A Study on applying the CSSM API for Security Interface to Electronic Commerce

Yeong Deok Kim*, Dong Myung Shin**, Yong Rak Choi***

Abstract

There are many security problems with Electronic Commerce since insecure public networks, especially Internet, are used. Therefore, for implementing secure Electronic Commerce, CAPIs (Cryptographic Application Programming Interfaces) is expected to use various form of security applications. The Cryptographic Application Programming Interface supports cryptographic services for each level and various security services. The CSSM API (Common Security Service Management Application Programming Interface) provides modularity, simplicity, and extensibility in terms of various add-in modules and interfaces in contrast to other CAPIs. This paper proposed an applying method of CSSM API having various extensibility and supporting multi-platforms to Electronic Commerce. we describe encryption, digital signature operation of CSSM API’s CSP interface and evaluate secureness by matching relation of threatening factors to security services.

* 대안보안전문대학 사무사동학과 조명교수
** 대안보안대학 대학원 컴퓨터공학과 (박사과정)
*** 대안보안대학 전자디지털정보공학부 교수
I. Introduction

The issue of protection of information in Electronic Commerce has been the subject of research and development in each application area according to its need. There have been attempts to solve the problem by using protocols based on Internet mainly although the basic problem-solving has been difficult. Further, there has been the problem with compatibility and availability according to the development of non-formalized or non-standardized information protection mechanisms of vendors; it is necessary to develop separately the method of combination of an application and cryptographic modules for developers; there is an added burden of development since a considerable amount of knowledge in cryptography is required for developers.

II. Factors threatening Electronic Commerce

The greatest factor for threatening Electronic method of Internet payment Commerce is illegal use of a person's information by a third user through interception of an important message such as personal information, credit card information, etc. being transmitted, exposure of information, or modification or fraud of information. In as much as Electronic Commerce is trading through networks contrary to the conventional commerce, there may be the third attack, i.e., the business attack, that can occur due to threatening factors such as forgery, modification, tapping, etc. of the network on communication lines as well as characteristics of commercial trading. Based on this, the factors threatening Electronic Commerce are classified as follows [1]

- System attack: Threats that can occur due to unlawful use of a computer by an outsider by entering the system, flow-out of information, destruction of information, abuse of the authority by an insider including misuse of the authority, utilization of intended reliability, misuse of a privileged program, etc.
- Data attack: The data attack in Electronic Commerce is divided into two: One is an attack to the data stored in a system, and another is an attack to the data floating around on the network.
- Business attack: There may be the third attack that can occur due to the characteristics of commercial trading in Electronic Commerce, which is often called the business attack. It is possible to have a fraud that can occur only in commercial trading. Since it is not possible to stop all these only with the cryptography or system, there should be external supplement to the electronic system such as an institutional device, legal assurance, insurance, etc.
- Internet banking system
  - Threatening factor in client security
  - Threatening factor in trade processing security
  - Threatening factor in server security
  - Threatening factor in application security
  - Threatening factor in internal security
Threatening factors in the
- Delivery of a credit card number through Form-GET: By using clear text
- Delivery of a credit card number through Form-GET: By using Netscape SSL
- Subscriber based home page

Table 1. Comparison of widely-used security APIs

<table>
<thead>
<tr>
<th>Criteria for comparison</th>
<th>IDUP-GSS-API</th>
<th>GCS-API</th>
<th>Cryptoki</th>
<th>CryptaAPI</th>
<th>CSSM-API</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm independency</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Application independency</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Cryptographic module independency</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Degree of knowledge in cryptography</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Module design and auxiliary services</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key management</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Cryptographic module verification</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>User certification</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Certification management</td>
<td>Some</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Query ability</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Installation/Uninstallation ability</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Safe programming (added value of 1-6)</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Security perimeter</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Each CAPI evaluated provides cryptographic modules, algorithm, and application independency, but is somewhat different from each other in view of the degree of knowledge in cryptography of an application developer, module design, and auxiliary services(3,4).

The CSSM-API retains more superior security functions than GSS-API, Cryptoki, or CryptaAPI since general requirements for security APIs are met. And its value of existence is not in doubt contrary to GCS-API, it has a superior extensibility as the add-in module is used, functional extensibility is added by selecting an element in each layer in terms of separated modules, and it follows an open structure for the application of standards and adoption to industries.
IV. Applying CSSM API for Security Interface

Many threatening factors in Electronic Commerce have been analyzed, and the information protection services against such threatening factors analyzed have been studied until now. Still there are many problems in the application of the information protection technologies thus developed to Electronic Commerce in addition to direct security problems in Electronic Commerce. That is, although mechanisms of various methods have been developed through gradual development of the information protection technologies, it is still difficult to graft various information protection techniques, and the present information protection services that are very burdensome to developers are obstacles to the development of Electronic Commerce. Many types of information protection services for each application program and each Internet protocol are provided with in order to solve such important problems in Electronic Commerce.

As most methods of Electronic Commerce are developed based on Internet at present, secure protocols such as S-HTTP, SSL, SET v.2.0, etc. are proposed and implemented, which have the following problems as the information protection services for each application program: The first, the problems with compatibility, availability, and commercialization in Electronic Commerce are presented due to the non-standardization work such as the development of protocols for each vendor, implementation of information protection mechanisms, etc. The Second, there have been an increased burden of internal expenses for development due to an increased use of cryptography commercially and inconsistent development. The Third, a program developer is able to provide a security service which is proper for each level if only he/she knows about detailed cryptographic support sub-structure for an application requiring for various cryptographic knowledge. The Fourth, it is necessary to match each application program with cryptographic modules for Electronic Commerce.

![Diagram of Electronic Commerce application using CSSM API](image)

Therefore, it is possible to offer more effective commercial trading services by applying the security API technology which enables reduction of a burden of cryptographic knowledge for a program developer, cut-down of unnecessary expenses for the development and investment, and facilitation of matching between the cryptographic service and application to Electronic Commerce. Also, various services may be offered more safely and readily if security APIs are further developed and become more elaborate. The security APIs can provide high-level efficiency and availability commercially since they can provide cryptographic services in each level for the applications in various Electronic Commerce that require for cryptographic knowledge and are compatible with various platforms.
and cryptographic algorithms. Particularly, application of the CSSM API proposed in this paper enables offer of cryptographic modulation as well as various extensibility and simplicity through many interfaces[5].

Figure 3 shows that it is possible to provide safe Electronic Commerce technologies through CSSM API supporting various technologies and platforms.

```
CSP_Encrypt(CC, Plaintext)
  check_ContextType(CC)
    If ContextType = CSSM_ALGCLASS_SYMOMETRIC and ContextAlgorithmType = CSSM_ALGCLASS_DES
      Call CSP_EncryptDataInit(Plaintext)
      Clear CSSM_DATA_PTR
      For Plaintext_Size bigger than Block_Size
        Call CSP_EncryptDataUpdate(Plaintext)
        Encrypt_Padding_Data <- Encrypt_Data
      Otherwise,
        Call_EncryptDataFinal(Plaintext)
        Padding_Data <- Plaintext + padding_bit
        Encrypt(Padding_Data)
        CSSM_DATA_PTR <- Encrypt_Data
      End
    Else
      Send_Service_Reject
  End
```

Fig 2. CSP Encryption Operation

Among the CSP interfaces, Interface related to encryption include CSP_EncryptDataInit(), CSP_EncryptDataUpdate(), CSP_EncryptDataFinal() functions. This function take process in order like a Figure 2. First, Encryption Process is initialized by calling CSP_EncryptDataInit() and then divide input data with block size if block-cipher algorithm is used. CSP_EncryptDataUpdate() function process real encryption procedure repeatedly until data size is smaller than the block size. When All the encryption process is finish, CSP_EncryptDataFinal() function is called for completion encryption process.

```
CSP_Signature(CC, Plaintext)
  check_ContextType(CC)
    If ContextType = CSSM_ALGCLASS_SIGNATURE and ContextAlgorithmType = CSSM_ALGID_HASH
      Call CSP_SignatureDataInit(Plaintext)
      Clear CSSM_DATA_PTR
      Call CSP_SignatureDataUpdate(Plaintext)
      Signature_Data <- Hash_Plaintext
      CSSM_DATA_PTR <- Signature_Data
      Call CSP_SignatureDataFinal()
      SignAlgorithm_type, plaintext, signature_data
    Else
      Send_Service_Reject
  End
```

Fig 3. CSP Digital Signature Operation

Interface related to digital signature include CSP_SignDataInit(), CSP_SignDataUpdate(), CSP_SignDataFinal() functions. This function take process in order like a Figure 3. First, Signature Process is initialized by calling CSP_SignDataInit(), and CSP_SignDataUpdate() function perform signature the hash data. When All the signature process is finish, CSP_SignDataFinal() function is called for completion signature process.

IV. Evaluation of Applying Method

In this paper, a method of application of security APIs for the secure environment for Electronic Commerce is reviewed. Particularly, the threatening factors for information protection in Electronic Commerce are established as follows by setting up threatening factors, services, and mechanisms for Electronic Commer-

ce and analyzing their interrelationships:

- Threat 1 (A1): An act of intentionally delaying the flow of information or of modulating the order
- Threat 2 (A2): An event of changed information on the transmission line
- Threat 3 (A3): An event of exposed user identification information during transmission
- Threat 4 (A4): An act of denying the fact of sending or receiving the information
- Threat 5 (A5): An act of disguising an unauthorized party as a lawful user by using the network information
- Threat 6 (A6): An act of offering information unlawfully by an internal user of a company to an outside user
- Threat 7 (A7): An act of harming confidentiality of the stored information
- Threat 8 (A8): An act of maintaining integrity according to real-time access to information

The information protection services which are required for Electronic Commerce in order to minimize or remove damages from above-described threatening factors are as follows, which should be provided during transmission and sharing of messages:

- Service 1 (S1): Confidentiality service for the prevention of exposure of important sending and receiving information on the network in Electronic Commerce
- Service 2 (S2): Integrity service for the confirmation of possible change of the message transmitted
- Service 3 (S3): Certification service for the confirmation of the identity of a user desiring to use the information and system resources
- Service 4 (S4): Non-repudiation service for the prevention of denial of the fact of sending or receiving when a message is sent successfully

- Service 5 (S5): Access controlling service for allowing only an authorized user to use resources

The following mechanisms are provided in order to offer information protection services which are required for Electronic Commerce through the applied security API:

- Security mechanism 1 (M1): A mechanism for a confidentiality service for sending the data by a sender to a receiver without their exposure supporting CSP_Encryption() in CSSM API
- Security mechanism 3 (M2): A mechanism for an integrity service that the data transmitted to a receiver are transmitted with no change supporting CSP_Digest() in CSSM API
- Security mechanism 3 (M3): A mechanism for a certification service in order to identify that a sender of the data is a lawful user offering CSP_Sign() in CSSM API
- Security mechanism 4 (M4): A mechanism for a non-repudiation service in order to block denial of sending or receiving of the data of a sender or receiver offering CSP_Sign() in CSSM API
- Security mechanism 5 (M5): A key man-
agement mechanism performing the request, generation, distribution, disposal, etc. of keys processed through the interface for the user key management and certification-related key management offering CSP_Keypair() and CSP_DeriveKey() in CSSM API

Table 2 and Table 3 show correlations between the information protection threatening factors and services and between the services and mechanisms which are set in order to study the method of safe Electronic Commerce.
Table 2. Relationship between threatening factors and services in EC

<table>
<thead>
<tr>
<th>Service</th>
<th>Threatening factor</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
<th>A7</th>
<th>A8</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Relationship between Information protection services and mechanisms in EC

<table>
<thead>
<tr>
<th>Mechanism</th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>S5</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>□</td>
<td>□</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
</tbody>
</table>

Table 4. Relationship between mechanisms and threatening factors in EC

<table>
<thead>
<tr>
<th>Threatening factor</th>
<th>Mechanism</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
<th>A7</th>
<th>A8</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VI. Conclusion

Among CAPIs, which can be modulated, CSSM API has a sufficient extensibility, and conforms to the CAPI evaluation criteria, is thought to be proper for Electronic Commerce applications. This CSSM API can support four security services of confidentiality, integrity, certification, and non-repudiation which are necessary commonly for various types of Electronic Commerce. Particularly, it has an advantage of accommodating a multi-layered architecture.

Continuous study on the security APIs for the construction of basis of secure Electronic Commerce, extended application of the updated cryptographic algorithm and the reliable mutual certification system in the implementation of security modules of CSSM API, and the study on standardization of extended security services are reserved for future study.
References