Utah State University

*E-mail : http://ym830@netian.com
**E-mail : http://sskgang@chirinu.ac.kr
***E-mail : http://wankyoo@chirinu.ac.kr
****E-mail : http://jksong@chirinu.ac.kr

요 약

본 논문에서는 개방형 통신망에서 전송중인 데이터를 암호화시켜 정보의 누출을 방지하고 송신자 가 인정한 수신자만이 이러한 정보를 받을 수 있도록 한 암호화 알고리즘을 제시한다. 암호화 방
법에는 크게 랜덤키 암호화 방법과 공개키 암호화 방법으로 나누는데 본 논문에서는 랜덤키 암호화
방식의 개념을 이용했다. 이 알고리즘은 통신사간과 저장공간을 적절하기 위해 전송할 데이터를 암
호화한 다음 암호화키에 의해 복호화가 가능하게 되며 암호화 key를 생성하기 위한 파라미터로서 키를 생성하게 하는 것이
특징이다. 파라미터는 키 값이 생성됨과 동시에 전송되며 암호화된 파라미터를 복호화 키로 변경시켜 키를 제
생시키는다. 암호키의 구성요소인 random number 는 table 형태로 저장되는데 기가 40자마다
table을 재설정시 key의 보안을 강화하였다. 이렇게 생성된 키와 원래 키는 인증과정을 거쳐 암호
화가 이루어진다. 복호화는 전송된 파라미터를 조사하여 복호화 키로 구한 다음 암호화 키로 인식으
로 수행한다. 본 논문에서 제시한 알고리즘을 구현 및 평가결과는 100KB 배치시 0.0152/sec 정도로
빠른 수행이 되었다.

abstract

In this paper we propose an encryption algorithm for security in data communication, this algorithm acts
encryption operation after the compression of data in order to reduce the transmission time and storage
an encryption key is generated by using a parameter. as soon as key value is generated the parameter is
transmitted and key is recreated every 26 times of parameter changing, the random number which is a
constituent unit of encryption key is stored in a table , the table is reorganized when the key is generated
40 times in order to intensity the security of encryption key, the encryption of data is made through the
operation process of the generated key and sour data and the decryption performs the revers operation of
encryption after getting decryption key by searching the transmitted parameter, as this algorithm is
performed fast it is possible to be used in practice.

1. 서 론

전 세계는 하나의 전자 시장으로 연결되었다. 더
히, 인터넷을 통해 쉽게 정보를 얻 수 있다. 그
러나 해킹으로 인한 위협 등 보안상의 위협으로 가
상공간의 사기와 속임수가 발생할 가능성이 높.
이런 상황에서 전자결제 방식으로는 수의 경우

와 같이 결제 결과에 자금을 날리 두고 이에 대한 지
불을 요청하는 방식인 e-debt방식과 크레디트 카드
와 같이 신용을 이용하여 제공하고 사용은 통행하는
형태의 e-credit 방식과 그 자체가 현금과 동일한
e-cash방식 등이 있다. 이러한 방식은 거래당당자, 
업자간 및 전자기관 등 3 자 이상의 관계로 진행
되는 의무의 시이다. 그러나 기본적으로 전자결제에
서 반드시 필요한 것이 신뢰성, 확실성, 비밀성 등으
로 이러한 요건을 충족하는 방식으로는 현재까지 약
화 및 전자서명의 방법론이기 때문에 전자결제의 경우에도 암호화 및 전자서명이 널리 사용되고 있다.

제한하는 전자결제시스템에서의 보안기법은 암호화 key를 생성하기 위해, 적정 파라미터 값을 주어 그 값에 의해 key를 생성하는 2차 key 알고리즘을 제한하고 key값을 구성하는 random number table의 내용을 일정한 횟수마다 변경시켜 암호화 key의 보안을 강화하였다.

본 논문의 구성은 다음과 같다. 2장에서는 전자결제시스템에서의 보안기법으로 암호화 방식에 대해 요약한다. 그리고 3장에서는 전자결제 보안 알고리즘 설계 및 구조에 대해 요약했으며, 4장에서는 구축한 알고리즘에 대해 수행효과 결과를 요약하고, 마지막으로 5장에서는 결론과 향후 연구계획에 제시한다.

II. 전자결제 시스템에서의 보안기술

2-1 전자결제 시스템 개요

전자결제시스템은 유 - 무선 통신장치와 컴퓨터 등을 통신회로선으로 연결하고 하드웨어나 소프트웨어 프로그램을 이용한 거래 및 결제를 할 수 있는 전자시스템이다.

특히, 공문서의 기안, 결제, 전송, 공문서 송·수신 내정을 관리하기 위한 공문서 결제 시스템과 금융기관의 예·수입에 따른 금융 결제 시스템 등이 있다. 이러한 시스템 상에서 결제 방식은 거래당사자, 인증기관 및 결제기관 등에 따라 그 관계를 형성하는 의사표현이 다를 수 있다. 그러나 기본적으로 전자결제 방식은 신용요건과 유효성, 확실성, 비밀성 등으로 이러한 요건을 충족하는 방식으로는 암호화 및 전자서명의 방법론이기에 핵심화하여 보안화 및 전자서명이 널리 사용되고 있다. 현재 개발형 통신망 상에서 안전한 전자결제가 설계되므로 하한 시스템 개발이 이루지가 있다. [2]

2-2 전자결제 시스템에서의 전자결제 및 암호화 2-2-1 전자결제 개요

전자결제는 전자적 데이터에 대해 비밀적이고 해석이 불가능한 표시로 인증기관에서 발행한 인증된 정보로 독특화될 수 있고 서명이 소유자 및 데이터 변경되지 않음을 확인 할 수 있는 것이다.[1]

2-2-2 전자서명의 종류

(1) 디지털 전자서명

디지털 전자서명은 송신자와 받자자의 문서에 담긴 서명 모양으로 수신자와 컴퓨터 또는 중간계체자에게 미래 데이터에 대한 유효성 확인이 가능하게 만든 후, 특수(coder)로 변환, 컴퓨터 또는 하드웨어에 서명하여 문서와 함께 송신하면 수신자 컴퓨터에 그 정보 그대로의 서명이 나타나게 되고 수신자 컴퓨터는 그 정보를 등록하여 있는 송신자의 서명과 비교하여 이를 확인하는 방식이다. 단점은 미래 서명을 등록하여야 한다는 점이다.[2,13]

(2) 기타 전자서명

수신자 서명을 미리 수신자 컴퓨터 또는 중간계체자에게 등록시킨 다음 송신자 또는 전자서명 작성 후 컴퓨터에 부착된 문서-sign지에 송신자라며, 마지막 서명이 수신자 컴퓨터 또는 중간계체자에게 도달하면 컴퓨터가 미리 등록된 지문과 대조하여 확인하는 방식이다. 단점은 미래 서명을 등록하여야 한다는 점이다.[2,13]

1) 대칭 암호방식을 이용한 전자서명

대칭 암호방식을 이용한 전자서명은 심화에서 전자서명기능 수행목적으로 사용되는 방식으로서 송신자와 수신자는 각자의 컴퓨터에 동일한 병렬적 암호프로그램을 설치한 다음 각자의 비밀번호를 가진다. 송신자는 전자문서(A)를 작성한 후 자신의 비밀번호(X)와 상대방 비밀번호(Y)를 입력한다. 그러면 코드의 숫자 값과 X Y에 대해 암호화 알고리즘을 적용하여 결과 값(S)을 만드는 방식이다. 복호화는 전자문서에 대해 코드의 숫자 값과 송신자 암호알로고리즘과 동일한 방식으로 결과 값을 만드는 방식이다. 이 방식은 위·변수 사실 확인을 하기 위해 코드 값 비교한다.[2,13]

2) 비대칭 암호방식을 이용한 전자서명

비대칭 암호방식을 이용한 전자서명은 송신자는 우선 비밀키와 공개키를 자신의 컴퓨터 스스로 만들거나 공용서비스 사업자로부터 받는다. 비밀키는 개인 비밀번호에 동일한 것으로 공개키는 그 비밀키가 개개인의 것이 아닌지 여부를 확인하기를 원한다. 이 방식은 송신자(가) 로서 전자문서(A)를 작성하여 암호를 선택한 후 그 전자문서에 대해 해ราย 암호키(Z)를 적용하여 암호를 만든다. 해려 암호키는 현재의 메시지 암호화의 일정과 비밀번호로 대손된 보안을 제공한다. 이 방식은 위·변수 사실 확인을 하기 위해 공개키와 민밀한 비밀키가 있는 메시지 암호화가 해야만이 비대칭 암호방식으로 표준화된다. 또한 해려 암호키는 그 메시지 암호화가 그 메시지와 유일하게 대조되는 일정이 비밀화를 표현하며 해려 암호키는 그 메시지 암호화와 해려 암호키를 이용하여 공개키를 얻을 수 있다. 이 방식은 공개키를 이용하여 암호화를 하여도 해려 암호키를 이용하여 암호를 해독할 수 있기 때문에 보안을 제공한다. 이 방식은 위·변수 사실 확인을 하기 위해 공개키와 민밀한 비밀키가 있는 메시지 암호화가 해야만이 비대칭 암호방식으로 표준화된다. 이러한 시스템 개발에 의하여 보안화 및 전자결제가 보강될 수 있다.
자와 비밀키와 암호화하는 학이 아니 경우 코드 같은 다르게 된다.[2]

2-2-3 전자문서의 보안 전자문서의 보안기법이 제공 할 수 있는 기능은 다음과 같다.
(1) 신분확인 서비스 신분확인서비스(IDENTIFICATION/AUTHENTICATION)는 네트워크 보안을 위해 상대방의 신분을 확인하는 것이다. 자신의 신분을 증명하는 것을 신분확인이라 하며, 대칭해시, 방식해시 신분확인을 나뉜다.[3,4,5,6]
(2) 액세스 제어 서비스 액세스제어 서비스(ACCESS CONTROL SERVICE)는 사용자와 대출한 실제의 액세스를 위한 자원과 액세스 동의 유형에 대한 정보를 검증하는 서비스이다. 특권의 최소화(LEASE PRIVILEGE), 백나무의 통산(ECONOMY OF MECHANISM), 수용성(ACCEPTABILITY), 품질 문제(COMPLETEST MEDIATION), 개방성 설계(OPEN DESIGN)와 설계원칙을 따르다.[3,4,7] 10]
(3) 저작의 무결성 저작의 정확성(DATA INTEGRITY)은 전송되는 데이터의 정확성을 지키는 서비스이다. 암호화 절차를 정확히 하는 새로운 푸스템(CONTEXT INTEGRITY)과 전송되는 전문의 순서를 검증하는 순시 무결성(SEQUENCE INTEGRITY)으로 나뉜다.[3,4]
(4) 비밀보장 비밀보장(DATA CONFIDENTIALITY)은 네트워크에 데이터가 불법적으로 그 내용이 노출되는 것을 방지하는 서비스로 전자문서에 대한 영향으로 선택적 필드에 대한 암호화로 구분되며, 암호화 메커니즘을 사용하여 전송되는 데이터의 내용을 감추는 것이다.[3,4]
(5) 부인 통제 서비스 부인부인(NON-REPUDIATION ORIGIN)과 수신 부인(NON-REPUDIATION, DELIVERY)이 있다. 발신자는 전송된 데이터 내용에 대한 발신 사실을 부인 할 수 없으며, 수신자는 수신된 데이터 내용에 대한 수신 사실을 부인 할 수 없게 한다.[3,4,12]

2-2-4 전자문서 암호화 전자문서에 대한 암호화는 다음과 같다.
(1) 전자문서 암호화 개요와 기법 전자결제 시스템에서 토론은 위트, 위조, 수정 등 이에 따른 안전대책이 하나로 암호화는 통합해시방식이 된다. 암호화(CRYPTOGRAPHY)란 송신된 자료가 식별되지거나 복사되지거나 누군가에 의해 조작되지 않도록 자료를 전송하기 전에 그 의미를 알 수 없는 암호문(CRYPTOGRAM)으로 변형시켜 전송한 후, 그것을 다시 해독하도록 하는 것을 말한다. 따라서 암호화 암호기계에 대한 입력 DATA로써 송신자가 수신자에게 보내는 보통의 통신문은 평문(PLAIN TEXT CLEAR TEXT)이라고 하고, 이 평문은 암호화 암호기계를 통해 나온 암호 데이터인 암호문(CIPHER TEXT)으로 변환시키는 작업을 암호화(ENCRYPTION)라고 한다. 암호화된 데이터의 형성적인 형태로 즉, 평문의 평문으로 조작시키는 과정을 복호화(DECRYPTION)이라고 한다. 복호화는 암호화된 데이터를 원래의 데이터로 원복하여 암호화 역할을 합니다. 암호화된 데이터를 열리는 방법으로 평문을 암호화 암호기계와 암호화 key로 암호문을 생성하여 수신자가 암호문의 원본을 받기 위해 송신자를 통해 수신자가 받고 있던 평문을 받게 된다.[3,4,9]
부적합은 왜일도의 누출 등으로 일어 날 수 있으므로, 이런 위험에 대비한 대책의 하나로서 암호화 가 이용된다. 또한 침묵의 고해도 사고 등의 체제에 의한 토론문서의 보안, 편의, 복잡 등에 대비한 물리적인 대책도 중요하지만 한계가 있으므로 이에 대한 보완력으로 암호화는 좋은 해결책이나 된다.[4,12]
암호시스템의 개발과정은 다음과 같이 3 단계로 구분할 수 있다.
제1 단계 : 암호 알고리즘의 개발(암호화 및 복호화 부분의 개발) 제2 단계 : 보안성 분석 및 효율성 평가 제3 단계 : 암호시스템 활용 암호시스템의 기본적인 기능은 데이터의 비밀성 사용자 사용이 보호할 수 있다는 것이다. 정보를 가진 전자문서의 보호를 위해 암호화를 사용한다. 암호화 알고리즘과 암호화 KEY로 암호문을 생성하여 전송한 후, 암호화를 통해 수신자에게 보내며, 수신자는 암호문을 받아 풀개 암호화 알고리즘과 복호화 KEY로 암호문이 복호화되며, 수신자는 암호화 알고리즘의 평문을 암호문으로 변환하여 두 부분의 임계를 하며 암호화 알고리즘과 복호화 KEY로 받아 송신자가 보내고 받은 내용을 얻게 된다. 암호화 문제의 기본기는 암호화 시제의 크기가 얼마나 암호화 기법은 크게 관용 KEY 암호화방식과 공개KEY 암호화방식으로 나눌 수 있다. 이러한 암호화 알고리즘의 평문을 암호문으로 바꾸어주는 관용KEY 암호화 방식은 암호화 및 복호화에 필요한 KEY가 다르다. 일반적으로 관용KEY 암호화 시스템은 수행 속도는 빠르나 인증성이 결려되어 공개KEY 암호화 방식은 암호화 및 복호화에 필요한 KEY가 단일이다. 일반적으로 관용KEY 암호화 시스템은 수행속도는 빠르나 인증성이 결려되어 공개KEY 암호화 시스템은 인증성은 있으나 수행속도가 느린 단점이 있다.[4,12]
1) 관용 암호화 방식 관용 암호화 방식(CONVENTIONAL CRYPTO SYSTEM)은 오래 전부터 널리 사용되던 방식으로 송신측과 수신측에 복잡한 KEY를 준비하여 암호화 및 복호화를 한다. 관용 암호화 방식이란 용이
한국해양정보통신학회 2001 추계종합학술대회 제5권 제2호

이 1996년 공개 KEY 암호화 방식의 개념이 발표되면서, 이에 대응하여 빠르게 이들로 위치교환방법, 정보교환방법, 대수적방법, 합성암호방법 등이 있이다.[3,4,11,16]

1) 위치교환방법(TRANSPOSITION)
위치교환방법은 정보의 각 자료들의 위치를 바꾸어 놓는 방법으로 정보에서 사용된 각 자료는 모두 그대로 사용되나 그 위치가 달라질 경우의 뜻을 알 수 없고, 이를 전체적 이를 병렬방법이라고 부른다. 이 방법은 다른 암호화 방식과 조합하여 사용함으로써 충분한 효과를 얻을 수 있으며, 다음 3가지 종류가 있다.

가) 각자의 순서를 반대로 나열하는 방법
나) 순서변환을 들로 나누어 적은 다음, 열(column) 순서로 나열
다) 정렬된 matrix의 크기에 따라 각자 위치를 바꾸는 방법

이 방법은 matrix를 만들 때 변환함 수 있으며 정보의 형태에서 각자 정보의 순서를 바꾸는 방법으로 정보의 형태와 맞게 요소와 요소를 반복할 수 있다.

2) 대수적방법
대수적방법은 (SUBSTITUTION)이 정보의 각 자료를 다른 위치에 다시 위치시키는 것으로 주변에 KEY TABLE에 의해 원래의 자료를 다른 위치로 위치시키며, 본래의 위치는 변하지 않으나 자료를 바꾸어 얻은 놓이 실제 행방법이 다.[3,4]

3) 변환암호
변환암호(PRODUCT CIPHER)는 전자 및 전자 암호화 방식의 혼합형이 미국이 개발한 DES (Data Encryption Standard)를 일본의 NTT가 보안하지 않으나 개발한 FEAL (Fastdata Encryption Algorithm)이 여기에 속한다. 미국 산업부 표준국에서 안전성 key를 만드는 것 또한 전자 및 전자 암호학에서 공개 모듈이며 IBM사의 W. L. Tuchman이 개발한 암호방식 1977년 D DES(Data Encryption Standard)에 재정정하였다.[3,4,11,15,17]

2) 공개키 암호화 방식
공개키 암호화 방식(PUBLIC-KEY CRYPTO SYSTEM)은 1976년에 Diffie와 Hellman에 의하여 발표된 명령어의 개념으로서 DES 등의 방식 암호화에서 중요시되고 있는 것이 매우 많은 것을 필요하다.

공개키와 해독키(key)를 별도로 가지고 있는 것으로서 사용자는 자기의 암호키(key)를 공개하고 해독키(key)는 비공개로 관리한다.

수신자는 자신의 암호키(key)와 해독키(key)를 사용하여 암호키(key)를 관리적으로 송신한다. 송신자는 종종을 암호키(key)로 암호문을 수신자에게 송신하고, 수신자는 이 암호키를 비밀리에 보관하여 부호키(key)로 송신하고, 수신선을 암호키(key)로 해독하여 송신문을 영동하도록 한다.

즉 암호키 가입자는 자신의 암호키(key)와 해독키(key)를 서로 다르게 하여 각자의 암호키(key)를 공개키(key)에 등록하고 부호키(key)를 보관한다.

예를 들어 인력의 송신자 A가 수신자 B에게 반복문을 전송할 때는 송신자의 공개 암호키(key)를 사용하여 암호문을 작성한 후에 B에게 송신한다. 이 때 B의 이외의 다른 가입자도 이 암호키를 사용할 수는 있지만 이 암호키는 복호화 할 수 있는 해독키(key)가 가지지 않기 때문에 복호화 할 수 없다.

즉 수신자 B만이 복호화 key를 가지고 있기 때문에 송신문을 복호화 할 수가 없다.

이러한 시스템은 암호화하기 위한 key와 복호화하기 위한 key를 소유하여 부호키(key)는 비밀로 하고 암호키(key)는 공개한 시스템이다. 암호키는 누구라도 작성하더라도 송신할 수 있지만 그것을 해독할 수 있는 것은 해독선이 될 것이다.

즉 암호키 가입자는 각자의 암호키(key)와 해독키(key)를 서로 다르게 하여 각자의 암호키(key)를 공개키(key)에 등록하고 부호키(key)는 본인이 관리한다. 이 시스템은 key 관리가 용이하며, 디지털 서명이 가능하다. 각 가입자는 자신의 공개키(key)를 가지고 있다고 해서 송신문을 받을 때 자신의 id와 공개 key로 자신을 보증한다.

따라서 이러한 공개 key 암호방식의 개념을 처음으로 설명하기로 건축한 RSA(미국의 MIT의 Rivest, shamir, adleman)에 의해 1978에 발표되었다.

RSA는 두 개의 큰 소수 p, q와 구하는 것은 쉽지 않지만, 공격에 주어졌을 때 인증을 받아 두 개의 소수를 구하는 것은 일반적인 소인수 분해의 이론을 이용하여 “trap door function”을 사용한 암호방식이 다.[3,4,11,14,16]
다. 이렇게 생성된 key 값과 원래 데이터는 연산을 거쳐 암호화가 이루어지게 되며, 40회마다 table 내용이 random number 생성에 의해 변경되도록 하여 key의 분석을 어렵게 하여 암호화 정도를 높이 하였다.

3-2 암호화 설계 및 구성
3-2-1 암호화 처리과정
암호화 처리과정은 [그림 1]과 같다.

[그림 1 암호화 처리과정]
여기에서 암호화 시각을 구하기 위해 시작시간 (get time)을 구한 후 저장공간을 통신시간을 점검하기 위해 입력 데이터를 암호화. 난수 생성기는 10 개의 6행 5열의 table 형태로 저장되는데 암호화 key 생성을 위한 parameter 값으로 이러한 random number의 위치를 나타내는 table number와 array 값이 전송된다. 전송된 parameter의 변환에 의해 암호화 key가 생성되고 입력 데이터와 암호화 key는 연산과정을 거쳐 8 byte 단위로 암호화가 이루어진다. table number와 array 값은 26행과 변경시켜 암호화 key를 재생성한다. 또한 암호화 key의 강도를 높이기 위해 random number를 40회마다 다시 생성하여 table 내용을 변경시켰다.
3-2-2 저장장치 암호화 알고리즘 설계
본 논문에 제안하는 암호화 알고리즘 (Encryption Algorithm)은 다음과 같다.
(1) compress input data using zip
(2) store in buffer A the value of generated by random number generator
(3) generate contents of buffer B the value from the fifth byte of buffer A
(4) store the first byte of buffer A rotated in the first 4 byte of buffer B
(5) store the value of key generated in the rest 4 byte of buffer B
(6) As soon as generate the contents of buffer B, buffer A is transferred
(7) change the end byte of contents of buffer A every 26 times
(8) store transferring data in buffer C
(9) buffer B and buffer C is computed using XOR and rotate
(10) change the contents of table by random number generator every 40 times

암호화 알고리즘의 구체적인 내용은 [그림 2]와 같다.

[그림 2. 암호화 알고리즘]
① 입력데이터를 8바이트 단위로 data buffer에 저장한다.
② 파라미터를 별도의 key 생성을 위한 head 부분 전송한다.
③ 저장된 data 형태로 변환시켜 다른 구조의 형태로 공통 이용하도록 바이어날 temp 로 복사한다.
④ 5 random number table의 32비트 (unsigned int) data를 받는다.
⑤ 암호화 key를 생성하기 위해 데이터와 key값과 rotate 시킨다.
⑥ xor 연산 data 생성한다.
⑦ 암호화된 data를 파일로 전송한다.

1) 난수(Random Number) 생성
key 생성을 위해 R · N 초기값 123456으로 저장
6행5열 구성 10개 데이터 그룹 R · N 암호화 key 참조 40회마다 다시 생성한다.
2) key 생성 파라미터 암호화 key를 생성하기 위한 변수 값의 구성은 [표 1]과 같다.

[표 1. key 생성 parameter 구조]

<table>
<thead>
<tr>
<th>?</th>
<th>table no</th>
<th>key 생성</th>
<th>array값</th>
</tr>
</thead>
<tbody>
<tr>
<td>3byte</td>
<td>1byte</td>
<td>4byte</td>
<td></td>
</tr>
</tbody>
</table>

3byte 는 수신자가 key 생성을 위한 parameter 값이 전송된 것이라는 것을 알기 위한 head 부분이 다.
그 다음 1byte 는 random number 가 위치한 것
을 나타내는 것으로서 10개의 table 중 어느 하나를
 Lösing table number 이다. 나머지 4 byte는 6행 5열의 table 에 순차적으로 위치한 random number
중 어느 하나를 참조하기 위한 array 값을 나타낸다. 이 parameter 는 26 회마다 변경시켜 key을 재 생성 시킨다.
3) 난수참조
10개의 table에서 저장된 random number는 참조되며, 40회마다 재생성된다.
4) 키 생성
   buffer A에서 파라미터 값으로 암호화 키를 생성하는 데는 buffer A의 구조에서 상위 4byte는 2byte씩 나누어 키 생성 생성된 8byte는 원복으로 2번, 두 번째 2byte는 원복으로 4번 rotate 시키고 buffer의 일부 4byte에 저장한다. 그리고 buffer A에서 구성한 table 넘 array 넘에 의해 생성된 random number 값은 4byte에 넣는다. 이렇게 생성된 키는 [표 2]와 같은 구조를 갖는다.

   [표 2. 키의 구조]

<table>
<thead>
<tr>
<th>상위 4byte</th>
<th>로타토된 값</th>
<th>생성된 키 값</th>
</tr>
</thead>
<tbody>
<tr>
<td>4byte</td>
<td>4byte</td>
<td>buffer B</td>
</tr>
</tbody>
</table>

   키 값 생성은 빌런 파라미터 값을 저장한 buffer A를 buffer B의 내용 값으로 생성과 동시에 전송한다.

5) 데이터 암호화 연산
   암호화 시키 전송할 data를 8byte 단위로 buffer C에 반복적인 수 buffer B에 구성된 키와 연산하여 암호화가 이루어진다. 전송된 data의 원복과 값을 임계한 buffer에 위치시켜 키가 반복되고 동시에 암호화를 시키게 되는데 XOR과 rotate를 이용하여 암호화 키의 원복과 플롯을 연산한다.

   먼저 8byte의 데이터를 16bit 4개의 프레이엄으로 구성하여 data의 첫 번째 프레이임은 1번 rotate 시켜 3번째 프레이임은 4번 rotate시킨다. 그리고 2번째 4번째 프레이임은 XOR시킨다.

   이렇게 변환된 data는 키와 다시 연산을 하게 되는데 키도 16bit 4개 프레이임으로 나누 후 수행한다.

3-2-3 복호화
   본 논문에서 제안하는 복호(Decryption Algorithm) 알고리즘 설계는 다음과 같다.
   
   (1) generator random number by the random number generator
   (2) get the parameter of key and inspect head
   (3) generator key by the value of parameter
   (4) reverse compute the generated key encrypted data
   (5) decompress the decrypted data

   복호화 과정은 다음과 같다.
   
   복호화 과정은 암호화 과정에서 전송되어온 파라미터 head 부분을 잡기 위해 8byte를 바꾸어 저장한다. 비비에 내용은 암호화 키 생성 파라미터를 복구하기 위해 자료구조와 변경되어 복구된다. 이에 반하도록 인식을 이용해 데이터를 복구하여 그 값은 키 값에 복구하여 rotate 시키고 생성된 파라미터를 원복의 암호화 키로 복구시킨다. 그리고 암호화된 실제 data 8byte를 data 비비에 임여들인다. 그리고 data 비비내용과 암호화 키를 XOR하여 파일에 저장한다.

   복호화 알고리즘의 구체적인 내용은 다음과 같다.

   [그림 3. 복호화 알고리즘]
   1) 전송 파라미터의 head 부분을 잡기 위해 비비에 8byte 임 저장한다.
   2) 비비에 내용은 암호화의 생성을 위해 자료구조가 변경되어 head 부분을 복사 전송한다.
   3) 3) 자료구조가 변경된 temp key를 이용 R-N table 참조 key 값에 복사 rotate 한 후 key 생성 파라미터 복구
   4) 암호화 data 8byte를 읽는다.
   5) 비비 data와 암호화 키를 rotate 및 xor 연산 data 복구
   6) des-res. zip에 저장한다.

IV 구현 및 평가

4-1 구현
   폴럼 체 부분에는 key 생성을 위한 parameter 값이 전송되었는지를 나타내는 head 부분이고 다 음 8 byte는 암호화 키에 의해 암호화된 부분이 다. 이러한 암호화는 매 26회마다 다른 값을 head 부분을 전송하게 된다. 폴럼 체 암호문은 [그림 4]와 [그림 5]가 같다.

   (1) 폴럼

   전 세계가 하나의 전자 시장으로 연결되어 있고 인터넷을 통해 쉽게 정보를 액세스할 수 있다. 그러나 해킹으로 인한 위협 등 보안상의 위협으로 가상공간의 사기와 속임수가 발생할 가능성이 높다.

   [그림 4. 폴럼]

(2) 암호문

   [그림 5. 암호문]

   복호화는 수신된 key 생성을 위한 파라미터 값을
이용하여 암호화 Key를 생성한 후 수신된 암호문으로 복호화 무언의 수행에 의하여 명문을 얻게 된다.

4-2 레가
본 논문에서 제안한 암호화 알고리즘은 인텔(R) C/C++ 프로세서, Memory 64MB, Windows ME 환경에서 수행결과를 평가하였다. 본 논문에서 암호화 알고리즘은 tcmC/c++ 프로세서, Memory 64MB, Windows ME 환경에서 수행결과를 평가하였다. 그 결과는 다음과 같다.

(1) 데이터 변동시 암호화 수행
 데이터 변동시 암호화 수행시간은 [표 1]과 같다.

<table>
<thead>
<tr>
<th>표 1. 데이터 변동시 데이터 변동 암호화 수행시간</th>
</tr>
</thead>
<tbody>
<tr>
<td>단위: 초</td>
</tr>
<tr>
<td>태이블변경암호화</td>
</tr>
<tr>
<td>10KB</td>
</tr>
<tr>
<td>암호화</td>
</tr>
</tbody>
</table>

(2) 태이블 변동시 복호화 수행
 데이터 변동시 복호화 수행시간은 [표 2]과 같다.

<table>
<thead>
<tr>
<th>표 2. 데이터 변동시 데이터 변동 복호화 수행시간</th>
</tr>
</thead>
<tbody>
<tr>
<td>단위: 초</td>
</tr>
<tr>
<td>태이블변경복호화</td>
</tr>
<tr>
<td>10KB</td>
</tr>
<tr>
<td>복호화</td>
</tr>
</tbody>
</table>

4-2-2 압도적평가
암호화하기로 분무화기를 갖추어야 함으로, 가장 중요한 것은 암호화 복호화가 정상에 맞는 점이다. 암호화 시스템이 일방적으로 사전화를 수행하는 암호화 간호에는 shannon이 정의한 무조건 안전(ungnitional secure)과 개선적 안전(computational secure)이 있다. 무조건 안전은 암호 해독자의 해독능력은 무한하다고 할지라도 암호해독에 이용할 수 있는 많은 병목문항에 암호화 시스템과, 계산적 안전은 암호해독자가 이용할 수 없는 정보의 양이 충분하여 연결자는 볼 수 있지만 그 해독과정이 복잡하고 시간과 경비가 많이 요구되며 경제적으로 무한이 부적합한 암호화 시스템을 말하는데 현재 대부분 암호화 방식은 계산적 안전을 중시하고 있다.[1,4,11]

본 논문에서는 정보를 짧게 위해 수행시간의 차이로 모든 경우를 비교하여 사용된 key를 찾는 전수함석에 대한 암호화 간호를 평가하였다.

64비트의 파라미터를 이용하여 64비트의 key를 생성하여 2의 64승 개의 가능한 모든 key 값을 조합하여 사용한 key를 본격적으로 검증하였다.

[표 5. 제안방식과 공개키 방식 수행결과]
V. 결 론

개발형 푸스틱이 점차 확장되고 발달에 따라 정보의 누출과 공격이 유효하게 되었으며, 전자결제시스템 상에서 보안문제에 대한 것은 암호화 과정이 가장 효율적이면 알래져야 한다.

본 논문에서는 개발형 푸스틱에서 얻은 전자결제시스템으로 보안을 위한 효율적인 암호화 알고리즘을 제시하였다. 그 특징은 다음과 같다.

1. 암호 KEY 관리데모서 KEY를 전송하지 않고 KEY생성을 위한 파라미터 값을 암호화하여 전송한다.

2. 그리고 파라미터 값은 나주에 의해 임의적으로 생성되고 나주는 주기적으로 바뀌도록 하여 데이터 양에 따라 암호화 강도를 높였다.

3. 양호화 알고리즘의 효율성 측면에서 처리시간 단축을 위해 과정을 압축하였다.

4. 그리고 자료처리 및 데이터 저장 측면에서 비용을 효율화가 가능하다.

본 논문에서 제시한 알고리즘은 메시지가 100KB일 때 암호화 및 복호화 시간은 0.0152/sec 소요되었고, 8 바이트 단위의 메세지를 나노초 속도로 컴퓨터로 해독하는데 약 34개월이 소요된다. 그리고 key를 생성하기 위한 파라미터 값이 주기적으로 변경되므로 데이터의 양에 따라 암호화의 강도는 더욱 향상되어 실무에서의 강력하고 효율적인 보안 알고리즘으로서의 활용이 가능하다고 판단되었다.

앞으로 보안의 강도를 강화하기 위한 보다 효율적이고 더욱 안전한 암호화 알고리즘을 위해 Key 생성을 위한 parameter 값을 범으로 유지하는 문제와 key로 생성 데이터 제생성에 따른 문제에 대한 지속적인 연구가 요구된다.

참고문헌