHashAlgo
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcTobePath [in] 원본 데이터 경로
pcCertPath [in] 서명용 인증서 경로
pcKeyPath [in] 서명용 인증서 개인키 경로
pcKeyPwd [in] 개인키 비밀번호
ppucOutSignature [out] 원본에 대한 서명값
pnOutSignatureLen [out] 원본에 대한 서명값 길이
ppucOutMakeSignatureDate [out] 서명값을 생성한 일시
pnOutMakeSignatureDateLen [out] 서명값을 생성한 일시 길이
nHashAlgo [in] 해쉬 알고리즘

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

전자서명 검증을 통합서명서버에서 하지 않고 로컬에서 수행한다.

· **요구사항**

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.
#define ISM_HASH_ALG_SHA1 0x05 /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03 /* MD5 */
#define ISM_HASH_ALG_SHA256 0x09 /* SHA256 */

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· **참고**

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe

NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe NAK_S_API_VerifySignatureHashBig
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.2 전자서명 검증

5.3.4.2.1 NAK_S_API_VerifySignTobe

통합전자서명 서버에 전자서명 검증 요청을 한다.

```
int NAK_S_API_VerifySignTobe(
        void * pVoid,
        char * pcID,
        char * pcPass,
        unsigned char * pucInTobe,
        int nInTobeLen,
        unsigned char * pucInSignedData,
        int nInSignedDataLen,
        unsigned char ** ppucOutSignerCert,
        int * pnOutSignerCertLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pucInTobe [in] 원본 데이터
nInTobeLen [in] 원본데이터의 길이
pucInSignedData [in] 원본에 대한 전자서명 데이터
nInSignedDataLen [in] 원본에 대한 전자서명 데이터 길이
ppucOutSignerCert [out] 전자서명한 인증서
pnOutSignerCertLen [out] 전자서명한 인증서 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

전자서명 검증을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 통합전자서명 서버에 등록이 되어 있어야 한다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_VerifySignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.2.2 NAK_S_API_VerifySignHashBig

원본 데이터가 대용량일 경우 서버에 전자서명 검증 요청을 한다.

```
int NAK_S_API_VerifySignHashBig(
        void * pVoid,
        char * pcID,
        char * pcPass,
        char * pcInTobePath,
        char * pcInSignedHashPath,
        unsigned char ** ppucOutSignerCert,
        int * pnOutSignerCertLen,
        int nHashAlgo
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pcInTobePath [in] 원본 데이터의 경로
pcInSignedHashPath [in] 원본의 Hash에 대한 전자서명 데이터의 경로
ppucOutSignerCert [out] 전자서명한 인증서
pnOutSignerCertLen [out] 전자서명한 인증서 길이
nHashAlgo [in] 해쉬 알고리즘

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

전자서명 검증을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 서버에 등록이 되어 있어야 한다.

· 요구사항

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.

```
#define ISM_HASH_ALG_SHA1 0x05  /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03  /* MD5 */
#define ISM_HASH_ALG_SHA256  0x09  /* SHA256 */
```

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_VerifySignatureHashBig

NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.2.3 NAK_S_API_VerifySignatureTobe

통합전자서명 서버에 원본 데이터에 대한 전자서명 검증 요청을 한다.

```
int NAK_S_API_VerifySignatureTobe(
        void * pVoid,
        char * pcID,
        char * pcPass,
        unsigned char * pucInTobe,
        int nInTobeLen,
        unsigned char * pucInSignature,
        int nInSignatureLen,
        unsigned char * pucInMakeSignatureDate,
        int nInMakeSignatureDateLen,
        unsigned char ** ppucOutSignerCert,
        int * pnOutSignerCertLen,
        int nHashAlgo
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pucInTobe [in] 원본 데이터
nInTobeLen [in] 원본 데이터의 길이
pucInSignature [in] 원본에 대한 서명값
nInSignatureLen [in] 원본에 대한 서명값의 길이
pucInMakeSignatureDate [in] 서명값을 생성한 일시
nInMakeSignatureDateLen [in] 서명값을 생성한 일시의 길이
ppucOutSignerCert [out] 전자서명한 인증서
pnOutSignerCertLen [out] 전자서명한 인증서 길이
nHashAlgo [in] 해쉬 알고리즘

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

전자서명 검증을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 서버에 등록이 되어 있어야 한다.

· 요구사항

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.
#define ISM_HASH_ALG_SHA1 0x05 /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03 /* MD5 */
#define ISM_HASH_ALG_SHA256 0x09 /* SHA256 */

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

서명값을 생성한 일시는 반드시 'yyyymmdd' 8자리 형식으로 입력해야 한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_VerifySignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.2.4 NAK_S_API_VerifySignatureHashBig

원본 데이터가 대용량일 경우 서버에 전자서명 검증 요청을 한다.

```
int NAK_S_API_VerifySignatureHashBig(
        void * pVoid,
```

```
        char * pcID,
        char * pcPass,
        char* pcTobePath,
        unsigned char * pucInSignature,
        int nInSignatureLen,
        unsigned char * pucInMakeSignatureDate,
        int nInMakeSignatureDateLen,
        unsigned char ** ppucOutSignerCert,
        int * pnSignerCertLen,
        int nHashAlgo
)
```

· **파라미터(Parameters)**

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
pcTobePath [in] 원본 데이터 경로
pucInSignature [in] 원본에 대한 서명값
nInSignatureLen [in] 원본에 대한 서명값의 길이
pucInMakeSignatureDate [in] 서명값을 생성한 일시
nInMakeSignatureDateLen [in] 서명값을 생성한 일시의 길이
ppucOutSignerCert [out] 전자서명한 인증서
pnSignerCertLen [out] 전자서명한 인증서 길이
nHashAlgo [in] 해쉬 알고리즘

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· **비고**

전자서명 검증을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 서버에 등록이 되어 있어야 한다.

· **요구사항**

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.
#define ISM_HASH_ALG_SHA1 0x05 /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03 /* MD5 */

#define ISM_HASH_ALG_SHA256 0x09 /* SHA256 */

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

서명값을 생성한 일시는 반드시 'yyyymmdd' 8자리 형식으로 입력해야한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal
NAK_S_API_GetCert

5.3.4.2.5 NAK_S_API_VerifySignatureLocal

전자서명 검증 요청을 한다.

```
int NAK_S_API_VerifySignatureLocal(
        void * pVoid,
        char* pcTobePath,
        char* pcCertPath,
        unsigned char * pucInSignature,
        int nInSignatureLen,
        unsigned char * pucInMakeSignatureDate,
        int nInMakeSignatureDateLen,
        int nHashAlgo
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcTobePath [in] 원본 데이터 경로
pcCertPath [in] 서명용 인증서 경로
pucInSignature [in] 원본에 대한 서명값

nInSignatureLen [in] 원본에 대한 서명값의 길이
pucInMakeSignatureDate [in] 서명값을 생성한 일시
nInMakeSignatureDateLen [in] 서명값을 생성한 일시의 길이
nHashAlgo [in] 해쉬 알고리즘

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

전자서명 검증을 하기 위해서는 미리 인증서와 개인키, 비밀번호가 서버에 등록이 되어 있어야 한다.

· 요구사항

nHashAlgo Type은 헤더파일에 다음과 같이 정의되어 있다.
#define ISM_HASH_ALG_SHA1 0x05 /* SHA1 */
#define ISM_HASH_ALG_MD5 0x03 /* MD5 */
#define ISM_HASH_ALG_SHA256 0x09 /* SHA256 */

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

서명값을 생성한 일시는 반드시 'yyyymmdd' 8자리 형식으로 입력해야한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe
NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_GetCert

5.3.4.3 인증서 요청

5.3.4.3.1 NAK_S_API_GetCert

해당 사용자의 인증서를 요청한다.

```
int NAK_S_API_VerifySignatureLocal(
        void * pVoid,
        char * pcID,
        char * pcPass,
        unsigned char ** ppucOutCert,
        int * pnOutCertLen
)
```

· 파라미터(Parameters)

pVoid [in] 초기화 API를 이용하여 할당한 컨텍스트 변수
pcID [in] 사용자 아이디
pcPass [in] 사용자 비밀번호
ppucOutCert [out] 사용자 인증서
pnOutCertLen [out] 사용자 인증서 길이

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조
int NAK_C_API_GetErrorInfo() : 일반 에러 정보
int NAK_C_API_GetDetailErrorInfo() : 상세 에러 정보

· 비고

입력한 사용자 아이디에 해당하는 인증서를 통합서명 서버에 요청하여 받아 온다.

· 요구사항

out 되는 데이터는 사용이 끝난 후에는 NAK_C_API_FreeItem()를 이용하여 반드시 메모리 해제해 주어야 한다.

· 참고

NAK_S_API_SetISMServerInfo
NAK_S_API_SignTobe
NAK_S_API_VerifySignTobe

NAK_S_API_SignHashBig
NAK_S_API_VerifySignHashBig
NAK_S_API_SignatureTobe
NAK_S_API_VerifySignatureTobe
NAK_S_API_SignatureHashBig
NAK_S_API_SignatureLocal
NAK_S_API_VerifySignatureLocal

5.3.5 공통 API

5.3.5.1 장기검증 공통 API

5.3.5.1.1 NAK_C_API_Initialize

장기검증 API를 사용하기 위한 컨텍스트를 생성한다.

```
int NAK_C_API_Initialize(
        void ** ppVoid
)
```

· **파라미터(Parameters)**

ppVoid [out] 초기화에 필요한 컨텍스트 변수

· **리턴값(Return Values)**

성공: 0

실패: ERROR CODE 참조

· **비고**

장기검증 API를 사용하기 전에 가장 먼저 호출되어야 하며, 이 부분이 성공하면 내부적으로 필요한 메모리를 생성하여 성공한다. 보통 클라이언트 어플리케이션이 구동할 때 한번 호출 하고 나서 나머지 API를 사용하면 된다.

· **요구사항**

해당사항 없음

· **참고**

NAK LTVS Toolkit API

5.3.5.1.2 NAK_C_API_Release

장기검증 API를 사용하기 위해 생성된 컨텍스트를 해제한다.

```
int NAK_C_API_Release(
        void * pVoid
)
```

· 파라미터(Parameters)

pVoid [int] 초기화에 필요한 컨텍스트 변수

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조

· 비고

초기화 했던 장기검증 API를 더 이상 사용하지 않을 때 컨텍스트를 해제하여 내부에 할당했던 메모리를 해제한다.

· 요구사항

해당사항 없음

· 참고

NAK LTVS Toolkit API

5.3.5.1.3 NAK_C_API_GetErrorInfo

API 호출에 대한 에러 발생 시 에러 메시지를 반환한다.

```
int NAK_C_API_GetErrorInfo(
        void * pVoid,
        unsigned char ** ppszOutErrorInfo
)
```

· 파라미터(Parameters)

pVoid [in] 초기화에 필요한 컨텍스트 변수
ppszOutErrorInfo [out] 일반 에러 메시지

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조

· 비고

API 호출에 대한 에러 발생 시 에러 메시지를 반환한다.

· 요구사항

API 호출에 대한 에러 발생 시 에러 메시지를 반환하기 위한 함수이다.

· 참고

NAK LTVS Toolkit API

5.3.5.1.4 NAK_C_API_GetDetailErrorInfo

API 호출에 대한 에러 발생 시 상세 에러 메시지를 반환한다.

```
int NAK_C_API_GetDetailErrorInfo(
        void * pVoid,
        unsigned char ** ppszOutErrorInfo
)
```

· 파라미터(Parameters)

pVoid [in] 초기화에 필요한 컨텍스트 변수
ppszOutErrorInfo [out] 일반 에러 메시지

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조

· 비고

API 호출에 대한 에러 발생 시 상세 에러 메시지를 반환한다.

· 요구사항

API 호출에 대한 에러 발생 시 상세 에러 메시지를 반환하기 위한 함수이다.

· 참고

NAK LTVS Toolkit API

5.3.5.1.5 NAK_C_API_FreeItem

API 호출시 할당한 변수의 메모리를 해제한다.

```
int NAK_C_API_FreeItem(
        unsigned char ** ppucDeleteItem
)
```

· 파라미터(Parameters)

ppucDeleteItem [in] 메모리를 할당받은 변수

· 리턴값(Return Values)

성공: 0

실패: ERROR CODE 참조

· 비고

API 호출시 out변수는 내부에서 메모리를 할당하기 때문에 사용 후 반드시 메모리를 해제해주어야 한다.

· 요구사항

해당사항 없음

· 참고

NAK LTVS Toolkit API

6 에러 코드

6.1 에러 코드의 범위

에러코드의 범위는 아래와 같으며, 에러 코드 범위 별로 다음과 같은 의미를 갖는다.
에러코드에 대한 세부사항은 6.2에서 정의한다.

에러코드 범위	설명
1000 ~ 1099	API 초기화와 API 사용 중 포괄적으로 일어날 수 있는 에러코드
1100 ~ 1199	환경설정 검증과 관련한 에러코드
1200 ~ 1299	인증서 검증과 정보 조회와 관련한 에러코드
1300 ~ 1399	개인키 암/복호화와 관련한 에러코드
1400 ~ 1499	저장매체와 관련한 에러코드
1500 ~ 1599	유무선 서명, 암호 메시지와 관련한 에러코드
1600 ~ 1699	시점확인 서비스와 관련한 에러코드
1700 ~ 1799	본인확인과 관련한 에러코드
1800 ~ 1899	보안 알고리즘과 관련한 에러코드
1900 ~ 1999	BASE64 인코딩/디코딩과 관련한 에러코드
2000 ~ 2099	디렉토리 서버(LDAP)과 관련한 에러코드
2100 ~ 2199	통합검증 서버와 관련한 에러코드

10000 ~ 19999	장기검증처리 관련 에러코드
20000 ~ 29999	통합서명/검증처리 관련 에러코드

6.2 세부 정의

6.2에서는 에러코드에 대한 상세 정의를 포함하며, 각 시스템에서 의미하는

코드는 전체 ERROR CODE에 대해 각각 유일한 값을 가진다.

6.2.1 에러코드[1000 ~ 1099]

API 초기화와 API 사용 중 포괄적으로 일어날 수 있는 에러코드

1000	Error Name	ERR_ALREADY_INITIALIZED
	설명	API가 이미 초기화 되어 있음
	해결방법	1. 초기화 시, 환경변수(*ppCtx)의 값을 NULL로 설정하였는지를 확인한다. 2. 이미 초기화 함수를 호출하여 환경변수가 이미 초기화 되어있지 않은지 확인한다.
1001	Error Name	ERR_API_NOT_INITIALIZED
	설명	API를 초기화해야 함
	해결방법	GPKI_API_Init() 함수를 호출하여 API를 초기화 한 후, 사용하고자 하는 함수를 호출한다.
1002	Error Name	ERR_SET_WORK_DIR
	설명	작업 디렉토리 경로 지정해야 함
	해결방법	GPKI_API_Init() 함수 호출 시, 작업 디렉토리 경로를 지정한다.
1003	Error Name	ERR_MEM_ALLOC
	설명	메모리 할당 실패
	해결방법	해당 시스템에서 더 이상 메모리를 할당 할 수 없습니다. 물리적으로 메모리 사이즈를 늘려주거나, 사용하지 않는 메모리를 환원해 준다.
1004	Error Name	ERR_INSUFFICIENT_ALLOC_SIZE
	설명	포인터에 할당한 메모리의 사이즈가 충분하지 않음
	해결방법	포인터에 할당한 메모리 사이즈를 늘여준다.
1005	1005	ERR_NO_ERR_MSG
	설명	추가적인 에러 정보 없음
1006	Error Name	ERR_INVALID_INPUT
	설명	올바르지 않은 인자 값

	해결방법	함수의 구조대로 입력 값이 들어오지 않았을 때 발생한다. 에러 정보메시지에서 언급하는 입력 인자의 데이터를 확인한다.
1007	Error Name	ERR_NOT_SUPPORTED_FUNCTION
	설명	해당 시스템에서 지원되지 않는 함수 임
	해결방법	동적 라이브러리를 이용한 저장매체 지원은 Window 시스템에서만 가능하다.
1008	Error Name	ERR_TIME_OUT
	설명	통신하는 서버에서 응답이 없어 세션을 끊음
	해결방법	1. Time Out 시간은 환경파일(gpkiapi.conf)에서 설정할 수 있다. 환경파일에 설정되어있는 Time Out 시간이 짧을 경우 시간을 늘린다.2. VerifyCert 클래스의 verify 매소드 또는 IdentifyUser 클래스의 identify 메소드를 이용한 경우에 발생한 에러이면 통합검증서버에서 응답이 없는 것입니다.3. TimeStamp 클래스의 reqTimeStampToken 메소드를 이용한 경우에 발생한 에러이면 시점확인서버에서 응답이 없는 것입니다.

6.2.2 에러코드 [1100 ~ 1199]

라이센스 검증과 관련한 에러코드

1100	Error Name	ERR_LOAD_INFO
	설명	정보 파일 읽기 실패
	해결방법	정보 파일이 API 초기화 시 지정한 작업 디렉토리에 있는지 확인한다.
1101	Error Name	ERR_WRONG_INFO
	설명	잘못된 정보 파일
	해결방법	정보 파일에 잘못된 부분이 있습니다. 라이센스 관리자에게 문의한다.
1102	Error Name	ERR_INFO_AUTHORITY
	설명	정보 파일이 변조 됨
	해결방법	정보 파일이 변조되었습니다. 라이센스 관리자에게 문의한다.

6.2.3 에러코드 [1200 ~ 1299]

인증서 검증과 정보 조회와 관련한 에러코드

1200	Error Name	ERR_WRONG_CERT
	설명	잘못된 인증서 형식
	해결방법	1. 해당 인증서를 Windows에서 확장자를 "der"로 바꾸어서 더블클릭하여 잘 열리는지 확인한다. 2. 잘못 만들어진 인증서 이거나 인증서 데이터가 손상되었을 수 있다. 3. 통합검증서버 이용시 발생하였다면 검증 또는 본인확인을 요청하였던 인증서가 올바른지 확인한다.
1201	Error Name	ERR_LOAD_CERT
	설명	정보를 확인할 인증서를 설정하지 않음
	해결방법	함수 GPKI_CERT_Load()을 이용하여 정보를 확인할 인증서를 설정한다.
1202	Error Name	ERR_NOT_EXIST_FIELD
	설명	정보를 확인하고자 하는 필드가 인증서에 존재하지 않음
	해결방법	해당 에러는 해당 필드가 단지 인증서에 존재하지 않음을 나타내는 것으로 처리 오류는 아니다.
1203	Error Name	ERR_EXPIRED_CERT
	설명	해당 인증서의 유효기간이 만료됨
	해결방법	유효기간이 만료된 인증서이므로 더 이상 사용할 수 없는 인증서이다.
1204	Error Name	ERR_WRONG_CERTS
	설명	인증서 목록 구조체 처리 실패
	해결방법	1. 잘못 만들어진 인증서 목록이거나 데이터가 손상되었을 수 있다. 2. 시스템 장애나 메모리 부족으로 날 수 있는 에러이다.
1205	Error Name	ERR_EXCEED_INDEX
	설명	Index값이 데이터의 개수를 넘음
	해결방법	획득하고자 하는 데이터의 개수보다 index의 값이 작아야 합니다.

1206	Error Name	ERR_CERTS_INCLUDE_WRONG_CERT
	설명	복수의 인증서 구조체에 잘못된 인증서가 포함되어 있음.
	해결방법	1. 복수의 수신자를 위한 암호 메시지 생성하는 경우라면, 인증서 목록에 키분배용 인증서가 아닌 인증서가 포함되어 있지 않은지 확인한다. 2. 인증서 목록에 잘못 만들어져있는 인증서가 있거나 데이터가 손상되었을 수 있다.
1207	Error Name	ERR_NOT_EXIST_CRL_DP
	설명	인증서에 CRL 배포지점 정보가 없음
	해결방법	인증서에 CRL 게시위치 정보가 없어서 CRL을 획득할 수 없다.
1208	Error Name	ERR_WRONG_CRLDP
	설명	인증서 CRL 배포 지점 정보가 잘못되었음
	해결방법	인증서 CRL 게시위치 정보의 인코딩이 잘못되었거나 인증서 파일이 손상된 경우 해당 에러가 발생 할 수 있다.
1209	Error Name	ERR_NOT_CRL
	설명	인증서 폐지 목록으로 설정한 데이터가 인증서 폐지 목록이 아님
	해결방법	인증서 폐지 목록을 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1210	Error Name	ERR_NOT_EXIST_CTL_SIGNER_CERT
	설명	CTL 서명자의 인증서 정보가 없어서 CTL의 서명을 검증할 수 없음
	해결방법	1. CTL이 신뢰하는 인증기관에서 발급한 건지 확인한다. 2. 신뢰하는 인증기관에서 발급한 CTL이라면 신뢰하는 최상위 인증기관 인증서를 함수 GPKI_CERT_AddTrustedCert을 이용하여 추가한다
1211	Error Name	ERR_WRONG_CRL
	설명	잘못된 인증서 폐지 목록(CRL)
	해결방법	1. 해당 인증서 폐지 목록을 Windows에서 확장자를 "crl"로 바꾸어서 더블클릭하여 잘 열리는지 확인한다. 2. CRL의 인코딩이 잘못되었거나 CRL 파일이 손상된 경우 해당 에러가 발생 할 수 있다.

1212	Error Name	ERR_EXPIRED_CRL
	설명	유효기간이 만료된 인증서 폐지 목록임
	해결방법	디렉토리로부터 획득한 ARL/CRL/DeltaCRL이 만료된 인증서 입니다. 인증센터 담당자를 통하여 ARL/CRL/DeltaCRL 갱신이 제대로 이루어 지는지 확인한다.
1213	Error Name	ERR_WRONG_CRL_VALIDITY
	설명	인증서 폐지 목록의 유효기간 정보가 잘못되었음
	해결방법	디렉토리로부터 획득한 ARL/CRL/DeltaCRL의 갱신시간이 현재 시스템 시간보다 이후입니다. 시스템 시간이 CA의 시간보다 늦는 경우에 발생 할 수 있다. 시스템 시간과 CA시간을 동기화 한다.
1214	Error Name	ERR_HOLDED_CERT
	설명	효력 정지된 인증서임
	해결방법	1. 검증대상 인증서나 인증경로상의 인증서가 효력정지 되어있는 상태이다. 2. OCSP를 이용한 경우 OCSP 서버의 인증서가 효력정지 되었거나, OCSP 서버의 인증경로상의 인증서가 효력정지 되어있는 상태이다.
1215	Error Name	ERR_REVOKED_CERT
	설명	폐지된 인증서임
	해결방법	1. 검증대상 인증서나 인증경로상의 인증서가 폐지 되어있는 상태이다. 2. OCSP를 이용한 경우 OCSP 서버의 인증서가 폐지 되었거나, OCSP 서버의 인증경로상의 인증서가 폐지 되어있는 상태이다.
1216	Error Name	ERR_CONNECT_OCSP
	설명	OCSP 서버에 접속하는데 실패
	해결방법	에러정보를 확인하여 접속을 시도했던 OCSP 서버의 IP와 Port를 확인하고 telnet을 이용하여 해당 OCSP에 접속 가능한지 확인한다.
1217	Error Name	ERR_OCSP_REQ_SEND
	설명	OCSP 요청 메시지 전송을 실패
	해결방법	해당 OCSP 서비스 위치가 OCSP 서비스가 제공되고 있는 위치인지 확인한다.

1218	Error Name	ERR_OCSP_MSG_REC
	설명	OCSP 응답 메시지를 받는데 실패
	해결방법	해당 OCSP 서비스 위치가 OCSP 서비스가 제공되고 있는 위치인지 확인한다.
1219	Error Name	ERR_COMPOSE_OCSP_REQ_MSG
	설명	OCSP 요청 메시지 구성 실패
	해결방법	1. OCSP 요청 메시지의 서명에 사용하는 인증서 또는 개인키 데이터의 손상이 없는지 확인한다. 2. 시스템 장애나 메모리 부족으로 날 수 있는 에러이다.
1220	Error Name	ERR_WRONG_OCSP_RES_MSG
	설명	잘못된 OCSP 응답 메시지
	해결방법	OCSP에서 받은 데이터가 변조되어 서명검증에 실패하였거나, 요청한 인증서에 대한 응답 메시지가 아닐 수 있다.
1221	Error Name	ERR_OCSP_REQ_NOT_GRANTED
	설명	OCSP 요청이 승인되지 않음
	해결방법	OCSP 요청 메시지가 잘못되었거나, 승인되지 않은 사용자가 요청 하였거나, 요청 메시지에 서명을 하지 않았거나, OCSP 서버의 내부 문제로 인하여 거절 되었을 수 있다.
1222	Error Name	ERR_UNKNOWN_CERT
	설명	해당 OCSP 서버에서 해당 인증서에 대한 폐지 여부 확인 서비스를 지원하지 않음
	해결방법	인증서에 포함되어있는 OCSP 정보가 잘못되었거나, OCSP를 사용하기 위해서 지정한 OCSP 서비스 정보가 잘못되어있지 않은지 확인한다.
1223	Error Name	ERR_SAVE_CERT_PATH
	설명	획득한 인증서 경로를 저장하는데 실패
	해결방법	에러 정보를 확인하여 인증서 경로를 저장하려고 했던 위치를 확인하여 해당 위치의 디렉토리에 쓰기 권한이 있는지 혹은 같은 이름의 디렉토리 존재하지 않는지 확인한다.
1224	Error Name	ERR_FAIL_CONSTRUCT_PATH
	설명	인증서 경로 구성 실패

	해결방법	경로 구성을 위해 지정하였던 CA 인증서 목록(CaPubs)이 검증하고자 하는 인증서의 CA 인증서 목록이 맞는지 확인한다.
1225	Error Name	ERR_SET_TRUST_ROOT_CERT
	설명	신뢰하는 최상위 인증기관 인증서를 지정해야 함
	해결방법	함수 GPKI_CERT_AddTrustedCert()을 이용해 신뢰하는 최상위 인증기관 인증서를 지정한다.
1226	Error Name	ERR_FAIL_READ_CONF_FILE
	설명	환경 파일을 읽는데 실패
	해결방법	설정한 환경 파일 경로에 환경 파일이 존재하는지 확인한다.
1227	Error Name	ERR_FAIL_READ_CTL_URL_FROM_CONF_FILE
	설명	환경파일에서 CTL의 게시 정보를 획득하는데 실패
	해결방법	환경파일에 섹션[VALIDATOIN_OPTION]의 "CTL_URL" 엔트리에 값이 있는지 확인한다.
1228	Error Name	ERR_FAIL_GET_CTL_FROM_LDAP
	설명	환경파일의 CTL의 게시 위치에서 CTL을 획득하는데 실패
	해결방법	환경파일에서 지정한 CTL 게시 위치에는 CTL이 게시되어있지 않다. CTL이 게시되어있는 위치를 확인하고 환경 파일 정보를 수정한다.
1229	Error Name	ERR_SAVE_CTL
	설명	획득한 신뢰 인증서 목록(CTL) 저장 실패
	해결방법	에러메시지를 확인하여 저장하고자 하였던 위치의 디렉토리 쓰기 권한과 파일과 같은 이름의 디렉토리가 존재하지는 않는지 확인한다.
1230	Error Name	ERR_WRONG_CTL
	설명	잘못된 신뢰 인증서 목록
	해결방법	1. CTL 데이터의 인코딩이 잘못되었거나 손상되었을 수 있다. 2. CTL이 변조되어 서명값 검증에 실패하였다.
1231	Error Name	ERR_NOT_TRUST_CTL_ISSUER
	설명	CTL 발급자를 신뢰할 수 없음
	해결방법	신뢰하는 최상위 인증기관에서 발급된 CTL 이라면 해당 최상위 인증기관을 함수 GPKI_CERT_AddTrustedCert()을 이용하여 추가

		한다.
1232	Error Name	ERR_NOT_TRUST_ROOT_CERT
	설명	구축된 인증서 경로의 최상위 인증기관 인증서를 신뢰할 수 없음
	해결방법	검증하는 인증서를 신뢰하는 인증기관을 통해 발급된 인증서이면 신뢰하는 최상위 인증기관의 인증서를 함수 GPKI_CERT_AddTrustedCert()을 이용하여 추가한다.
1233	Error Name	ERR_PATH_VALIDATION
	설명	인증서 경로 검증 실패
	해결방법	1. 해당 에러는 현재는 발생할 수 없는 에러이며 추후, 인증서 경로 검증의 흐름이 변경된 경우에 발생할 수 있다. 2. 통합검증서버를 이용하여 인증서를 검증한 경우에는 검증을 요청한 인증서가 "정부인증기관"이나 "공인인증기관"에서 발급된 인증서가 맞는지 확인한다.
1234	Error Name	ERR_PATH_VALIDATION_VALIDITY
	설명	인증서 경로 검증 시, 유효기간 검증 실패
	해결방법	1. 검증대상 인증서나 경로상의 인증기관 인증서 중 유효기간이 만료된 인증서가 없는지 확인한다. 2. 상위 인증기관 인증서 보다 하위 인증기관 인증서가 그 이전에 발급되었거나 그 이후에 만료되지는 않는지 확인한다.
1235	Error Name	ERR_PATH_VALIDATION_KEY_USAGE
	설명	인증서 경로 검증 시, 키 용도 검증 실패
	해결방법	1. 검증대상 인증서의 용도에 맞게 아래과 같이 인증서 키 용도가 설정되어있는지를 확인한다. 서명용 : digitalSignature 암호화용 : keyEncipherment 또는 keyAgreement OCSP용 : 확장키 용도 - OCSPSigning, 키 용도 : digitalSignature TSA용 : 확장키 용도 - TimeStamping, 키 용도 : digitalSignature 2. 경로상의 인증기관 인증서 중 "keyCertSign"용도가 없는 인증서가 있는지를 확인한다.
1236	Error Name	ERR_PATH_VALIDATION_BASIC_CONSTS
	설명	인증서 경로 검증 시, 기본 제한 검증 실패

	해결방법	1. 경로상의 인증기관 인증서 중 cA값이 false인 인증서가 있는지 확인한다. 2. 경로상의 인증기관 인증서 중 pathLenConstraints 값보다 더 많은 하위 인증기관 인증서가 있는지를 확인한다.
1237	Error Name	ERR_PATH_VALIDATION_NAME_CONSTS
	설명	인증서 경로 검증 시, 이름 제한 검증 실패
	해결방법	1. 인증기관 인증서에서 사용을 권고한 이름을 하위 인증서의 주체이름이나 주체 대체이름에서 사용하였음을 확인한다. 2. 인증기관 인증서에서 사용을 금지하는 이름을 하위 인증서의 주체이름이나 주체 대체이름에서 사용하지 않았음을 확인한다.
1238	Error Name	ERR_PATH_VALIDATION_CERT_POLICIES
	설명	인증서 경로 검증 시, 인증서 정책 검증 실패
	해결방법	1. 검증 대상 인증서는 해당 서비스에서 허용하지 않는 인증서이다. 2. 해당 인증서가 허용되는 인증서라면 해당 인증서에 대한 인증서 정책을 허용하는 인증서 정책에 추가한다. 3. 통합검증서버를 이용하여 인증서를 검증한 경우에는 검증을 요청한 인증서가 "정부인증기관"이나 "공인인증기관"에서 발급된 인증서가 맞는지 확인한다.
1239	Error Name	ERR_NEED_OCSP_INFO
	설명	OCSP 서버스 위치 정보가 필요함
	해결방법	1. 인증서 검증 시, 지정하는 구조체의 pszOCSPURL 필드에 OCSP 서비스 위치 URL을 지정하거나, 2. 환경파일의 [VALIDATOIN_OPTION] 섹션의 "OCSP_SERVER" 엔트리에 OCSP 서비스 위치 URL을 지정한다.
1240	Error Name	ERR_SAVE_CRL
	설명	디렉토리에서 획득한 CRL / ARL을 저장하는데 실패
	해결방법	에러 정보를 확인하여 저장하려고 했던 위치의 디렉토리의 쓰기 권한을 확인하고, 파일이름과 같은 이름의 디렉토리가 없는지를 확인한다.
1241	Error Name	ERR_NOT_EXIST_OCSP_CERT
	설명	OCSP 인증서가 존재하지 않음
	해결방법	OCSP 응답 메시지에 OCSP 서버의 인증서가 포함되어있지 않음

1242	Error Name	ERR_WRONG_TIME
	설명	검증 시간 형식 또는 날짜가 잘못되었음
	해결방법	1. 설정한 시간 형식이 "YYYYMMDDhhmmss" 인지 확인한다. 2. 설정한 시간이 1970년 이전이거나 2038년 1월 19일 03:14:07 이후 가 아님을 확인한다.
1243	Error Name	ERR_FAIL_OPTAIN_CERT_PATH
	설명	인증서 경로 획득 실패
	해결방법	1. 환경 파일이 지정되어있지 않고 cache에 저장된 인증서 경로가 없는 경우에 발생한다. 2. 통합검증서버를 이용하여 인증서를 검증한 경우에는 검증을 요청한 인증서가 "정부인증기관"이나 "공인인증기관"에서 발급된 인증서가 맞는지 확인한다.
1244	Error Name	ERR_FAIL_OPTAIN_CTL
	설명	디렉토리 서버로부터 CTL을 획득하는데 실패
	해결방법	1. 환경파일에 설정되어있는 CTL의 게시 위치가 올바른지 확인한다. 2. 해당 디렉토리 서버가 떠 있는지 확인한다.
1245	Error Name	ERR_NOT_CERT
	설명	인증서로 설정한 데이터가 인증서가 아님
	해결방법	1. 인증서를 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다. 2. 통합검증서버를 이용한 본인확인 시, 해당 에러가 발생하면 통합검증서버의 키분배용 인증서(cache/IVS/IVS_KmCert.der)가 올바른지 확인한다.
1246	Error Name	ERR_NOT_CTL
	설명	인증서 신뢰 목록으로 설정한 데이터가 인증서 신뢰 목록이 아님
	해결방법	인증서 신뢰 목록을 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1247	Error Name	ERR_NOT_CERTS
	설명	인증서 목록으로 설정한 데이터가 인증서 목록이 아님
	해결방법	인증서 목록을 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.

1248	Error Name	ERR_UPDATE_CERT
	설명	유효기간 만료일이 얼마 남지 않았으므로 인증서를 갱신해야 함
	해결방법	해당 인증서를 갱신해야 한다.
1249	Error Name	ERR_EXPIRED_CTL
	설명	인증서 신뢰 목록이 만료되었음
	해결방법	디렉토리로부터 획득한 CTL이 만료된 인증서 입니다. 인증센터 담당자를 통하여 CTL 갱신이 제대로 이루어지는지 확인한다.
1250	Error Name	ERR_WRONG_CTL_VALIDITY
	설명	인증서 신뢰 목록의 유효기간 정보가 잘못되었음
	해결방법	디렉토리로부터 획득한 CTL의 갱신시간이 현재 시스템 시간보다 이후입니다. 시스템 시간이 CA의 시간보다 늦는 경우에 발생 할 수 있다. 시스템 시간과 CA시간을 동기화 한다.
1251	Error Name	ERR_FAIL_OPTAIN_CRL
	설명	CRL을 획득하는데 실패하였습니다.
	해결방법	통합검증서버에서 리턴하는 에러 코드로서 통합검증서버 관리자에게 문의합니다.

6.2.4 에러코드 [1300 ~ 1399]

개인키 암/복호화와 관련한 에러코드

1300	Error Name	ERR_WRONG_PRIKEY
	설명	잘못된 복호화된 개인키
	해결방법	개인키 데이터의 인코딩이 잘못되었거나 손상되었을 수 있다.
1301	Error Name	ERR_WRONG_PASSWORD
	설명	개인키 비밀번호가 틀렸음
	해결방법	개인키 비밀번호를올바르게 넣었는지 확인한다.
1302	Error Name	ERR_ENCRYPT_PRIKEY

	설명	개인키 암호화 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1303	Error Name	ERR_NOT_MATCHED_KEY_PAIR
	설명	키 쌍이 맞지 않음
	해결방법	키 쌍을 올바르게 설정하였는지 확인한다.
1304	Error Name	ERR_WRONG_ENC_PRIKEY
	설명	잘못된 암호화된 개인키
	해결방법	1. 암호화된 개인키 데이터의 인코딩이 잘못되었거나 손상되었을 수 있다. 2. 암호화된 개인키가 표준보안API에서 지원하지 않는 알고리즘으로 암호화되었을 수 있다.
1305	Error Name	ERR_NOT_PRIKEY
	설명	복호화된 개인키로 설정한 데이터가 복호화된 개인키가 아님
	해결방법	복호화된 개인키로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1306	Error Name	ERR_NOT_ENC_PRIKEY
	설명	암호화된 개인키로 설정한 데이터가 암호화된 개인키가 아님
	해결방법	암호화된 개인키로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.

6.2.5 에러코드 [1400 ~ 1499]

저장매체와 관련한 에러코드

1400	Error Name	ERR_READ_CERT
	설명	인증서를 저장매체에서 읽는데 실패
	해결방법	1. 디스크에서 읽는 경우에는 해당 위치에 인증서 파일이 존재하는지, 해당 디렉토리의 읽기 권한이 있는지 확인한다.

		2. 동적 저장매체를 이용하여 읽는 경우에는 저장매체가 해당 시스템과 연결되어있는지 저장매체의 PIN값을 올바르게 입력하였는지를 확인한다.
1401	Error Name	ERR_READ_PRIKEY
	설명	개인키를 저장매체에서 읽는데 실패
	해결방법	1. 디스크에서 읽는 경우에는 해당 위치에 개인키 파일이 존재하는지, 해당 디렉토리의 읽기 권한이 있는지 확인한다. 2. 동적 저장매체를 이용하여 읽는 경우에는 저장매체가 해당 시스템과 연결되어있는지 저장매체의 PIN값을 올바르게 입력하였는지를 확인한다.
1402	Error Name	ERR_READ_FILE
	설명	해당 경로의 파일을 읽는데 실패
	해결방법	1. 해당 경로에 파일이 존재하는지 확인한다. 2. 해당 경로의 파일에 대한 읽기 권한이 있는지 확인한다.
1403	Error Name	ERR_SAVE_FILE
	설명	해당 경로에 파일을 저장하는데 실패
	해결방법	1. 해당 경로에 쓰기 권한이 있는지 확인한다. 2. 해당 경로에 동일한 이름의 디렉토리가 없는지를 확인한다.
1404	Error Name	ERR_EMPTY_FILE
	설명	읽고자 하는 파일이 비어있음
	해결방법	해당 경로의 파일의 사이즈가 0이 아닌지를 확인한다.
1405	Error Name	ERR_INVALID_STORAGE
	설명	지원하지 않는 저장매체 종류
	해결방법	함수 인자로 지정한 저장매체 종류 값을 확인한다.
1406	Error Name	ERR_LOAD_LIBRARY
	설명	동적 라이브러리를 로드해야 함
	해결방법	GPKI_STORAGE_Load() 함수를 이용하여 동적 라이브러리를 로드한다.
1407	Error Name	ERR_ALREADY_LOADED
	설명	이미 동적라이브러리가 로드되어 있음

	해결방법	GPKI_STORAGE_Unload() 함수를 이용해 이미 로드되어있는 라이브러리를 해제한다.
1408	Error Name	ERR_FAIL_LOAD_LIBRARY
	설명	동적 라이브러리 로드 실패
	해결방법	동적 라이브러리 위치를 지정한 곳에 동적 라이브러리가 존재하는지 확인한다.
1409	Error Name	ERR_FAIL_FREE_LIBRARY
	설명	동적 라이브러리 해제 실패
	해결방법	윈도우 용 함수 FreeLibrary()을 사용하는데 실패하였음. 에러 정보의 에러코드를 통하여 정확한 원인을 파악한다.
1410	Error Name	ERR_FAIL_LOAD_FUNCTION
	설명	동적 라이브러리로부터 함수를 로드하는데 실패
	해결방법	SmartCard 클래스 생성자 호출 시, 설정한 동적 라이브러리가 올바른 라이브러리인지 확인한다.
1411	Error Name	ERR_DELETE_FILE
	설명	지정한 파일을 저장매체에서 삭제하는데 실패
	해결방법	1. 디스크에 있는 파일을 지우고자 한 경우에는 해당 위치의 파일을 삭제할 수 있는 권한이 있는지 확인한다.2. 스마트카드에 있는 파일을 지우고자 한 경우에는 에러 정보를 확인한다.
1412	Error Name	ERR_SAVE_CERT
	설명	인증서 파일을 저장매체에 저장하는데 실패
	해결방법	1. 디스크에 파일을 저장하고자 한 경우에는 해당 위치에 파일을 저장할 수 있는 권한이 있는지 확인한다.2. 스마트카드에 인증서를 저장하고자 한 경우에는 에러 정보를 확인한다.
1413	Error Name	ERR_SAVE_PRIKEY
	설명	개인키 파일을 저장매체에 저장하는데 실패
	해결방법	1. 디스크에 파일을 저장하고자 한 경우에는 해당 위치에 파일을 저장할 수 있는 권한이 있는지 확인한다.2. 스마트카드에 개인키를 저장하고자 한 경우에는 에러 정보를 확인한다.
1414	Error Name	ERR_CHECK_CONNECTION
	설명	스마트카드가 PC에 꽂혀있지 않음

	해결방법	스마트카드가 PC에 꽂혀있는지 확인한다.
1415	Error Name	ERR_WRONG_PIN
	설명	잘못된 스마트카드의 PIN 번호를 입력함
	해결방법	입력한 스마트카드의 PIN 번호가 올바른지 확인합니다.

6.2.6 에러코드 [1500 ~ 1599]

유무선 서명, 암호 메시지와 관련한 에러코드

1500	Error Name	ERR_WRONG_SIGNED_DATA
	설명	잘못된 유선용 서명 메시지
	해결방법	1. 서명 메시지가 변조되어 서명값 검증에 실패하였을 수 있다. 2. 표준보안API에서 지원되지 않는 알고리즘으로 서명되어있을 수 있다. 3. 서명 메시지가 잘못 인코딩 되었거나 손상되었을 수 있다.
1501	Error Name	ERR_COMPOSE_SIGNED_DATA
	설명	서명 메시지 구성 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1502	Error Name	ERR_WRONG_ENVELOPED_DATA
	설명	잘못된 암호 메시지
	해결방법	1. 데이터가 표준보안API에서 지원되지 않는 알고리즘으로 암호화되어있을 수 있다. 2. 암호 메시지가 잘못 인코딩 되었거나 손상되었을 수 있다.
1503	Error Name	ERR_COMPOSE_ENVELOPED_DATA
	설명	암호 메시지 구성 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1504	Error Name	ERR_WRONG_SIGNED_AND_ENVELOPED_DATA
	설명	잘못된 서명 및 암호 데이터
	해결방법	1. 데이터가 표준보안API에서 지원되지 않는 알고리즘으로 암호화

		되어있을 수 있다. 2. 표준보안API에서 지원되지 않는 알고리즘으로 서명되어있을 수 있다. 3. 서명 및 암호 메시지가 변조되어 서명값 검증에 실패하였을 수 있다. 4. 서명 및 암호 메시지가 잘못 인코딩 되었거나 손상되었을 수 있다.
1505	Error Name	ERR_COMPOSE_SIGNED_AND_ENVELOPED_DATA
	설명	서명 및 암호 데이터 구성 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1506	Error Name	ERR_NOT_EXIST_SIGNER_CERT
	설명	서명자의 인증서가 없음
	해결방법	1. 서명 메시지 또는 서명 및 암호 메시지에 서명자의 인증서가 없음을 말한다. 2. TSP 모듈에서는 응답메시지에 TSA 서버의 인증서가 없음을 말한다.
1507	Error Name	ERR_CANNOT_DECRYPT_DATA
	설명	해당 인증서로는 암호 데이터를 처리 할 수 없음
	해결방법	암호 데이터를 암호화할 때 사용했던 수신자 인증서를 확인한다.
1508	Error Name	ERR_WRONG_RECIPIENT_CERT
	설명	잘못된 수신자 인증서
	해결방법	서명 및 암호 메시지를 생성하기 위해서 지정한 수신자의 인증서가 손상되었거나 인코딩이 잘못 되었을 수 있다.
1509	Error Name	ERR_WRONG_SIGNER_CERT
	설명	잘못된 서명자 인증서
	해결방법	서명 및 암호 메시지를 생성하기 위해서 지정한 서명자의 인증서가 손상되었거나 인코딩이 잘못 되었을 수 있다.
1510	Error Name	ERR_NOT_SIGN_CERT
	설명	서명용 인증서가 아님
	해결방법	서명용으로 사용하기 위해 설정한 인증서의 용도를 확인한다.
1511	Error Name	ERR_NOT_KM_CERT

	설명	암호화용 인증서가 아님
	해결방법	암호화용으로 사용하기 위해 설정한 인증서의 용도를 확인한다.
1512	Error Name	ERR_NOT_SIGNED_DATA
	설명	서명 메시지로 설정한 데이터가 서명 데이터가 아님
	해결방법	서명 메시지로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1513	Error Name	ERR_NOT_ENVELOPED_DATA
	설명	암호 메시지로 설정한 데이터가 암호 데이터가 아님
	해결방법	암호 메시지로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1514	Error Name	ERR_NOT_SIGNED_AND_ENVELOPED_DATA
	설명	서명 및 암호 데이터로 설정한 데이터가 서명 및 암호 데이터가 아님
	해결방법	서명 및 암호 데이터로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1515	Error Name	ERR_NOT_ENCRYPTED_DATA
	설명	대칭키 암호 데이터가 아님
	해결방법	대칭키 암호 데이터로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1516	Error Name	ERR_WRONG_ENCRYPTED_DATA
	설명	잘못된 대칭키 암호 데이터 임
	해결방법	대칭키 암호 메시지를 처리하기 위해서 지정한 대칭키 암호 메시지가 손상되었거나 인코딩이 잘못 되었을 수 있다.
1517	Error Name	ERR_COMPOSE_ENCRYPTED_DATA
	설명	대칭키 암호 데이터 구성 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1518	Error Name	ERR_WRONG_SIGNER_CERTS
	설명	잘못된 서명자 인증서 집합
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.

1519	Error Name	ERR_WRONG_SIGNED_CONTENET
	설명	잘못된 무선용 서명 메시지 형식
	해결방법	서명 메시지가 잘못 인코딩 되었거나 손상되었을 수 있다.
1520	Error Name	ERR_COMPOSE_SIGNED_CONTENET
	설명	무선용 서명 메시지 생성 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1521	Error Name	ERR_WRONG_WAP_ENV_DATA
	설명	잘못된 무선용 비대칭키 암호 메시지 형식
	해결방법	1. 데이터가 표준보안API에서 지원되지 않는 알고리즘으로 암호화 되어있을 수 있다. 2. 암호 메시지가 잘못 인코딩 되었거나 손상되었을 수 있다.
1522	Error Name	ERR_COMPOSE_WAP_ENV_DATA
	설명	무선용 비대칭키 암호 메시지 생성 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1523	Error Name	ERR_NOT_SUPPORTED_SIGNER_INFO
	설명	무선용 서명 메시지에서 사용된 서명자 정보 정보를 처리할 수 없음
	해결방법	표준보안API에서는 무선용 서명 메시지의 서명자 정보로 x509_certificate 만을 지원한다.

6.2.7 에러코드 [1600 ~ 1699]

시점확인 서비스와 관련한 에러코드

1600	Error Name	ERR_COMPOSE_TSP_REQ_MSG
	설명	TimeStamp 요청 메시지를 생성하는데 실패
	해결방법	거의 일어날 수 없는 에러이며, 해당 에러가 발생한 경우에는 개발자의 실수이거나 시스템 문제일 수 있다.
1601	Error Name	ERR_CONNECT_TSA
	설명	TSA 서버에 접속 실패

	해결방법	접속하고자 하는 TSA 서버의 IP, Port를 telnet을 통하여 접속 가능한지 확인한다.
1602	Error Name	ERR_TSA_REQ_SEND
	설명	TimeStamp 요청 메시지를 전송하는데 실패
	해결방법	해당 TSP 서비스 위치에 TSP 서비스가 제공되고 있는 위치인지 확인한다.
1603	Error Name	ERR_TSA_MSG_REC
	설명	TimeStamp 응답 메시지를 받는데 실패
	해결방법	해당 TSP 서비스 위치에 TSP 서비스가 제공되고 있는 위치인지 확인한다.
1604	Error Name	ERR_NOT_TSP_RES_MSG
	설명	TSA 응답 메시지로 설정한 데이터가 TSA 응답 메시지가 아님
	해결방법	TSA 응답 메시지로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1605	Error Name	ERR_WRONG_TSP_RES_MSG
	설명	잘못된 TimeStamp 응답 메시지 입니다.
	해결방법	1. TimeStmap 응답 메시지가 변조되어 서명 검증에 실패하였을 수 있다. 2. TimeStamp 응답 메시지가 표준보안API에서 지원하지 않는 서명 알고리즘으로 서명 되었을 수 있다. 3. TimeStamp 응답 메시지의 인코딩이 잘못되었거나 손상되었을 수 있다.
1606	Error Name	ERR_WRONG_TOKEN
	설명	잘못된 TimeStamp 토큰
	해결방법	1. TimeStmap 응답 메시지가 변조되어 서명 검증에 실패하였을 수 있다. 2. TimeStamp 응답 메시지가 표준보안API에서 지원하지 않는 서명 알고리즘으로 서명 되었을 수 있다. 3. TimeStamp 응답 메시지의 인코딩이 잘못되었거나 손상되었을 수 있다.
1607	Error Name	ERR_TSP_REQ_NOT_GRANTED
	설명	TimeStamp 요청이 승인되지 않음
	해결방법	TSA 서버에 보낸 TimeStamp 요청 메시지에 서명을 요구하거나,

		TSA 서버의 내부 문제 등으로 인해서 거절 되었다.
1608	Error Name	ERR_RES_NOT_SIGN_TOKEN
	설명	TimeStamp 토큰이 서명 메시지 형식이 아님
	해결방법	TSA 서버에서 규격에 맞추어 TimeStamp 토큰을 생성하지 않았다.
1609	Error Name	ERR_NOT_TIME_STAMP_TOKEN
	설명	TSA 응답 메시지로 설정한 데이터가 TSA 응답 메시지가 아님
	해결방법	TSA 응답 메시지로 설정해야 할 인자에 다른 데이터를 지정하지 않았는지 확인한다.
1610	Error Name	ERR_REQ_INFO_NOT_EXIST
	설명	TimeStamp 요청 정보를 확인할 수 없음
	해결방법	TimeStamp를 우선 요청 한 후, 응답 메시지를 처리한다. 이때, 요청한 후 환경 변수의 값을 해제하고 다시 초기화하지 않았는지 확인한다.
1611	Error Name	ERR_DIFFERENT_MESSAGE_IMPRINT
	설명	TimeStamp를 요청했던 메시지와 TimeStamp 받은 메시지가 다름
	해결방법	TimeStamp를 요청했던 메시지에 대한 응답 메시지를 지정하였는지를 확인한다.
1612	Error Name	ERR_DIFFERENT_NONCE
	설명	TimeStamp 요청에 대한 응답 메시지가 아님
	해결방법	TimeStamp를 요청 메시지에 대한 응답 메시지를 지정하였는지를 확인한다.
1613	Error Name	ERR_NOT_PERMITTED_POLICY
	설명	TimeStamp 응답메시지가 요청자가 허용하지 않는 인증서 정책으로 발급됨
	해결방법	TimeStamp를 요청 메시지에 대한 응답 메시지를 지정하였는지를 확인한다
1614	Error Name	ERR_NOT_TOKEN_FOR_DOCUMENT
	설명	해당 문서에 대한 시점확인 토큰이 아님
	해결방법	설정한 문서가 해당 TimeStamp 토큰에 대한 문서가 맞는지 확인한다.

6.2.8 에러코드 [1700 ~ 1799]

본인확인과 관련한 에러코드

1700	Error Name	ERR_INVALID_VID
	설명	식별번호에 대한 인증서 소유자가 아님
	해결방법	1. 검증하는 인증서에 대한 식별 번호를 바르게 지정하였는지를 확인한다. 2. 검증하는 인증서에 대한 랜덤값을 바르게 지정하였는지를 확인한다.
1701	Error Name	ERR_NOT_EXIST_VID_IN_CERT
	설명	인증서 안에 본인확인 정보가 없음
	해결방법	1. 랜덤값을 획득하고자 했던 인증서가 서명용인지 확인한다. 2. 행정용에는 VID를 이용한 본인확인 서비스가 제공되지 않으므로 해당 값이 존재 하지 않는다.
1702	Error Name	ERR_NOT_EXIST_RANDOM_IN_PRIKEY
	설명	개인키에 본인확인 정보인 랜덤값이 없음
	해결방법	1. 랜덤값을 획득하고자 했던 개인키가 서명용인지 확인한다. 2. 행정용에는 VID를 이용한 본인확인 서비스가 제공되지 않으므로 해당 값이 존재 하지 않는다.

6.2.9 에러코드 [1800 ~ 1899]

보안 알고리즘과 관련한 에러코드

1800	Error Name	ERR_INVALID_SYM_ALG
	설명	지원하지 않는 대칭키 알고리즘
	해결방법	함수 인자에 설정한 대칭키 알고리즘이 API에서 제공하는 알고리즘 이외의 값을 설정하지 않았는지 확인한다.
1801	Error Name	ERR_INVALID_KEY_LEN
	설명	지정한 알고리즘에 부적당한 Key 길이 임
	해결방법	대칭키 알고리즘에 따라 다음과 같은 길이의 키를 지정해야 함 SEED : 16Byte, NEAT : 16Byte, 3DES : 24Byte, DES : 8Byte
1802	Error Name	ERR_INVALID_IV_LEN

	설명	지정한 알고리즘에 부적당한 IV 길이 임
	해결방법	대칭키 알고리즘에 따라 다음과 같은 길이의 IV를 지정해야 함 SEED : 16Byte, NEAT : 16Byte, 3DES : 8Byte, DES : 8Byte
1803	Error Name	ERR_INVALID_HASH_ALG
	설명	지원하지 않는 해쉬 알고리즘
	해결방법	함수 인자에 설정한 해쉬 알고리즘이 API에서 제공하는 알고리즘 이외의 값을 설정하지 않았는지 확인한다.
1804	Error Name	ERR_INVALID_MAC_ALG
	설명	지원하지 않는 MAC 알고리즘
	해결방법	함수 인자에 설정한 MAC 알고리즘이 API에서 제공하는 알고리즘 이외의 값을 설정하지 않았는지 확인한다.
1805	Error Name	ERR_NOT_SET_KEY_IV
	설명	대칭키 알고리즘의 Key와 IV가 설정되지 않음
	해결방법	GPKI_CRYPT_SetKeyAndIV() 함수를 이용하여 Key와 IV를 설정한다.
1806	Error Name	ERR_ENCRYPT_DATA
	설명	데이터를 해당 키로 암호화하는데 실패
	해결방법	1. 대칭키를 이용한 암호화일 경우에는 시스템 문제이거나 개발자의 실수일 수 있다. 2. 비대칭키를 이용한 암호화일 경우에는 암호화하고자하는 데이터의 크기가 큰 경우에 발생한다. RSA V2.0 1024bit 인 경우에는 86Byte까지만 암호화 할 수 있다. RSA V1.5 1024bit 인 경우에는 117Byte까지만 암호화 할 수 있다. 3. 통합검증서버를 이용한 본인확인일 경우에는 통합검증서버의 암호화용 인증서(cache/IVS/IVS_KmCert.der)가 올바른지 확인한다.
1807	Error Name	ERR_DECRYPT_DATA
	설명	데이터를 해당 키로 복화하는데 실패
	해결방법	1. 시스템 문제이거나 개발자의 실수 일 수 있다. 2. 통합검증서버를 이용한 본인확인 시, 발생하였다면 환경파일에

		지정한 통합검증서버의 게시위치가 맞는지 확인하고 맞다면, 통합검증서버의 키분배용 인증서가 바뀐것입니다. 로컬에 저장되어 있는 암호화용 인증서(cache/IVS/IVS_KmCert.der)를 삭제하고 다시 수행한다.
1808	Error Name	ERR_KCDSA_SIGN_NEED_CERT
	설명	KCDSA 서명을 위해서는 인증서 정보를 지정
	해결방법	KCDSA 서명값을 생성하기 위해서는 인증서 정보가 필요하므로 인증서를 지정하는 인자에 인증서 정보를 지정한다.
1809	Error Name	ERR_SIGN
	설명	서명 값 생성 실패
	해결방법	1. 개인키가 표준보안API에서 지원하지 않는 알고리즘의 개인키일 수 있다. 2. 시스템 문제이거나 개발자의 실수 일 수 있다.
1810	Error Name	ERR_VERIFY_SIGNATURE
	설명	서명 검증에 실패
	해결방법	1. 서명 시, 사용된 개인키와 쌍인 인증서를 맞게 설정하였는지 확인한다. 2. 에러 메시지를 확인한다.
1811	Error Name	ERR_DIGEST_DATA
	설명	메시지 다이제스트 생성 실패
	해결방법	시스템 문제이거나 개발자의 실수 일 수 있다.
1812	Error Name	ERR_VERIFY_MAC
	설명	MAC 값을 검증하는데 실패
	해결방법	MAC값 생성 시, 사용한 데이터와 비밀번호를 올바르게 설정하였는지 확인한다.
1813	Error Name	ERR_GEN_MAC
	설명	MAC 값을 생성하는데 실패
	해결방법	시스템 문제이거나 개발자의 실수 일 수 있다.
1814	Error Name	ERR_CHECK_KEY_PAIR
	설명	키쌍 확인 필요
	해결방법	1. 키쌍 확인 함수 GPKI_PRIKEY_CheckKeyPair()을 이용하여 함

		수의 인자로 설정한 개인키와 인증서의 키쌍을 확인한다. 2. 비대칭키 복호화의 경우에는 추가적으로 복호화하고자 하는 데이터를 바르게 지정하였는지 확인한다.
1815	Error Name	ERR_GEN_RANDOM
	설명	랜덤값 생성 실패
	해결방법	시스템 문제이거나 개발자의 실수 일 수 있다.
1816	Error Name	ERR_NOT_SUPPORTED_ALG
	설명	해당 개인키나 공개키의 알고리즘을 지원하지 않음
	해결방법	1. 무선용 서명 메시지 생성 시에는 서명용 개인키가 RSA 또는 ECDSA 인지를 확인한다. 2. 무선용 비대칭키 암호 메시지 생성 시에는 키분배용 인증서의 공개키가 RSA 인지를 확인한다.
1817	Error Name	ERR_INVALID_KEY_TYPE
	설명	정의되어진 값 이외의 키종류를 저장하였음
	해결방법	키 종류로 정의된 KEY_TYPE_PRIVATE, KEY_TYPE_PUBLIC 이외의 값을 지정하지 않았는지를 확인한다.
1818	Error Name	ERR_WRONG_PUBKEY
	설명	설정한 공개키 또는 인증서의 공개키 정보가 잘못됨.
	해결방법	1. 지정한 공개키 또는 인증서가 올바른 데이터 인지를 확인한다. 2. 잘못 만들어진 공개키(인증서) 이거나 공개키(인증서) 데이터가 손상되었을 수 있다.

6.2.10 에러코드 [1900 ~ 1999]

BASE64 인코딩/디코딩과 관련한 에러코드

1900	Error Name	ERR_BASE64_ENCODE
	설명	BASE64 인코딩 실패
1901	Error Name	ERR_BASE64_DECODE
	설명	BASE64 디코딩 실패
	해결방법	함수 인자에 설정한 BASE64 인코딩된 데이터 안에 "0~9", "a~z", "A~Z", '+', '/', '='(마지막에만 올 수 있음) 이외의 문자가 없음을 확인한다.

6.2.11 에러코드 [2000 ~ 2099]

디렉토리 서버(LDAP)과 관련한 에러코드

2000	Error Name	ERR_WRONG_URL
	설명	URL에서 문자열 "://"을 찾을 수 없음
	해결방법	지정한 URL에 "ldap://" 부분을 설정하였는지 확인한다.
2001	Error Name	ERR_INVALID_PROTOCOL
	설명	URL에서 지정한 프로토콜을 지원하지 않음
	해결방법	표준보안API에서는 LDAP 프로토콜만 지원함.
2002	Error Name	ERR_LDAP_NO_DATA
	설명	찾고자 하는 데이터가 해당 엔트리에 존재하지 않음
	해결방법	획득하고자 하는 데이터가 게시되어있는 위치를 확인하여 게시 위치를 올바르게 지정한다.
2003	Error Name	ERR_INVALID_DATA_TYPE
	설명	지원되지 않는 데이터 타입
	해결방법	함수 인자에 설정한 데이티 종류를 API에서 제공하는 데이터 종류 이외의 값을 설정하지 않았는지 확인한다.
2004	Error Name	ERR_LDAP_INIT
	설명	LDAP 초기화(ldap_init)에 실패
2005	Error Name	ERR_LDAP_SIMPLE_BIND_S
	설명	LDAP 함수 ldap_simple_bind_s 실패

	해결방법	telnet을 이용하여 접속하고자 하는 디렉토리에 접속이 가능한지 확인한다.
2006	Error Name	ERR_LDAP_SET_OPTION
	설명	LDAP 함수 ldap_set_option 실패
	해결방법	에러 메시지를 통하여 정확한 원인을 파악한다.
2007	Error Name	ERR_LDAP_SEARCH_S
	설명	LDAP 함수 ldap_search_s 실패
	해결방법	1. GPKI_LDAP_GetAnyDataByURL()함수에서 에러가 난 경우라면 찾고자하는 데이터의 속성값을 올바르게 지정하였는지 확인한다. 2. 에러 메시지를 통하여 정확한 원인을 파악한다.
2008	Error Name	ERR_LDAP_FIRST_ENTRY
	설명	LDAP 함수 ldap_first_entry의 리턴값이 NULL
	해결방법	획득하고자 하는 데이터가 게시되어있는 위치를 확인하여 게시 위치를 올바르게 지정한다.
2009	Error Name	ERR_LDAP_GET_VALUES_LEN
	설명	LDAP 함수 ldap_get_values_len의 리턴값이 NULL
	해결방법	획득하고자 하는 데이터가 게시되어있는 위치를 확인하여 게시 위치를 올바르게 지정한다.
2010	Error Name	ERR_GET_CRL_FROM_CERT
	설명	인증서를 이용해 CRL 획득 실패
	해결방법	인증서에 있는 CRL 배포지점을 이용해서 CRL을 회득하는데 실패하였다. CRL 게시 위치의 디렉토리 서버가 서비스를 하고 있는지를 확인한다.
2011	Error Name	ERR_NOT_EXIST_LDAP_INFO
	설명	발급자에 대한 LDAP 정보가 없음
	해결방법	환경정보에 LDAP 정보가 부족하다. 에러정보를 확인하여 필요한 정보를 추가한다.
2012	Error Name	ERR_NOT_EXIST_ISSUER_CERT
	설명	해당 게시 위치에 유효한 발급자 인증서 없음
	해결방법	발급자 인증서가 게시되어있는 위치를 확인하여 환경파일에 정보가 올바르게 설정되어있는지 확인한다.

2013	Error Name	ERR_READ_ENTRY
	설명	환경파일에서 정보 획득 실패
	해결방법	환경파일에 LDAP 정보를 올바르게 지정하였는지 확인한다.

6.2.12 에러코드 [2100 ~ 2199]

통합검증 서버와 관련한 에러코드

2100	Error Name	ERR_CONNECT_IVS
	설명	통합검증서버에 접속 실패
	해결방법	해당 API가 구동되고 있는 시스템에서 명령어"telnet ivs.gpki.go.kr 8080"을 수행시켜, 통합검증서버에 접속이 가능한지 확인한다.
2101	Error Name	ERR_IVS_REQ_SEND
	설명	통합검증서버에 요청 메시지 전송 실패
	해결방법	환경파일에 지정된 통합검증서버의 IP, Port에서 통합검증서버의 서비스가 제공되고 있는지 확인한다.
2102	Error Name	ERR_IVS_MSG_REC
	설명	통합검증서버로부터 응답메시지 수신 실패
	해결방법	환경파일에 지정된 통합검증서버의 IP, Port에서 통합검증서버의 서비스가 제공되고 있는지 확인한다.
2103	Error Name	ERR_WRONG_IVS_RES_MSG
	설명	잘못된 통합검증서버의 응답 메시지
	해결방법	1. 통합검증서버에서 받은 데이터가 변조되어 서명검증에 실패하였을 수 있다. 2. 정확한 원인은 API에서 제공하는 에러 정보를 확인한다.
2104	Error Name	ERR_NOT_SERVICE_CERT
	설명	통합검증서버에서 서비스를 제공하는 범위의 인증서가 아님
	해결방법	1. 인증서 검증 시 - 검증하고자한 인증서가 "정부인증기관"이나 "공인인증기관"에서 발급된 인증서 인지를 확인한다. 2. 본인확인 시 - 본인확인 하고자 한 인증서가 "정부인증기관"에서 발급된 인증서 인지를 확인한다.

2105	Error Name	ERR_SYSTEM_INTERNAL_ERROR
	설명	통합검증서버의 내부 에러 발생
	해결방법	통합검증서버의 내부 에러입니다. "통합검증서버" 관리자에게 문의한다.
2106	2106	ERR_REQUESTER_CERT_REVOKED
	설명	폐지된 서비스 요청자의 인증서임
	해결방법	서비스 요청 시 사용한 인증서는 이미 폐지된 인증서입니다. 해당 인증기관을 통하여 인증서를 재발급 받아서 서비스를 이용한다.
2107	Error Name	ERR_REQUESTER_CERT_INVALID
	설명	유효하지 않은 요청자 인증서
	해결방법	서비스 요청 시 사용한 인증서는 유효하지 않은 인증서입니다. 해당 인증기관을 통하여 인증서를 재발급 받아서 서비스를 이용한다.
2109	Error Name	ERR_REQUESTER_CERT_UNAUTHORIZED
	설명	서비스를 이용할 수 없는 요청자임
	해결방법	서비스를 이용하기 위해서 사용한 요청자 인증서는 서비스 이용을 위해서 등록이 되지 않았습니다. "통합검증서버" 관리자를 통하여 요청 서비스 이용을 위해서 등록합니다.
2110	Error Name	ERR_REQ_MSG_FORMAT
	설명	잘못된 통합검증서버에 전송한 요청 메시지
	해결방법	통합검증서버에 전송한 메시지가 변조되었거나 통합검증서버에서 받아들이는 요청 메시지의 포맷이 바뀌었을 수 있습니다.
2111	Error Name	ERR_UNKNOWN_IVS_CODE
	설명	통합검증서버에서 응답한 에러코드는 알수 없는 값임
	해결방법	API에서 제공하는 에러정보를 이용하여 통합검증서버에서 리턴한 에러코드와 에러설명값을 확인한다.
2112	Error Name	ERR_FAIL_READ_IVS_IP_FROM_CONF_FILE
	설명	환경파일에서 통합검증서버의 IP 정보를 획득하는데 실패하였음
	해결방법	환경파일에 "[IVS]" 섹션에 "IP"의 값이 지정되어있는지 확인한다.
2113	Error Name	ERR_FAIL_READ_IVS_PORT_FROM_CONF_FILE
	설명	환경파일에서 통합검증서버의 Port 정보를 획득하는데 실패하였

		음
	해결방법	환경파일에 "[IVS]" 섹션에 "PORT"의 값이 지정되어있는지 확인한다.
2114	Error Name	ERR_FAIL_READ_IVS_CERT_FROM_CONF_FILE
	설명	환경파일에서 통합검증서버의 인증서 게시 위치를 획득하는데 실패
	해결방법	환경파일에 "[IVS]" 섹션에 "SVR_KM_CERT_URL"의 값이 지정되어있는지 확인한다.
2115	Error Name	ERR_FAIL_GET_IVS_KM_CERT_FROM_LDAP
	설명	통합검증서버의 키분배용 인증서 획득 실패
	해결방법	환경파일에서 지정한 통합검증서버의 게시위치가 올바른지 확인한다.
2116	Error Name	ERR_FAIL_GET_IVS_SIGN_CERT_FROM_LDAP
	설명	통합검증서버의 서명용인증서 획득 실패
	해결방법	환경파일에서 지정한 통합검증서버의 게시위치가 올바른지 확인한다.
2117	Error Name	ERR_NOT_TRUST_IVS_CERT
	설명	응답메시지 서명시 사용하였던 통합검증서버의 서명용 인증서를 신뢰할 수 없음
	해결방법	1. 환경파일에서 지정한 통합검증서버의 게시위치가 올바른지 확인한다. 2. 게시위치가 맞다면, "cache" 디렉토리의 "IVS" 디렉토리에 있는 "IVS_SignCert.der" 파일을 삭제하고 다시 수행한다.

6.2.13 에러코드 [10000 ~ 19999]

장기검증처리 관련 에러코드

10010	Error Name	EMR_ERR_NOT_INITIALIZED
	설명	초기화 하지 않은 경우이다.
	해결방안	초기화 API를 호출한 다음 사용하여야 한다.
10011	Error Name	EMR_ERR_INVALID_PARAMETER
	설명	인자값(매개변수)이 정확하지 않은 경우이다.

	해결방안	API사용시 인자값이 정확한지 확인 한후 다시 시도 한다.
10020	Error Name	LIB_ERR_CRYPTO_ENCRYPT_NOT_SUPPORT_ALGORITHM
	설명	지원하지 않는 암호화 알고리즘입니다.
	해결방안	암호 알고리즘이 지원되지 않는 알고리즘을 사용했으므로 알고리즘의 설정을 확인한다.
10021	Error Name	LIB_ERR_CRYPTO_ENCRYPT_NOT_SUPPORT_ALGORITHM_MODE
	설명	지원하지 않는 암호화 알고리즘 모드입니다.
	해결방안	암호 알고리즘이 지원되지 않는 알고리즘 모드을 사용했으므로 알고리즘 모드의 설정을 확인한다.
10022	Error Name	LIB_ERR_CRYPTO_ENCRYPT_ENCRYPTDATA
	설명	암호알고리즘 수행시 에러가 발생 하였습니다.
	해결방안	암호 알고리즘에 사용되는 암호키를 확인하신후 수행 하시기 바랍니다.
10023	Error Name	LIB_ERR_CRYPTO_DECRYPT_NOT_SUPPORT_ALGORITHM
	설명	지원하지 않는 암호화 알고리즘입니다.
	해결방안	복호화 알고리즘이 지원되지 않는 알고리즘을 사용했으므로 알고리즘의 설정을 확인한다.
10024	Error Name	LIB_ERR_CRYPTO_DECRYPT_NOT_SUPPORT_ALGORITHM_MODE
	설명	지원하지 않는 암호화 알고리즘 모드입니다.
	해결방안	복호화 알고리즘이 지원되지 않는 알고리즘 모드을 사용했으므로 알고리즘 모드의 설정을 확인한다.
10025	Error Name	LIB_ERR_CRYPTO_DECRYPT_DECRYPTDATA
	설명	복호화 알고리즘 수행시 에러가 발생 하였습니다.
	해결방안	복호화 알고리즘에 사용되는 암호키를 확인하신후 수행 하시기 바랍니다.
10026	Error Name	LIB_ERR_CRYPTO_RANDOM_GENERATERANDOM
	설명	암복호화에 사용되는 암호키 생성 에러 입니다.
	해결방안	암복호화에 사용되는 암호키 생성시 발생한 에러로서 시스템 관리자에게 문의 하시기 바랍니다.
10030	Error Name	LIB_ERR_ENVELOP_ADD_RECIPIENT_BY_CERT

	설명	Envelop 암호화 수행시 대상 인증서가 잘못되었습니다.
	해결방안	Envelop암호화 수행시 사용되는 대상인증서의 상태를 확인하시기 바랍니다.
10031	Error Name	LIB_ERR_ENVELOP_MAKE_ENVELOPED_DATA
	설명	Envelop 암호화 수행시 에러가 발생 하였습니다.
	해결방안	Envelop 암호화 수행시 사용되는 대상인증서의 인증서 상태 및 Contents가NULL상태 인지 확인 하십시오
10032	Error Name	LIB_ERR_OPENENVELOP_SET_RECIPIENT_IDENTIFIER
	설명	Envelop 데이터 복호화시 에러가 발생 하였습니다.
	해결방안	자신의 인증서 및 개인키의 상태를 확인 하신후 다시 시도 하시기 바랍니다.
10033	Error Name	LIB_ERR_OPENENVELOP_PARSE_ENVELOPED_DATA
	설명	Envelop 데이터의 포맷이 잘못 되었거나 Envelop상태로 암호화 되지 않은 데이터를 읽으려 했습니다.
	해결방안	Envelop데이터의 포맷 상태를 확인하시기 바랍니다.
10034	Error Name	LIB_ERR_OPENENVELOP_GET_CONTENT
	설명	Envelop 암호화 데이터를 복호화 하여 원문을 꺼낼수 없습니다.
	해결방안	Envelop데이터의 포맷 상태를 확인하시기 바랍니다.
10040	Error Name	LIB_ERR_VERIFYVID_WITH_CERT
	설명	인증서의 식별번호가 잘못되었습니다.
	해결방안	인증서에 포함되는 개인(법인) 식별번호를 확인 하신후 다시 시도 하시기 바랍니다.
10041	Error Name	LIB_ERR_PARSE_CERTIFICATE
	설명	인증서를 파싱중에 에러가 발생 하였습니다.
	해결방안	인증서 정보가 올바른지 확인하신 후 다시 시도하시기 바랍니다.
10042	Error Name	LIB_ERR_PARSE_CERTIFICATE_NOT_POLICY
	설명	인증서에 정책이 없습니다.
	해결방안	현재 사용중인 인증서에 인증서 정책이 포함되어 있지 않습니다. 이 인증서는 사용할 수 없는 인증서 입니다.
10043	Error Name	LIB_ERR_CHECK_VALIDITY

	설명	인증서 유효기간이 해당기간에 유효하지 않습니다.
	해결방안	현재 사용중인 인증서의 유효기간이 해당일에 유효하지 않습니다.
10045	Error Name	LIB_ERR_INVALID_CRL
	설명	해당인증서의 CA에서 배포한 CRL이 아닙니다.
	해결방안	현재 사용중인 인증서의 CA에서 배포한 CRL인지 확인하시기 바랍니다.
10046	Error Name	LIB_ERR_PARSE_CRLINFO
	설명	CRL 파싱중에 에러가 발생하였습니다.
	해결방안	CRL 정보가 올바른지 확인하신 후 다시 시도하시기 바랍니다.
10047	Error Name	LIB_ERR_ISREVOKED
	설명	인증서 폐기여부 체크시 에러가 발생하였습니다.
	해결방안	현재 인증서가 폐기되었는지 확인하신 후 다시 시도하시기 바랍니다.
10050	Error Name	LIB_ERR_PARSEPFX_SET_PKCS12_DATA
	설명	PFX포맷을 생성중에 에러가 발생 하였습니다.
	해결방안	PFX를 생성에 사용되는 Contents 및 PFX의 비밀번호를 확인하시기 바랍니다.
10051	Error Name	LIB_ERR_PARSEPFX_GET_KEY_COUNT
	설명	한쌍 이상의 인증서, 개인키 쌍이 있습니다.
	해결방안	PFX는 한쌍의 인증서만을 사용하실 수 있습니다. PFX를 생성에 사용되는 Contents를 확인하시기 바랍니다.
10052	Error Name	LIB_ERR_PARSEPFX_GET_KEY_AND_CERT
	설명	PFX로부터 인증서 및 개인키를 꺼낼수 없습니다.
	해결방안	PFX를 생성에 사용되는 Contents 및 PFX의 비밀번호를 확인 하시기 바랍니다.
10053	Error Name	LIB_ERR_MAKEPFX_SET_KEY_AND_CERT
	설명	PFX생성 중에 오류가 발생 하였습니다.
	해결방안	PFX생성에 사용되는 비밀번호 및 Contents를 확인하시기 바랍니다.
10054	Error Name	LIB_ERR_MAKEPFX_MAKE_PKCS12_DATA
	설명	PFX 생성중에 오류가 발생하였습니다.

	해결방안	PFX생성에 사용되는 비밀번호 및 Contents를 확인하시기 바랍니다.
10060	Error Name	LIB_ERR_PRIVATEKEY_DECRYPT_GETPRIVATEKEYINFO
	설명	개인키의 비밀번호가 잘못되었습니다.
	해결방안	개인키의 비밀번호를 확인하시기 바랍니다.
10061	Error Name	LIB_ERR_PRIVATEKEY_ENCRYPT_NOT_SUPPORT_ALGORITHM
	설명	지원되지 않는 PBE 알고리즘 입니다.
	해결방안	개인키 암호화시 지원되지 않는 알고리즘을 사용하였습니다. 시스템 관리자에게 문의하시기 바랍니다.
10062	Error Name	LIB_ERR_PRIVATEKEY_ENCRYPT_SET_PRIVATE_KEY_INFO
	설명	개인키 암호화시 에러가 발생 하였습니다.
	해결방안	개인키 암호화시 에러가 발생 하였습니다. 시스템 관리자에게 문의하시기 바랍니다
10063	Error Name	LIB_ERR_PRIVATEKEY_ENCRYPT_GET_PKCS8_DATA
	설명	개인키 암호화시 에러가 발생 하였습니다.
	해결방안	개인키 암호화시 에러가 발생 하였습니다. 시스템 관리자에게 문의하시기 바랍니다
10064	Error Name	LIB_ERR_PRIVATEKEY_RVALUE_V1_CONVERT_V2
	설명	개인키 버전 변환시 에러가 발생 하였습니다.
	해결방안	개인키 버전 변환시 에러가 발생 하였습니다. 시스템 관리자에게 문의 하시기 바랍니다.
10065	Error Name	LIB_ERR_PRIVATEKEY_PARSE_RVALUE
	설명	개인키에서 Rvalue 추출 시 에러가 발생 하였습니다.
	해결방안	개인키에서 Rvalue 추출 시 에러가 발생 하였습니다. 시스템 관리자에게 문의하시기 바랍니다.
10070	Error Name	LIB_ERR_MAKESIGNEDDATA_NOT_SUPPORT_SIGN
	설명	지원하지 않는 서명 포맷입니다.
	해결방안	전자서명값의 포맷을 확인하시기 바랍니다.
10071	Error Name	LIB_ERR_MAKESIGNEDDATA_SET_SIGNED_DATA

	설명	전자서명할 원본 데이터를 설정중에 오류가 발생 하였습니다.
	해결방안	전자서명할 원본 데이터가 NULL인지 확인하시기 바랍니다.
10072	Error Name	LIB_ERR_MAKESIGNEDDATA_ADD_SIGNER_CERT
	설명	전자서명을 할 개인의 인증서 및 개인키가 잘못되었습니다.
	해결방안	전자서명을 할 개인의 인증서 및 개인키가 올바른지 확인하시기 바랍니다.
10073	Error Name	LIB_ERR_MAKESIGNEDDATA
	설명	전자서명을 수행중에 에러가 발생 하였습니다.
	해결방안	전자서명을 할 개인의 인증서 및 개인키가 올바른지 확인하시기 바랍니다.
10074	Error Name	LIB_ERR_PARSE_SIGNED_DATA
	설명	전자서명값을 검증시 에러가 발생 하였습니다.
	해결방안	전자서명값의 형식이 아니거나 전자서명되지 않은 데이터를 검증하려 했습니다. 전자서명데이터를 확인하시기 바랍니다.
10075	Error Name	LIB_ERR_DONOT_EXIST_CERT
	설명	전자서명에 사용된 인증서가 존재하지 않습니다.
	해결방안	전자서명값의 형식이 아니거나 전자서명되지 않은 데이터를 검증하려 했습니다. 전자서명데이터를 확인하시기 바랍니다.
10080	Error Name	LIB_ERR_NOT_SUPPORT_HASH_ALGORITHM
	설명	지원하지 않는 Hash 알고리즘입니다.
	해결방안	Message Digest 생성시 지원되지 않는 알고리즘을 사용 하였습니다. 시스템 관리자에게 문의하시기 바랍니다.
10081	Error Name	LIB_ERR_HASH_DIGEST_DATA
	설명	Message Digest 수행시 에러가 발생 하였습니다.
	해결방안	알고리즘의 타입 및 원본의 데이터가 NULL인지 확인하시기 바랍니다.
10090	Error Name	LIB_ERR_OID_MISMATCH
	설명	입력받은 OID와 인증서에서 추출한 OID가 다른 경우이다.
	해결방안	입력하신 OID가 인증서의 OID와 다릅니다. 입력 하신 OID를 확인하시기 바랍니다.
10091	Error Name	LIB_ERR_PARSE_RVALUE

	설명	복호화된 개인키에서 인증서 소유자의 R값을 추출 할수 없습니다.
	해결방안	잘못된 인증서의 개인키 입니다. 인증서와 개인키를 확인하신후 다시시도 하시기 바랍니다.
10092	Error Name	LIB_ERR_VERIFY_CERT
	설명	인증서 검증에 실패 하였습니다.
	해결방안	인증서의 유효기간이 만료 되었거나 사용할 수 없는 인증서 입니다.
10100	Error Name	LIB_ERR_APPDATA_WRONG_TYPE
	설명	Handshake 프로토콜 버전이 올바르지 않습니다.
	해결방안	Handshake 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
10101	Error Name	LIB_ERR_APPDATA_WRONG_LENGTH
	설명	Application Data Message 길이가 잘못되어 있습니다.
	해결방안	Handshake 프로토콜의 길이가 잘못되었습니다.
10110	Error Name	LIB_ERR_HANDSHAKE_WRONG_TYPE
	설명	Handshake 타입이 올바르지 않습니다.
	해결방안	Handshake 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
10111	Error Name	LIB_ERR_HANDSHAKE_WRONG_LENGTH
	설명	Secure Message 길이가 잘못 되어 있습니다.
	해결방안	Secure Message 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
10120	Error Name	LIB_ERR_ALERT_WRONG_LENGTH
	설명	Alert Message 길이가 잘못 되어 있습니다.
	해결방안	Alert Message 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
10130	Error Name	LIB_ERR_PLAIN_WRONG_LENGTH
	설명	Plain Message 길이가 잘못 되어 있습니다.
	해결방안	Plain Message 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다
10148	Error Name	LIB_ERR_UNKNOWN_HANDSHAKE_TYPE

	설명	Handshake 타입 오류 입니다.
	해결방안	로밍서버와 Handshake시 오류가 발생 하여 더 이상 진행 할수 없습니다. 시스템 관리자 에게 문의 하시기 바랍니다.
10150	Error Name	LIB_ERR_SOCKET_RECV_FAIL
	설명	서버와의 통신시 데이터를 수신하지 못했습니다.
	해결방안	로밍서버와의 통신 장애가 발생하였습니다. 로밍서버의 상태를 확인하시기 바랍니다.
10151	Error Name	LIB_ERR_SOCKET_CALLOC_FAIL
	설명	메모리 할당중에 에러가 발생 하였습니다.
	해결방안	사용자 PC의 메모리를 더 이상 사용할 수 없으므로 사용하지 프로그램을 종료하신 후 다시 시도하시기 바랍니다.
10152	Error Name	LIB_ERR_SOCKET_SEND_FAIL
	설명	서버와의 통신시 데이터를 전송할수 없습니다.
	해결방안	로밍서버와의 통신 장애가 발생 하였습니다. 로밍서버의 상태를 확인하시기 바랍니다.
10153	Error Name	LIB_ERR_SOCKET_FAIL
	설명	사용자 PC에서 통신모듈 생성시 오류가 발생 하였습니다.
	해결방안	사용자 PC의 네트웍 상태가 현재 통신이 가능한 상태인지 확인하시기 바랍니다.
10154	Error Name	LIB_ERR_SOCKET_INET_FAIL
	설명	설정하신 IP주소로 통신 모듈을 생성할 수 없습니다.
	해결방안	API초기화에 사용하신 IP가 올바른지 확인하시기 바랍니다.
10155	Error Name	LIB_ERR_SOCKET_CONNECT_FAIL
	설명	설정하신 IP주소로 서버에 접속중에 오류가 발생 하였습니다.
	해결방안	API초기화에 사용하신 IP가 올바른지 확인하시기 바랍니다.
10160	Error Name	LIB_ERR_SIGNEDDATA_COUNT_WRONG
	설명	로밍메시지의 결과 리스트가 1개 미만 입니다.
	해결방안	로밍서버와의 통신시 Contents List의 개수를 확인하시기 바랍니다.
10161	Error Name	LIB_ERR_SMARTCARD_WRITE_DATA

	설명	스마트 카드에 데이터 저장을 실패했습니다.
	해결방안	인증서, 스마트 카드 및 리더기의 연결상태를 확인하시기 바랍니다.
10162	Error Name	LIB_ERR_SMARTCARD_READ_DATA
	설명	스마트 카드에서 데이터 리드를 실패했습니다.
	해결방안	인증서, 스마트 카드 및 리더기의 연결상태를 확인하시기 바랍니다.
10163	Error Name	LIB_ERR_SMARTCARD_DELETE_DATA
	설명	스마트 카드에서 데이터 삭제를 실패했습니다.
	해결방안	스마트 카드 및 리더기의 연결상태를 확인하시기 바랍니다.
10170	Error Name	LIB_ERR_CERT_PASSWORD_WRONG
	설명	현재 인증서 암호가 잘못되었습니다.
	해결방안	개인키의 비밀번호를 확인하시기 바랍니다.
10171	Error Name	LIB_ERR_WRITE_CERT_TO_USB_TOKEN
	설명	USB토근에 인증서를 저장할 수 없습니다.
	해결방안	USB토큰이 올바른 상태 인지 확인하시기 바랍니다.
10172	Error Name	LIB_ERR_FOUND_NOT_CERT
	설명	인증서 파일을 찾을 수 없습니다.
	해결방안	인증서 파일의 경로를 확인하시기 바랍니다.
10173	Error Name	LIB_ERR_FOUND_NOT_KEY
	설명	개인키 파일을 찾을 수 없습니다.
	해결방안	개인키 파일의 경로를 확인하시기 바랍니다.
10180	Error Name	LIB_ERR_ALERT_MESSAGE_FROM_SERVER
	설명	서버로부터 수신된 ALERT Message를 알 수 없습니다.
	해결방안	서버로부터 수신된 ALERT Message의 형식을 확인하시기 바랍니다.
10190	Error Name	LIB_ERR_ROAMING_MESSAGE_NULL
	설명	서버로부터 수신된 로밍메시지가 NULL 입니다.

	해결방안	서버로부터 수신된 로밍 메시지를 확인하시기 바랍니다.
10191	Error Name	LIB_ERR_ROAMING_MESSAGE_WRONG_LENGTH
	설명	로밍 메시지 길이가 잘못 되어 있습니다.
	해결방안	서버로부터 수신된 로밍 메시지의 길이를 확인하시기 바랍니다.
10192	Error Name	LIB_ERR_MAKE_ROAMING_MESSAGE
	설명	로밍 메시지를 생성할 수 없습니다.
	해결방안	다시 시도하시기 바랍니다.
10201	Error Name	LIB_ERR_ROAMING_IP_PORT_NULL
	설명	IP 또는 port번호가 잘못되었습니다.
	해결방안	API 초기화시 입력하신 IP, PORT번호를 확인하시기 바랍니다.
10202	Error Name	LIB_ERR_EMR_VERITY_SIGNED_FILE_SAVE_FAIL
	설명	파일 데이터에 대한 서명 검증후 원본을 파일에 저장할 때 실패했습니다.
	해결방안	파일 데이터를 저장할 디스크의 공간을 확인하시기 바랍니다.
10203	Error Name	LIB_ERR_EXIST_CERT
	설명	식별번호에 해당하는 인증서를 찾을 수 없습니다. 식별번호를 확인하십시오
	해결방안	식별번호에 해당하는 인증서가 등록 되어 있지 않습니다. 인증서 등록후 사용하시기 바랍니다.
10204	Error Name	LIB_ERR_MAKE_SERVER_CONFIG
	설명	서버 환경 요청문을 생성 할 수 없습니다.
	해결방안	다시 시도하시기 바랍니다.
10205	Error Name	LIB_ERR_SIGN_FILE_PATH_IS_NULL
	설명	파일서명을 위한 파일의 경로를 알 수 없습니다.
	해결방안	서명대상 파일이 경로에 존재 하는지 확인하시기 바랍니다.
10206	Error Name	LIB_ERR_SIGN_FILE_CONTENTS_EMPTY
	설명	파일서명을 위한 파일에 내용이 없습니다.
	해결방안	서명대상 파일의 내용이 존재 하는지 확인하시기 바랍니다.

10207	Error Name	LIB_ERR_MEMORY_SIZE_MISMATCH
	설명	EMR_GetSignValue에서 서명값 반환시 할당된 메모리의 크기가 실제 데이터의 길이 보다 작습니다.
	해결방안	전자서명 결과값의 길이만큼 메모리를 할당 했는지 확인하시기 바랍니다.
10208	Error Name	LIB_ERR_HASH_MISMATCH
	설명	서명된 Hash값과 원본 Hash값이 다릅니다. 서명검증에 실패 하였습니다.
	해결방안	원본문서와 전자서명된 Hash값이 일치 않아 서명검증에 실패 하였습니다. 원본 문서를 확인하시기 바랍니다.
10209	Error Name	LIB_ERR_SAVE_PFX_BY_ID
	설명	인증서 등록시 인증서 파일을 저장중에 에러가 발생하였습니다.
	해결방안	인증서 저장 디렉토리의 디스크 크기를 확인하시기 바랍니다.
10210	Error Name	LIB_ERR_OUTPFX_WRONG_LENGTH
	설명	식별번호에 해당하는 인증서를 찾았으나 PFX의 길이가 0 입니다.
	해결방안	인증서를 재등록 또는 백업 데이터를 복원후 사용하시기 바랍니다.
10211	Error Name	LIB_ERR_GET_PFX_LIST
	설명	인증서 목록을 검색중에 오류가 발생 하였습니다.
	해결방안	인증서가 저장된 디렉토리의 상태를 확인 하시기 바랍니다.
10212	Error Name	LIB_ERR_DELETE_PFX_BY_ID
	설명	저장된 인증서(PFX) 삭제시 오류가 발생 하였습니다.
	해결방안	관리자가 더 이상 사용하지 않는 인증서에 대한 삭제시 발생한 오류로서 디렉토리에서 직접 해당 인증서를 삭제하시기 바랍니다. 삭제 하지 않으셔도 업무 진행에는 문제가 발생 하지 않습니다.
10213	Error Name	LIB_ERR_GET_PFX_COUNT
	설명	서버에 등록된 인증서의 개수를 확인 중에 오류가 발생 하였습니다.
	해결방안	인증서가 저장된 디렉토리의 상태를 확인하시기 바랍니다.
10214	Error Name	LIB_ERR_CHECK_CA_NETWORK_STATUS
	설명	로밍서버와 공인인증기관간의 통신상태가 두절되어 있습니다.
	해결방안	로밍서버에서 Out Bound 로 389 포트가 Open되어 있어야 합니

		다.
10215	Error Name	LIB_ERR_CHECK_PFX_BY_ID
	설명	입력한 식별번호에 해당하는 인증서가 존재 하지 않습니다.
	해결방안	사용할 인증서를 등록하신 후 다시 시도하시기 바랍니다.
10216	Error Name	LIB_ERR_FILE_OPEN_ERROR
	설명	로밍서버의 환경설정 파일을 찾을 수 없습니다.
	해결방안	로밍서버의 환경설정파일이 존재 하는지 확인하시기 바랍니다.
10217	Error Name	LIB_ERR_FILE_KEY_NOT_FOUND
	설명	로밍서버의 환경설정 파일이 올바르지 않습니다.
	해결방안	로밍서버의 환경설정 파일의 내용을 확인하시기 바랍니다.
10218	Error Name	LIB_ERR_FILE_NOT_FOUND
	설명	환경파일에 정의된 디렉토리에서 KM 인증서를 찾을 수 없습니다.
	해결방안	KM인증서의 위치와 환경설정 파일에 정의된 경로를 확인하시기 바랍니다.
10220	Error Name	DST_ERR_INITIALIZE
	설명	DSTOOLKIT의 초기화에 실패 하였습니다.
	해결방안	DSTOOLKIT의 환경설정 파일이 올바른지 확인하시기 바랍니다.
10221	Error Name	DST_ERR_API_GETINFO
	설명	DSTOOLKIT의 라이센스 및 버전을 확인 중에 오류가 발생 하였습니다.
	해결방안	사용중인 DSTOOLKIT의 버전과 라이센스 파일이 틀리거나 라이센스가 만료되었는지 확인하시기 바랍니다.
10222	Error Name	DST_ERR_SET_CONFIG_FILE
	설명	DSTOOLKIT의 환경파일을 설정중에 오류가 발생 하였습니다.
	해결방안	DSTOOLKIT의 환경설정 파일이 올바른지 확인하시기 바랍니다.
10223	Error Name	DST_ERR_ADD_TRUSTED_CERT
	설명	DSTOOLKIT에 Trusted 인증서를 설정중에 오류가 발생 하였습니다.
	해결방안	공인인증체계의 ROOT인증서의 상태를 확인하시기 바랍니다.

10224	Error Name	DST_ERR_SET_VERIFY_ENV
	설명	DSTOOLKIT의 인증서 검증정보 설정중에 오류가 발생하였습니다.
	해결방안	다시 시도하시기 바랍니다.
10225	Error Name	DST_ERR_CERT_VERIFY
	설명	인증서 검증에 실패하였습니다.
	해결방안	사용중인 인증서의 유효기간이 만료되었는지 확인하시고, Trusted 인증서의 위치 및 상태를 확인 하십시오.
10230	Error Name	EMR_ERR_UNKNOWN_OPCODE
	설명	알 수 없는 OP코드입니다.
	해결방안	서버에서 처리할 수 없는 작업을 요청 하였습니다. 로밍 메시지가 잘못 되었습니다.
10240	Error Name	SYS_ERR_PARSE_ROAMING_MESSAGE
	설명	로밍메시지가 잘못되었습니다.
	해결방안	서버에서 처리할 수 없는 작업을 요청 하였습니다. 로밍 메시지가 잘못 되었습니다.
10241	Error Name	SYS_ERR_MALLOC
	설명	현재 사용중인 PC의 메모리를 더 이상 사용 할 수 없습니다.
	해결방안	현재 사용중인 PC에서 사용하지 않는 프로그램을 종료하신 후 다시 시도하시기 바랍니다.
10250	Error Name	SYS_ERR_BASE64_ENCODE
	설명	BASE64 인코딩에 실패했습니다.
	해결방안	다시 시도하시기 바랍니다.
10251	Error Name	SYS_ERR_BASE64_DECODE
	설명	BASE64 디코딩에 실패했습니다.
	해결방안	다시 시도하시기 바랍니다.
12000	Error Name	LTVS_ERR_INVALID_PARAM
	설명	인자값(매개변수)이 정확하지 않은 경우이다.
	해결방안	API사용시 인자값이 정확한지 확인한 후 다시 시도한다.
13001	Error Name	LTVS_ERR_NOT_VALID_CERT_USE_TERM

	설명	인증서의 유효기간검증 에러
	해결방안	인증서의 유효기간을 확인한다.
13002	Error Name	LTVS_ERR_INVALID_CERT_USE_TERM
	설명	인증서의 유효기간검증 에러
	해결방안	인증서의 유효기간을 확인한다.
13003	Error Name	LTVS_ERR_NOT_EXIST_TIMESTAMPTOKEN_OF_ESA
	설명	ESA 타임스템프가 존재하지 않음
	해결방안	검증데이터 형식이 ESA 인지 확인한다.
13004	Error Name	LTVS_ERR_VERIFY_REVOCATION_VALUES
	설명	폐기목록 검증 에러 발생
	해결방안	폐기목록 검증시 에러발생
19000	Error Name	LTVS_ERR_NO_SEARCH_VALIDATION_DATA
	설명	검증데이터가 존재하지 않는다.
	해결방안	해당 인증서가 등록되어 있는지 확인한다. 등록된 인증서 일 경우 검증데이터 요청일이 인증서의 유효기간에 포함되는지 확인한다.
19001	Error Name	LTVS_ERR_NO_SEARCH_DATA
	설명	데이터가 존재하지 않는다.
	해결방안	데이터가 조회하는지 파악한다.
19002	Error Name	LTVS_ERR_EXIST_DATA
	설명	검증 데이터가 존재하지 않는다.
	해결방안	검증 데이터가 조회하는지 파악한다.
19100	Error Name	LTVS_ERR_CERT_VALIDITY
	설명	검증데이터 생성시 인증서 유효기간
	해결방안	증명서 생성시간 당시의 인증서 유효기간 오류
19101	Error Name	LTVS_ERR_CHAIN_VALIDATION
	설명	인증서 체인 검증 오류

	해결방안	인증서 체인 검증 오류
19102	Error Name	LTVS_ERR_PARSING_LTV
	설명	증명서 파싱 오류
	해결방안	증명서 파싱 오류
19103	Error Name	LTVS_ERR_VERIFY_LTV
	설명	장기서명 검증 에러
	해결방안	장기서명 검증 에러
19104	Error Name	LTVS_ERR_ORG_DATA
	설명	원본 데이터 오류
	해결방안	원본 데이터 오류
19105	Error Name	LTVS_ERR_HASH
	설명	원본 검증 오류 : 해쉬가 다름
	해결방안	원본 검증 오류 : 해쉬가 다름
19106	Error Name	LTVS_ERR_CERT
	설명	검증 데이터 검증시 인증서 파싱 에러
	해결방안	검증 데이터 검증시 인증서 파싱 에러
19107	Error Name	LTVS_ERR_CERT_REVOKED
	설명	서명인증서가 폐기된 인증서
	해결방안	해당 시점에 서명인증서의 유효기간이 올바른지 확인한다.
19108	Error Name	LTVS_ERR_TSA_CERT_VALIDITY
	설명	증명서 생성시간 당시의 TSA인증서 유효기간 오류
	해결방안	증명서 생성시간 당시의 TSA인증서 유효기간 오류
19109	Error Name	LTVS_ERR_TSA_PARSE_TOKEN
	설명	토큰 파싱 에러
	해결방안	타임스템프 토큰형식이 올바른지 확인한다.
19110	Error Name	LTVS_ERR_TSA_CHAIN_VALIDATION

	설명	TSA인증서 체인 검증 오류
	해결방안	TSA 인증서 경로 구축이 올바른지 확인한다.
19111	Error Name	LTVS_ERR_TSA_HASH
	설명	TSA 원본 검증 오류 : 해쉬가 다름
	해결방안	TSA 원본 검증 오류 : 해쉬가 다름
19112	Error Name	LTVS_ERR_TSA_CERT
	설명	TSA인증서 오류
	해결방안	TSA 인증서가 정상적인지 확인한다.
19113	Error Name	LTVS_ERR_VERIFY_CRL
	설명	인증서 폐기 목록 검증 에러
	해결방안	해당 인증서의 발행기관에서 발행한 인증서 폐기목록이 아니거나 폐기된 인증서 또는 유효기간이 지난 인증서인지 확인한다.
19114	Error Name	LTVS_ERR_USED_TSA_CERT
	설명	이미 사용중인 TSA 인증서
	해결방안	전자서명 장기검증 서버에서 이미 사용중인 TSA 인증서를 등록할 때 발생한다.
19115	Error Name	LTVS_ERR_NOT_REGISTER_TSA_CERT
	설명	등록되지 않은 TSA 인증서
	해결방안	등록되지 않은 TSA 인증서
19116	Error Name	LTVS_ERR_VERIFY_AUDIT_INFO
	설명	감사로그 서명 데이터 검증 에러
	해결방안	감사로그 서명 데이터 검증 에러
19119	Error Name	LTVS_ERR_ETC
	설명	그 밖의 에러
	해결방안	그 밖의 에러

6.2.14 에러코드 [20000 ~ 29999]

통합서명/검증처리 관련 에러코드

20010	Error Name	EMR_ERR_NOT_INITIALIZED
	설명	초기화 하지 않은 경우이다.
	해결방안	초기화 API를 호출한 다음 사용하여야 한다.
20011	Error Name	EMR_ERR_INVALID_PARAMETER
	설명	인자값(매개변수)이 정확하지 않은 경우이다.
	해결방안	API 사용시 인자값이 정확한지 확인 한 후 다시 시도 한다.
20020	Error Name	LIB_ERR_CRYPTO_ENCRYPT_NOT_SUPPORT_ALGORITHM
	설명	지원하지 않는 암호화 알고리즘입니다.
	해결방안	암호 알고리즘이 지원되지 않는 알고리즘을 사용했으므로 알고리즘의 설정을 확인한다.
20021	Error Name	LIB_ERR_CRYPTO_ENCRYPT_NOT_SUPPORT_ALGORITHM_MODE
	설명	지원하지 않는 암호화 알고리즘 모드입니다.
	해결방안	암호 알고리즘이 지원되지 않는 알고리즘 모드를 사용했으므로 알고리즘 모드의 설정을 확인한다.
20022	Error Name	LIB_ERR_CRYPTO_ENCRYPT_ENCRYPTDATA
	설명	암호알고리즘 수행시 에러가 발생 하였습니다.
	해결방안	암호 알고리즘에 사용되는 암호키를 확인하신 후 수행하시기 바랍니다.
20023	Error Name	LIB_ERR_CRYPTO_DECRYPT_NOT_SUPPORT_ALGORITHM
	설명	지원하지 않는 암호화 알고리즘입니다.
	해결방안	보호화 알고리즘이 지원되지 않는 알고리즘을 사용했으므로 알고리즘의 설정을 확인한다.
20024	Error Name	LIB_ERR_CRYPTO_DECRYPT_NOT_SUPPORT_ALGORITHM_MODE
	설명	지원하지 않는 암호화 알고리즘 모드입니다.
	해결방안	복호화 알고리즘이 지원되지 않는 알고리즘 모드를 사용했으므로 알고리즘 모드의 설정을 확인한다.
20025	Error Name	LIB_ERR_CRYPTO_DECRYPT_DECRYPTDATA
	설명	복호화 알고리즘 수행시 에러가 발생 하였습니다.
	해결방안	복호화 알고리즘에 사용되는 암호키를 확인하신 후 수행하시기 바

		랍니다.
20026	Error Name	LIB_ERR_CRYPTO_RANDOM_GENERATERANDOM
	설명	암복호화에 사용되는 암호키 생성 에러입니다.
	해결방안	암복호화에 사용되는 암호키 생성시 발생한 에러로서 시스템 관리자에게 문의하시기 바랍니다.
20030	Error Name	LIB_ERR_ENVELOP_ADD_RECIPIENT_BY_CERT
	설명	Envelop 암호화 수행시 대상 인증서가 잘못되었습니다.
	해결방안	Envelop암호화 수행시 사용되는 대상인증서의 상태를 확인하시기 바랍니다.
20031	Error Name	LIB_ERR_ENVELOP_MAKE_ENVELOPED_DATA
	설명	Envelop 암호화 수행시 에러가 발생 하였습니다.
	해결방안	Envelop 암호화 수행시 사용되는 대상인증서의 인증서 상태 및 Contents가 NULL상태인지 확인 하십시오
20032	Error Name	LIB_ERR_OPENENVELOP_SET_RECIPIENT_IDENTIFIER
	설명	Envelop 데이터 복호화시 에러가 발생 하였습니다.
	해결방안	자신의 인증서 및 개인키의 상태를 확인 하신 후 다시 시도하시기 바랍니다.
20033	Error Name	LIB_ERR_OPENENVELOP_PARSE_ENVELOPED_DATA
	설명	Envelop 데이터의 포맷이 잘못 되었거나 Envelop상태로 암호화 되지 않은 데이터를 읽으려 했습니다.
	해결방안	Envelop데이터의 포맷 상태를 확인하시기 바랍니다.
20034	Error Name	LIB_ERR_OPENENVELOP_GET_CONTENT
	설명	Envelop 암호화 데이터를 복호화 하여 원문을 꺼낼수 없습니다.
	해결방안	Envelop데이터의 포맷 상태를 확인하시기 바랍니다.
20040	Error Name	LIB_ERR_VERIFYVID_WITH_CERT
	설명	인증서의 식별번호가 잘못되었습니다.
	해결방안	인증서에 포함되는 개인(법인) 식별번호를 확인 하신 후 다시 시도하시기 바랍니다.
20041	Error Name	LIB_ERR_PARSE_CERTIFICATE
	설명	인증서를 파싱중에 에러가 발생 하였습니다.
	해결방안	인증서 정보가 올바른지 확인하신 후 다시 시도하시기 바랍니다.

20042	Error Name	LIB_ERR_PARSE_CERTIFICATE_NOT_POLICY
	설명	인증서에 정책이 없습니다.
	해결방안	현재 사용중인 인증서에 인증서 정책이 포함되어 있지 않습니다. 이 인증서는 사용할 수 없는 인증서 입니다.
20043	Error Name	LIB_ERR_CHECK_VALIDITY
	설명	인증서 유효기간이 해당기간에 유효하지 않습니다.
	해결방안	현재 사용중인 인증서의 유효기간이 해당일에 유효하지 않습니다.
20045	Error Name	LIB_ERR_INVALID_CRL
	설명	해당인증서의 CA에서 배포한 CRL이 아닙니다.
	해결방안	현재 사용중인 인증서의 CA에서 배포한 CRL인지 확인하시기 바랍니다.
20046	Error Name	LIB_ERR_PARSE_CRLINFO
	설명	CRL 파싱중에 에러가 발생하였습니다.
	해결방안	CRL 정보가 올바른지 확인하신 후 다시 시도 하시기 바랍니다.
20047	Error Name	LIB_ERR_ISREVOKED
	설명	인증서 폐기여부 체크시 에러가 발생하였습니다.
	해결방안	현재 인증서가 폐기되었는지 확인하신 후 다시 시도하시기 바랍니다.
20050	Error Name	LIB_ERR_PARSEPFX_SET_PKCS12_DATA
	설명	PFX포맷을 생성중에 에러가 발생 하였습니다.
	해결방안	PFX를 생성에 사용되는 Contents 및 PFX의 비밀번호를 확인하시기 바랍니다.
20051	Error Name	LIB_ERR_PARSEPFX_GET_KEY_COUNT
	설명	한쌍 이상의 인증서, 개인키 쌍이 있습니다.
	해결방안	PFX는 한쌍의 인증서만을 사용하실 수 있습니다. PFX를 생성에 사용되는 Contents를 확인하시기 바랍니다.
20052	Error Name	LIB_ERR_PARSEPFX_GET_KEY_AND_CERT
	설명	PFX로부터 인증서 및 개인키를 꺼낼수 없습니다.
	해결방안	PFX를 생성에 사용되는 Contents 및 PFX의 비밀번호를 확인하시기 바랍니다.

20053	Error Name	LIB_ERR_MAKEPFX_SET_KEY_AND_CERT
	설명	PFX생성 중에 오류가 발생 하였습니다.
	해결방안	PFX생성에 사용되는 비밀번호 및 Contents를 확인하시기 바랍니다.
20054	Error Name	LIB_ERR_MAKEPFX_MAKE_PKCS12_DATA
	설명	PFX 생성중에 오류가 발생하였습니다.
	해결방안	PFX생성에 사용되는 비밀번호 및 Contents를 확인하시기 바랍니다.
20060	Error Name	LIB_ERR_PRIVATEKEY_DECRYPT_GETPRIVATEKEYINFO
	설명	개인키의 비밀번호가 잘못되었습니다.
	해결방안	개인키의 비밀번호를 확인하시기 바랍니다.
20061	Error Name	LIB_ERR_PRIVATEKEY_ENCRYPT_NOT_SOPPORT_ALGORITHM
	설명	지원되지 않는 PBE 알고리즘 입니다.
	해결방안	개인키 암호화시 지원되지 않는 알고리즘을 사용 하였습니다. 시스템 관리자에게 문의하시기 바랍니다.
20062	Error Name	LIB_ERR_PRIVATEKEY_ENCRYPT_SET_PRIVATE_KEY_INFO
	설명	개인키 암호화시 에러가 발생 하였습니다.
	해결방안	개인키 암호화시 에러가 발생 하였습니다. 시스템 관리자에게 문의하시기 바랍니다
20063	Error Name	LIB_ERR_PRIVATEKEY_ENCRYPT_GET_PKCS8_DATA
	설명	개인키 암호화시 에러가 발생 하였습니다.
	해결방안	개인키 암호화시 에러가 발생 하였습니다. 시스템 관리자에게 문의하시기 바랍니다
20064	Error Name	LIB_ERR_PRIVATEKEY_RVALUE_V1_CONVERT_V2
	설명	개인키 버전 변환시 에러가 발생 하였습니다.
	해결방안	개인키 버전 변환시 에러가 발생 하였습니다. 시스템 관리자에게 문의하시기 바랍니다.
20065	Error Name	LIB_ERR_PRIVATEKEY_PARSE_RVALUE
	설명	개인키에서 Rvalue 추출시 에러가 발생하였습니다.
	해결방안	개인키에서 Rvalue 추출시 에러가 발생하였습니다. 시스템 관리자

		에게 문의하시기 바랍니다.
20070	Error Name	LIB_ERR_MAKESIGNEDDATA_NOT_SUPPORT_SIGN
	설명	지원하지 않는 서명 포맷입니다.
	해결방안	전자서명값의 포맷을 확인하시기 바랍니다.
20071	Error Name	LIB_ERR_MAKESIGNEDDATA_SET_SIGNED_DATA
	설명	전자서명 할 원본 데이터를 설정중에 오류가 발생 하였습니다.
	해결방안	전자서명 할 원본 데이터가 NULL인지 확인하시기 바랍니다.
20072	Error Name	LIB_ERR_MAKESIGNEDDATA_ADD_SIGNER_CERT
	설명	전자서명을 할 개인의 인증서 및 개인키가 잘못되었습니다.
	해결방안	전자서명을 할 개인의 인증서 및 개인키가 올바른지 확인하시기 바랍니다.
20073	Error Name	LIB_ERR_MAKESIGNEDDATA
	설명	전자서명을 수행중에 에러가 발생 하였습니다.
	해결방안	전자서명을 할 개인의 인증서 및 개인키가 올바른지 확인하시기 바랍니다.
20074	Error Name	LIB_ERR_PARSE_SIGNED_DATA
	설명	전자서명값을 검증시 에러가 발생 하였습니다.
	해결방안	전자서명값의 형식이 아니거나 전자서명 되지 않은 데이터를 검증하려 했습니다. 전자서명데이터를 확인하시기 바랍니다.
20075	Error Name	LIB_ERR_DONOT_EXIST_CERT
	설명	전자서명에 사용된 인증서가 존재하지 않습니다.
	해결방안	전자서명값의 형식이 아니거나 전자서명 되지 않은 데이터를 검증하려 했습니다. 전자서명데이터를 확인하시기 바랍니다.
20080	Error Name	LIB_ERR_NOT_SUPPORT_HASH_ALGORITHM
	설명	지원하지 않는 Hash 알고리즘입니다.
	해결방안	Message Digest 생성시 지원되지 않는 알고리즘을 사용하였습니다. 시스템 관리자에게 문의하시기 바랍니다.
20081	Error Name	LIB_ERR_HASH_DIGEST_DATA
	설명	Message Digest 수행시 에러가 발생 하였습니다.
	해결방안	알고리즘의 타입 및 원본의 데이터가 NULL인지 확인하시기 바랍

		니다.
20090	Error Name	LIB_ERR_OID_MISMATCH
	설명	입력받은 OID와 인증서에서 추출한 OID가 다른 경우이다.
	해결방안	입력하신 OID가 인증서의 OID와 다릅니다. 입력 하신 OID를 확인하시기 바랍니다.
20091	Error Name	LIB_ERR_PARSE_RVALUE
	설명	복호화된 개인키에서 인증서 소유자의 R값을 추출할 수 없습니다.
	해결방안	잘못된 인증서의 개인키 입니다. 인증서와 개인키를 확인하신 후 다시시도 하시기 바랍니다.
20092	Error Name	LIB_ERR_VERIFY_CERT
	설명	인증서 검증에 실패 하였습니다.
	해결방안	인증서의 유효기간이 만료 되었거나 사용할 수 없는 인증서 입니다.
20100	Error Name	LIB_ERR_APPDATA_WRONG_TYPE
	설명	Handshake 프로토콜 버전이 올바르지 않습니다.
	해결방안	Handshake 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
20101	Error Name	LIB_ERR_APPDATA_WRONG_LENGTH
	설명	Application Data Message 길이가 잘 못 되어 있습니다.
	해결방안	Handshake 프로토콜의 길이가 잘못되었습니다.
20110	Error Name	LIB_ERR_HANDSHAKE_WRONG_TYPE
	설명	Handshake 타입이 올바르지 않습니다.
	해결방안	Handshake 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
20111	Error Name	LIB_ERR_HANDSHAKE_WRONG_LENGTH
	설명	Secure Message 길이가 잘못되어 있습니다.
	해결방안	Secure Message 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
20120	Error Name	LIB_ERR_ALERT_WRONG_LENGTH
	설명	Alert Message 길이가 잘못되어 있습니다.

	해결방안	Alert Message 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다.
20130	Error Name	LIB_ERR_PLAIN_WRONG_LENGTH
	설명	Plain Message 길이가 잘못 되어 있습니다.
	해결방안	Plain Message 프로토콜이 잘못되었거나 수신된 프로토콜이 잘못되었습니다
20148	Error Name	LIB_ERR_UNKNOWN_HANDSHAKE_TYPE
	설명	Handshake 타입 오류 입니다.
	해결방안	로밍서버와 Handshake시 오류가 발생하여 더 이상 진행 할 수 없습니다. 시스템 관리자 에게 문의하시기 바랍니다.
20150	Error Name	LIB_ERR_SOCKET_RECV_FAIL
	설명	서버와의 통신시 데이터를 수신하지 못했습니다.
	해결방안	로밍서버와의 통신 장애가 발생하였습니다. 로밍서버의 상태를 확인하시기 바랍니다.
20151	Error Name	LIB_ERR_SOCKET_CALLOC_FAIL
	설명	메모리 할당중에 에러가 발생 하였습니다.
	해결방안	사용자 PC의 메모리를 더 이상 사용할 수 없으므로 사용하지 프로그램을 종료하신 후 다시 시도하시기 바랍니다.
20152	Error Name	LIB_ERR_SOCKET_SEND_FAIL
	설명	서버와의 통신시 데이터를 전송할 수 없습니다.
	해결방안	로밍서버와의 통신 장애가 발생 하였습니다. 로밍서버의 상태를 확인하시기 바랍니다.
20153	Error Name	LIB_ERR_SOCKET_FAIL
	설명	사용자 PC에서 통신모듈 생성시 오류가 발생 하였습니다.
	해결방안	사용자 PC의 네트웍 상태가 현재 통신이 가능한 상태인지 확인하시기 바랍니다.
20154	Error Name	LIB_ERR_SOCKET_INET_FAIL
	설명	설정하신 IP주소로 통신 모듈을 생성할 수 없습니다.
	해결방안	API초기화에 사용하신 IP가 올바른지 확인하시기 바랍니다.
20155	Error Name	LIB_ERR_SOCKET_CONNECT_FAIL
	설명	설정하신 IP주소로 서버에 접속중에 오류가 발생 하였습니다.

	해결방안	API초기화에 사용하신 IP가 올바른지 확인하시기 바랍니다.
20160	Error Name	LIB_ERR_SIGNEDDATA_COUNT_WRONG
	설명	로밍메시지의 결과 리스트가 1개미만 입니다.
	해결방안	로밍서버와의 통신시 Contents List의 개수를 확인하시기 바랍니다.
20161	Error Name	LIB_ERR_SMARTCARD_WRITE_DATA
	설명	스마트 카드에 데이터 저장을 실패했습니다.
	해결방안	인증서, 스마트 카드 및 리더기의 연결상태를 확인하시기 바랍니다.
20162	Error Name	LIB_ERR_SMARTCARD_READ_DATA
	설명	스마트 카드에서 데이터 리드를 실패했습니다.
	해결방안	인증서, 스마트 카드 및 리더기의 연결상태를 확인하시기 바랍니다.
20163	Error Name	LIB_ERR_SMARTCARD_DELETE_DATA
	설명	스마트 카드에서 데이터 삭제를 실패했습니다.
	해결방안	스마트 카드 및 리더기의 연결상태를 확인하시기 바랍니다.
20170	Error Name	LIB_ERR_CERT_PASSWORD_WRONG
	설명	현재 인증서 암호가 잘못 되었습니다.
	해결방안	개인키의 비밀번호를 확인하시기 바랍니다.
20171	Error Name	LIB_ERR_WRITE_CERT_TO_USB_TOKEN
	설명	USB토큰에 인증서를 저장할 수 없습니다.
	해결방안	USB토큰이 올바른 상태 인지 확인하시기 바랍니다.
20172	Error Name	LIB_ERR_FOUND_NOT_CERT
	설명	인증서 파일을 찾을 수 없습니다.
	해결방안	인증서 파일의 경로를 확인하시기 바랍니다.
20173	Error Name	LIB_ERR_FOUND_NOT_KEY
	설명	개인키 파일을 찾을 수 없습니다.
	해결방안	개인키 파일의 경로를 확인하시기 바랍니다.

20180	Error Name	LIB_ERR_ALERT_MESSAGE_FROM_SERVER
	설명	서버로부터 수신된 ALERT Message를 알수 없습니다.
	해결방안	서버로부터 수신된 ALERT Message의 형식을 확인하시기 바랍니다.
20190	Error Name	LIB_ERR_ROAMING_MESSAGE_NULL
	설명	서버로부터 수신된 로밍메시지가 NULL 입니다.
	해결방안	서버로부터 수신된 로밍 메시지를 확인하시기 바랍니다.
20191	Error Name	LIB_ERR_ROAMING_MESSAGE_WRONG_LENGTH
	설명	로밍 메시지 길이가 잘못 되어 있습니다.
	해결방안	서버로부터 수신된 로밍 메시지의 길이를 확인하시기 바랍니다.
20192	Error Name	LIB_ERR_MAKE_ROAMING_MESSAGE
	설명	로밍 메시지를 생성할 수 없습니다.
	해결방안	다시 시도하시기 바랍니다.
20201	Error Name	LIB_ERR_ROAMING_IP_PORT_NULL
	설명	IP 또는 port번호가 잘못되었습니다.
	해결방안	API 초기화시 입력하신 IP, PORT번호를 확인하시기 바랍니다.
20202	Error Name	LIB_ERR_EMR_VERITY_SIGNED_FILE_SAVE_FAIL
	설명	파일 데이터에 대한 서명 검증후 원본을 파일에 저장할 때 실패했습니다.
	해결방안	파일 데이터를 저장할 디스크의 공간을 확인하시기 바랍니다.
20203	Error Name	LIB_ERR_EXIST_CERT
	설명	식별번호에 해당하는 인증서를 찾을 수 없습니다. 식별번호를 확인하십시오
	해결방안	식별번호에 해당하는 인증서가 등록 되어 있지 않습니다. 인증서 등록후 사용하시기 바랍니다.
20204	Error Name	LIB_ERR_MAKE_SERVER_CONFIG
	설명	서버 환경 요청문을 생성 할 수 없습니다.
	해결방안	다시 시도하시기 바랍니다.

20205	Error Name	LIB_ERR_SIGN_FILE_PATH_IS_NULL
	설명	파일서명을 위한 파일의 경로를 알 수 없습니다.
	해결방안	서명대상 파일이 경로에 존재 하는지 확인하시기 바랍니다.
20206	Error Name	LIB_ERR_SIGN_FILE_CONTENTS_EMPTY
	설명	파일서명을 위한 파일에 내용이 없습니다.
	해결방안	서명대상 파일의 내용이 존재 하는지 확인하시기 바랍니다.
20207	Error Name	LIB_ERR_MEMORY_SIZE_MISMATCH
	설명	EMR_GetSignValue에서 서명값 반환시 할당된 메모리의 크기가 실제 데이터의 길이 보다 작습니다.
	해결방안	전자서명 결과값의 길이만큼 메모리를 할당 했는지 확인하시기 바랍니다.
20208	Error Name	LIB_ERR_HASH_MISMATCH
	설명	서명된 Hash값과 원본 Hash값이 다릅니다. 서명검증에 실패 하였습니다.
	해결방안	원본문서와 전자서명된 Hash값이 일치 않아 서명검증에 실패하였습니다. 원본 문서를 확인하시기 바랍니다.
20209	Error Name	LIB_ERR_SAVE_PFX_BY_ID
	설명	인증서 등록시 인증서 파일을 저장중에 에러가 발생하였습니다.
	해결방안	인증서 저장 디렉토리의 디스크 크기를 확인하시기 바랍니다.
20210	Error Name	LIB_ERR_OUTPFX_WRONG_LENGTH
	설명	식별번호에 해당하는 인증서를 찾았으나 PFX의 길이가 0 입니다.
	해결방안	인증서를 재등록 또는 백업 데이터를 복원후 사용하시기 바랍니다.
20211	Error Name	LIB_ERR_GET_PFX_LIST
	설명	인증서 목록을 검색중에 오류가 발생 하였습니다.
	해결방안	인증서가 저장된 디렉토리의 상태를 확인하시기 바랍니다.
20212	Error Name	LIB_ERR_DELETE_PFX_BY_ID
	설명	저장된 인증서(PFX) 삭제시 오류가 발생 하였습니다.
	해결방안	관리자가 더 이상 사용하지 않는 인증서에 대한 삭제시 발생한 오류로서 디렉토리에서 직접 해당 인증서를 삭제하시기 바랍니다. 삭제 하지 않으셔도 업무 진행에는 문제가 발생 하지 않습니다.

20213	Error Name	LIB_ERR_GET_PFX_COUNT
	설명	서버에 등록된 인증서의 개수를 확인 중에 오류가 발생하였습니다.
	해결방안	인증서가 저장된 디렉토리의 상태를 확인하시기 바랍니다.
20214	Error Name	LIB_ERR_CHECK_CA_NETWORK_STATUS
	설명	로밍서버와 공인인증기관간의 통신상태가 두절되어 있습니다.
	해결방안	로밍서버에서 Out Bound 로 389 포트가 Open되어 있어야 합니다.
20215	Error Name	LIB_ERR_CHECK_PFX_BY_ID
	설명	입력한 식별번호에 해당하는 인증서가 존재 하지 않습니다.
	해결방안	사용할 인증서를 등록하신 후 다시 시도하시기 바랍니다.
20216	Error Name	LIB_ERR_FILE_OPEN_ERROR
	설명	로밍서버의 환경설정 파일을 찾을 수 없습니다.
	해결방안	로밍서버의 환경설정파일이 존재 하는지 확인하시기 바랍니다.
20217	Error Name	LIB_ERR_FILE_KEY_NOT_FOUND
	설명	로밍서버의 환경설정 파일이 올바르지 않습니다.
	해결방안	로밍서버의 환경설정 파일의 내용을 확인하시기 바랍니다.
20218	Error Name	LIB_ERR_FILE_NOT_FOUND
	설명	환경파일에 정의된 디렉토리에서 KM 인증서를 찾을 수 없습니다.
	해결방안	KM인증서의 위치와 환경설정 파일에 정의된 경로를 확인하시기 바랍니다.
20220	Error Name	DST_ERR_INITIALIZE
	설명	DSTOOLKIT의 초기화에 실패하였습니다.
	해결방안	DSTOOLKIT의 환경설정 파일이 올바른지 확인하시기 바랍니다.
20221	Error Name	DST_ERR_API_GETINFO
	설명	DSTOOLKIT의 정보 확인 중에 오류가 발생 하였습니다.
	해결방안	사용중인 DSTOOLKIT의 정보 파일이 틀렸는지 확인하시기 바랍니다.
20222	Error Name	DST_ERR_SET_CONFIG_FILE

	설명	DSTOOLKIT의 환경파일을 설정중에 오류가 발생 하였습니다.
	해결방안	DSTOOLKIT의 환경설정 파일이 올바른지 확인하시기 바랍니다.
20223	Error Name	DST_ERR_ADD_TRUSTED_CERT
	설명	DSTOOLKIT에 Trusted 인증서를 설정중에 오류가 발생하였습니다.
	해결방안	공인인증체계의 ROOT인증서의 상태를 확인하시기 바랍니다.
20224	Error Name	DST_ERR_SET_VERIFY_ENV
	설명	DSTOOLKIT의 인증서 검증정보 설정중에 오류가 발생하였습니다.
	해결방안	다시 시도하시기 바랍니다.
20225	Error Name	DST_ERR_CERT_VERIFY
	설명	인증서 검증에 실패하였습니다.
	해결방안	사용중인 인증서의 유효기간이 만료되었는지 확인하시고, Trusted 인증서의 위치 및 상태를 확인 하십시오.
20230	Error Name	EMR_ERR_UNKNOWN_OPCODE
	설명	알 수 없는 OP코드 입니다.
	해결방안	서버에서 처리할 수 없는 작업을 요청하였습니다. 로밍 메시지가 잘못 되었습니다.
20240	Error Name	SYS_ERR_PARSE_ROAMING_MESSAGE
	설명	로밍메시지가 잘못 되었습니다.
	해결방안	서버에서 처리할 수 없는 작업을 요청 하였습니다. 로밍 메시지가 잘못 되었습니다.
20241	Error Name	SYS_ERR_MALLOC
	설명	현재 사용중인 PC의 메모리를 더 이상 사용 할수 없습니다.
	해결방안	현재 사용중인 PC에서 사용하지 않는 프로그램을 종료 하신 후 다시 시도하시기 바랍니다.
20250	Error Name	SYS_ERR_BASE64_ENCODE
	설명	BASE64 인코딩에 실패했습니다.
	해결방안	다시 시도하시기 바랍니다.
20251	Error Name	SYS_ERR_BASE64_DECODE
	설명	BASE64 디코딩에 실패했습니다.

	해결방안	다시 시도하시기 바랍니다.