韓國 OA學會 論文誌
第5卷 第3號, 2000. 9.

2000-5-3-3-4

안전한 인터넷 사용을 위한 접근제어 메커니즘 설계

이 호* 정 진 욱**

A Design of Access Control Mechanism for the Secure Use of Internet

Ho Lee* Jin-Wook Jung**

요 약

본 논문은 인터넷 환경에서 접근제어 요구 사항의 복잡한 문제를 해결하기 위한 접근제어 메커니즘의 설계 방안을 제시한다. 본 논문에서는 자원의 기밀성, 무결성 및 가용성의 공통적 목적을 달성하기 위한 접근제어 메커니즘을 제안하고, 신분-기반, 규칙-기반 및 직무-기반의 관점에서 각 관련된 정책과 규칙을 정의하였으며, 필요한 접근제어 오퍼레이션들을 구현하였다. 제안된 접근제어 메커니즘은 보안 레이블, 무결성 등급, 직무 및 소유권 등의 다단계 보안 정책을 기반으로 하여 자원에 대한 불법적인 접근을 방어 할 수 있다.

Abstract

This paper presents a design of an access control mechanism that can resolves the complicated problems of access control requirements in internet environment. In this paper, we proposed an access control mechanism which can satisfy the combined goals of confidentiality, integrity and availability of any resource. We defined an access control mechanism from the viewpoints of identity-based, rule-based and role-based policy and implemented 6 access control operations. The proposed access control mechanism can protect resources from unauthorized accesses based on the multi-level security policies of security label, integrity level, role and ownership.

* 송호대학 정보산업계열 부교수
** 성균관대학교 전기전자 및 컴퓨터공학부 교수

Ⅰ. 서 론

보안에 관한 문제는 인터넷의 급격한 기술 발전으로 정보의 공유와 다양한 정보 서비스에 대한 욕구가 증가됨에 따라서 이러한 서비스들을 위하여 함께 해결해야 할 중요한 문제로서 인식되고 있다.

인터넷을 이용한 응용 서비스에는 정보에 대한 불법적 사용, 노출, 파괴 및 변경 등의 위협 요소들이 있다[3]. 이와 같은 보안 문제 해결을 위한 기본적인 요구사항은 권한 없는 자에 대한 정보의 노출 및 유통을 방지하기 위한 기밀성(confidentiality) 보장, 권한 없는 자의 불법적인 정보 변조를 방지하여 정보의 일관성을 유지하는 무결성(integrity) 보장과 정당한 사용자에게 정보와 자원의 사용을 보장하는 가용성(availability) 보장의 세 가지로 구분할 수 있다[3, 5, 6].

본 논문에서는 안전한 인터넷 사용을 위한 보안 모델로써 기밀성과 무결성을 보장할 수 있는 접근제어 메커니즘을 제안한다. 제안하는 접근제어 모델은 계층적 분류 체계인 다단계 보안 등급을 사용하여 각 등급간의 정보 유통과 정보의 생성, 삭제 및 변경에 대해 엄격히 제한한다. 또한, 비계층적 분류 체계인 보안 범주(category)와 직무(role)를 사용하여 주체가 접근할 수 있는 범위에 대하여 수행 가능한 역할을 정의한다.

Ⅱ. 접근제어 정책

본 절에서는 제안하는 접근제어 메커니즘의 무결성과 기밀성 보장을 위한 보안 정책을 정의한다. 기밀성 보장을 위해서 BLP 모델의 보안 성질(security property)인 임의-보안 (discretionary-security), 단순-보안(simple-security) 및 스타-보안(star-security) 성질을 이용하고[4, 7, 8, 9, 10, 11], 무결성 보장을 위해서는 Biba 모델에서 엄격한 무결성 정책(strict integrity policy)과 접근제어 리스트(access control list)를 이용한 새로운 정책 및 접근제어 규칙을 정의한다[4, 12].

1. ACI 관리정책

ACI를 구성하는 항목 : identifier, owner, security label, integrity level, role, 통신망상의 위치(IP address).

- 모든 실체는 ACI에 존재한다.
- 모든 실체는 ACI에 보안 레이블을 명시한다.
- 모든 실체는 ACI에 무결성 등급을 명시한다.
- 모든 실체는 ACI에 직무를 명시한다.
- 모든 실체는 ACI에 소유권자를 명시한다.
- 새로운 실체 생성시 ACI에 등록한다.
- 실체의 보안 정보 수정 시 ACI에도 수정한다.
- 실체의 삭제 시 ACI에서도 삭제한다.
- 실체 생성시 생성된 실체는 생성자의 ACI를 상속한다.

2. 소유권자 관리정책

- 모든 실체는 자신의 소유권자가 있다.
- 생성된 실체는 생성자의 소유권자를 상속한다.
- 소유권자의 변경은 소유권자 및 권한자만이 변경한다.
- 정보의 전달 후 전달된 정보에 대한 소유권자는 정보를 전달받은 객체가 된다.

3. 보안 레이블 관리정책

- 모든 실체는 자신의 보안 레이블을 소유한다.
- 보안 레이블에는 보안범주(category), 보안등급을 명시한다.
- 새로운 실체 생성시 생성자의 보안 레이블을 상속받는다.
- 주체는 자신의 보안 레이블을 변경할 수 없다.
- 주체는 객체의 보안 레이블을 변경할 수 없다.
- 객체는 객체의 보안 레이블을 변경할 수 없다.
- 실체의 보안 레이블 변경은 보안 레이블 변경 권한자만이 할 수 있다.

4. 신분 기반 정책

- 모든 실체는 자신이 소유한 객체에 대한 접근모드 설정권을 가진다.
- 주체는 객체의 보안 범주를 지배한다.

5. 규칙 기반 정책

(1) 다음을 만족시킬 때 observe 동작을 수행할 수 있다.

- 주체의 보안 레이블은 객체의 보안 레이블을

지배한다.
- 객체의 무결성 등급이 주체의 무결성 등급을 지배한다.

(2) 다음을 만족시킬 때 modify 동작을 수행할 수 있다.
- 주체는 객체의 소유권자이다.
- 주체와 객체의 보안 레이블이 일치한다.
- 주체와 객체의 무결성 등급이 일치한다.

(3) 다음을 만족시킬 때 delete 동작을 수행할 수 있다.
- 주체는 객체의 소유권자이다.
- 주체의 보안 레이블이 객체의 보안 레이블을 지배한다.
- 주체의 무결성 등급이 객체의 무결성 등급과 일치한다.

(4) 다음을 만족시킬 때 move 동작을 수행할 수 있다.
- 주체는 객체 정보의 소유권자이다.
- move를 수행하는 주체는 객체정보와 객체의 보안 레이블을 지배한다.
- 객체 정보와 객체의 보안 레이블이 일치한다.
- 객체 정보와 객체의 무결성 등급이 일치한다.

6. 직무 기반 정책

- 주체는 객체의 소유권자이다.
- 주체의 직무로서 수행 가능한 프로그램이어야 한다.
- 주체의 보안 레이블이 요구한 프로그램의 보안 레이블을 지배한다.
- 주체의 무결성 등급이 요구한 프로그램의 무결성 등급과 일치한다.

Ⅲ. 접근제어 메커니즘 설계

1. 접근제어 메커니즘 구조 및 구성 방법

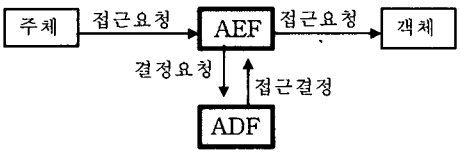

그림 1. 접근제어 메커니즘 구조
Fig 1. Structure of Access Control Mechanism

본 메커니즘은 주체가 접근을 요청하면 그 접근 요청에 대한 접근제어 결정을 요구하고, 결정된 접근제어를 시행하는 접근제어 시행 함수인 AEF와 AEF로부터의 접근제어 결정 요구가 있을 때 접근제어 규칙에 따라 접근제어 결정을 수행하는 함수인 ADF의 두 가지 요소로 이루어진다. 이러한 메커니즘의 개념적 구조는 그림 1 및 그림2와 같다.

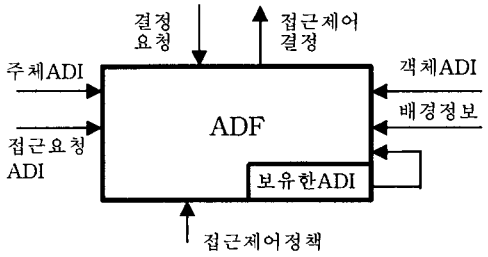

그림 2. ADF 모델
Fig 2. ADF Model

이 모델을 이용하여 구성하고자 하는 접근제어 구조는 그림3과 같이 주체와 객체에 AEF를 포함하고, 외부의 제 3 기관에 ADF를 위치시키는 입력 접근제어와 출력 접근제어의 혼합형으로 구성한다.

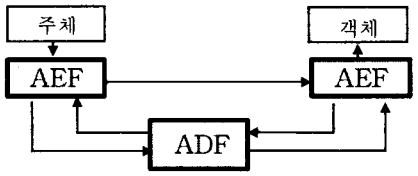

그림 3. 접근제어 구성 방법
Fig 3. Construction Method of Access Control

그림3의 접근제어 구성 방법은 주체의 출력(outgoing) 접근제어와 객체의 입력(incoming) 접근제어를 순차적으로 수행하여 주체 지역의 접근제어 요소들의 객체에 대한 불법적 접근제어를 방어할 수 있고, 객체 지역의 접근제어 시스템에 대한 신뢰도와 관계없이 안전한 접근제어를 제공할 수 있으며, 권한 없는 제3자의 위장 접근제어를 방어할 수 있다.

2. 접근제어 규칙

새로 정의한 접근제어 정책을 효율적으로 수행하기 위한 접근제어 규칙을 정의할 필요가 있다. 제안한 메커니즘에서 제공하는 접근 모드로는 observe, modify와

execute가 있다.

(1) 임의 접근제어 규칙(discretionary access control rule)

```
discret_acr(i, t, m) =
TRUE : if permit(s, o, m)) and
       dominate(Category(i), Category(t))
FALSE : otherwise
```

(2) 단순 접근제어 규칙(simple access control rule)

```
simple_acr(i, t, m) =
TRUE : if dominate(S_Label(s), S_Label(o))
       and dominate(I_Level(o), I_Level(s))
FALSE : otherwise
```

(3) 강력 접근제어 규칙(strict access control rule)

```
strict_acr(s, o, m) =
TRUE : if m = 'o' and
       dominate(S_Label(s), S_Label(o))
       and dominate(I_Level(o),
       I_Level(s))
TRUE : if m = 'm' and equal(S_Label(s),
       S_Label(o)) and equal(I_Level(s),
       I_Level(o))
TRUE : if m = 'e' and
       dominate(S_Label(s), S_Label(o))
       and equal(I_Level(s), I_Level(o))
FALSE : otherwise
```

(4) 흐름 제어 규칙(flow control rule)

```
flow_cr(s, o1, o2) =
TRUE : if dominate(S_Label(s),
       S_Label(o1)) and
       dominate(S_Label(s), S_Label(o2))
       and equal(S_Label(o1), S_Label(o2))
       and equal(I_Level(o1), I_Level(o2))
FALSE : otherwise
```

(5) 실행 제어 규칙(execute control rule)

```
execut_cr(s, o) =
TRUE : if equal(Role(s), Role(o)) and
       dominate(S_Label(s), S_Label(o))
       and equal(I_Level(s), I_Level(o))
FALSE : otherwise
```

3. 접근제어 오퍼레이션 설계

(1) login operation

```
login(identifier, S_Label_V, I_Level_V)
 {
  C_SLabel ← S_Label_V
  C_ILevel ← I_Level_V
  if    exist_ACI(identifier)
  then  login start
  Login_ACI = get_ACI(identifier)
  if C_SLabel = S_Label(Login_ACI)
    and C_ILevel = I_Level(Login_ACI)
  then    C_Role  Role(Login_ACI)
  login OK
  endif
 }
```

(2) create operation

```
create(s, o1, o2)
 {
  if    discret_acr(s, o1)  and
        simple_acr(s, o1, 'm')  and
        strict_acr(s, o1, 'm')
  then  s create  o2  at  o1
        create_ACI(o2)
        inherit(s, o2, S_Label(s))
        inherit(s, o2, I_Level(s))
        inherit(s, o2, owner(s))
  endif
 }
```

(3) observe operation

```
observe(s, o)
 {
   if    discret_acr(s, o)  and
         simple_acr(s, o, 'o')  and
         strict_acr(s, o, 'o')  and
   thens observe  o
   endif
 }
```

(4) modify operation

```
modify(s, o)
 {
   if s = owner(o)  and
```

```
            discret_acr(s, o)  and
            simple_acr(s, o, 'm')  and
            strict_acr(s, o, 'm')
      then  s  modify  o
      endif
}
(5) delete operation
    delete(s, o)
      {
       if     s = owner(o)  and
              discret_acr(s, o)  and
              simple_acr(s, o, 'm')  and
              strict_acr(s, o, 'm')  and
      then s  delete  o  and
           delete_ACI(o)
      endif
      }
(6) execute operation
    execute(s, o)
      {
       If     discret_acr(s, o)  and
              simple_acr(s, o, 'e')  and
              strict_acr(s, o, 'e')  and
              execut_cr(s, o)
    then     s  execute  o
     endif
    }
(7) move operation
    move(s, o1, o2)
      {
      if     s = owner(o1)  and
            discret_acr(s, o1)  and
            simple_acr(s, o1, 'o')  and
            strict_acr(s, o1, 'o')  and
            discret_acr(s, o2)  and
            simple_acr(s, o2, 'm')  and
            strict_acr(s, o2, 'm')  and
            flow_cr(s, o1, o2)
    then    i  move  (o1o2)
    endif
    }
```

4. 설계한 접근제어 메커니즘의 구조

본 논문에서 제안한 접근제어 메커니즘의 논리적 구조는 그림4와 같다. 이 구조에서 주체 지역의 AEF는 주체로부터의 출력 접근제어를 수행하고, 객체 지역의 AEF는 객체에 대한 입력 접근제어를 수행함으로써, 각 주체와 객체간의 접근제어 시스템에 대한 상호 신뢰가 미약해도 안전한 접근제어를 수행할 수 있다.

제안한 구조와 같은 입력과 출력 접근제어가 중재 된 형태의 접근제어 구조는 정보 시스템상의 모든 주체와 객체간의 접근에 대하여 수행된다. 그러므로, 모든 실체에 대하여 접근제어 정책 및 규칙이 수행될 수 있으므로, 모든 실체에 대한 기밀성과 무결성이 보증된다.

제안한 접근제어 메커니즘의 안전성을 위해서는, ADF와 ACI에 대한 접근이 엄격히 차단됨으로써 접근제어 결정의 일관성 유지를 위한 ACI나 ADF 정책 등의 무결성, 기밀성 및 정확성을 보장할 수 있다.

정보 시스템 상에서 이루어진 모든 접근제어 요구나 결정에 관한 자료는 감사 파일에 기록되므로 정보 시스템에 대한 서비스 부인이나 부당한 접근에 대한 감사 자료를 제공한다.

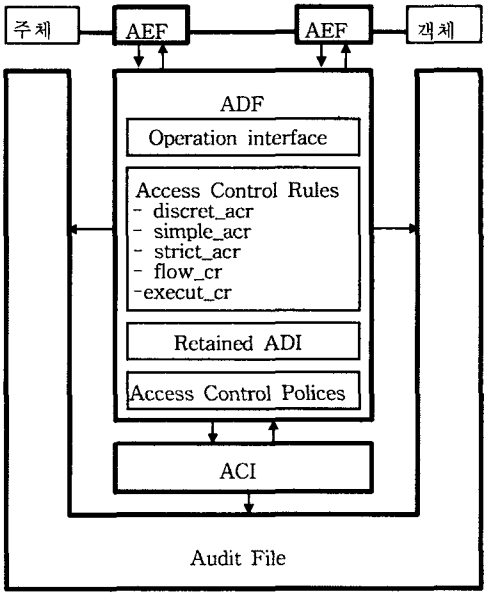

그림 4. 접근제어 메커니즘 구조
Fig 4. Structure of Access Control Mechanism

Ⅳ. 결 론

본 연구에서 제안한 접근제어 메커니즘은 보안 레이블, 무결성 등급, 직무, 소유권 등을 이용하는 다단계 보안 체계를 이용하여 권한의 불법적 사용을 방지하였다. 이러한 다단계 보안 체계를 이용하여 각 보안 등급간의 정보의 흐름을 제한함으로써 인터넷 환경에서 정보의 불법적 유통을 차단하였다. 그리고, 정보의 전송 경로에서의 정보의 불법적 노출에 대한 보안 문제는 전송되는 정보가 인증이나 암호화 기법에 의해서 암호화된 상태의 정보이므로 권한 없는 사용자에 대한 정보의 노출 위협은 없다.

제안한 접근제어 메커니즘 구성 방법을 사용함으로써 통신망 상에서 이루어지는 모든 접근제어에 대한 감시 및 감사가 용이하고, 전체 통신망에 대한 ACI의 관리가 용이해진다. AEF를 포함하고 있는 모든 주체와 객체는 자신의 시스템에 접근하여 접근제어를 수행한 모든 주체에 대한 감사 파일 작성이 용이하며, 제3기관에서 보유하고 있는 감사 파일과 ADI를 이용하여 접근제어에 대한 부인 봉쇄를 제공한다.

본 논문에서 제안한 접근제어 메커니즘 및 구성 방법은 국가 기관이나 군사 기관의 기밀성 보장을 위한 접근제어 메커니즘이나 기업이나 은행 등의 무결성 보장을 위한 접근제어 메커니즘으로써 사용할 수 있다. 또한, 기밀성과 무결성이 모두 필요한 전자상거래 시스템에도 적절한 메커니즘으로써 적용이 가능하다.

참고문헌

[1] ISO/IEC DIS 10181-1 Information Technology-Open Systems Interconnection -Security Frameworks in Open Systems-Part 1: Security Frameworks Overview, 1993.

[2] ISO/IEC DIS 10181-3 Information Technology-Open Systems Interconnection-Security Frameworks in Open Systems-Part 3: Access Control, 1993.

[3] Warwick Ford, Computer Communications Security - Principles, Standard Protocols and Techniques, Prentice Hall, 149-176, 1994.

[4] Silvanna Castono, Mariagrazia Fugini, Giancarlo Martella, Pierangela Samarati, Database Security, Addison-Wesley, 39-142, 1995.

[5] Shari Lawrence Pfleeger, A Framework for Security Requirements, Computer & Security, Vol. 10, 511-523, 1991.

[6] Wen-Pal Lu, Maluk K. Sundareshan, A Model for Multilevel Security in Computer Networks, IEEE Transactions on Software Engineering, Vol. 16, No. 6, June 1990.

[7] Bell D. E., LaPadula L. J., Secure Computer Systems : Mathematical Foundations, ESD-TR-73-278, Vol 1-2, 1974

[8] Bell D. E., LaPadula L. J., Secure Computer System : a Refinement of the Mathematical Model, Technical Report ESD-TR-73-278, Vol. 3, 1974.

[9] Bell D. E., LaPadula L. J., Secure Computer Systems : Mathematical Foundations and Model, Technical Report M74-244, The MITRE Corp., 1974.

[10] Bell D. E., LaPadula L. J., Secure Computer Systems : Unified Exposition and Multics Interpretation, The MITRE Corp., 1975.

[11] E.E.O. Roos Lindgreen, I.S. Herschberg, On the Validity of the Bell-LaPadula Model, Computer & Security, Vol. 13, 317-338, 1994.

[12] Biba K. J., Integrity Considerations for Secure Computer Systems, ESD-TR-76-372, The MITRE Corp., 1977.
[13] Leonard J. LaPadula, Formal Modeling in a Generalized Framework for Access Control, IEEE, 100-109, 1990.
[14] Ingrid M. Olson, Marshall D. Abrams, Computer Access Control Policy Choices, Computer & Security, Vol. 9, 699-714, 1990.
[15] Jones A. K., Protection Mechanism Models : Their Usefulness, In Foundation of Secure Computation, 1978.
[16] Fugini M. G., Martella, ACTEN : A Conceptual Model for Security System Design, Computer & Security, Vol. 3, 1984.
[17] Landwer C. E., Formal Models for Computer Security, ACM Computing Surveys, Vol. 13, 1981.
[18] McLean J., The Specification and Modeling of Computer Security, IEEE Computer, Vol. 23, 1990.
[19] Millen J. K., Cerniglia C. M., Computer Security Models, Technical Report n. MTR9531, The MITRE Corporation, Bedford MA, 184.

저 자 소 개

이 호
1989년 벨기에 V.U.B. 대학원 공학석사
1995년 성균관 대학교 대학원 정보공학과 박사과정 수료
1982-1991 한국전자통신연구원 선임연구원
현재 송호대학 정보산업계열 부교수

정진욱
1979년 성균관 대학교 대학원 공학석사
1991년 서울 대학교 대학원 계산통계학과 이학박사
1973-1985 한국과학기술연구소 실장
현재 성균관 대학교 전기전자 및 컴퓨터 공학부 교수