컴퓨터 보안 훈련을 위한 웹 기반 교수 시스템

최진우 우종우
jwchoi@cs.kookmin.ac.kr cwwoo@kookmin.ac.kr

Web-based ITS for Training Computer Security

Jin-woo Choi Chong-woo Woo
School of Computer Science, Kookmin University

요 약

최근 컴퓨터 해킹이 커다란 사회적 문제로 대두되고 있다. 물론 시스템 보호를 위한 많은 상용 제품들이 존재하지만, 침입피해 상황에서는 대부분의 경우, 시스템 관리자의 현장 경험에 의존하는 실정이다. 따라서, 시스템 관리자는 기존의 침입에 관한 해결방법 뿐만 아니라, 새로운 위협들에 대한 대처방안을 항상 준비 하여야 한다. 이러한 침입상황을 시스템 관리자들에게 교육하기 위하여, 본 논문에서는 모의 훈련환경을 설계하고 구현 하였다. 본 시스템의 특징은 우선, 지식베이스로부터 동적으로 생성되는 학습 주제들로 이루어진 교과과정을 학습자에게 제시한다. 학습자에 의해 선택된 학습 주제는 학습목표로 간주되고, 이 주제는 교수 계획에 의해 다수의 임무(mission)들을 생성한다. 학습자는 각 임무에서 주어지는 상황을 가상의 UNIX 명령어들을 직접 사용하여 모의 실험해 봄으로써 임무 완수에 필요한 지식을 숙지할 수 있게 된다. 시스템은 임무 완수에 요구되는 해 경로(solution paths)를 유지함으로써, 학습자의 문제 해결 과정을 감독할 수 있고, 도움을 요구하거나 실수를 할 때 적절한 힌트를 제공한다. 시스템은 웹 기반의 클라이언트/서버 구조로 설계되어, 학습자는 브라우저만으로도 학습이 가능하고, 자바 애플릿으로 이루어진 가상 운영체제 하에서 직접 침입대서 상황을 학습 할 수 있다.

1. 서 론

컴퓨터 통신 네트워크의 급속한 발전은 우리 사회전반에 걸쳐 많은 이익을 가져왔다. 그러나 한편으로는 이러한 새로운 통신기술은 악의적인 컴퓨터 침입의 증가라는 부작용을 초래하고 있다. 따라서 최근에는 이러한 침입을 탐지하는 침입 탐지 시스템(Intrusion Detection System: IDS)이 활발히 연구 개발되고 있다[5][9]. IDS가 침입에 대한 경보를 알리면, 피해 시스템의 시스템 관리자는 시스템의 손상된 일부분을 복구하거나 다시 설치하는 등의 모든 대응 과정에 세심한 주의가 필요하다. 그러나 대부분의 경우 이러한 작업 과정은 시스템 관리자의 현장 경험에 의존하는 실정이며, 따라서 시스템 관리자는 기존의 침입뿐만 아니라 새로운 위협에도 항상 대처할 수 있도록 훈련할 필요가 있다.

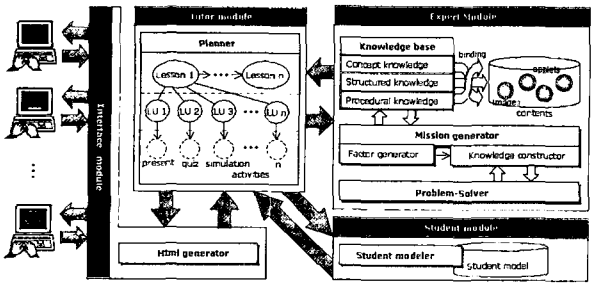
시스템 관리자를 위한 보안 교육에 관한 대표적인 연구는 ID-Tutor[7]가 있다. 이 시스템은 AI 계획 기법을 사용하여 감사 파일(audit file)을 생성하고, 학습자는 교수 규칙에 따라 모의실험을 한다. 모의 실험은 시스템 메커니즘을 이해함에 있어 보다 포괄적인 통찰력을 증진시킬 수 있으므로, 이 시스템은 보안 교육을 위한 훌륭한 프레임워크(framework)를 제공한다[6][10]. 그러나 ID-Tutor는 몇 가지 제약이 있는데, 예를 들면, 시스템이 메뉴 위주 방식이므로 학습자는 응답을 즉 시 입력할 수 없다. 또한 학습자의 단일 행동에 따른 교육 또는 힌트 제공에 그 중점을 두었기 때문에, 학습자의 학습성과를 모델화 하거나 분석하지 않는다.

본 논문에서는 컴퓨터 침입에 대한 시스템 관리자 교육을 위한 보안 교육 시스템에 관하여 기술한다. 시험 영역으로 UNIX 보안 문제를 선정하였다. UNIX 보안 주제들은 매우 다양하므로 이들 전반을 관장하는 단일 교수 시스템의 구현에 앞서, 본 논문에서는 취약점(vulnerability)문제로 국한 하였다.

2. 시스템 설계

시스템은 우선 학습자에게 교과과정(curriculum)을 제공한다. 교과과정은 학습 주제들로 구성되는데, 이러한 학습 주제(topic)는 지식베이스로부터 동적으로 생성된다. 학습자에 의해 선택된 학습 주제는 학습 목표(goal)로 간주되고, 이러한 학습주제는 교수 계획(instructional plan)에 의해 다양한 임무(mission)들을 생성한다. 학습자는 각 임무에서 주어지는 가상의 상황을 UNIX 명령어들을 직접 사용하여 모의 실험함으로써 임무 완수에 필요한 지식을 숙지하게 된다. 또한, 시스템은 임무 완수에 요구되는 해답 경로(solution paths)를 유지함으로써, 학습자의 문제 해결 과정을 감독할 수 있으며, 학습자가 도움(help)을 요구하거나, 실수를 범할 때, 적절한 힌트를 제공할 수 있는 기반을 제공 한다. 시스템 구조는 전문가 모듈(expert module),

교수모듈(tutor module), 학습자 모듈(student module), 그리고 인터페이스 모듈(interface module)을 중심으로 하는 전형적인 ITS 구조를 기반으로 하여 다음 [그림 1] 과 같이 설계하였다.



[그림 1] 시스템 구조

2.1 시스템 구조

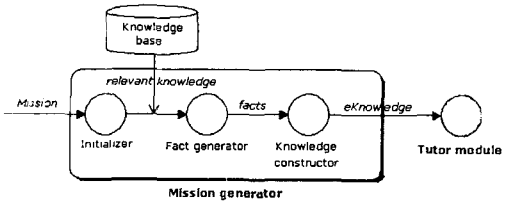
전문가 모듈은 영역 지식(domain knowledge base), 문제 해결기(problem solver), 그리고 임무 생성기(mission generator)로 구성된다. 영역 지식은 객체 지향 개념으로 설계되고, 개념 지식(concept knowledge)과 단계적 지식(structure knowledge), 그리고 절차적 지식(procedural knowledge) 들을 포함한다. 개념 지식은 학습자의 질의에 상응하는 설명을 제공하기 위한 지식이며, 단계적 지식은 학습자의 단계별 학습 진행과 필요 시 제공되는 힌트에 관한 지식이다. 그리고 절차적 지식은 학습 주제에 수반되는 수행 절차에 관련 된 지식이다. 그 외 다른 형태의 이벤트 지식이 또한 포함되는데, 이는 보다 심화된 상황을 연출하기 위한 부가적인 지식이다. 문제 해결기와 임무 생성기는 다음절에서 상세히 기술한다.

교수 모듈은 기본적으로 전체 학습물 (lesson content) 들을 생성하는 역할을 담당한다. 본 시스템에 내에서, 교수 모듈은 먼저 교과 과정을 제공하고, 학습자에 의한 선택에 준하여 개념 설명, 단계적 연습, 그리고 모의 실험과 같은 학습 계획을 생성한다. 개념들은 기본적인 보안에 관련된 정의들이나, UNIX 명령어의 사용 등과 같은 것들로 이루어져 있다. 또한 퀴즈와 같은 간단한 문제들을 제공하고, 마지막으로 학습자에게 모의 실험 연습을 통하여 직접 실행해 보는 과정을 제공한다. 시스템 제어는 기본적으로 혼합주도형(mixed initiative) 방식으로써 학습의 초기 시점에서, 주어진 메뉴에서 학습자가 직접 선택할 수 있으며, 또는 시스템에게 학습 주제를 제안하도록 요구할 수 있다. 두 가지 제어방식을 혼용하도록 설계 함으로서, 학습자에게 보다 편리하게 학습할 수 있는 환경을 제시하였다.

학습자 모델은 기본적으로 오버레이 (overlay) 기법을 사용하여 단순하게 구현하였으며, 학습자 모델은 주어진 학습의 학습자의 진행과 학습 성과에 대하여 기록하게 된다.

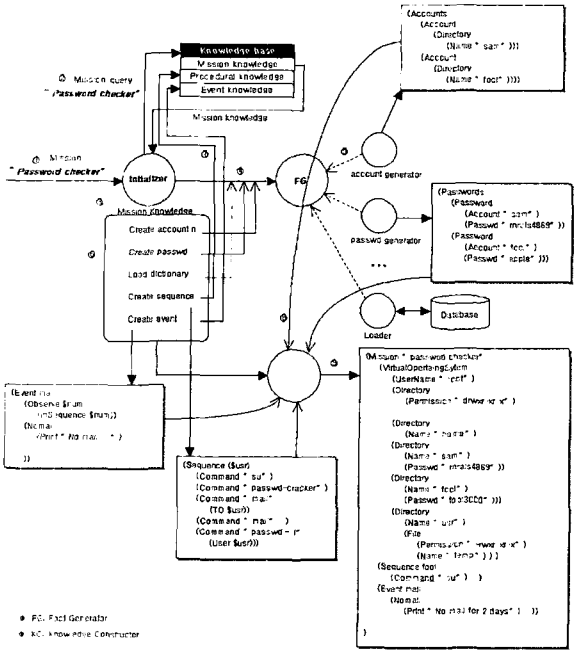
2.2 임무 생성기

임무 생성기는 선택된 임무를 위해 지식 베이스로부터 필요한 지식을 수집을 담당하며, initializer, factor generator, 그리고 knowledge constructor로 구성된다[그림 2].



[그림 2] 임무 생성기

initializer 는 선택된 임무에 관계하는 자바 객체를 초기화하고, 지식 베이스로부터 필요한 지식을 수집한다. Fact generator는 주어진 임무에 관련된 사실들을 생성하며, 다른 여러 객체들을 대표하는 객체로 존재한다. 예를 들어, 주어진 임무가 '*password checker*'라 가정하면, 생성되는 사실로는 '*(userID, userPW)*'이 된다. knowledge constructor는 fact generator로부터 유입되는 사실들로부터 지식을 S-expression 구조로 조직한다. [그림 3]은 보다 자세한 지식 생성 과정을 도식화한 것이며, 상세한 절차는 다음과 같다.



[그림 3] 임무 생성기의 내부 과정

① 선택된 임무 '*password checker*'가 임무 생성기로의 유입
② 임무 생성기는 초기화와 지식 베이스로부터 임무 지식을 수집

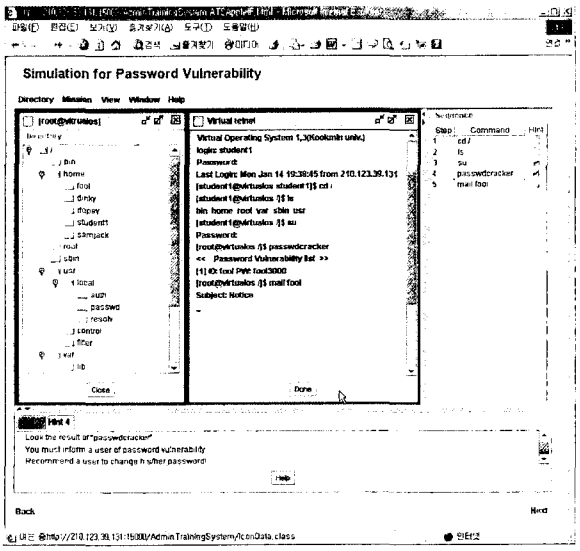
③ 지식 해석 과정 후 ④ 사실 생성기에게 일련의 해석 결과를 요구
⑤ 사실 생성기는 선택된 객체를 로드 하여 그 결과를 반환하고, ⑥ 이러한 과정(④⑤)을 반복 수행
⑦ 절차 지식의 수집
⑧ 지식 구성기로의 확장된 지식들의 전송
⑨ 확장 지식들을 S-expression 구조로 조직화

2.3. 절차적 이벤트

시스템은 절차적 지식에 의존적인 또 다른 형태 지식인 이벤트 지식을 가진다. 이벤트 지식은 학습자들의 수행에 관하여 다양성을 제공하여, 동일한 임무를 수행하더라도 동적 이벤트 지식을 기반으로 다양하게 수행될 수 있으며, 학습자에게 기대되는 절차가 증가될 수 있다. 예를 들어, "*password checker*" 임무에서 준비된 이벤트가 "newmail" 또는 "nomail"일 수 있다. 또한 이러한 이벤트 지식은 더 이상 시스템 모듈에서 사용 되지 않고 학습자가 사용하는 시뮬레이션 내의 이벤트 제어기(event controller)에 의해 동작된다.

3. 시스템 구현

본 시스템의 전체 모듈은 JDK 1.3.1을 사용하여 자바로 설계 및 구현하였으며, 서버는 Microsoft NT 4.0, 학습자 모델은 MS Access를 이용하였다.



[그림 4] 모의 실험 (mission '*password checker*')

[그림 4]에서, 화면의 상단에는 몇 가지 메뉴를 제공하는데, 메뉴의 내용은 가상 OS의 조작 및 임무에 관한 간략한 소개와 학습자에게 도움이 되는 몇 가지 기능을 제시한다. 화면의 좌측 윈도우는 가상 OS의 디렉터리들을 트리 구조로 보여주고, 중앙의 윈도우는 마치 실제 터미널과 같이 학습자가 UNIX 명령어들을 직접 입력할 수 있다. 그리고 우측 화면은 실행한 명령어를 기록하고, 하단의 윈도우에는 힌트와 도움말들이 제공된다.

4. 결론

본 연구에서는 컴퓨터 침입(computer intrusion)에 관한 훈련 또는 시스템 관리자의 교육을 목적으로 하는 웹 기반의 교수 시스템을 설계 및 구현하였다.
시스템의 중요성은 다음과 같이 기술할 수 있다. 첫째, 학습자는 실제와 같은 UNIX 명령어들을 사용해 볼 수 있다. 둘째, 시스템은 계획 기법을 사용하여 선택된 시나리오를 동적으로 다양한 보안 임무들을 생성한다. 그리고 모의 실험을 통하여 적절한 UNIX 명령어들을 사용하여 생성된 임무를 수행해 볼 수 있다. 셋째, 시스템은 임무 완수에 요구되는 해답 경로(solution paths)를 유지함으로써, 학습자의 문제 해결 과정을 감독 할 수 있으며, 또한 도움을 요구하거나 또는 실수를 범할 때 적절한 힌트를 제공한다. 넷째, 시스템은 웹 기반의 클라이언트/서버 구조로 설계되어, 학습자는 브라우저만으로도 학습이 가능하고, 학습 시나리오에 따라 자바 애플릿으로 이루어진 가상 운영체제 하에서 직접 조작 할 수 있다.

참고 문헌

1. Alpert, S., Singley, K., and Fairweather, P.: Porting a Standalone Intelligent Tutoring System on the Web. Proceedings of ITS'2000 workshop, pp. 1-11, 2000.
2. Brusilovsky, P., Eklund, J., and Schwarz, E.: Web-based education for all: A tool for developing adaptive courseware. Proceedings of the 7th WWW conference, pp. 291-300, 1998.
3. Farmer, D., and Spafford, E. H.: The COPS Security Checker System. Proceedings of the Summer USENIX Conference, pp. 165-170, 1990.
4. Hume, G., Michael, J., Rovick, A., and Evens, M.: Hinting as a tactic in one-on-one tutoring. The Journal of Learning Sciences, 5(1), pp. 23-47, 1996.
5. Kumar, S.: Classification and Detection of Computer Intrusions. PhD thesis, Purdue University, 1995.
6. Reddy, Y., Fox, M., Hussain, N., and McRoberts, M.: The Knowledge-based Simulation System. IEEE Software, Vol 3. No.2, pp. 26-37, 1986.
7. Rowe, N. C. and Schiavo, S.: An Intelligent Tutor for Intrusion Detection on Computer Systems. Computers and Education, pp. 395-404, 1998.
8. Rowe, N. C. and Galvin, T.: An authoring system for intelligent tutors for procedural skills. IEEE Intelligent Systems, 13, 3, pp. 61-69, 1998.
9. Sebring, M., Shellhouse, E., Hanna, M. and Whitehurst, R.: Expert Systems in Intrusion Detection: A Case Study. Proceedings of the 11th National Computer Security Conference, pp. 74-81, 1988.
10. Shannon, R. E.: Introduction to Simulation. Proceedings of the 1992 Winter Simulation Conference, pp. 65-73, 1992.