韓國 軍事運營分析 學會誌
第12卷 第1號, 1986. 6.

效率的 秘話 DBMS를위한 크립토그래픽 모델

남 길 현*

Abstract

Cryptography attempts to protect information by altering its form to make it unreadable to all but the authorized readers. DBMS is a most important computer application area requiring data security, but only a few cryptosystems are suggested for the database encryption.

This research develops a new Residue-Coded Cryptosystem based on the Chinese Remainder Theorem, which is considered to be more efficient than the database encryption scheme introduced by Davida, Wells and Kam in 1981.

1. 서 론

근대에 있어서 컴퓨터 산업은 이론과 응용분야의 급속한 발전과 함께 다량의 자료가 수집되어 컴퓨터 데이타베이스에 저장되고 있으며 복잡한 통신 네트워크를 통하여 컴퓨터 상호간 혹은 터미날 사이에 자료전송이 이루어지고 있다. 이와함께 보안을 요구하는 자료들에 대해서는 적절한 보안대책이 강구 되어야만 안심하고 컴퓨터를 사용하여 비밀자료를 처리 및 송수신 할수 있게 된다.

크립토그래피 (cryptography: 비화학)란 보통문(plaintaxt)을 암호문(ciphertext)으로 변형시킴으로서 비밀키를 알고 있는 인가된 사람 이외에는 해독을 할 수 없도록 하는 크립토시스템(cryptosystem)에 관한 분야를 연구하는 과학이다. 크립토시스템이라 함은 보통문을 암호화 하고 암호문을 해독하여 다시보통문으로 복귀시키는 시스템을 일컫는다. 대부분의 과학자들은 유선시설, 인공위성, 또는 마이크로 웨이브 등을 이용하는 전화통신 또는 컴퓨터 네트워크를 통하여 송수신되는 자료의 보안을 위해서는 크립토그래피를 응용하는 방안이 가장 효과적인 방책이라고 말하고 있다. 또한 저장된 자료의 분실이 우려되는 경우에도 크립토그래피는 자료보안을 위하여 효과적인 방안이 될 수 있다.

* 국방대학원

한편 현대의 컴퓨터 응용분야에서 빼놓을 수 없도록 매우 중요한 위치를 차지하고 있는 것이 데이타베이스 관리 시스템(DBMS)이다. DBMS는 자료의 저장뿐만 아니라 컴퓨터 네트워크를 통하여 끊임없이 자료의 송수신이 필요하다. 따라서 보안성을 필요로 하는 자료들을 관리해야 하는 DBMS의 경우에 있어서도 크립토그래피는 효율적인 해결책을 제공할 수 있다고 생각된다. 그러나 DBMS를 위하여 특별히 제시된 크립토시스템은 매우 드물다.

본 연구에서는 DBMS를 위한 크립토시스템의 특징과 DBMS를 위하여 제안되었던 DWK 크립토시스템을 알아보고 더욱 효과적이라고 할 수 있는 레시듀(residue)코드를 이용한 새로운 크립토시스템을 소개하려고 한다.

가. 자료보안과 크립토그래피

일반적으로 자료보안을 위한 통제방안들로서는 자료를 실제적으로 안전한 장소에 보관하고 사용자의 허가 여부는 신분증, 지문, 음성 등의 물리적 방법으로 판단하는 물리적 통제방법과 컴퓨터에 저장된 자료를 사용하거나 송수신할때 패스워드(password)나 문답형식등을 이용하여 사용자의 허가 여부를 판단할 수 있도록 하는 컴퓨터 시스템에 의한 통제방안등이 사용되고 있다. 그러나 이러한 방법들만으로서는 본래의 자료가 누구나 알아 볼 수 있는 암호화되지 않은 형태로 저장되거나 송수신되기 때문에 자료의 분실이나 통신중에 일어날 수 있는 도청의 경우에 대한 자료보안성은 매우 희박하다고 볼 수 있다. 이와같이 분실이나 도청으로 부터의 자료보안을 달성하기 위해서는 크립토그래피를 응용하는 방안이 매우 효과적이라고 할 수 있다. 즉 보통문이 암호화되어 저장되거나 통신망을 통하여 송수신 된다고 하면 자료가 분실되거나 도청된다 하더라도 비밀키를 모르고서는 해독이 거의 불가능하다고 인정되기 때문에 정보유출을 방지할 수 있다고 보는 것이다.

1977년 미국 표준국(National Bureau of Standards)은 IBM 회사가 제안한 DES (Data Encryption Standard)를 채택하여 비밀로 분류되지 않은 정부기관의 문서통신과 일반상업용으로 사용할 수 있는 크립토시스템으로 공포하였다. DES는 모든 알고리즘을 일반에게 알려주고 사용자는 비밀키만을 간직함으로서 암호화 및 해독을 할 수 있게 하였으며 특별히 제작된 하드웨어를 사용하거나 일반 컴퓨터를 이용하는 소프트웨어를 이용할 수 있도록 하였다. 보안성과 관련한 비밀키의 길이에 대한 논란이 있기는 하지만 DES는 세계적으로 널리 알려진 가장 대표적인 크립토시스템이라고 할 수 있다.

나. 통신의 신뢰성

컴퓨터 통신에 의한 자료 전송시에는 송신자가 보내는 자료와 실제로 수신자가 받게되는 자료에는 차이가 있을 수 있다. 첫째로는 통신채널의 잡음에 의한 자료변형을 들수 있고 둘째로는 적이 거짓정보를 제공하거나 자료를 무능화 시키기 위하여 고의적으로 자료를 변형시키는 경우라고 할 수 있다.

이와 같은 경우에 거짓정보를 검출해 내고 잡음에 의한 에러를 검출 또는 수정하여 컴퓨터 통신의 신뢰도를 높여주기 위해서는 에러 수정코드가 크립토그래피와 함께 사용되어야 한다.

즉 잡음에 의한 일부분의 미세한 자료변형은 에러 통제코드로서 검출 및 수정이 가능하고 암호화 할때는 비밀키를 이용하여 암호문을 만들어야 하기 때문에 거짓정보를 보내는 것은 매우 어렵게 된다. 대표적인 에러 통제코드 들로서는 Hamming 코드, BCH 코드, Reed-Solomon 코드, Residue 코드 등을 들수 있다.

2. DBMS와 크립토시스템의 응용

1970년대 부터 크립토그래피에 대한 연구가 활발히 진행되고 있지만 DBMS 응용에 대한 연구는 소홀히 되고 있는 형편이다. 일반적으로 사용하는 그립토시스템을 그대로 DBMS에 사용할수도 있지만 DBMS 고유의특성에 알맞는 크립토시스템이 개발될 수 있다면더욱 바람직한 일이 될것이다. 먼저 DBMS에 크립토시스템을 응용할 경우 고려해야 할 사항들과 그 운영조직을 살펴보기로 한다.

가. DBMS를 위한 크립토시스템의 특징

비 인가자에 의해서 자료가 직접 읽히거나 수정 또는 입력되지 않아야 한다는 점에서 볼때 크립토시스템을 이용한다는 것은 자료보안을 위해 적절한 방책이라고 할 수 있다. 그러나 자료보안을 위한 시스템 때문에 데이타베이스 구조 자체를 변경시키는 일이 있어서는 안될 것이다.

(1) 보안성

크립토시스템은 비밀키를 모르는 적이 불법적으로 암호문을 해독하려고 할때는 이론적으로 혹은 계산하는데 소요되는 시간이 너무 많기 때문에 해독이 거의 불가능하도록 해야 한다. 본 연구에서는 새로운 시스템의 보안성이 최소한 DES만큼은 좋아야 한다는 기준에서 분석하고저 한다.

(2) 암호화 단위

일반적으로 데이타베이스의 구조가 관계모형 계층적 모형 또는 네트워크 모형이거나를 막론하고 그 주소의 단위는 레코드(record) 이며 각 레코드는 여러개의 필드(field)들로 일정한 형식에 의해 구성되어 있다. 따라서 암호화할때의 단위는 각 레코드 또는 필드가 되어야 한다. 즉 n 번째의 레코드 혹은 n 번째 레코드의 어떤 필드는 그 전후에 있는 레코드를 암호화하거나 해독하지 않고 독립적으로 암호화될 수 있어야 하며 동시에 해독도 가능해야 한다. 이 조건을 충족시키기 위해서 연속성 암호방법(stream cipher)은 DBMS에 사용이 매우 곤란할 것이다. 암호단위가 레코드인가 아니면 필드인가에 따라서 두가지 방법으로 나누어 볼 수 있다.

(가) 레코드 단위 암호방법

레코드 전체가 하나의 단위가 됨으로서 각 필드의 값들은 그 레코드 속에 숨겨진 상태로 암호화 되는 방법이다. 따라서 레코드를 암호화 하기 위해서는 비밀키 셑이 필요하지만 각 필드를 해독하기 위해서는 각 필드에 따라 개별적인 비밀키만으로써 해독할 수 있다.

(나) 필드 단위 암호방법

레코드에 포함한 각 필드가 기본단위가 되며 각 필드별로 개별적인 비밀키를 사용하여 암호화 하고 해독도 하게 하는 방법이다. 따라서 필드단위 암호방법은 레코드를 더욱 짧게 세분하여 암호화학는 방법이라 할 수 있다.

(3) 각 필드단위의 해독키 보유

DBMS에서 사용자는 레코드내의 각 필드단위로 읽고 쓰는 권한이 주어진다. 즉 어떤 사람은 이름과 주소만을 다룰 수 있게 하고 어떤 사람은 고유번호와 급여액만을 취급 할 수 있도록 허가되어 있다. 따라서 각 필드단위로서로 다른 해독키를 갖고 있어야만 인가된 필드 이외에는 해독을 할 수 없게 된다.

(4) DBMS의 계산성 오버헤드(computational overhead)

아무리 DBMS를 위한 크립토시스템이라 하더라도 암호화하고 해독하는데 너무 많은 시간이 소요된다면 효율성 문제를 고려하지않을 수 없게 된다. 실제적으로 DBMS에 크립토시스템을 적용하지 못하는 가장 큰 원인이 바로 계산성 오버헤드 때문이라고 해도 과언이 아니다.

특히 DBMS에서는 해독하는데 더욱 빠른시

간이 요망된다. 왜냐하면 DBMS 사용자들의 요구사항을 살펴보면 일반적으로 읽는 경우(해독 : decryption)가 쓰는 경우(암호화 : encryption)보다 상대적으로 자주 일어나기 때문이다. 첨가해서 사용자의 요구 사항이 어떤 필드의 연산을 요구하는 경우 만약 그 필드를 해독하지 않고 암호화된 상태에서 그대로 연산을 해서 결과를 얻을 수 있다면 훨씬 효율적이라고 볼수 있다. 예를 들면 어떤 회사사원들의 연령 평균을 계산하기 위해서는 통상 연령을 나타내는 필드를 전부 해독한 연후에 서로 합산을 한후 평균을 구한다. 그러나 해독을 하지 않고 암호화 되어있는 그 자체를 합산한 후 결과만을 해독하여 평균을 얻을 수 있다면 해독하는데 소요되는 시간을 절약할 수 있기 때문에 매우 효율적이라고 본다.

(5) 저장의 효율성

자료보안이나 에러통제를 하기 위해서 본래의 정보 이외에 추가적인 정보가 첨가 되어야 한다는 것은 불가피한 일이지만 이 추가된 정보가 20～25% 이상 된다고 하면 저장의 효율성이 좋지 않다고 본다. 추가된 정보가 많을수록 저장비용의 증가 뿐아니라 계산속도에도 지대한 영향을 주기 때문에 과도한 양의 추가 정보는 피해야 한다.

(6) 통신의 효율성

일반적으로 컴퓨터에 의한 계산비용 보다는 통신비용이 높기 때문에 꼭 필요한 자료만을 송수신 함으로서 통신망을 통하여 전송되는 자료량을 최소한으로 줄이는 것이 바람직 하다.

(7) 패턴 비교방법 (pattern matching attacks)

해독키를 모르고 있는 경우라 할지 라도 이미 수집된 암호문과 해독문을 갖고 비교해 봄으로서 해독문을 추측하는 방법을 패턴 비교방법이라고 한다. 이러한 패턴 비교방법으로 부터 자료를 보호하기 위해서는 보통문에서는 미세한 차이가 있더라도 암호문에서는 아주 많은 차이가 나도록 해야하며 두개의 똑같은 보통문이라 할지라도 암호화 되었을때는 서로 다른 형태가 되도록 하는것이 요망된다.

(8) 치환 방법 (substitution attacks)

해독을 하지 않고 암호문 자체를 다른 암호문과 치환 함으로서 본래의 자료를 변경시키는 방법을 치환 방법이라고 한다. 예를 들면 사장의 봉급 암호문을 어떤 사원의 봉급란에다 그대로 베껴 넣은 경우를 생각한 수 있다. 이와 같은 치환 방법으로부터 자료를 보호하기 위해서는 레코드 전체를 한 단위로 묶어서 암호화하는 레코드 단위 암호방법을 사용하면 각 필드만을 따로 치환할 수 없게되며 완전한 레코드의 치환은 한 레코드의 중복이라고 볼 수 있으므로 의미가 없어지게 된다. 그러나 이경우에 있어서도 근래의 수정된 자료대신 옛날에 수집되었던 자료로 치환할 수 있기 때문에 치환 방법을 완벽하게 막아주는 시스템을 개발하기는 매우 어렵다고 본다.

(9) 에러통제 및 자동 반복 요구 (ARQ : automatic repeat request)

만약 통신망을 통하여 멀리 떨어져 있는 사용자가 DBMS를 이용할 경우 통신 채널에서 발생되는 에러를 검출하거나 수정할 수 있다면 통신 채널의 신뢰성을 높일 수 있을 것이다 에러 검출보다는 에러수정을 위한 코드는 더 많은 추가 정보를 요구하기 때문에 많은량의 자료 송수신을 위해서는 에러 검출만 할 수 있어도 효과적일 수 있다. 즉 에러가 발견될때 자동적으로 송신을 다시 하도록 요구하는 ARQ 시스템을 이용하면 이는 에러 발생 확율이 매우 작은 환경에서는 아주 효과절일 수 있다.

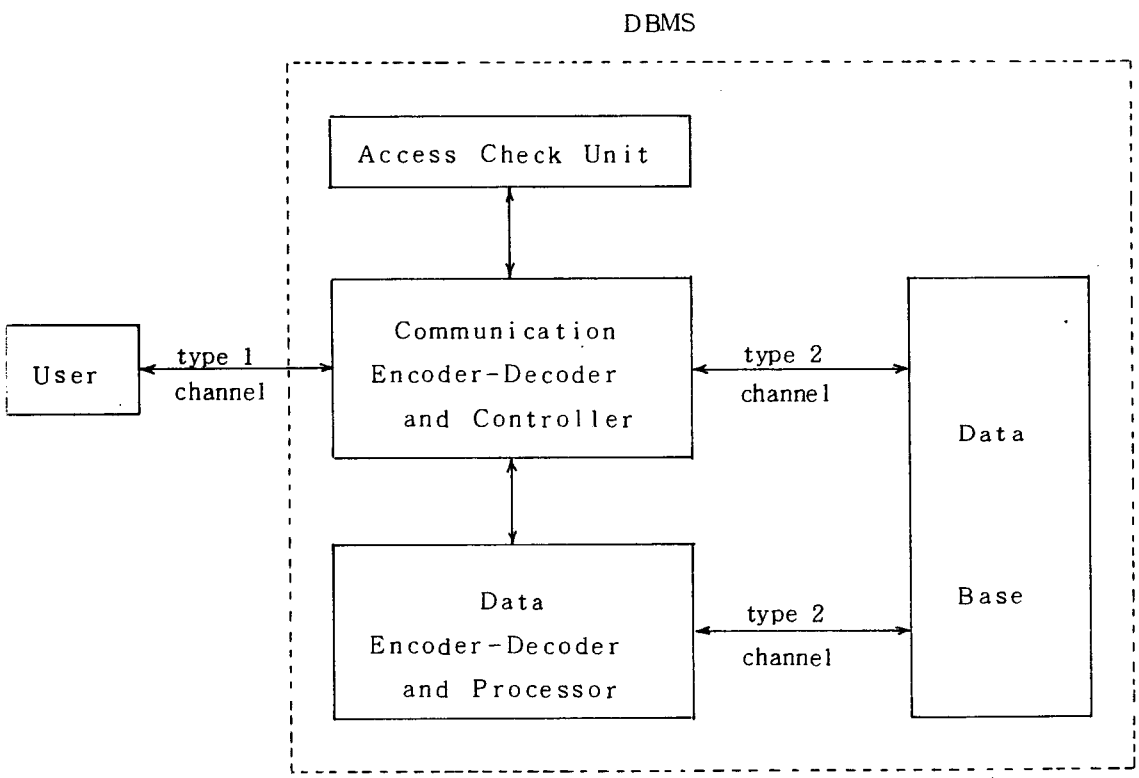
지금까지 DBMS를 위한 크립토시스템의 특징들을 알아보았지만 우리는 이러한 요구사항을 동시에 모두 만족시켜주는 시스템의 개발은 불가능 하다는 것을 알아야 한다. 왜냐하면 어

떤사항들은 서로 이율배반적 성질을 갖고 있기 때문이다. 예를 들어보면 치환 방법을 사용할 수 없게 하기 위해서는 레코드 단위 암호방법을 사용하는것이 좋지만 그렇게 되면 한 필드만을 해독하기 위해서도 레코드 전체를 사용해야 하기 때문에 계산성 오버헤드를 감당하기 어렵게 되는 것이다. 지금까지 어떤 시스템도 앞의 조건들을 모두 만족 시켜주도록 개발된 것은 없었다. 따라서 개발하고저 하는 크립토시스템은 DBMS의 주위 여건들을 고려하여 그 중에서 가장 적합한 방법이 될 수 있도록 연구되어야 한다.

나. DBMS의 운영조직

그림 1은 본 연구에서 제시된 크립토시스템을 응용할 수 있는 일반적인 DBMS의 운영조직 구조를 나타내 주고 있다. 물론 여러가지 주위여건에 따라 수정될 수 있으리라고 본다.

DBMS에 응용할 수 있는 크립토시스템이라 함은 원거리 송수신에 사용되는 제1형태의 채널 혹은 내부 데이타베이스에 자료보관용으로 사용되는 제2형태의 채널에 사용될 수 있다는 것을 뜻한다. 각 형태별로 다른 크립토시스템을 선택할지 또는 동일한 크립토시스템을 사용할 것인가는 DBMS의 성격을 분석하고 효율성을 감안하여 판단되어야 할 것이다. 구체적인 각 부분의 임무에 대한 설명은 여기에서 생략하기로 한다. 다만 데이타베이스에 저장된 자료는 반드시 암호화 되어야 하며 원거리 사용자와 송수신을 위한 자료는 암호화 되는것에 부가해서 통신의 신뢰성을 높이기 위해 에러통제 코드를 사용해야 한다고 가정한다. 에러 통제 코드라 함은 에러 검출 또는 에러 수정코드를 뜻한다.



〈그림 1〉 DBMS의 운영조직

다. 크립토시스템의 선택시 고려사항

시스템 설계자는 시스템의 특정한 주위 여건을 분석하여 가장 적절한 크립토시스템을 선택하여야 한단는 것은 앞에서 애기한 바와 같다. 선택기준의 비중은 각 시스템에 따라 다르기 때문에 일괄적으로 애기할 수 없지만 전체 시스템의 성취도에 영향을 주는 요소들은 다음과 같다.

(1) 요구된 자료의 보안성과 신뢰성

(2) 암호화 및 해독에 소요되는 계산상의 오버헤드

(3) 사용자의 요구사항에 따르는 자료량

(4) 계산비용과 통신비용

(5) 비밀키의 관리 유지 문제

(6) 사용자들의 특징과 필요사항

3. DWK 크립토시스템 (DWKC)

1981년 미국의 Davida, Wells, 그리고 Kam 세사람이 Chinese Remainder 이론에 기초를 둔 종속키(subkey)를 사용하는 크립토시스템을 데이타베이스 시스템에 적합하다고 소개하였다. 이 시스템을 본 연구에서는 DWKC라 부르기도 한다. DWKC는 레코드 단위 암호방법을 사용함으로서 적으로부터 필드 치환 방법을 못쓰게 할수는 있지만 다른 효율성에 있어서는 많은 문제점들을 안고 있다. 그러나 DWKC가 특별히 데이타베이스를 위한 크립토시스템으로 소개 되었다는 점에서 그 중요성을 인정 받고 있다고 볼 수 있다.

가. 암호화 방법 (encryption)

하나의 레코드 M을 K개의 필드($F_1, F_2, \cdots, F_K$)들로 구성되어 있는 보통문이 라고 가정하자. 그러면 M에 대응하는 암호문 C는 다음과 같은 공식에 의하여 얻을 수 있다.

$$C = \sum_{i=1}^{K} e_i (X_i \| F_i) \bmod D$$

여기에서

$D = \pi_{i=1}^{K} d_i$, d_i는 F_i에 대응하는 비밀 해독키(decryption key)

$\|$는 두 자료의 접합을 나타내고

X_i는 F_i를 위하여 생성된 임의의 난수이며 e_i는 비밀 암호키(encryption key)로서 다음식을 만족시켜야 한다.

$$e_i = \frac{D}{d_i} b_i \ , \ \frac{D}{d_i} b_i (\bmod d_i) = 1$$

난수들을 각 필드 앞에다 접합시키는 것은 보다 높은 보안성을 유지하기 위한 방책으로서 사용되었다.

나. 해독 방법 (decryption)

각각의 필드 F_i는 다음 공식에서 해독키 d_i를 이용하여 얻어질 수 있다.

$X_i \| F_i = C \mod d_i$, $i=1, \cdots, k$

즉 d_i를 이용하여 $X_i \| F_i$를 구한 후에 난수부분인 X_i를 제거해 버리면 구하고자 하는 F_i를 얻을 수 있다.

각 필드의 해독키 d_i는 필드값 F_i의 가장 큰값 보다 큰 소수(prime)중에서 임의로선택한다. 만약 난수 X_i가 필드값 F_i에 접합되지 않는다면 적에 의해 수집된 보통문과 대응하는 암호문들을 분석하여 d_i가 쉽게 발견될 수 있기 때문에 최소 32 비트 이상 크기를 갖는 난수를 암호화 할때 각 필드에 첨가 시켜야 한다

다. 문제점 및 해결방안

Chinese Remainder 이론에 기초를 둔 DWKC는 해독키를 알고 있는 사람들이 서로 타협할 경우 암호화키를 쉽게 계산할 수 있기 때문에 DBMS에 응용할때 자료의 수정이나 입력시에는 반드시 DBMS 자체에서 통제하도록 하여야 한다. 만약 개인이 암호화키를 이용해서 마음대로 자료를 수정하거나 입력시킬 수

있도록 한다면 자료의 판독(reading)만 허가된 즉 해독키만을 알고 있는 사람도 암호화키를 계산해서 자료 수정을 할 수 있는 가능성이 있기 때문이다. 이와 같은 기본적인 사항 이외에 파생되는 문제점과 해결방안을 알아 보기로 한다.

(1)에러 통제 능력 결여

DWKC는 에러의 수정이나 검출을 할 수 없기 때문에 컴퓨터 통신망을 통하여 자료를 송수신 할때는 에러 통제(에러 검출 또는 수정) 코드를 추가적으로 사용해야 한다는 불편이 있다. 이 점을 해결하기 위해서는 만약 에러 수정이나 검출을 할 수 있는 레시듀(Residue) 코드를 암호화하는데 이용하면 레시듀 코드 본래의 에러 통제능력을 갖는 크립토시스템으로 발전시킬 수 있을 것이다.

(2)레코드 단위 암호방법

DWKC는 레코드 단위 암호방법을 사용하기 때문에 거기에 따르는 많은 문제점들을 갖고있다. 첫째 암호화하고 해독하는데 소요되는 계산상의 오버헤드가 과다하다는 점이다. 즉 한개의 필드만을 해독하기 위해서도 전체의 레코드 값을 사용하여야 하기 때문에 만약 레코드가 여러개의 필드들로 구성되어 있을때는 레코드 전체의 길이가 필드와 비교 할때 상대적으로 매우 크며 따라서 암호화 하고 해독 하는데 훨씬 많은 시간을 필요로 한다. 둘째로는 자료 전송시 오버헤드가 높다. 이것은 첫번째와 마찬가지로 한개의 필드만이 요구되었다 하더라도 레코드 전체를 전송해야 하기 때문에 통신비용이 많이 소요된다. 세번째는 데이타베이스를 자주 수정해 주어야 한다는 점이다. 한개의 필드만을 이용할 수 있었던 사람이 그 자격을 상실할 경우 그 필드를 포함하고 있는 데이타베이스의 레코드 전체를 다시 써 주어야 하기 때문에 사용자가 빈번히 교체되는 경우에는 그 손실이 매우 클 수 밖에 없다.

이와 같은 일련의 문제점들은 모두 DWKC가 레코드 단위 암호방법을 사용하기 때문에 일어난다고 할 수 있으므로 만약 필드 단위 암호방법을 사용하는 크립토시스템을 개발하면 이러한 문제점들은 해소 시킬 수 있다고 본다.

(3)암호화된 자료계산의 문제점

데이타베이스를 이용하는 사용자의 상당부분은 단순한 정수 계산을 요구한다. 즉 어떤정수 필드의 총계 또는 평균을 구하는 것은 정수 계산이라고 볼 수 있다. 그러나 이와 같은 정수 계산을 하기 위해서도 반드시 해독을 한 후에 계산을 할 수 밖에 없다. 지금까지 제안된 어떠한 크립토시스템도 해독을 하지 않고 암호화된 상태에서 직접 어떤 계산을 할 수 있도록 하는 방안을 제시하지는 못하였다. 그러나 만약 가장 간단한 정수의 덧셈이나 곱셈만이라도 해독하지 않고 직접 암호문의 계산으로 할 수 있다면 이는 커다란 통계 숫자를 다루는 특수한 데이타베이스를 위해서는 매우 효과적인 결과를 얻을 수 있을 것이다.

지금까지 DWKC의 문제점들과 그 해결방안들을 간추려 보았다. 본 연구에서는 그 해결방안에 부합된다고 할 수 있는 레시듀 코드를 이용한 크립토시스템을 소개하고저 한다.

4. 레시듀 코드화된 크립토시스템 (RCC)

RCC는 에러의 검출 혹은 수정 능력을 갖고 있는 (n,k) 레시듀(Residue) 코드를 이용하여 암호화 할 수 있도록 개발 되었다.

DWKC가 레코드 단위 암호방법을 사용하는데 반하여 RCC는 필드 단위 암호방법을 사용하도록 하였으며 RCC를 DBMS에 응용할때의 다른 장점으로서는 정수의 덧셈이나 곱셈을 암호화된 자료를 해독하지 않고도 실행할 수 있다는 점이다.

가. 암호화 방법

레코드의 어떤 필드 하나를 나타내는 보통문을 M 이라고 하자, 그러면 암호문 C는 n-k 개의 에러통제용 레시듀를 포함하여 n개의 레시듀들로 구성된다.

즉 보통문(M)

$M=[\quad M\quad]$

암화문(C)

$C=[C_1][C_2][\cdots][C_{K+1}][\cdots][C_n]$

$C_1,\cdots\cdots,C_K$: 실자료 레시듀

$C_{K+1},\cdots\cdots,C_n$: 에러통제용 레시듀

암호화키들은 암호문의 각 레시듀 $C_1,C_2,\cdots C_n$에 대응하는 $d_1,d_2,\cdots,d_n$ 들로서다음식을 만족시키도록 하는 상대적 소수들(relative prime) 중에서 선택한다.

$$\prod_{i=1}^{K} d_i \geq \max(M) * Z_c$$

$$d_{K+j} > d_i,\quad j=1,2,\cdots,n-k$$
$$i=1,2,\cdots,k$$

여기에서 max(M)은 M의 최대값을 뜻하며 Z_c는 보안성을 높이기 위한 32비트 이상의 정수를 나타낸다.

암호문의 각 레시듀에 대응하는 암호화키가 선택되면 암호문은 다음 공식을 이용하여 얻을 수 있다.

$$C_i = (Z \| M) \bmod d_i,\quad i=1,2,\cdots n$$

Z는 Z_c 보다 작은 일정 길이를 갖는 난수이며 '||'는 두 수의 접합을 나타낸다.

나. 해독 방법

암호문을 해독하여 보통문을 얻기 위한 공식은 다음과 같다.

$$Z \| M = \left(\sum_{i=1}^{K} e_i \cdot c_i\right) \bmod D$$

여기서

$$D=\prod_{i=1}^{K} d_i$$

e_i = 비밀 해독키

$$=\frac{D}{d_i} b_i,\ \left(\frac{D}{d_i} b_i \pmod{d_i}=1\right)$$

따라서 보통문 M을 얻기 위해서는 윗 공식을 이용하여 Z||M을 구한 다음 Z부분을 없애면 된다.

암호문을 해독하는 외에 에러 검출이나 수정을 위해서는 신드롬(syndrome) 벡터(vector)를 계산하여야 한다.

한개의 레시듀 에러 검출을 위해서는 한개의 추가 레시듀가 필요하며 한개의 레시듀 에러 수정을 위해서는 두개의 레시듀가 추가되어야 한다. 에러 검출은 매우 쉽게 할 수 있지만 에러 수정은 훨씬 복잡하기 때문에 여기에서는 자세한 설명을 생략 하고저 한다.

그림 2는 RCC의 전반적인 암호화 및해독 과정을 나타내 주고 있다. 실제적으로 예를 들어보자

만약 $d_1=7,\ d_2=5,\ d_3=11$을 선택하면 $D=385, e_1=330,\ e_2=231,$

$e_3=210$이 된다.

보통문 $M=15=01111_2$,

임의의 난수 $Z=2=010_2$ 라고 한면

$Z \| M = 01001111_2 = 79$가 된다.

따라서

암호문 $C=(C_1,C_2,C_3)$

$=(79 \bmod 7,\ 79 \bmod 5,$

$79 \bmod 11)$

$=(2,4,2)$가 된다.

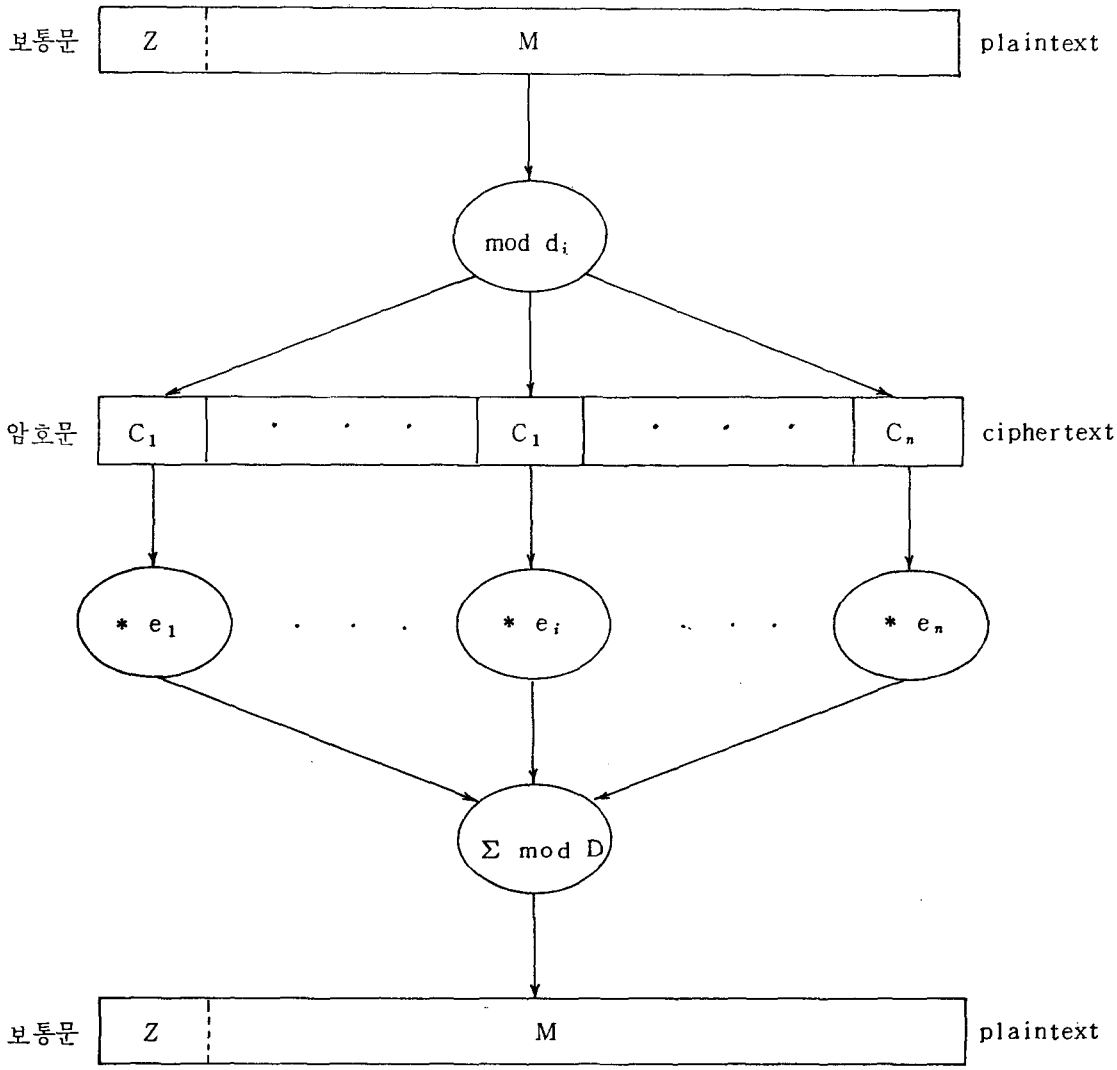
다시 보통문 M을 얻기 위해서

$Z \| M = (2 * 330 + 4 * 231 + 2 * 210) \bmod 385$

$= 2004 \bmod 385$

$= 79$

$= 01001111_2$



〈그림 2〉 RCC에서 암호화 및 해독과정

해독하는 사람은 난수의 길이를 알고 있기때문에 앞의 3비트를 제거하고 보통문 $M = 01111_2 = 15$를 얻게 된다.

다. RCC에서의 자료 연산

전형적인 DBMS 크립토시스템에서는 가장 단순한 정수의 연산을 할 경우에도 모든 자료들을 먼저 해독한 이후에 연산을 할 수밖에 없으며 이 결과로 DBMS에 미치는 계산성 오버헤드는 매우 크다고 할 수 있다. 그러나 RCC에서는 레시듀 숫자로 암호화되어 있기 때문에 암호문을 해독하지 않고 레시듀 숫자의 연산방법을 이용하여 직접 정수 연산을 할수 있다는 이론적 근거를 제공한다. 즉 레시듀 숫자의 덧셈, 뺄셈 그리고 곱셈은 다음 공식대로 아주쉽게 할 수 있다.

두개의 정수 U와 V가 레시듀 숫자로 표현되었다고 하면,

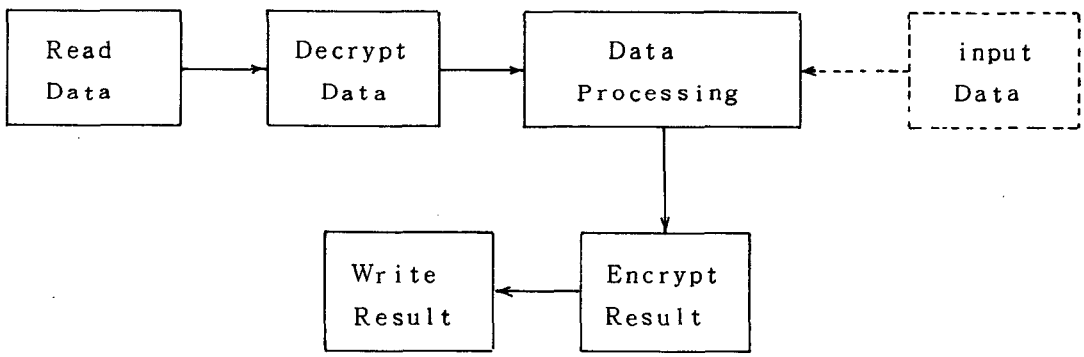
$$U = (u_1, u_2, \cdots, u_n)$$
$$V = (v_1, v_2, \cdots, v_n)$$

$$U + V = (u_1, \cdots, u_n) + (v_1, \cdots, v_n)$$
$$= ((u_1 + v_1) \bmod d_1, \cdots, (u_n + v_n) \bmod d_n)$$
$$U - V = (u_1, \cdots, u_n) - (v_1, \cdots, v_n)$$
$$= ((u_1 - v_1) \bmod d_1, \cdots, (u_n - v_n) \bmod d_n)$$
$$U * V = (u_1, \cdots, u_n) * (v_1, \cdots, v_n)$$
$$= ((u_1 * u_1) \bmod d_1, \cdots, (u_n * v_n) \bmod d_n)$$

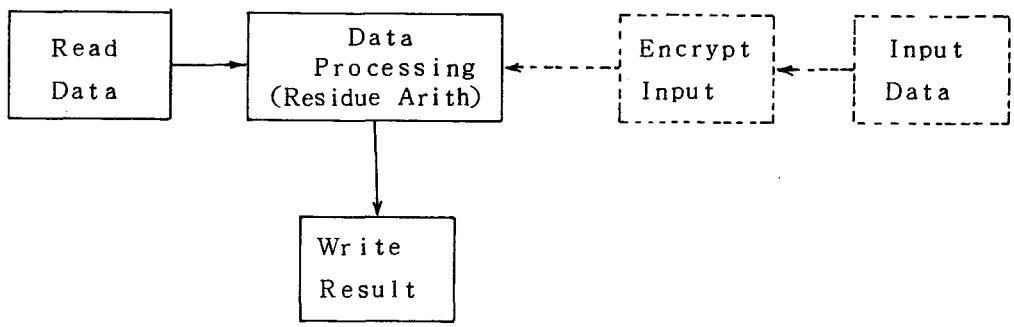
로 나타낼 수 있다.

그러나 레시듀 숫자의 연산방법을 DBMS에서 사용하기 위해서는 여러가지 제한적 요소들을 고려해야 할 것이다.

왜냐하면 레시듀 연산에서는 나눗셈이 어렵고 양수와 음수의 구분이 되지 않으며 두 숫자의 크기를 비교하기 어려운 점 등 여러가지 문제점들을 내포하고 있기 때문이다. 구체적인 연구는 아직 이루어지지 않았지만 그림 3에서 보여주는 것 처럼 RCC는 단순한 정수 연산은 해독을 하지 않고 레시듀 숫자의 연산방법을 이용할 수 있다고 본다. 특히 큰 정수 연산을 많이 필요로 하는 통계적 데이타베이스를 위해서는 더욱 효과적일 수 있다.



a) 전통적인 크립토시스템



b) RCC (정수 자료의 연산)

〈그림 3〉 DBMS에서 자료연산 과정

5. RCC와 DWKC의 비교 및 결론

지금까지 데이타베이스의 암호시스템으로제안되었던 DWKC를 소개하고 새로운 시스템 RCC를 제안하였다.

크립토시스템의 한 중요 쟁점은 자료의 보안성에 있지만 DWKC와 RCC의 보안성은 첨부되는 난수의 크기(길이)에 달려 있기 때문에 보안성을 임의대로 높일 수 있으며 만약 각 필드별로 32 비트의 난수를 첨가 시키면 DES과 거의 같은 수준의 보안성을 제공한다고 판단된다.

DWKC와 RCC가 모두 DBMS를 위한 크립토시스템으로 제안 되었기 때문에 여기에서는 두 시스템을 비교하여 장단점을 분석해 보려고 한다.

(1) 암호화 및 해독에 소요되는 시간

DWKC에서는 해독하는 것이 암호화 하는것보다 용이하고 RCC는 암호화 하는것이 해독하는것 보다 용이하다. 그러나 DWKC는 레코드 단위 암호방법을 사용하고 RCC는 필드 단위 암호방법을 사용하도록 되어 있기때문에 암호화하고 해독해야 하는 자료의 크기가 다르며 따라서 DWKC는 상대적으로 훨씬 많은 시간을 필요로 한다. 만약 한개의 레코드 길이가 4096 비트이고 각 레코드는 8개의 같은 크기의 필드들로 구성되어 있다면 현대의 계산기술상으로 볼때 DWKC는 RCC 보다 한개의 필드를 암호화 할때 약 180 배, 해독할때는 약 40 배나 더 많은 시간을 소요하게 된다.

(2) 자료전송시 오버헤드

자료전송시에도 DWKC는 전체 레코드를 보내주어야 하기 때문에 요구사항이 몇개의 필드일때는 그만큼 필요없는 자료를 보내야 하는 결과를 초래하여 통신비용을 높여주는 역할을 한다.

(3) 데이타베이스의 재 수정

한개의 필드만 인가된 사용자가 그 권한을 취소할때 RCC에서는 해당되는 필드만 새로운 키로서 수정해주면 되지만 DWKC는 전체의 레코드를 모두 다시 암호화 해주어야 한다.

(4) 에러 통제능력

RCC는 에러 수정이나 검출 능력을 갖고 있으며 특히 한개의 레시듀 에러 검출은 매우 쉽게 할 수 있지만 DWKC는 에러 통제능력이 전혀없기 때문에 만약에 자료전송시 에러 검출이 필요할때는 에러 검출 코드를 추가적으로 사용해야 하는 불편함이 있다.

(5) 치환 방법의 해결

두 시스템 모두가 적이 치환 방법으로 자료를 변경시키는 것을 완벽하게 막을 수가 없지만 이 사항에 있어서 만은 DWKC가 비교적 좋은 해결책이라고 볼 수 있다.

(6) 자료 연산의 효율성

아직 구체적인 응용방안을 연구 개발시키지는 못했지만 RCC에서는 간단한 정수의 연산은 암호문을 해독하지 않고 실행할 수 있는 방안을 제시 하였다. 만약 간단한 정수의 덧셈만이라도 해독하지 않고 가능하다면 데이타베이스의 정수 연산은 60% 이상이 덧셈이라는 점을 감안 할때 매우 효과적일 것이다.

결론적으로 본 연구에서는 DBMS를 위한크립토시스템으로 소개되었던 DWKC의 문제점들을 분석하고 그 문제점들을 해결할 수 있도록 필드 단위 암호방법을 사용하고 레시듀 코드를 이용한 RCC를 개발하여 DWKC와 비교해 보았으며 RCC가 DWKC 보다 훨씬 효율적이라는 결과를 얻을 수 있었다.

그러나 RCC를 실용화 하기 위해서는 실제적으로 다른 일반적인 크립토시스템 (예를 들면 DES 같은)을 DBMS에 응용했을 때와의 효율성을 비교검토하고 주위여건들을 고려하여앞으로 더욱 구체적인 연구가 이루어져야할 것이다

參 考 文 獻

1. Dorothy E. Denning, Cryptography and Data Security, Addison-Wesley, 1982.
2. National Bureau of Standards, "Data Encryption Standard", FIPS publication 46, Jan. 1977, pp. 1-18.
3. Whitfield Diffie and Martin E. Hellman, "Exhaustive Cryptanalysis of the NBS Data Encryption Standard", Computer, Vol. 10, No. 6, June 1977, pp. 74-84.
4. Donald E. Knuth, The Art of Computer Programming: Volume 2. Seminumerical Algorithms, Addison-Wesley Publishing Co., 1981.
5. Shu Lin and Daniel J. Costello, Jr., Error Control Coding: Fundamentals and Applications, Prentice-Hall, 1983.
6. David Mandelbaum, "Error Correction in Residue Arithmetic", IEEE Trans. Computers, Vol. c-21, Jun. 1972, pp. 538-545.
7. George I. Davida, David L. Wells and John B. Kam, "A Database Encryption System with Subkeys", ACM Trans. on Database System, Vol. 6, No. 2, June 1981, pp. 312-328.
8. Kil-Hyun Nam and T.R.N. Rao, "Cryptographic Models for DBMS Communication", Proc. Pacific Computer Communication Symposium, Seoul, Korea, Oct. 1985, pp. 255-261.
9. T.R.N. Rao, "An (n, k) Code for Residue Arithmetic", Proceedings of Second Annual Princeton Conference, 1968.
10. Charles C. Wood, "Future Application of Cryptography", Proc. of the 1981 Symposium on Security and Privacy, Apr. 1981, pp. 70-74.