通信情報保護學會誌
第1卷, 第3號, 1991. 12

데이타 베이스에서의 정보보호기법

Database Protection Techniques

나 민 영*

1. 서 론

최근 들어 중형 컴퓨터에서 뿐만 아니라 개인용 컴퓨터에서도 데이타 베이스 시스템의 사용이 계속 증가하고 있다. 데이타 베이스 시스템 (Data Base System : DBS)이란 데이타를 저장하고 관리해서 정보를 생성하는 컴퓨터 중심의 시스템을 말하는 것으로서 이는 데이타 베이스 (DB), 데이타 베이스 관리 시스템(DBMS) 그리고 데이타 언어, 데이타 베이스 관리자 등을 모두 포함하는 포괄적인 시스템이다[Elma 89]. 이중 데이타 베이스는 일반적으로 우리가 말하는 데이타 베이스의 집합으로서 통상 디스크에 저장되어 있으며, 여러 사용자에 의해 공유되고 있다. 다른 컴퓨터 구성품, 예를 들면 프로그램, 프로시쥬어, 터미날, 디스크 등과 마찬가지로 데이타 베이스도 고의적 또는 우연히 변화를 맞게 된다. 이러한 변화는 통상 권한을 가지고 있는 사용자에 의해 또는 불법적으로 보안 장치를 통과한 사용자에 의해 이루어진다. 불법적인 사용자는 비밀 정보를 훔치거나, 개인의 사생활 정보를 빼내거나, 또는 불법적인 목적에 이용하기 위하여 정보를 액세스 할 수 있다. 따라서 데이타 베이스의 보안이 점차 중요한 문제로 대두되어 여러 연구가 진행되고 있다. 즉 다중 레벨 릴레이션 기법을 이용하거나[Jajo 90], 시맨틱 데이타 모델을 사용하거나 [Smit 90], 또는 지문을 사용하는 기법 [Wagn 90] 등에 관한 연구가 그것이다. 그러나 이러한 기술적인 문제들도 근본적인 정보 보호 기법에 토대를 둔 개념에서 출발하므로 근본적인 기법을 먼저 살펴보는 것이 필요하다.

본고에서는 데이타 베이스에서의 정보를 보호하는데 사용될 수 있는 기법중 가장 기본이 되는 세 가지 기법을 살펴본다. 먼저 2장에서는 데이타 베이스 액세스 레벨에서의 데이타 베이스 보호 기법인 View와 Authorization에 대해 알아보고, 3장에서는 운영체제 레벨에서의 데이타 보호 기법인 액세스 제어를 살펴본 다음, 4장에서는 좀 더 적극적인 보호 방법인 암호화 기법을 다룬다. 마지막으로 5장에서는 결론을 맺는다.

2. View와 Authorization

2. 1 View

View란 중요하고 민감한 데이타를 그 데이타를

* 육군사관학교 조교수

액세스할 권한이 없는 사용자로부터 감추는 기법으로서 WHERE 절내에 적당한 조건을 표시하고 사용자가 볼 수 있도록 허용된 열들만 SELECT 절에 포함시킴으로서 얻어진다[Elma 89]. 예를 들어, EMP 릴레이션이 (EMP#, NAME, ADDRESS, DEPT#, SALARY) 등의 애트리뷰트들로 구성되어 있고 이중 SALARY>50K 정보는 모든 데이타 베이스 사용자에게는 노출되기를 원하지 않는다고 하자. 이런 경우에는 아래와 같이 View를 만들어 사용함으로서 허용되지 않은 사용자들로부터 보호할 수 있다.

```
DEFINE VIEW BASICINFO
  AS SELECT EMP#, NAME, ADDRESS,
            DEPT#, SALARY
     FROM   EMP
     WHERE  SALARY≤50000
```

2. 2 Authorization

이 기법은 일단 데이타 베이스 액세스 허락을 받은 사용자가 데이타 베이스를 액세스할 때 어카운트가 가질수 있는 권한, 또는 릴레이션에 대한 액세스 권한등을 제한하여 정보를 보호하는 방법으로서 권한의 부여와 취소에 기초를 두고 있다[Grif 76].

데이타 베이스 시스템에서 사용되는 권한 부여 방법에는 두가지 레벨이 있다. 첫째는 어카운트 레벨로서 이 레벨에서는 각 어카운트가 데이타 베이스내의 릴레이션에 무관하게 독자적으로 가질 수 있는 권한이 명시된다. 예를 들어 어떤 어카운트가 CREATE 권한을 가지고 있지 않다면 이 어카운트로부터는 어떤 릴레이션도 생성될 수가 없다. 두번째는 릴레이션 레벨로서 이 레벨에서는 각 명령이 적용될 수 있는 릴레레이션을 명시함으로서 각 릴레이션을 액세스하는 권한이 조정된다. 현재 대부분의 관계형 DBMS에서는 이 둘을 구분하지 않고 오직 릴레이션 레벨에서만 권한을 부여하고 있다.

릴레이션 권한의 부여와 취소를 제어하기 위해 데이타 베이스에 있는 각 릴레이션은 오너(owner) 어카운트를 지정하게 되는데, 오너 어카운트란 그 릴레이션이 첫번째 생성되었을때 사용된 어카운트를 말한다. 한 릴레이션의 오너는 그 릴레이션 상에서의 모든 권한을 가지고 있다. 이것은 한 어카운트로부터 생성된 모든 릴레이션은 그 어카운트에 의해 그 어카운트에 부여된 모든 액세스 권한을 사용해서 액세스 될 수 있음을 의미한다. 오너 어카운트를 갖고 있는 사용자는 이 권한들을 다른 사용자에게 GRANT 명령을 사용해서 전해줄 수 있다. 이러한 전달을 효과적으로 제한할 수 있는 기법들도 연구되었으나[Elma 89], 아직 대부분의 시스템에서는 구현되지 못하고 있다.

다음의 예는 데이타 베이스 질의어 중에 권한의 부여와 취소를 허용하는 부수적인 문장들을 포함시킴으로써 릴레이션에 대한 액세스 권한이 제어되는 것을 보여준다.

예 : 다음은 SQL에서 사용되는 authorization 명령들이다.

```
GRANT CREATETAB TO A1
```

CREATETAB 권한은 어카운트 A1에게 새로운 데이타 베이스 테이블(릴레이션)을 생성할 수 있는 권한을 준다. 즉 이것은 "어카운트 권한"이다. 이제 A1이 두 릴레이션 EMP와 DEPT를 생성했다 하자. 그러면 A1은 이 두 릴레이션의 오너가 되고 각각의 릴레이션에 대해 모든 "릴레이션 권한"을 갖는다.

다음 어카운트 A1이 어카운트 A2에게 이 두 릴레이션에 두플들을 삽입하고 삭제하는 권한을 주려고 한다 하자. 그러면 A1은 다음과 같은 명령을 수행한다.

```
GRANT INSERT, DELETE ON EMP, DEPT
TO A2 WITH GRANT OPTION
```

여기서 WITH GRANT OPTION은 A2가 GRANT를 사용해서 다른 어카운트에게 권한을 전파할 수 있음을 의미한다. 권한의 취소는 다음과 같은 명령으로 이루어진다.

```
REVOKE INSERT ON EMP FROM A2
```

그러면 DBMS는 A2로 부터 EMP 릴레이션에

대한 INSERT 권한을 취소하게 된다.

3. 액세스 제어

데이타 베이스에 대한 보호는 적절한 운영 체제와 밀접한 관계가 있다.[Fern 81]. 왜냐하면 통상 DBMS는 운영체제상에서 수행되기 때문이다. 가장 간단한 액세스 제어는 각 데이타 베이스 사용자가 데이타 베이스 시스템을 사용하고자 할때 허락을 받고 사용하도록 하는 것이다. 따라서 데이타 베이스 시스템을 맨 처음 액세스하려고 할때 제일 먼저 해야 하는 일은 어카운트와 패스워드를 신청해서 받는 일이다. 그러면 사용자는 필요할 때 어카운트와 패스워드를 사용해서 DBMS를 로그인 할 수 있다. 이때 DBMS는 반드시 어카운트와 패스워드가 맞는가를 검사해서 사용자가 데이타 베이스를 액세스할 수 있도록 허용되었나를 확인하게 된다.

그러나 일단 시스템에 들어와서는 보호되어져야 할 object를 액세스하려고 하는 subject로부터 보호해야 한다. 여기서 object란 액세스가 통제되어져야할 엔티티를 가리키는 것으로, 데이타 베이스 화일, 테이블(릴레이션), 레코드(투플), 또는 필드(애트리뷰트) 등이 object가 될 수 있다[Maek 87]. Subject란 object를 액세스하는 사람 또는 엔티티를 말한다. object의 보호는 액세스 행렬(Access Matrix) A에 의한 표현을 이용하여 object에 대한 액세스를 제한함으로써 가능하다. 그림 1은 액세스 행렬의 한 예를 보여준다. 이 그림에서 보는 바와 같이 행렬의 각 행은 subject를 나타내고 각 열은 object를 나타낸다. 원소 A[S, X]는 object X에 대한 subject S가 갖는 액세스 권한을 정의한다. 예를 들어, subject S_1은 object S_2에 대한 'wait'와 'signal' 권한을 가지고 있고, 또한 object S_4를 종료시키고, object X_1을 읽으며, object X_2를 읽고 수행할 수 있는 권한도 가지고 있다.

Subjects \ Objects	S_1	S_2	S_3	S_4	X_1	X_2
S_1		wait, signal		terminate	read	read, execute
S_2	wait, signal		wait, signal send, terminate		append	write
S_3		wait, signal receive		wait, signal terminate	execute	
S_4			wait, signal	control		

〈그림 1〉

이러한 액세스 행렬을 이용하여 각 object monitor는 다음과 같이 object에 대한 액세스를 보호한다.

1) subject S가 object X 상에서 operation α를 수행하려고 한다.

2) S, X, α의 3가지 정보가 종합되어 X의 object monitor에 전달된다.

3) object monitor는 A[S, X]에 α가 존재하는가 살핀다. 만일 존재하면 액세스가 허용되고 그렇지 않으면 위반 신호가 발생하게 된다.

4. 암호화

암호화 기법은 보조 메모리에 저장된 데이타를 보호하는데 쉽게 이용될 수 있는 기법이다. 데이타 베이스를 이루는 데이타들이 암호 형태로 저장되어

있어 그 암호를 풀 수 있는 키가 없는 사용자는 읽을 수 없기 때문이다. 따라서 보조 메모리를 훔치거나 불법으로 복사한다해도 데이타 베이스 내용을 알 수가 없다.

암호화 변환은 암호화할 때 데이타의 구조가 유지되는 기법과 데이타의 구조가 변하는 기법의 두 가지로 크게 나뉠 수 있는데 여기서는 암호화 변환의 주종을 이루는 전자에 대해서 다음의 세가지 기법만[Sebe 89] 살펴보기로 한다.

(1) 치환(Substitution)

이 방법은 데이타 원소를 변환시키는 것으로 변환되는 방식은 다음과 같은 식으로 표현된다.

$d^{(2)}=E_k(d^{(1)})$; $d^{(1)}\in F^{(1)}$, $d^{(2)}\in F^{(2)}$

여기서 $d^{(1)}$, $d^{(2)}$는 원소를, $F^{(1)}$, $F^{(2)}$는 애트리뷰트를, E_k는 암호변환을 의미하고, (1)은 원래의 형태, (2)는 암호화되어 변환된 형태를 의미한다.

예 : 다음 그림 2와 같은 논리 레코드 (릴레이션) LR이 있다고 하자.

$LR^{(1)}$	
EMPLOYEE NAME	SALARY
Green	3500
Brown	1500
Grey	2200

〈그림 2〉

이제 세 가지 암호 키 (K=0, 1, 2)를 위해 정의된 변환 E_k를 생각해 보자. 세가지 변환 E_0, E_1, E_2는 다음 그림 3과 같이 기술된다.

이때 만일 각 데이타 원소에 다음과 같은 방법으로 정수들이 주어진다면

Green ⟷ 0

Brown ⟷ 1

Grey ⟷ 2

이 예에서의 변환 함수는 다음과 같이 나타낼 수 있다.

$E_k(d)=(d+k) \bmod 3$

예를 들어 d=Brown이고 키 K=1 이면 $E_1(1)=2 \bmod 3=2$, 따라서 암호화된 데이타 원소는 Grey가 된다.

(2) 전치(Transposition)

전치는 데이타 원소의 형태를 바꾸는 것이 아니라 각 물리 레코드 안에서 단순히 데이타 원소의 위치를 바꾸는 기법이다. 이러한 종류의 암호화는 브라우징(browing)으로 부터 데이타 베이스를 보호할 수 있다.

n개의 물리 레코드(투플)로서 구성된 논리 레코드 $LR^{(1)}$을 생각해 보자. 그러면 $LR^{(1)}=\{PR_1, PR_2, \cdots, PR_n\}$이라 표현할 수 있고 각 물리 레코드 $PR_r^{(1)}$은 순서가 있는 연속 즉

$PR_r^{(1)}=\{d_{r,1}^{(1)}, dr,_2^{(1)}, \cdots, d_{r,k}^{(1)}\}$;

$r=1, 2, \cdots, n$

이다. 데이타 원소의 전치는 다음과 같은 방법으로 정의된 암호화 방법이다.

$d_{r,l}^{(2)}=d_{r,Ek^{(r)}(l)}^{(1)}$

여기서 $r=1, \cdots, n$이고 $E_k^{(r)}$은 정수 l ($l=0, \cdots, k$)을 위해 결정된 암호변환이다. 암호화 변환의 형태는 r에 의존하므로 각 물리 레코드는 각각 다른

E_0	
d	$E_0(d)$
Green	Green
Brown	Brown
Grey	Grey

E_1	
d	$E_1(d)$
Green	Brown
Brown	Grey
Grey	Green

E_2	
d	$E_2(d)$
Green	Grey
Brown	Green
Grey	Brown

〈그림 3〉

암호변환을 사용한다. 예를 들어 첫번 레코드는 $E_k^{(1)}$, 두번째는 $E_k^{(2)}$와 같다.

(예) 다음 그림 4와 같은 논리 레코드 $LR^{(1)}$를 생각해 보자.

$LR^{(1)}$

EMPLOYEE #	EMPLOYEE NAME	POSITION	SALARY	PROJECT
4129	Green	professor	3500	a12
3909	Brown	tutor	1500	a12
2457	Grey	lecturer	2200	tr1
1234	Smith	professor	3100	tr1
3467	Jones	associate professor	2700	x243

〈그림 4〉

암호 변환은 다음과 같은 형태라 가정한다.

$E_k(M)=(KM+3)$ Mod 5 ; K=1, 2, 3, 4

여기서 M은 원문을 나타낸다. 키 K=2가 첫번째 물리 레코드의 암호화에 사용되도록 결정되었다 하자. 먼저 $l=1, 2, 3, 4$, 에 대해 $E_k(l)$을 계산한다 (그림 5).

l	$E_k(l)$ for K=2
0	3
1	0
2	2
3	4
4	1

〈그림 5〉

물리 레코드 $PR_1^{(2)}=(d_{1,0}^{(2)}, d_{1,1}^{(2)}, d_{1,2}^{(2)}, d_{1,3}^{(2)}, d_{1,4}^{(2)})$이 되고, 이는

$d_{1,0}^{(2)}=d_{1,Ek(0)}^{(1)}=d_{1,3}^{(1)}$
$d_{1,1}^{(2)}=d_{1,Ek(1)}^{(1)}=d_{1,0}^{(1)}$
$d_{1,2}^{(2)}=d_{1,Ek(2)}^{(1)}=d_{1,2}^{(1)}$
$d_{1,3}^{(2)}=d_{1,Ek(3)}^{(1)}=d_{1,4}^{(1)}$
$d_{1,4}^{(2)}=d_{1,Ek(4)}^{(1)}=d_{1,1}^{(1)}$

이 된다.

다시 말해서 첫번 레코드 $PR_1^{(1)}=$(4129, Green, Professor, 3500, a12)는 조합된 레코드 $PR_1^{(2)}=(d_{1,3}^{(1)}, d_{1,0}^{(1)}, d_{1,2}^{(1)}, d_{1,4}^{(1)}, d_{1,1}^{(1)})$가 되어 (3500, 4129, professor, a12, Green)가 된다. 그림 6은 각 레코드에 대한 키가 2, 4, 2, 1, 3일때 얻어진 새로운 논리 레코드를 보여주고 있다.

3500	4129	professor	a12	Green
1500	tutor	Brown	3909	a12
2200	2457	lecturer	tr1	Grey
3100	tr1	1234	Smith	professor
2700	Jones	x243	associate professor	3467

〈그림 6〉

(3) 액세스 주소의 암호화(Encipherment of access address)

이 암호화 변환은 주로 논리 레코드에 관한 것이다. 만일 암호화 이전의 원시 논리 레코드가 다음과 같은 형태라면

$LR^{(1)}=I(F_1, F_2, \cdots, F_n)$

현재 우리가 고려중인 액세스 주소 암호화 기법을 적용한 후에는 다음과 같은 두개의 논리 레코드를 얻는다.

$LR_1^{(2)}=I_1(F_1, \cdots, F_j, E_k(AD))$, $LR_2^{(2)}=I_2(F_{j+1}, \cdots, F_n)$

여기서 AD는 논리 레코드의 두번째 부분의 주소를 나타내고, $E_k(AD)$는 그 주소의 암호화된 값을 나타낸다.

예 : 다음 그림 7과 같은 논리 레코드를 생각해 보자.

EMPLOYEE #	EMPLOYEE NAME
4129	Green
3909	Brown
2457	Grey

〈그림 7〉

액세스 주소에 암호화를 적용하는 논리 레코드는 다음 그림 8과 같은 형태를 갖는다.

EMPLOYEE #	$E_k(AD)$	EMPLOYEE NAME
4129	$E_k(a_1)$	Green
3909	$E_k(a_2)$	Brown
2457	$E_k(a_3)$	Grey

〈그림 8〉

여기서 애트리뷰트(또는 항목) $AD=\{a_1, a_2, a_3\}$는 각 물리 레코드의 두번째 부분의 주소를 가리키고 있다.

(4) 암호화 변환 기법의 적용

이제 이러한 암호화 변환 기법들이 어떻게 데이타 베이스 보호에 적용되는가를 살펴보자. 데이타 베이스 시스템은 데이타 독립성을 제공하기 위하여 보통 외부 스키마, 논리 스키마, 그리고 물리 스키마 등의 다중 레벨 스키마로 표현되는데, 한 스키마로부터 다른 스키마로 변환시 암호화를 사용함으로서 불법적인 액세스로부터 정보를 보호할 수 있다. 먼저 사용자 레벨로부터 논리 레벨로 변환시에는 치환기법이 사용되어질 수 있으며, 논리 레벨로부터 물리 레벨로 변환시에는 치환, 전치, 액세스 주소의 암호화 등의 기법들이 사용되어질 수 있다.

5. 결 론

본고에서는 데이타 베이스 보호 기법중 기본이 되는 몇가지 기법들을 살펴보았다. 먼저 데이타 베이스 액세스 레벨에서의 보호 기법인 View와 Authorization에 대해 알아 보았다. 그러나 이러한 기법들보다 더 적극적인 보호 방법은 데이타 자체를 암호화하는 것이다. 따라서 우리는 데이타 보호에 스이는 암호화 기법과 이 기법이 어떻게 데이타 베이스에 적용될 수 있는가를 살펴 보았다. 이러한 기법들을 토대로 앞으로의 데이타 베이스 보호에 관한 연구는 다음과 같은 방향, 즉 (1) 기존의 데이타 베이스를 보안 개념이 있도록 확장하거나, (2) 데이타 보호에 적합한 새로운 모델을 개발하는 방향으로 이루어질 것이다.

이러한 데이타 베이스 보호는 중앙 집중형 데이타 베이스 시스템에서의 문제만이 아니다. 분산 데이타 베이스 시스템에 있어서는 더욱 중요한 이슈가 되고 있다. 왜냐하면 분산 데이타 베이스 시스템에 있어서는 네트워크, 질의어 처리, 데이타 할당 등에 관한 사항이 추가적으로 고려되어야 하기 때문이다 [Thur 90]. 따라서 현재 분산 데이타 베이스의 연구가 점차 증가되고 있으므로 데이타 베이스의 보호 문제는 앞으로 더욱 중요시되고 또한 많이

연구되어질 것이라 생각된다.

참 고 문 헌

[Elma 89] Elmasri, R., and Navathe, S. B., Fundamentals of Database Systems, Benjamin/Cummings, pp. 587-604, 1989.

[Fern 81] Fernandez, E. B., Summers, R. C., and Wood, C., Database Security and Integrity, Addison Wesley, pp. 217-221, 1981.

[Grif 76] Griffiths, P., and Wade, B., "An Authorization Mechanism for a Relational Database System," ACM Trans. on Database Systems, Vol. 1, No. 3, pp. 242-255, 1976.

[Jajo 90] Jajodia, S., Sandhu, R., and Sibley, E., "Update Semantics for Multilevel Relations," The Sixth Computer Security Applications Conference, pp. 103-112, December 1990.

[Maek 87] Maekawa, M., Oldehoeft, A. E., and Oldehoeft, R. R., Operating Systems : Advanced Concepts, Benjamin/cummings, pp. 301-308, 1987.

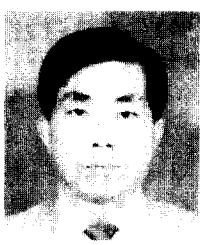
[Sebe 89] Seberry, J., and Pieprzyk, J., Ctyptography : An Introduction to Computer Security, Prentice Hall, pp. 233-245, 1989.

[Smit 90] Smith G. W., "The Semantic Data Model for Security : Representing the Security Semantics of an Application," International Conference on Data Engineering, pp. 322-329, February 1990.

[Thur 90] Thuraisingham, B., and Kamon, A., "Secure Query Processing in Distributed Database Management Systems," The Sixth Computer Security Applications Conference, pp. 88-101, December 1990.

[Wagn 90] Wagner, N. R., Fountain, R. L., and Hazy R. J., "The Fingerprinted Database," International Conference on Data Engineering, pp. 330-336, February 1990.

□ 著者紹介



나 민 영 (正會員)

1978. 3 육군사관학교 졸업
1983. 2 서울대학교 컴퓨터공학과 졸업
1986. 2 서울대학교 대학원 컴퓨터공학과(공학석사)
1990. 12 University of Florida, Dept. of Computer and Information Sciences(Ph. D.)
1991－현재 육군사관학교 수학과 조교수

관심분야 : 데이타 베이스 디자인, 분산 데이타 베이스, 데이타 베이스 보안, 연합 데이타 베이스, 데이타 모델링.