표준 CryptoLib 제작 과제 기본설계

[개요]

- 적용 대상 샘플
  - Sample_Registration_Encryption
  - 설정파일 암호화 및 서명
  - 통신암호화 및 디바이스 인증 – 예시만.

- 암호 연산
  - TPM 활용/SoftwareOnly
  - 파일 암호화, 통신암호화, 인증
  - 다이제스트 생성
  - 서명
  - 서명 검증
  - 복호화

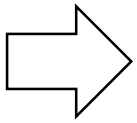
- 테스트 계획
  - 소스코드 함수 수준의 단위테스트
  - 테스트애플리케이션을 이용하는 표준암호 연산 결과 비교
  - 암·복호화 전후 내용 비교

- 산출물
  - API 5 종의 소스코드, unit test 포함
  - API Manual
  - Sample Application(파일암호화, 통신암호화, 디바이스인증)
  - Test Application
  - TPM 초기화 매뉴얼
  - TPM 개발 가이드

암호 연산

- 적용 대상 샘플
  - Sample_Registration_Encryption
  - 설정파일 암호화 및 서명

- 암호 연산
  - TPM 활용/SoftwareOnly
  - 암호화
  - 해시
  - 서명
  - 서명 검증
  - 복호화



Init( )	TPM의 RSA키, 바인딩키등을 확인하고 없으면 초기화
cipher = Encrypt(plain, length)	TPM의 바인딩키를 이용해서 암호화
hash = Digest(data, length)	해시함수로 다이제스트 생성
signature = Sign(data, length)	TPM으로 서명하여 시그니처 생성
result = Verify(signature, data, length)	TPM의 공개키로 서명 검증
plain = Decrypt(cipher, length)	TPM의 바인딩키로 복호화

Sample Application

[디바이스인증]

설정파일 암호화:

```
WriteEncryptedConfig(filename, Data)
{
    Init( )
    Hash = Digest(Data, DataLen)
    Signature = Sign(Hash, Hashlen)
    CipherText = Encrypt(Data + Signature)
    writeConfigFile(filename, CipherText, CipherLen)
}
```

설정파일 복호화:

```
ReadEncryptedConfig(filename)
{
    Init( )
    CipherText = readFile(filename)
    PlainText = Decrypt( CipherText, CipherLen )
    Signature = getSignature(PlainText)
    Result = Verify(Signature, PlainText, PlainLen)
    If Result is not True then Fail
    Return getData(PlainText)
}
```

설정파일 예)

Encrypt( )

```
ServerIP="61.250.21.211";
ServerPort=31002;
DomainCode="ThingPlug";
GWAuthID="002655EDE8F1";
AuthKey="SP10030583key";
GWID="SC10007135";
DeviceID="";
Message="Temperature=28C";
```

Sign(Digest( ))

Sample Application

[통신암호화]

통신데이터 암호화:

```
EncryptComm( Data )
{
    Init( )
    CipherText = Encrypt(Data)
    Return CipherText
}
```

통신데이터 복호화:

```
DecryptComm( Cipher )
{
    Init( )
    Data = Decrypt( Cipher )
    Return Data
}
```

[디바이스인증]

인증정보 요구

```
RequestAuth()
{
    Init( )
    Nonce = random( )
    Send(Nonce)
    Data = Recv( )
    Result = Verify( Data,
                     Nonce + DeviceID, NonceLen + IDLen )
    Return Result
}
```

인증 요구에 대한 응답

```
ProcessAuth()
{
    Init( )
    Nonce = Recv( )
    Signature = Sign(Nonce + ID )
    Send(Signature)
}
```