객체의 안전한 보안등급의 하강을 위한 접근통제 메커니즘의 설계 및 구현

박춘구*, 신욱*, 강정민*, 이동익*

*광주과학기술원, 정보통신공학과

A Design and Implementation of Access Control Mechanism for Secure Downgrading of Objects

Chun-Goo Park*, Shin Wook*, Jung-Min Kang*, Dong-Ik Lee*

*Department of Information and Communications,

Kwang-Ju Institute of Science and Technology

요 약

다중등급보안(MLS: Multi-Level Security)기반 안전한 운영체제는 정보의 흐름을 안전하게 통제하기 위하여 주체 및 객체의 보안등급변화를 허용하지 않는다. 하지만 안전한 운영체제의 사용성(Usability) 측면에서 주체 및 객체의 보안등급변화는 고려되어야 한다. 주체 및 객체의 보안등급변화에 관련된 요구사항은 시스템의 환경 및 보안정책에 따라 다양하게 발생할 수 있고, 이러한 보안등급변화에 관련된 다양한 요구사항들은 모두 해결하기 어렵다. 뿐만 아니라, 기존 접근통제 메커니즘은 보안등급변화에 관련된 요구사항을 해결할 수 없다. 따라서, 본 논문에서는 MLS 기반 안전한 운영체제에서 빈번하게 발생할 수 있는 보안등급 변화에 관련된 요구사항 중 특히, 시스템의 환경에 의해 주체의 보안등급이 하강되었을 때 해당 주체가 생성했던 객체들의 안전한 보안등급 하강과 관련된 보안요구사항을 해결할 수 있는 접근통제 메커니즘을 설계하고 구현한다.

I. 서론

대부분의 보안모델은 접근통제 시스템 내부의 주체 및 객체의 보안등급 변화를 허용하지 않는 평정 원리(Tranquility Principle)를 명시적으로 또는 묵시적으로 포함하고 있다[1,2,3,4]. 하지만 접근통제 시스템은 사용성 측면에서 주체 및 객체의 보안등급 변화를 필요로 한다.[5,6,7,8,10]. 특히 가장 널리 사용되는 접근통제 모델 중에 하나인 MLS 정책기반 BLP(Bell LaPadula)모델에서 주체 및 객체의 보안등급 변화를 허용하기 위한 시도가 이루어졌다[9,10]. 하지만 [9,10]에서 제안한 방법은 신뢰주체(Trusted Subject)가 보안등급변화가 예상되는 주체 및 객체를 미리 결정해야 하기 때문에 보안등급변화에 관련된 요구사항들을 효과적으로 해결할 수 없다.

주체 및 객체의 보안등급변화에 관련된 요구사항은 시스템의 환경 및 보안정책에 따라 다양하게 변화할 수 있고, 해당 요구사항을 해결하기 위한 방법 또한 달라지게 된다. 따라서 본 논문에서는 MLS기반 BLP모델을 적용한 시스템에서 자주 발생할 수 있는 보안등급의 변화에 관련된 요구사항들 중 특히, 시스템의 환경에 의해 주체의 보안등급이 하강되었을 때 해당 주체가 생성했던 객체들의 안전한 보안등급 하강을 고려하고[17], 그 요구사항을 해결하기 위한 향상된 접근통제메커니즘을 제안, 설계, 구현한다.

본 논문의 구성은 다음과 같다. 2장에서는 객체의 안전한 보안등급 하강을 위해 필요한 정보들에 대하여 기술하고, 3장에서는 2장에서 언급한 정보들을 포함하는 향상된 접근통제 메커니즘에 대하

여 기술한다. 4장에서는 새롭게 제안한 접근통제 메커니즘의 설계, 구현에 관련된 내용을 기술한다. 끝으로, 5장에서는 결론에 대하여 기술한다

II. 객체의 안전한 보안등급 하강을 위한 정보

접근통제 메커니즘은 보안모델의 보안규칙과 보안요구사항의 구현기능을 갖는다. BLP모델은 AM(Access Matrix)과 SL(Security Label)의 접근통제 메커니즘을 이용하여 주체와 객체의 접근통제를 수행한다[11,12]. 하지만 주체와 객체의 접근정보와 보안등급정보만을 포함한 기존 AM, SL 접근통제 메커니즘은 객체의 안전한 보안등급의 하강을 수행 할 수 없다[16]. 따라서, 이 장에서는 객체의 안전한 보안등급 하강을 위해 2ISL (Internal Information Security Level), DR (Downward Reference), Private and Public이라는 추가정보를 제안한다.

1. 2ISL

보안등급은 시스템내부의 주체와 객체의 보안중요도의 계층적인 관계를 표현한다. 대부분의 시스템은 정보의 흐름측면에서 객체의 보안등급 하강을 허용하지 않고, 단지 객체의 보안등급 상승만을 허용한다. 하지만 이러한 원칙에는 객체의 과등급화(Over-Classify)에 의해 예외상황이 발생할 수 있다[18,19]. 따라서, 과등급화된 객체는 보안관리자에 의해 안전한 보안등급 하강을 수행 할 수 있다. 효과적인 보안등급 하강을 위해서 과등급화된 객체는 객체내부의 등급정보를 나타내는 추가정보가 필요하다. 따라서, 본 논문에서는 객체내부정보의 중요도를 나타내기 위하여 내부정보 보안등급 (Internal Information Security Level : 2ISL)이라는 새로운 보안등급을 제안한다. 2ISL의 특징, 설정방법 및 규칙은 다음과 같다.

- 2ISL의 특징
  - 2ISL은 객체내부정보의 중요도를 나타내는 객체와 관련된 보안등급
  - 2ISL은 MLS의 객체와 같은 보안등급체계를 가지고 있다.
  - 시스템내의 객체는 기존의 보안등급과 2ISL 값을 모두 가지고 있다.
  - 2ISL은 접근통제를 위하여 사용되지 않고, 보안관리자에 의하여 객체의 보안등급이 하강될 때 사용된다.
- 2ISL의 설정방법[17,20]

먼저 2ISL 설정규칙을 정형화된 기법으로 표기하기 위한 구성요소를 살펴보자

- S : 주체(사용자)의 집합
- O : 객체의 집합
- L : 순서화 된 보안등급의 집합
- $XLev(o)$: 객체 o의 2ISL. $o \in O$
- $Lev(s)$: 주체 s의 보안등급. $s \in S$
- $Lev(o)$: 객체 o의 보안등급. $o \in O$

2ISL 설정규칙에 사용될 함수들은 다음과 같다[17,20].

- ***Set_2ISL*** : O × L → boolean
- ***Owner*** : S × O → boolean
- ***Dominates*** : L × L → boolean
- ***CreateFromRead*** : O × O → boolean
- ***AppendToObject*** : O × O → boolean
- ***WriteToObject*** : S × O → boolean

위에서 언급한 함수에 의한 2ISL설정 규칙은 다음과 같다.[17,20]

- $\forall o,o' \in O$, $XLev(o') \in L$: *Set_2ISL(o, XLev(o')) iff CreateFromRead(o,o')*
- $\forall s \in S$, $\forall o,o' \in O$, $Lev(s) \in L$: *Set_2ISL(o, Lev(s)) iff CreateFromRead(o,o')==false and Owner(s,o)*
- $\forall s \in S$, $\forall o,o' \in O$, $Lev(s) \in L$: *Set_2ISL(o, Lev(s)) iff AppendToObject(o,o') and Owner(s,o)*
- $\forall s \in S$, $\forall o \in O$, $XLev(o) \in L$ $Lev(s) \in L$: *Set_2ISL(o, Lev(s)) iff WriteToObject(s,o) and Dominates(Level(s), XLev(o))*

2. 하위참조

참조(Reference)는 UNIX의 심볼릭 링크 유틸리티처럼 주체가 허용된 기준 내에서 객체의 접근을 보다 편리하게 하기 위한 방법 중에 하나이다. MLS 보안정책에 의해, 하위등급의 객체는 상위등급의 객체를 접근할 수 없기 때문에 원칙적으로 상위참조를 허용하지 않는다. 그러나, 하위참조는 MLS 보안정책에 위배되지 않고 허용된 기준 내에서 주체가 객체의 접근을 편하게 하기 위하여 허용될 수 있다. 그러나 시스템에 의해 허용된 하위참조는 객체의 보안등급의 안전한 하강을 수행하기 위해서는 관리가 필요하다. 예를 들어, 자신의 보안등급보다 낮은 등급의 객체o'을 참조하는 객체 o가 등급변화의 허용에 의해 객체 o'보다 낮은 등급으로 낮아질 수 있다. 이런 경우 정보의 흐름측면에서 비밀성에 취약점이 발생한다. 왜냐

하면, 하위참조였던 객체*o*가 상위참조가 되어 버리기 때문이다.

객체의 하위참조는 다음과 같이 기술할 수 있다[17,20].

- ***Down*** : O × O → boolean
- ***Reference*** : O × L → boolean
- ∀o,o′∈O, Lev(o′)∈L : *Reference(o, Lev(o′)) iff Down(o,o′)*

3. Private and Public

개인메일 또는 일기파일과 같이 사용자의 개인정보를 포함하는 SECRET등급의 객체 *o*가 존재한다고 하자. 만약 보안관리자에 의해 객체*o*를 생성한 생성자의 보안등급이 낮아지면 객체*o*의 보안등급 또한 함께 낮아져야 한다. 만약 비밀성 측면만을 고려하여 객체*o*의 보안등급을 하강하지 않으면, BLP의 보안특성에 의해 보안등급이 변화된 주체는 자신의 개인메일 또는 일기에 해당하는 객체*o*을 접근 할 수 없게 된다. 따라서, 위와 같은 상황은 사용성 측면을 고려할 때 해결하여 할 문제이다.

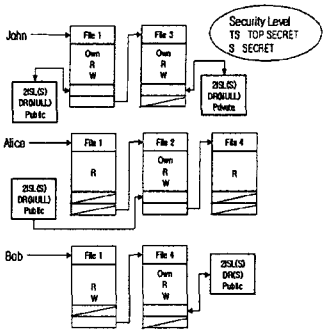
위에서 언급한 2ISL과 참조 정보만을 이용해서는 위에서 언급한 객체의 안전한 보안등급 하강을 수행할 수 없다. 하지만 객체를 접근하는 주체들의 행위를 파악할 수 있다면, 해당 객체가 개인적인 자료를 포함하는지 아닌지를 판단할 수 있다. 주체들의 행위에 의해 객체의 자료성격을 파악하기 위한 간단한 규칙은 다음과 같다[17,20].

- ***A*** : 접근모드의 집합.
- ***Allow*** : S × A × O → boolean
- ***Public*** : O → boolean
- ∀o∈O, s∈S, *α*∈A : *Public(o) iff Allow(s, α,o) and Owner(s,o)==false*

III. 확장된 접근통제 메커니즘

주체와 객체의 접근정보와 보안등급정보를 기반으로 하는 기존의 접근통제 메커니즘은 객체의 안전한 보안등급 하강을 수행할 수 없다.

따라서, 이 장에서는 객체의 안전한 보안등급 하강을 수행하기 위해 기존의 접근통제 메커니즘에 2ISL, DR, Private and Public정보를 포함한 확장된 접근통제 메커니즘인 ECL (Extended Capability List)을 제안한다. ECL는 [그림1]과 같다. 참고로 주체의 보안등급이 하강 될 때 해당 주체가 생성했던 객체들의 보안등급을 고려하기 때문에 객체의 권한영역에 Own값이 설정되어 있는 객체에 대해서만 해당정보를 추가한다. 추가된 정보는 다음과 같이 설정된다.



[그림 1] ECL(Extended CL)

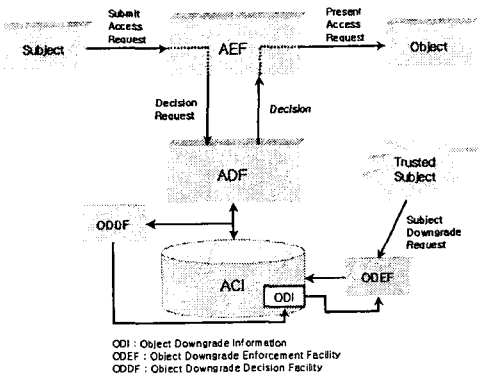
- ***2ISL(l)*** : *l* 등급의 2ISL을 갖는다.
- ***DR(l)*** : *l* 등급의 객체를 참조한다.
- ***Private*** : 개인정보를 포함하는 객체를 나타낸다.
- ***Public*** : 개인정보가 아닌 정보를 포함하는 객체를 나타낸다.

- 추가정보 설정규칙[17,20]
  - *∀o∈O*
    - *: 2ISL(XLev(o)) iff Set_2ISL (o,XLev(o))*
    - *: DR(Lev(o)) iff Reference(o,Lev(o))*
    - *: DR(NULL) iff Reference(o,Lev(o))== false*
    - *: Private iff Public(o)==false*
    - *: Public iff Public(o)*

지금까지 객체의 보안등급하강을 위해 추가된 정보들을 설정하는 방법에 대하여 논하였다. 이제 추가된 정보들이 포함된 다음의 알고리즘을 이용하여 객체의 보안등급 하강을 수행한다.

```
if ( Private ) then
    Downgrade Object
end if
if ( Level(DR(l)) != NULL ) then
    if ( Dominates(Level(After(s)), Level(DR(l))) ) then
        Downgrade object
    end if
end if
if ( Public ) then
    if ( Dominates(Level(After(s)), XLev(o)) ) then
        Downgrade object
    end if
end if
```

객체의 안전한 하강이 수행된 후, 보안관리자는 보안등급이 낮아진 객체들 중 특히 Private정보에 의해 보안등급이 낮아진 객체를 중점 관리하여야 한다. 만약 주체가 Private객체에 상위등급의 정보를 포함시킨 후 특정 주체들이 접근할 수 있게 한다면, Private객체에 의해 상위등급의 정보가 하위등급으로 유출 되게 된다. 따라서, 보안등급이 낮



[그림 2] 접근 제어 개념도

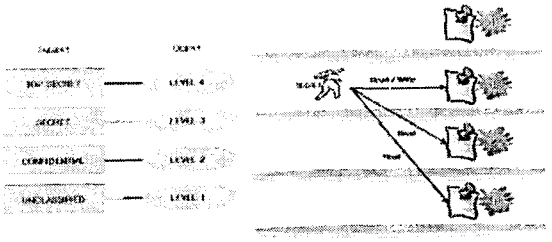
아진 Private객체는 보안등급의 변화이후 반드시 해당객체의 생성자만이 접근할 수 있게 관리되어야 한다.

IV. 설계 및 구현

기존 접근통제 시스템은 객체의 안전한 보안등급 하강을 위한 접근통제 메커니즘을 포함하고 있지 않다. 따라서, 기존 접근통제 시스템의 보안모듈에 객체의 안전한 하강을 수행하기 위한 보안모듈을 추가하여 보안 등급변화 요구사항을 해결하고자 한다. [그림2]는 객체의 안전한 보안등급 하강을 위해 추가된 구성요소와 기존 접근 제어 시스템의 구성요소와의 관계를 나타내고 있다. 객체의 안전한 보안등급 하강을 위해 추가된 구성요소는 다음과 같다.

- ODDF(Object Downgrade Decision Facility) : 객체 하강 결정 기능
- ODEF(Object Downgrade Enforcement Facility) : 객체 하강 실행 기능
- ODI(Object Downgrade Information) : 객체 하강 정보

ODDF는 ADF가 접근통제를 수행하기 위하여 사용했던 ACI와 접근통제수행 결과값을 이용하여 ODI를 설정한다. ODEF는 ODDF에 의하여 설정된 ODI를 3절의 알고리즘에 적용하여 객체의 안전한 보안등급 하강을 수행하도록 한다. 여기서 ODI는 2절에서 제시한 2ISL, DR, Private and Public 의 3가지 추가정보를 말한다. 이 장에서는 ODDF, ODEF, ODI의 추가모듈을 적용한 CSRL 시스템에 대하여 기술한다.



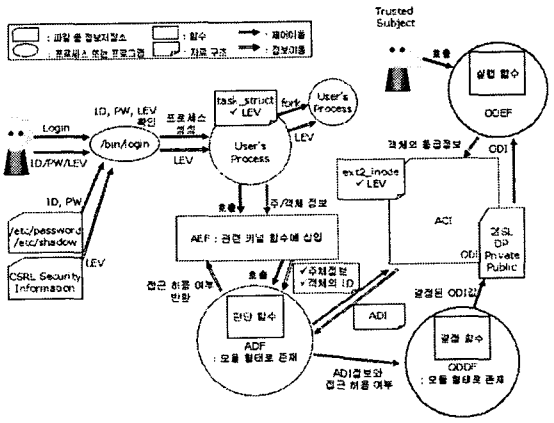
[그림 3] 주체와 객체의 보안등급

1. BLP모델 기반 CSRL 시스템

LINUX 커널 기반 CSRL(CSRL is SecuRe Linux) 시스템은 주체에 해당하는 사용자의 보안등급을 TOP SECRET에서 UNCLASSIFIED까지 4개의 등급으로 나누었고, 객체에 해당하는 파일의 보안등급을 4등급에서 1등급까지 4개의 등급으로 나누었다. 또한 향상된 BLP모델에 의해 사용자는 자신의 보안등급보다 낮거나 같은 파일은 읽을 수 있고, 자신의 보안등급과 동일한 등급의 파일만 쓸 수 있는 권한이 있다. 당연히 자신의 보안등급보다 높은 파일은 읽기, 쓰기 권한이 없다 [그림 3]. 현재 개발중인 CSRL 시스템은 주체와 객체의 부서(Category)정보에 관한 내용은 배제했고, 객체는 LINUX 시스템의 기본 파일 시스템인 ext2 파일 시스템에 한해 국한했다. CSRL의 전체적인 구조에 객체의 보안등급의 안전한 하강을 위한 보안모듈을 추가한 전체적인 구조와 기능은 [그림 4]와 같다.

2. 추가 접근통제 모듈

ODDF는 ADF에 의해 접근통제가 수행되는 동



[그림 4] CSRL 접근통제 시스템

안 2ISL, DR, Private/Public의 ODI를 설정하기 위한 정보를 CSRL 커널로부터 얻는다.

ODDF는 CSRL 커널의 4가지 시스템 콜 sys_open(fs/open.c),sys_read(fs/read_write.c), sys_write(fs/read_write.c),sys_symlink(fs/namei.c)로부터 ODI의 값을 설정하기 위한 정보를 얻는다. ODDF는 해당 정보들에 의해 2절에서 언급한 함수들을 수행하고, 그 함수들에 의해 2ISL, DR, Private/Public 정보를 설정한다. 시스템의 내부환경에 의해 특정사용자의 보안등급 하강이 필요하게 되면, 보안관리자는 해당 사용자가 생성했던 객체들의 보안등급을 처리하기 위하여 ODEF를 호출한다. ODEF는 ODDF에 의해 설정된 ODI를 3절의 알고리즘을 이용하여 보안등급의 변화 유무를 결정한다

3. 테스트

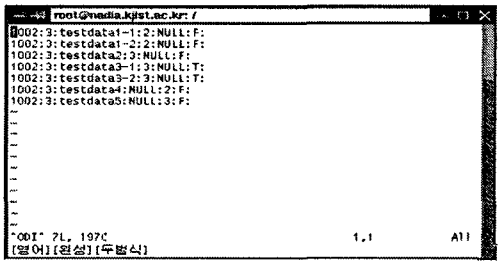
[그림5]는 SECRET등급의 사용자 madreach가 로그인하여 수행한 동작을 보여주고 있다. SECRET등급의 madreach는 3등급으로 로그인하여 1등급의 Level_1_Data, 2등급의 Level_2_Data, 3등급의 Level_3_Data 파일을 이용하여 파일(TestData1-1, TestData1-2. TestData2, TestData3-1, TestData3-2, TestData4, TestData5) 을 생성한다.

ODDF는 커널내부의 시스템 콜 정보를 이용하여 2ISL, DR, Private/Public 정보의 값을 설정한다.

[그림6]은 madreach가 생성한 파일들의 2ISL, DR, Private/Public 정보의 값을 보여준다. [그림6]의 자료구조는 다음과 같다.

UID: User's Level: FileName: 2ISL: DR: Private/Public

첫 번째 필드는 madreach의 UID를 나타내고,



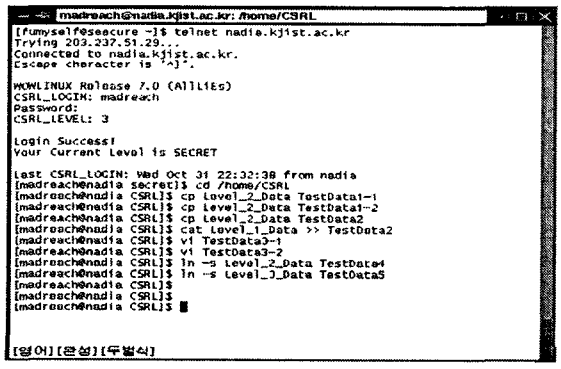
[그림 6] madreach가 생성한 파일들의 ODI 값

두 번째 필드는 madreach의 보안등급, 세 번째 필드는 madreach가 생성한 파일의 이름을 나타낸다. 네 번째 필드는 해당파일의 2ISL값을 나타낸다. 다섯 번째 필드는 하위참조의 유무를 나타낸다. NULL이면 하위참조가 존재하지 않는 것을 나타내고, 숫자 값이면 그 숫자에 해당하는 등급의 객체를 참조함을 나타낸다. 마지막 필드는 해당파일이 Private 인지 Public 인지를 나타낸다. T값이면 Private를 나타내고, F값이면 Public을 나타낸다.

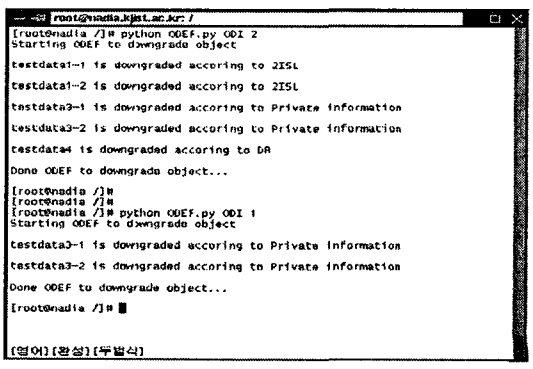
시스템에 의해 madreach의 보안등급이 2등급으로 하강되면, 보안관리자는 madreach가 생성했던 객체들의 보안등급을 처리하기 위하여 ODEF를 호출한다. [그림 7]은 ODEF가 [그림 6]의 ODI정보를 이용하여 2등급으로 보안등급이 낮아질 수 있는 파일의 리스트와 판단근거를 보여준다.

V. 결론 및 향후연구

주체 및 객체의 보안등급변화에 관련된 요구사항은 시스템의 환경에 따라 변화할 수 있고, 해당 요구사항을 해결하기 위한 방법 또한 달라진다. 따라서 본 논문에서는 가장 널리 사용되는 접근통제 모델 중 하나인 BLP모델기반 시스템 에서 자주 발생할 수 있는 보안등급의 변화에 관련된 요



[그림 5] madreach의 로그인과 파일접근



[그림 7] 파일의 보안등급 하강

구사항들 중 특히, 시스템의 환경에 의해 주체의 보안등급이 하강되었을 때 해당 주체가 생성했던 객체들의 보안등급 하강을 고려하고, 그 요구사항을 해결하기 위한 향상된 접근통제메커니즘인 ECL을 제안했다.

ECL에는 기존의 CL에 ODI라는 객체에 관련된 정보를 추가하고 있다. ODI는 객체 내부정보의 중요도를 나타내는 보안등급인 2ISL, 객체참조에 관련된 정보인 DR, 객체가 개인정보를 포함하는지 포함하지 않는지를 포함하고 있는지를 나타내는 Private/Public 정보로 구성된다. 본 논문에서는 ODI를 이용하여 객체의 보안등급을 하강하기 위해 ODDF와 ODEF의 모듈을 추가했다. ODDF는 ODI의 정보를 설정하고 ODEF는 ODI를 이용하여 객체의 안전한 보안등급의 하강을 수행한다.

향후, 사용자의 다양한 형태의 파일접근 및 생성과정으로부터 정확한 ODI를 설정하기 위한 방법이 보다 심도 깊게 연구되어야 한다.

참고문헌

[1] J. McLean, "Reasoning about security models," In IEEE Symposium on Security and Privacy, Oakland, 1987.

[2] J. Haigh and W. Young, "Extending the Noninterference Version of MLS for SAT," In EEE Symposium on Security and Privacy, Oakland, 1986

[3] D. Sutherland, "A Model of Information," In Proceedings of the 9th National Computer Security Conference, 1986.

[4] D. McCullough, "Noninterference and the Composability of Security Properties," In IEEE Symposium on Security and Privacy, Oakland, 1988.

[5] P, Bieber and F. Cuppens, "Secure Dependencies with Dynamic Level Assignments," Computer Security Foundations Workshop V, Pages(s): 63-75, 1992.

[6] S.N. Foley and Li Gong and Xiaolei Qian, "A security model of dynamic labelling providing a tiered approach to verification," IEEE Symposium on Security and Privacy, Page(s): 142 -153, 1996.

[7] J, McLean, "The algebra of security," IEEE Symposium on Security and Privacy, Page(s): 2 -7, 1988.

[8] Sutherland, I and Perlo, S and Varadarajau, R, "Deducibility Security with Dynamic Level Assignments," Computer Security Foundations Workshop II, Page(s): 3 -8, 1989.

[9] J, McLean, "The algebra of security," IEEE Symposium on Security and Privacy, Page(s): 2 -7, 1988.

[10] Li Gong and Xiaolei Qian, "Enriching the expressive power of security labels," IEEE Transactions on Knowledge and Data Engineering, Volume: 7 Issue: 5, Page(s): 839 -841, Oct, 1995.

[11] Dieter Gollmann, "Computer Security," JOHN WILEY & SONS, Aug, 1999.

[12] Edward G. Amoroso, "Fundamentals of Computer Security Technology," PRENTICE HALL PTR, 1994.

[13] Sandhu, R.S and Samarati, P. "Access Control : principle and practice," IEEE Communications Magazine, Volume: 32 Issue: 9, Page(s): 40-48, Sept, 1994.

[14] D.E. Denning, "Cryptography and Data Security, Addison-wesley," Reading (Mass), 1982.

[15] I Lung Kao, Chow, R. "An Extended Capability Architecture to Enforce Dynamic Access Control Policies," Computer Security Applications Conference, 12th Annual, Page(s): 148-157, 1996.

[16] ITU-T SG/7 & Working parties, Final text for recommendation X.812 Information Technology - Open systems interconnection Security framework for open systems : Access control framework. Geneva.26 June-7 July 1995.

[17] Chun-Goo Park, "A Security Mechanism for Secure Downgrading of Object," CSRL-Technical-Report-2001-8-fumyself, Aug, 2001.

[18] Robinson, C.L., Wiseman, S. R. "Using security models to investigate CMW design and implementation," Computer Security Applications Conference, Page(s) 278-287, 1994

[19] Berger, J.L., Piciotto, J., "compartmented mode workstation : prototype highlights, Software Engineering," IEEE Transactions on, Page(s) 608-618, June, 1990.

[20] 박춘구 외 3인, " MLS에서 객체의 안전한 보안등급의 하강을 위한 접근통제메커니즘에 관한 연구", 2001년 한국정보과학회 추계학술발표논문집, 2001.