SPR를 이용한 보안 모델의 안전성 평가[1)]

강미영*, 김일곤*, 최진영*, 강인혜**, 강필용***, 이완석***, Dmitry P. Zegzhda****

*고려대학교, 컴퓨터학과

**서울시립대학교, 기계정보공학과

***한국정보보호진흥원

****St-Petersburg State Polytechnical University

The Safety Evaluation for a Security Model using SPR

Mi-Young Kang*, Il-Gon Kim*, Jin-Young Choi*

*Department of Computer Science & Engineering, Korea Univ.

**In-Hye Kang

**Department of Mechanical and Information Engineering, Univ. of Seoul

Pil-Yong Kang, Wan S. Lee***

***Korea Information Security Agency

Dmitry P. Zegzhda****

****St-Petersburg State Polytechnical Univ.

요 약

시스템의 안전성을 평가하기 위해 안전성 문제 해결 도구인 SPR[1]을 이용하여 보안 모델을 프롤로그 기반의 명세 언어인 SPSL로 기술하여 안전성을 검증한다. 보안 모델은 시스템의 3가지 컴포넌트, 시스템 보안 상태(system security states), 접근 통제 규칙(access control rules), 그리고 보안 기준(security criteria)으로 구성된다. 본 논문에서는 보안 모델의 보안 기준에 NTFS의 다중 사용 권한을 적용하여 명세하고 안전성 문제 해결방법을 제시하고자 한다.

I. 서론

컴퓨터 보안은 가용성의 손실, 비인가 접근, 또는 데이터의 수정에 대한 컴퓨터 자원의 보호를 말한다. 컴퓨터 자원 보호를 위한 모델로는 접근 통제 모델(access control models)과 정보 흐름 모델(information flow models)이 있다. 접근 통제 모델은 임의적 접근 통제(Discretionary Access Control), 강제적 접근 통제(Mandatory Access Control), 역할 기반 접근 통제(Role-Based Access Control)등이 있다. 임의적 접근 통제는 주체의 접근 권한에 따라 객체에 대한 접근을 통제하는 방법이다. 강제적 접근 통제는 주체의 보안

1) 본 연구는 한국정보보호진흥원 위탁과제로 수행되었음.

레이블과 주체가 접근하고자 하는 객체의 보안레이블을 비교하여 보안 정책에 합당한 접근 통제 규칙에 의해 접근통제를 하는 방법이다. 임의적 접근 통제는 역할(role)에 기반을 두고 사용자의 시스템 자원에 대한 접근을 제어하는 방법이다[2]. 정보 흐름 모델은 Denning의 모델에서 정보 흐름에 대한 정의를 상태 기계(state machines)로 정형화한 것을 말한다[3]. 정형화된 모델로는 Noninterference[4], Restrictiveness[5], Nondeducibility[6], Separability[7]등이 있다. SPR(Safety Problem Resolving)은 접근 통제 모델의 안전성을 평가하는 도구이고 다음 상태의 안정성을 평가하기 위해서는 정보 흐름의 개념도 수용하였다. 그리고 시스템의 상태를 프롤로그[8]기반의 SPSL(Safety Problem Specification Language)로 상태 보안를 모델링하고 접근 통제 규칙(access control rules)을 작성하고 보안 기준(security criteria)에 따라 컴퓨터의 보안성을 평가할 수 있는 정형 검증 도구이다.

본 논문에서는 SPR을 이용하여 보안 시스템의 안전성을 평가하는 방법으로 SPR의 입력으로 정의된 보안 모델을 명세하고 NTFS 다중 사용 권한에 대한 보안 기준을 명세하여 '거부된 사용 권한은 다른 모든 사용 권한보다 우선한다'는 접근 규칙을 명세, 검증하고 발견된 보안상 결함을 수정하는 방법을 설명하고자 한다. 2장에서는 SPR과 SPSL에 대한 설명을 하고, 3장에서는 보안 모델에 대한 NTFS 다중 사용 권한을 SPSL로 명세, 검증하고 분석한 후 오류 수정의 방법을 제시한다. 4장에서 결론 및 향후 연구 방향을 제시한다.

II. SPR과 SEW

1) SPR

SPR(Safety Problem Resolver)은 프롤로그 기반의 규칙으로 구성된 보안 시스템의 안전성 평가 도구이다. SPR은 두 가지 측면에서 안전성 문제를 해결할 수 있다.

첫째, 만일 보안 기준에 대해 주어진 현 시스템 상태를 평가하기를 원한다면, 시스템 보안 상태와 보안 기준, 접근 규칙을 SPR에 입력하면 보안 기준에 따라서 시스템의 안전성 상태를 평가할 수 있다.

둘째, 만약 시스템의 안전성을 평가하기를 원한다면, 초기 상태에서 도달 가능한 시스템 안전 상태들을 생성하고 생성된 상태들의 안전성을 평가한다.

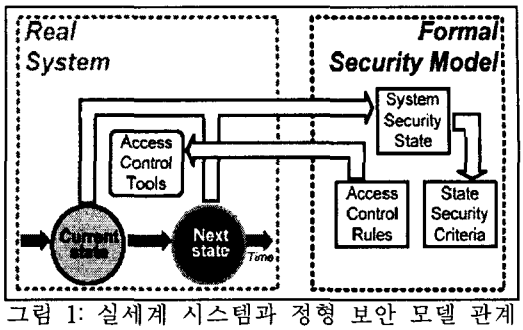

그림 1: 실세계 시스템과 정형 보안 모델 관계

그러므로, SPR은 안전성 평가을 위한 보안 모델을 입력으로 받아 들인다. SPR은 초기 상태의 안전성/비안전성을 검사하고, 현 상태에서 도달 가능한 모든 상태를 생성하고 도달 가능한 상태들의 안전성과 비안전성을 보여 줄 수 있다.

그림 1에서 실세계의 시스템의 정형 보안 모델을 표현하기 위해, 먼저 현 상태에서 시스템 보안 상태(system security state)를 추출하고, 안전성을 평가하기 위한 접근 통제 규칙(access control rules)을 작성하고, 모델의 안전성을 평가하기 위한 보안 기준(security security criteria)을 명세할 수 있다. 그리고 다음 상태를 나타내는 시스템 보안 상태, 현 상태에서 다음 상태로 정보가 흐르는 것을 접근 통제 규칙으로 기술하고, 다시 다음 상태의 안전성을 평가하는 보안 기준을 적용할수 있다. 보안 모델과 SPR의 관계는 그림 2로 나타낼 수 있다. 그림 2에서 보안 모델은 시스템 보안 상태와 접근 통제 규칙, 그리고 보안 기준으로 명세하고 SPR의 입력으로 사용된다. 보안 모델의 명세는 프롤로그 기반의 명세 언어인 SPSL(Safety Problem Specification Language)이 사용된다. 시스템 보안 상태는 시스템 상태의 추상화이다. 시스템의 요소들은 사용자의 계정, 수행중인 프로그램, 파일, 접근 권한등을 나타내며 주체(subject), 객체(object)로 표현할 수 있다. 시스템의 보안 상

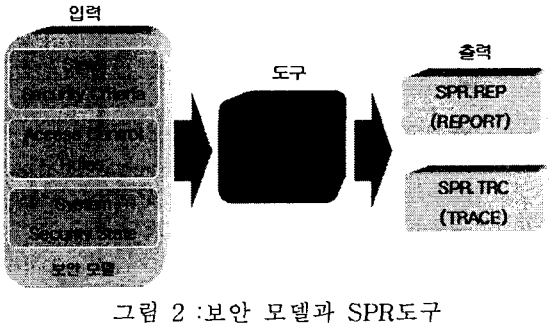

그림 2 :보안 모델과 SPR도구

태는 프롤로그 문법으로 데이터베이스의 사실(facts)로 표현한다. 접근 통제 규칙은 시스템 행동의 제한을 표현한다. 시스템의 상태 변화는 시스템 주체(subject)가 접근 통제에 의해 허락된 접근 후에 가능하다. 마지막으로 보안 기준은 안전/비안전 상태를 판별하기 위해 정의한다. 접근 통제 규칙과 보안 속성은 프롤로그의 규칙(rule)으로 명세한다. 보안 모델이 입력되면 SPR이 보안 기준에 의해 안전성을 판별하여 succeeded/failed의 결과를 SPR.REP의 파일로 출력하고 안전성 평가의 과정을 프롤로그의 trace형태의 파일(SPR.TRC)로 출력한다. 시스템의 보안 기준에 따라 failed가 출력되면 SPR.TRC파일을 분석하여 보안 기준에 맞지 않는 부분을 분석하는 작업을 수행하여야 한다. 하지만 보안 모델의 명세와 평가 결과의 분석은 전문적인 시스템 분석가에게도 어려운 작업이다. 그래서 각 단계의 명세와 분석을 도와주는 도구가 필요하다.

2) SEW

시스템 분석가와 시스템 설계자가 SPR도구를 사용하기 위해서는 각 단계를 도와주는 도구가 필요하다. SEW(Safety Evaluation Workshop)는 SPR을 이용하여 안전성을 평가하기 위한 주변 도구들로 구성되어 있다. SEW을 구성하고 있는 중요 컴포넌트들은 그림 3에 잘 나타나 있다.

SPR의 입력인 보안모델의 시스템 보안 상태를 프롤로그의 사실(facts)형태로 자동 추출을 해주는 시스템 상태 분석기(System State Analyzer)와 보안모델을 평가하기 위한 보안기준을 GUI 형태로 입력할 수 있도록 도와주는 보안 기준 관리기(Security Criteria Manager)가 있다. 그러나 보안 시스템을 기술하는 접근 통제 규칙을 자동 추출하는 도구는 없다. 그리고 SPR도구의 출력 결과물로는 SPR.TRC와 SPR.REP가 있다. 먼저 SPR.TRC의 분석을 도와주는 도구로는 보안 결함 탐색기(Security Flows Explorer)가 있다. 이 탐색기는 보안성 평가 결과 과정이 프롤로그의 trace로 출력된 결과에서 보안 결함 부분을 자동으로 찾아주는 도구이다. 그리고 SPR.REP을 좀더 쉽게 사용하기 위해 인터넷 파일 형식으로 출력해 주는 평가 보고서(Evaluation Reporter)가 필요하다.

Ⅲ. SPR를 이용한 보안 모델의 안전성 평가

Windows 2000은 자원의 보안과 사용자의 관리에 있어서 NTFS 파일 시스템 기반의 NTFS 사용 권한으로 파일과 폴더의 제어 관리를 지원한

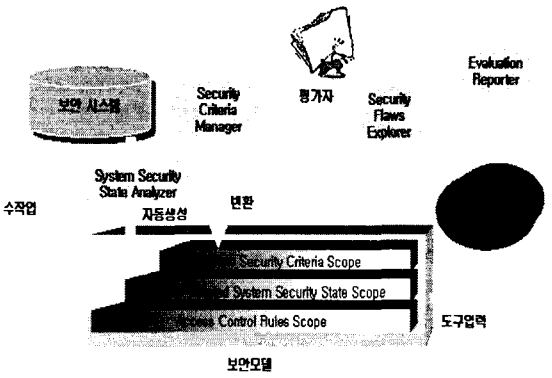

그림 3: SEW(Safety Evaluation Workshop)

다. NTFS는 Windows 2000과 Windows NT에서만 지원하는 파일 시스템이다. NTFS의 특징으로는 대용량 매체 지원, 긴 파일 이름 지원, 시스템 복원 등을 지원하고 NTFS 사용 권한을 이용하여 자원의 보안과 사용자 관리에 있어 공유된 폴더와 파일 뿐만 아니라 사용자 개인의 자원까지 관리가 가능하다.

NTFS 파티션에서 파일과 폴더를 안전하게 관리 하기 위해 NTFS 사용 권한을 사용하여 사용자와 그룹계정에 NTFS 권한 허용을 사용한다. 만약 사용자나 사용자가 속한 그룹에 접근 권한을 부여하지 않았다면, 그 사용자는 자원을 접근할 수 없다. NTFS 사용 권한은 자원의 보안과 사용자 관리를 지원하기 때문에 사용자 개인의 파일과 폴더에 대해 사용자에게 알맞은 권한 허용을 부여해야 한다. NTFS 사용 권한은 기본적으로 관리자나 파일 또는 폴더의 소유자는 Full Control(모든 권한)이 부여된다. 또한 NTFS 사용 권한은 사용자 계정에 부여하고 그룹에도 부여할 수 있다. 하나의 사용자는 다수의 그룹에 속할 수 있고, 그룹마다 다른 권한허용이 부여될 수 있다. 본 논문에서는 NTFS 다중 사용 권한에 대하여 보안 기준을 두고 교수(Profs)와 학생(Students) 그룹에 대한 상태 모델링을 하고 Windwos 2000의 규칙에 따라 모델의 안전성 검사를 제시한다

1) NTFS 다중 사용 권한

사용자에게 NTFS 허가가 다중으로 부여되었을 때 사용자에게 다중 사용 권한이 유효하다.[7]

㉠ 사용자에게 폴더 읽기 권한을 부여하고 그 사용자가 속한 그룹에 쓰기 권한이 부여된다면 사용자에게 유효한 권한은 읽기/쓰기가 가능하다.

㉡ NTFS 파일 사용 권한은 폴더 사용 권한 보

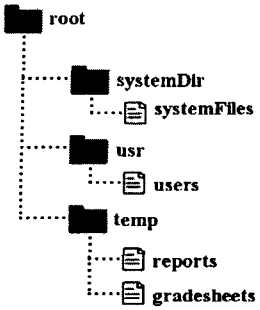

그림 4 : 객체의 초기 구조

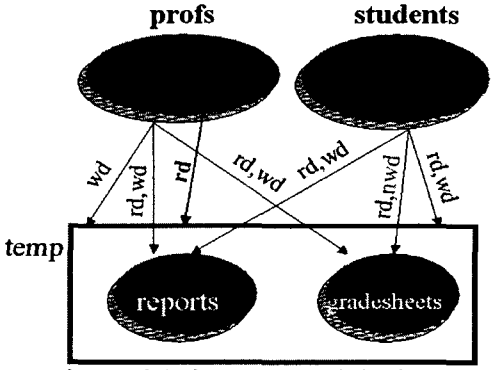

그림 5 : 사용자 그룹과 객체의 접근 권한

다 우선한다.

㉢ 폴더에 읽기 사용 권한이 부여된 사용자가 폴더에 있는 파일에 변경 가능한 사용 권한이 부여 되면 사용자는 파일 사용에 대한 권한인 변경 권한이 가능하다.

㉣ 거부된 사용 권한은 다른 모든 사용 권한보다 우선한다.

2) 시스템 보안 상태

객체의 초기 구조는 그림 4와 같이 root 폴더에 시스템 폴더와 시스템 파일, user 폴더와 users 파일, 그리고 temp 폴더와 reports 파일과 gradesheets 파일로 구성되어 있다. 주체(subjects)는 교수 그룹과 학생 그룹으로 구성되어 있다.

```
subject(pascal,[subjectGroups(profs)]).
subject(kant,[subjectGroups(profs)]).
subject(demian,[subjectGroups(students)]).
subject(kain,[subjectGroups(students)]).
objectAttr(profs).
objectAttr(students).
objectAttr(pascal).
objectAttr(kant).
object(temp,[objectType(dir),parentObject(root),objectOwner
(admin),administrators(rd,wd,ad,ds,d,rp,wp,wo),localUsers,
profs(wd,wp),students(rd,rp,wd),pascal(rd,rp),kant(rd,rp),
demian,kain,alice,bob,anthony,john,admin]).
object(reports,[objectType(file),parentObject(temp),objectOwner
(admin),administrators(rd,wd,ad,ds,d,rp,wp,wo),localUsers,profs
(rd,rp,wd),students(rd,rp,wd),pascal,
,kant(wp),demian,kain,alice,bob,anthony,john,admin]).
object(gradesheets,[objectType(file),parentObject(temp),
objectOwner(admin),administrators(rd,wd,ad,d,rp,wp),localUsers,
profs(rd,rp,wd),students(rd,rp,nwd),pascal,
kant(wp),demian,kain,alice,bob,anthony,john,admin]).
```

그림 6 : 초기 상태 명세

모델의 상태를 SPSL로 기술하면 그림 6과 같이 명세된다. 교수 그룹은 reports 파일에 읽기/쓰기 권한이 부여되고 학생그룹에는 reports에 읽기/쓰기가 부여되고 gradesheets에는 읽기와 쓰기 거부 권한이 부여된다. 그리고 temp 폴더에는 읽기 권한이 부여되고 교수그룹의 pascal에는 쓰기 권한만 부여한다. pascal에게는 다중 사용 권한에 의해 읽기/쓰기가 부여되고 학생그룹은 temp 폴더에

```
canWriteFile(S,O):-
        validSubject(S),
        isFile(O),
        canReadFile(S,O),
        pWriteData(S,O).
canReadFile(S,O):-
        validSubject(S),
        isFile(O),
        pReadData(S,O),
        canReadPermissions(S,O).
testState1(S,O):-
        validSubject(S),
        isStudents(S),
        canWriteFile(S,O).
testState2(S,O):-
        validSubject(S),
        isStudents(S),
        canWriteFile(S,gradesheets).
testState3(S,O):-
        validSubject(S),
        isProfs(S),
        not(canWriteFile(S,O);canReadFile(S,O)),!.
testState4(kant,O):-
        validSubject(S),
        isProfs(S),
        not(canWriteFile(S,O);canReadFile(S,O)),!.
```

그림 7 : 접근 통제 규칙

```
/*
*          SPR report file
*          File contains criteria and safety results
*/

testState1(_,_)                    failed
testState2(_,_)                    succeeded
testState3(_,_)                    succeeded
testState4(_,_)                    succeeded
```

그림 8 : SPR.REP

읽기/쓰기 권한이 부여되어 있음에도 불구하고 gradesheets에는 접근할 수 없다.

3)접근 통제 규칙과 보안 기준

접근 통제 규칙의 일부를 그림 7에 기술하였다. teststate는 보안 기준의 규칙으로 보안에 위배되는 규칙을 기술하였다. 보안 기준은 cr이라고 표현했을때, cr_i는 해당 시스템에서 발생하지 않아야 하는 속성을 의미한다.

cr_i iff !testState(_,_)∧...∧!testStateN(_,_) (1)

(1)의 !testState1(_,_)∧ ... ∧!testStateN(_,_) 에 대하여 true의 경우 시스템은 안전하다라고 결론을 내린다. 보안 범위를 기준으로 검증할 경우 testState1의 경우 failed의 결과가 STR.REP파일에 출력된다. 그림 7에서 testState1은 주체S가 학생 그룹의 kain로 가정할 경우 kain는 reports파일에 쓰기 권한이 가능한다. 그래서 보안에 위배되는 규칙이 실패되어야 하지만 성공이 나오므로 그림 8과 같이 failed의 결과가 출력된다. 그래서 그림 7의 testState2에서 canWriteFile(S,gradesheets)으로 '학생 그룹이 gradesheets를 읽을 수 있다'라고 바꾸면, unsafe상태가 fail로 발생하므로 testState2는 succeeded의 결과로 그림 8에 출력된다. testState3은 '사용자의 폴더 사용 권한에 읽기를 부여하고 그 사용자가 속한 그룹에 쓰기 사용 권한을 부여 된다면 사용자에게 유효한 권한은 읽기/쓰기가 가능하다'을 검증하기 위해 '교수 그룹의 kant, pascal은 temp파일에 대하여 읽기/쓰기가 가능하지 않다'라고 기준을 제시한다. not(canWriteFile(S,O);canRead File(S,O))는 fail이 발생하므로 그림8에서 succeeded 결과가 출력된다. testState4에서는 교수그룹중의 kant를 temp 폴더 안의 파일에 대하여 읽기/쓰기가 가능하지 않다'라는 기준을 제시하고 fail이 발생하므로 그림 8에서 succeeded 결과가 출력된다.

Ⅳ. 결론

시스템의 안정성을 평가하기 위해서, 실세계의 시스템에 대한 보안 모델이 필요하고 또한 보안 모델에 대한 정형적 검증 방법이 요구된다. 본 논문에서는 실세계 시스템의 보안 모델을 프롤로그 기반의 보안성 평가 도구인 SPR로 검증하는 예를 제시하였다. 보안 모델을 SPSL로 기술하였고 보안 기준은 NTFS 다중 사용 권한의 부분을 명세하고 검증하였다. 향후 연구로는 Windows 2000의 시스템을 SPSL로 명세하고 SPR로 안전성을 검증하며, 나아가 운영체제 시스템, IDS, Firewall등 보안 시스템의 안정성을 검증하고 분석하는 연구를 하고자 한다.

참고문헌

[1] http://www.ssl.stu.neva.ru/spr/whitepaper.htm

[2] R. Focardi, R. Gorrieri, *Foundations of Security Analysis and Design*, Springer-Verlag, 2001, pp. 137-196.

[3] R. V. Peri, "Specification and Verification of Security Policies", Doctor of Philosophy, 1996.

[4] J. A. Goguen, J. Meseguer, "Unwinding and Inference Control", Proceedings of the IEEE Symposium on Security and Privacy, April 1984.

[5] D. McCullough, "Specifications for Multi -level Security and Hook-up Property", Proceedings of the IEEE Symposium of Security and Privacy, 1987.

[6] D. Sutherland, "A Model of Information", 9th National Security Conference, 1986.

[7] J. McLean, "A general theory of Composition for Trace Sets Closed Under Selective Interleaving Functons", Proceedings of the IEEE Symposium on Research in Security and Privay, 1994.

[8] J. Wielemaker, SWI-Prolog 5.2 Reference Manual, http://swi-prolog.org, July 2003.

[9] 조성만 외 3, Windows 2000 Server, 혜지원, 2000.