MLS에서 객체의 안전한 보안등급의 하강을 위한 접근통제메커니즘에 관한 연구

박춘구[0], 신욱, 강정민, 이동익
광주과학기술원 정보통신공학과
(cgpark, sunihill, jmkang, dilee)@kjist.ac.kr

A Study on Access Control Mechanism for Secure Downgrading of Objects in the MLS System

Chun-Goo Park[0], Shin Wook, Jung-Min Kang, and Dong-Ik Lee
Dept. of Information and Communications,
Kwang-Ju Institute of Science and Technology

요 약

MLS 보안 시스템은 가용성측면에서 주체 및 객체의 보안등급의 변화를 고려하여야 한다. 하지만 보안등급의 변화에 관련된 모든 요구사항은 해결하기 어렵고, 보안규칙과 보안요구사항의 구현기능을 갖는 기존의 접근통제메커니즘 또한 이러한 보안등급의 변화에 관련된 요구사항을 해결할 수 없다. 따라서, 본 논문에서는 MLS 정책기반 안전한 운영체제에서 발생할 수 있는 보안등급의 변화에 관련된 요구사항 중 특히 시스템의 환경에 의해 주체의 보안등급이 하강 되었을 때 해당 주체가 생성했던 객체들 중 비밀성이 저해되지 않는 범위 내에서 객체의 보안등급의 안전한 하강과 관련된 보안요구사항을 해결할 수 있는 접근통제 메커니즘을 제안한다.

1. 서론

대부분의 보안모델에서는 시스템 내부의 주체 및 객체의 보안등급의 변화를 허용하지 않는 Tranquility Principle을 명시적 또는 묵시적으로 포함하고 있다[1,2,3]. 하지만 시스템의 가용성측면에서 비추어 볼 때, 실세계의 시스템에서는 주체 및 객체의 보안등급의 변화가 필요로 한다[4,5,7]. 특히 가장 널리 사용되는 접근통제 정책 중에 하나인 MLS 정책기반 BLP모델에서의 주체 및 객체의 보안등급의 변화를 위한 연구가 수행 되고 있다[6,7]. 하지만, 보안모델이 실세계의 시스템으로 구현될 때 시스템의 환경 및 보안등급변화에 관련된 요구사항에 따라 해당 정보를 결정하는 방법과 해당정보의 값이 달라진다. 다시 말해, 위에서 언급한 [6,7]의 보안모델을 시스템에 적용할 때, 시스템의 환경 및 보안등급 변화에 관련된 요구사항에 따라 Expressive Security Labels 및 보안등급의 변화를 허용 할 주체 및 객체의 집합을 결정하는 방법과 결과값이 달라진다. 따라서, 주체 및 객체의 보안등급 변화에 관련된 모든 요구사항을 위와 같은 방법으로 해결하기는 어렵다. 따라서, 본 논문에서는 [8]의 보안등급 변화에 관련된 요구사항과 같이 MLS 정책기반 안전한 운영체제에서 발생할 수 있는 보안등급의 변화에 관련된 요구사항 중 특히 시스템의 환경에 의해 주체의 보안등급이 하강 되었을 때 해당 주체가 생성했던 객체들 중 비밀성이 저해되지 않는 범위 내에서 객체의 보안등급 하강을 고려한다.

접근통제시스템은 보안규칙 및 보안요구사항의 구현기능을 갖는 접근통제 메커니즘에 의존한다. 그러나 BLP모델의 접근통제메커니즘인 AM(Access Matrix), SL(Security Label)은 위와 같은 보안 요구사항을 해결 할 수 없다[9,10,11].

따라서, 본 논문에서는 BLP모델에서 시스템의 환경에 의해 주체의 보안등급이 하강 되었을 때 해당 주체가 생성했던 객체들 중 비밀성이 저해되지 않는 범위 내에서 객체의 보안등급의 안전한 하강을 돕는 접근통제 메커니즘을 제안한다. 논문의 구성은 다음과 같다. 2장에서는 객체의 안전한 보안등급 하강을 위해 필요한 정보들에 대하여 기술하고, 3장에서는 2장에서 언급한 정보들을 포함하는 접근통제 메커니즘에 대하여 기술한다. 끝으로, 4장에서는 결론과 향후계획에 대하여 기술한다.

2. 객체의 안전한 하강을 위한 정보들

기존 AM과 SL 접근통제메커니즘의 보안등급과 접근권한을 이용하여 객체의 안전한 보안등급의 하강을 수행 할 수 없다. 따라서, 본 논문에서는 2ISL(Internal Information Security Level), DR(Downward Reference), Private의 3가지정보를 이용하여 객체의 안전한 하강을 수행하고자 한다.

2.1 2ISL

보안등급은 시스템내부의 각각의 주체와 객체들의 보안 중요도의 계층적인 관계를 표현한다. 대부분의 시스템은 정보의 흐름측면에서 보안등급의 하강은 허용하지 않고, 단지 객체의 보안등급의 상승(Upgrade)만을 허용한다. 하지만 이러한 원칙에는 객체의 과등급화(Over-Classify)와 같은 예외상황이 발생 할 수 있다[12,13]. 따라서, 보안관리자에 의해 객체의 하강을 수행할 때 과등급화된 객체는 해당 객체내부의 정확한 등급정보를 나타내는 추가적인 등급정보를 이용한다면 객체의 하강을 안전하게 수행 할 수 있게 된다. 본 논문에서는 객체내부정보의 정확한 중요도를 나타내기 위하여 내부정보보안등급(Internal Information Security Level : 2ISL)이라는 새로운 보안등급을 제시한다. 다음은 2ISL의 특징과 설정방법을 살펴보자.

- 2ISL의 특징
  - 2ISL은 객체내부정보의 중요도를 나타내는 객체와

관련된 보안등급이다.

- 2ISL은 MLS의 객체와 같은 보안등급체계를 가지고 있다.
- 시스템내의 각각의 객체는 기존의 보안등급과 2ISL 값을 모두 가지고 있다.
- 2ISL은 접근통제를 위하여 사용하지는 않지만, 보안관리자에 의하여 객체의 보안등급이 하강 될 때 사용된다.

• 2ISL의 설정방법

먼저 정형화된 표기법을 이용하여 2ISL 설정방법을 기술하기 위해 몇 가지 구성요소를 살펴보자.

- S : 주체(사용자)들의 집합. $s \in S$.
- O : 객체의 집합. $o, o' \in O$.
- L : 순서화 된 보안등급의 집합. $l, l' \in L$.
- $2ISL(o)$: 객체 o의 2ISL : $o \in O$, $2ISL(o) \in L$.
- $l(s)$: 주체 s의 보안등급 : $s \in S$, $l(s) \in L$.
- $f: A \rightarrow B$: f는 정의역 A의 임의의 원소에 대응하는 공역 B의 원소를 나타내는 함수.
- *boolean* : true, false값을 갖는 집합.

2ISL 설정에 사용될 함수들은 다음과 같다.

- $Set\_2ISL : o \times l \rightarrow boolean$
- $Owner : s \times o \rightarrow boolean$
- $Dominates : l \times l' \rightarrow boolean$
- $CreateFromRead : o \times o' \rightarrow boolean$
- $AppendToObject : o \times o' \rightarrow boolean$
- $WriteToObject : s \times o \rightarrow boolean$

위의 함수를 이용하여 4가지 경우의 2ISL설정 규칙을 기술하면 다음과 같다.

- $\forall o,o' \in O,\ 2ISL(o), 2ISL(o') \in L : Set\_2ISL(o, 2ISL(o')) \text{ iff } CreateFromRead(o,o')$

새로운 객체가 생성될 때 새로운 객체의 보안등급은 그 객체의 생성자의 행위에 의존적으로 설정된다. 그림 1과 같이, 주체가 다른 객체를 이용하여 새로운 객체 o를 생성한 경우 o의 2ISL은 이용되어진 객체 o'의 2ISL로 설정된다. 파일복사동작과 유사하게 주체는 단지 o'의 정보를 o에 가져온 것이 때문에 o와 o'은 같은 2ISL을 가질 수 있다.

- $\forall s \in S,\ \forall o,o' \in O,\ l(s) \in L : Set\_2ISL(o, l(s)) \text{ iff } CreateFromRead(o,o') == false \text{ and } Owner(s,o)$

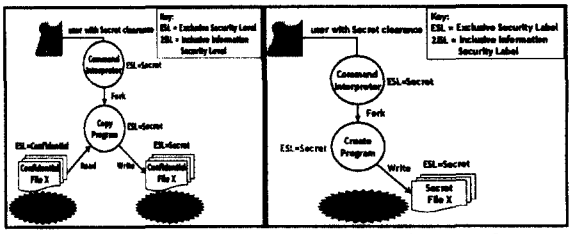
새로운 객체가 생성될 때 새로운 객체의 보안등급은 그 객체의 생성자의 행위에 의존적으로 설정된다. 그림 2과 같이, 주체 s가 다른 객체의 참조 없이 독립적으로 새로운 객체 o를 생성한 경우, 기존의 BLP의 보안등급설정방법과 동일하게[9,10,11] o의 2ISL은 s의 Clearance Level과 같게 된다.

- $\forall o,o' \in O,\ 2ISL(o), 2ISL(o') \in L : Set\_2ISL(o, 2ISL(o')) \text{ iff } AppendToObject(o,o') \text{ and } Dominates(2ISL(o'), 2ISL(o))$

객체의 2ISL은 주체의 추가 접근에 의해서 변경될 수 있다. 주체가 추가접근으로 다른 객체 o'의 정보를 객체 o에 추가할 경우, o의 2ISL은 o'의 2ISL값으로 변경된다.

- $\forall s \in S,\ \forall o \in O,\ 2ISL(o) \in L,\ l(s) \in L : Set\_2ISL(o, l(s)) \text{ iff } WriteToObject(s,o) \text{ and } Dominates(l(s), 2ISL(o))$

객체의 2ISL은 주체의 쓰기 접근에 의해서 변경될 수 있다. 주체 s가 다른 객체의 정보를 이용하지 않고 독립적으로 수행할 경우, 기존의 BLP의 보안등급설정방법과 동일하게[9,10,11] 해당 객체 o의 2ISL은 s의 Clearance Level을 상속 받는다.



[Figure 1]

[Figure 2]

2.2 하위참조(Downward Reference : DR)

참조(Reference)는 UNIX의 링크 유틸리티와 같이 주체가 허용된 기준 내에서 객체의 접근을 쉽게 하기 위하여 사용하는 방법 중에 하나이다. MLS 보안정책에 의해 하위등급의 객체가 상위등급의 객체를 접근할 수 없기 때문에 기본적으로 상위참조를 허용하지 않는다. 하위참조는 MLS 보안정책에 위배되지 않고 허용된 기준 내에서 주체가 객체의 접근을 쉽게 하기 위하여 허용될 수 있다. 그러나 시스템에 의해 허용된 하위참조는 객체의 보안등급의 안전한 하강을 수행하기 위해서는 관리가 필요하다. 예를 들어, 객체 o는 자신보다 보안등급이 낮은 객체 o'을 참조하는 하위참조라고 하자. 객체의 보안등급의 하강을 수행하다가 객체 o의 보안등급이 객체 o'보다 낮은 등급으로 낮아지는 경우가 발생할 수도 있다. 이런 경우 정보의 흐름측면에서 비밀성에 취약점이 발생한다. 하위참조 였던 객체 o가 상위참조가 되어 버리기 때문이다. 하위참조에 해당하는 객체는 다음과 같이 표현할 수 있다.

- $DR : o \times o' \rightarrow boolean$: 객체 o가 객체 o' 을 하위참조하면 DR(o,o')은 true이다.
- $Reference : l \rightarrow boolean$: 객체가 l 등급이 참조되고 있으면 Reference(l)은 true이다.

$\forall o,o' \in O,\ l(o') \in L : Reference(l(o')) \text{ iff } DR(o,o')$

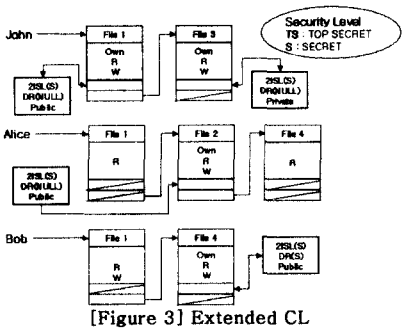
2.3. Private Objects vs. Public Objects

예를 들어, 객체가 SECRET 2ISL을 가지고 있지만 개인메일 또는 일기파일과 같이 사용자의 개인적인 정보만을 포함하고 있다면, 객체의 생성자의 보안등급이 낮아지면 해당객체의 보안등급 또한 낮아져야 한다. 만약 비밀성 측면만을 고려하여 해당 객체의 보안등급을 하강하지 않으면, BLP의 보안특성에 의해 보안등급이 하강 된 주체는 자신의 개인메일 또는 일기에 관련된 객체들을 접근 할 수 없게 된다. 다음과 같은 사항은 가용성 측면을 고려할 때 해결 하여 할 문제이다.

그러나 2ISL과 참조정보만을 이용해서는 해당객체의 안전한 하강을 수행할 수 없다. 만약 해당객체를 접근하는 주체들의 행위들을 파악할 수 있다면 해당객체가 개인적인 자료를 포함하는지 아닌지를 판단할 수 있을 것이다. 객체에 대한 주체들의 행위에 의해 해당객체의 자료성격을 파악하기 위한 간단한 규칙은 다음과 같다.

- A : 접근모드의 집합. { Read, Write, Append, Execute }. $a \in A$.
- $Allow : s \times a \times o \rightarrow boolean$: 예를 들어, my_file을 bill이 읽기 접근(read)을 할 수 있으면 Allow(bill, Read, my_file)은 true가 된다.
- $Private : o \rightarrow boolean$: 객체 o가 개인정보를 가지고 있는 객체이면 Private(o)는 true이다.

$\forall o \in O,\ s \in S,\ a \in A : Private(o) \text{ iff } Access(s,a,o) \text{ and } Owner(s,o)$



[Figure 3] Extended CL

객체*o*를 생성한 주체*s*만이 해당객체*o*를 접근*a*했을 경우, 객체*o*는 개인자료를 포함하는 객체이다.

3. 확장된 접근통제 메커니즘

접근통제시스템은 2ISL, DR, Private정보를 포함한 확장된 CL을 이용하여 시스템내부객체의 보안등급의 하강을 안전하게 수행 할 수 있다. 확장된 CL는 그림3과 같다.

주체의 보안등급이 하강 될 때 해당 주체가 생성했던 객체들의 보안등급을 고려하기 때문에 객체의 동작영역에 Own값이 설정되어 있는 객체에 대해서만 정보를 추가한다. 추가정보는 다음과 같이 설정된다.

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

$\forall o \in O$: *2ISL(2ISL(o)) iff Set_2ISL(o,2ISL(o))*

$\forall o \in O$: *DR(l(o)) iff Reference(l(o)) , DR(NULL) iff Reference(l(o)) == false*

$\forall o \in O$: *Private iff Private(o) , Public iff Private(o) == false*

다음 알고리즘을 이용하여 객체의 하강이 수행된다.

- *Level(After(s))* : 주체*s*의 보안등급이 낮아진 이후의 주체*s*의 보안등급.
- *Level(2ISL(o))* : 객체*o*의 2ISL 보안등급.
- *Level(DR(o))* : 객체*o*의 보안등급

```
If ( Private ) then
    Downgrade Object
end if
If ( Public ) then
    If ( Dominates(Level(After(s)), Level(2ISL(o))) ) then
        If ( Level(DR(o)) == NULL ) then
            Downgrade object
        else if ( Dominates(Level(After(s)), Level(DR(o))) ) then
            Downgrade object
        end if
    end if
end if
```

4. 결론 및 향후계획

Tranquility Principle을 포함하고 있는 대부분의 보안모델은 시스템에 구현되어 운영될 때 시스템의 가용성측면에서 주체 및 객체의 보안등급의 변화를 필요로 한다. 보안모델이 실세계의 시스템으로 구현될 때 시스템의 환경 및 보안등급변화에 관련된 요구사항에 따라 주체 및 객체의 변화를 처리하기 위한 방법이 달라진다. 따라서, 시스템이 보안등급변화에 관련된 모든 요구사항을 수용할 수 없다. 본 논문에서는 특히, MLS 정책기반 BLP 모델을 적용한 시스템에서 주체 및 객체의 보안등급 변화에 관련된 요구사항 중 특히 시스템의 환경에 의해 주체의 보안등급이 하강 되었을 때 해당 주체가 생성했던 객체들 중 비밀성이 저해되지 않는 범위 내에서 객체의 보안등급 하강을 고려했다. 또한, 해당객체의 보안등급의 안전한 하강을 돕는 확장된 CL 접근통제 메커니즘을 제안했다.

확장된 CL에는 기존의 CL에 3가지의 객체에 관련된 정보를 추가하고 있다. 첫째, 객체 내부정보의 중요도를 나타내는 보안등급인 2ISL. 둘째, 객체참조에 관련된 정보인 DR. 셋째, 객체가 개인정보를 포함하는지 포함하지 않는지를 포함하는지를 나타내는 Private/Public 정보. 3가지의 정보를 이용하여 앞에서 기술한 알고리즘을 이용하여 객체의 안전한 하강을 수행 할 수 있다. 단, 주의해야 할 사항은 Private정보라고 설정되어서 하강되었던 객체는 보안관리자에 의해 관리되어야 한다.

향후 연구과제로 객체의 안전한 하강을 돕는 2ISL, DR, Private에 해당하는 정보를 설정하기 위한 보안 모듈을 설계하고, 안전한 운영체제에 해당 모듈을 구현하여 본 논문에서 고려한 보안등급에 관련된 요구사항이 시스템의 안전성이 저해되지 않는 범위 내에서 이루어지는지를 확인하고자 한다.

5. 참고 문헌

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

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

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

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

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

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

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

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

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

[13]Berger, J.L., Picciotto, J., ompartmented mode workstation: prototype highlights, Software Engineering, IEEE Transactions on, Page(s) 608-618, June, 1990.