정보보호 시스템 보안성 자동 분석 방법 연구*

김점구** · 김태은***

요 약

국내 보안관리 시스템의 후진성은 선진기술을 가진 외국 보안업체에 의존하는 현상을 낳았고, 이는 국내 기업은 물론 공공기관의 기밀사항이 외국에 유출될 위험을 내포하고 있다. 따라서 본 논문은 국내 공공망의 안전성 유지를 위한 자동화 보안분석 시스템을 설계 구현함으로서 보안성 분석기술을 확보하고, 이를 이용 공공기관의 보안수준을 높이며 외국업체에 대한 의존도를 줄여 국가 보안 안전성 확보에 기여하고자 한다.

A Study on Scheme of Automatical Security Analysis Tools for Information Security System*

Jeom-Goo Kim** · Tae-Eun Kim***

ABSTRACT

The backwardness of Domestic security management system tend to depend on foreign security companies which have advanced technology. The appearance risk to flow out confidential affairs of domestic enterprises and public organizations to foreign countries. In this regard, this paper is implement and designed automatic security analysis system for secure public network. This system is to offer enhanced security quality of public organizations and reducing the dependence on foreign companies. And maintains security analysis technique for public network.

Key words : Security System, Information Security

* 본 연구는 산학협동재단 2007년도 연구비 지원에 의해서 이루어짐.
** 남서울대학교 컴퓨터학과
*** 남서울대학교 멀티미디어학과

1. 서 론

최근 국내·외에서 발생되고 있는 일련의 전산망 보안 침해 사고들에 대한 대책을 응용기술의 발전에 맞추어 준비해야 할 문제임은 틀림이 없을 것이다. 이러한 인터넷 활용 증가에 따른 편리함에 비례하여 역기능 요소 또한 증가해, 이에 따른 보안관리가 체계적으로 이루어져야 한다[6]. 그러므로 미국을 비롯한 정보 선진국들의 주요 기관에서는 응용 환경에 적합한 안전성과 보안수준 유지를 위해 보안관리 도구를 개발하여 활용하고 있고, 이에 관한 기술요건과 요구사항을 문서화하여 보안 관리의 정책과 모든 정보자원 보안 관리의 지침으로 활용하고 있다. 반면에 국내의 경우는 정보원에 대한 보안 관리정책과 보안관리 도구, 그리고 보안 분석 방법 등 보다 적극적이고 광범위한 보안관리 방안이 매우 미흡한 실정이다.

본 논문은 국내환경에 맞는 정보보호 시스템의 안전성을 위한 자동화 보안분석 도구를 설계 구현하여 보안관리의 효율성을 증대시키고, 시스템 개발에 적용된 보안성 분석 기법은 향후 국내 보안성 분석 기법 개발에도 기여하고자 한다.

2. 관련 연구

2.1 위협평가(threat assessment)방법

정보시스템에 대한 위협들은 공격대상인 자산과 자산에 미치는 충격의 정도가 각기 다르며 발생빈도 역시 다르다. 물론 동일한 위협일지라도 시스템의 종류나 업무의 성격에 따라 위협의 정도는 다르게 평가될 수 있다. 위협평가는 각 위협들의 발생빈도 및 위협정도를 측정하는 것으로 이 위협평가는 위협요소의 발생빈도, 충격을 주는 자산의 범위, 그리고 시스템에서 평가된 자산 가치와 위협이 발생하였을 때 입는 자산의 손상율에 위해 결정되며, 이 평가의 결과는 보안대책의 수립에 기초가 된다. 다음은 위협으로 인한 피해 정도를 산출하는 방법을 기술하였다[6].

- 발생빈도(F_j) : j번째 위협이 일정기간 동안 발생할 횟수를 예상, $1 \le i \le n$
- 성공률 (S_j) : j번째 위협이 성공할 확률, $1 \le i \le n$
- 위협대상의 가치(Vi) : i번째 위협대상이 되는 자산의 가치를 평가, $1 \le i \le k$
- 위협대상의 손상비율(Rij) : j번째 위협에 의해 입을 수 있는 i번째 위협대상의 손상비율 j번째 위협에 대한 평가는 다음의 식으로 산출될 수 있다. 여기에서 T_j는 일정기간동안 j번째 위협에 의한 자산피해액을 나타낸다.

$$T_j = \sum_{i=1}^{k} F_j * S_j * V_j * R_{ij}$$

그리고 일정기간 동안 모든 위협들에 의해 입을 수 있는 위협대상의 피해액(T)은 다음과 같이 나타낼 수 있다.

$$T = \sum_{j=1}^{n} T_j$$

위에서 평가된 각 위협요소들의 평가 값 들은 정보시스템의 보안정책 결정자로 하여금 적절한 보안대책을 수립할 수 있게 해줄 것이다.

2.2 취약성의 평가 방법

취약성은 앞의 위협평가 모형에서 보는 바와 같이 자산, 위협, 위협에 따른 영향 등과 밀접한 관련을 갖는다. 따라서 취약성 평가 시 위험분석의 타 평가와 연관성이 고려되어져야하며, 양적 및 질적 평가가 동시에 고려되어야 한다. 취약성의 평가는 취약성의 정의에 따라 그 방법이 달라 질 수

있다.

2.3 보안 분석 자동화 시스템 개발의 필요성

2.3.1 정량분석의 문제점

정량분석(Quantitative Analysis)의 가장 큰 장점이자 단점은 자산에 가해지는 위협에 대한 영향을 정량화(수량화) 시키는 것이다. 수량화된 분석결과는 위험이 직·간접적으로 특정 자산에 영향을 미치는 정도를 금전적으로 나타내주기 때문에 위협 발생 시 피해정도를 산술적으로 산출할 수 있는 장점이 있다. 그러나 유형 자산의 정량화는 어느 정도 가능하나 무형자산의 정량화는 많은 통계 데이터와 경험을 요구한다. 특히 전산 데이터와 같은 자산의 경우 정확한 정량화 수치를 구하는 것이 무척 어렵다. 정량화가 어려운 자산의 경우 분석이 용이하지 않는 단점이 있다. 또한 연간 기대손실치의 산출과정이 대부분 미국 국립표준기술연구원(NIST)의 FIBS-65에 근간을 두고 있기 때문에 산출결과가 국내환경에 맞지 않는 어려움이 있다[5].

2.3.2 정성분석의 문제점

정성분석(Qualitative Analysis)은 위와 같은 정량분석의 문제점을 해결하고자 하는 움직임에서 발전되었다. 일부 학자들은 정보 시스템 위험분석에 있어서 정량화를 시도하는 것이 매우 어렵다는 의견을 내놓게 되었다. 금융기관(은행, 증권회사, 투자회사)등과 같은 곳에서 금융자산을 대상으로 위험분석을 적용하였을 경우 대상자산은 이미 정량화되어 있기 때문에 정량분석이 적절하나 정보시스템의 경우 네트워크, 전자정보(Electrical Data)등은 이의 적용이 적절치 못하다. 따라서 정성분석은 위험을 수량화되지 않은 기술변수(예를 들면 상/중/하)로 나타냄으로서 수량화가 가져오는 오차를 줄이고 분석 과정에서 전문가의 의견을 최대로 반영할 수 있는 장점이 있다. 그러나 전문가의 주관적 판단이 지나치게 작용할 우려가 높을 뿐만 아니라 위험관리에 있어서 중요한 기능인 보안계획의 수립과 대응책 구현을 위한 비용효과 분석이 용이하지 않은 점은 단점이라고 볼 수 있다[5, 6].

2.3.3 위험분석 소프트웨어 개발의 필요성

구미와 달리 국내에서는 위험분석 자동화 도구의 개발 및 활용이 거의 전무하고 자체적인 위험분석 기법도 개발된 바 없다. 이는 정보보호에 있어서 가장 중요한 기능인 보안 관리의 체계가 이루어지지 않았다는 반증이기도 하다. 국내에서 외국 컨설팅 업체에 막대한 기술료를 지불하고 얻는 보안 컨설팅의 결과는 일시적으로 사용될 수 있을 뿐 장기적이지 못하다. 뿐만 아니라 국가 주요 공공기관의 위험분석 및 관리를 외국 컨설팅 업체에 맡긴다는 것은 더더욱 있을 수 없는 일이다.

외국 위험분석 소프트웨어를 자체 분석에 사용한다 하더라도 국내 환경과 일치하지 않는 외국도구의 검증 없는 사용은 분석결과의 신뢰를 떨어뜨릴 수 있다. 따라서 국내 위험분석 기술과 보안컨설팅 산업을 활성화하기 위해서는 표준화된 한국 위험분석 기법과 이를 바탕으로 한 자동화 위험분석 도구가 필수적으로 개발되어야 할 것이다[2].

3. 시스템 요구 분석 및 설계

3.1 시스템 요구사항

ISO/IEC JTC1/SC27에서 '정보기술보안관리지침(GMITS : Guidelines for the Management of IT Security)'으로 위험분석에 관한 표준화 보고서를 근거로 위험분석절차가 포함된 위험관리과정의 요구사항은 다음과 같다[3].

가. 자산 분류 및 평가 : 보호해야 할 전산자원들을 식별하고, 체계적인 분류를 해서, 소유하고 있

는 자산들의 가치를 평가하는 기본적인 단계이다. 여기서 자산이란 하드웨어, 소프트웨어, 데이터/데이터베이스, 사용자/전산요원, 시스템 관련문서, 전산자료 저장매체, 통신망 및 관련 장비, 등을 말한다.

나. 위협평가 : 위협은 자산에 해를 줄 수 있는 위험의 원천이다. 이와 같은 위협을 식별하고 분류해서, 발생빈도와 손실크기를 측정하는 것을 말한다.

다. 취약성 평가 : 취약성이란 정보시스템에 손해를 끼치는 원인이 될 수 있는 조직, 절차, 인력관리, 행정, 하드웨어와 소프트웨어의 약점을 뜻한다. 이와 같은 약점을 확인하고 분류하여 위협을 감소시키는 것이 취약성을 평가하는 목적이다.

라. 위험 측정 : 자산에 대한 손실을 분석하는 과정으로서, 위협의 발생확률과 손실크기를 곱해서 기대손실을 가능하면 계량적으로 계산한다. 손실크기를 화폐가치로 계산할 수 없으면, 정성적인 위험분석법을 이용한다.

마. 보안대책 선택 : 평가된 위협요소와 취약요소에 대해서 보안대책을 선택하는 단계이며, 여기서 선택해서 추진하는 비용까지 계산해야 한다.

바. 비용효과 분석 : 발생 가능한 위험요소에 대응해서 보안대책을 수립했을 경우 감소되는 위험수준을 화폐가치로 측정/평가한다.

3.2 시스템 설계

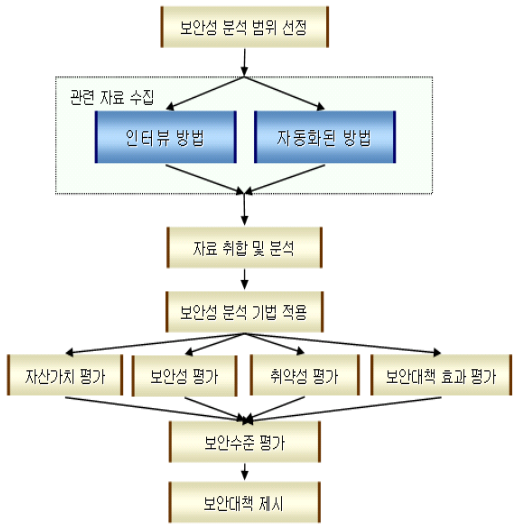
3.2.1 자동화 방법

위험분석의 대상이 되는 기업전산망의 규모가 커지고 복잡해짐에 따라 인터뷰 방법에 의한 위험관련 자료수집은 많은 시간과 노력을 필요로 한다. 그래서 정보시스템에 대한 위험관련 자료수집 시 자동검색을 통한 자산 식별, 취약성 식별, 그리고 보안대책 식별이 필요하다.

가. 자동화 자산 식별 : 정보시스템이 가진 자산을 기반프로그램 응용프로그램 개발프로그램, 데이터베이스, 문서, 일반데이터 등으로 분류하여 그 각각에 대하여 자동검색을 이용 자산을 식별하고 자산평가에 필요한 정보를 수집한다.

나. 자동화 취약성 식별 : 정보시스템이 가진 취약성 항목들을 분류하고 각 취약성 항목별로 자동검색할 세부 취약성 항목들을 설정한다. 이들 세부 취약성 항목들에 대해서 자동 취약성 검사를 이용 취약성을 식별하고 취약성 평가에 필요한 정보를 수집한다.

다. 보안대책 식별 : 이용 가능한 보안대책 항목들을 분류하고, 이들 각 보안대책의 설치 여부와 효과 평가에 필요한 정보를 자동 검색을 통하여 수집한다. 즉, 이들 보안대책들이 보안 정책에 따라 정확히 구현되었는지를 자동 검색을 통하여 검사한다.



(그림 1) 보안성 분석 체계

이러한 자동화된 방법을 통하여 위험분석에 필요한 많은 정보들을 적은 시간과 노력으로 수집할 수 있는 장점이 있지만 자동검색을 통하여 수집할 수 있는 자료가 제한적이기 때문에 보다 정확하고 다양한 자료 수집을 위해서는 (그림 1)과

같이 인터뷰 방식을 병행해서 사용하는 것이 바람직하다[6].

3.2.2 시스템 모듈 설계

시스템 개발의 효율성과 기능 확장성 확보를 위하여 접근통제, 네트워크정책, 인증정책, 서비스정책 모듈로 나누어 <표 1>~<표 4>와 같이 분야별 보안 정책과 점검 내용으로 모듈별 평가될 수 있도록 한다.

〈표 1〉 접근통제 보안 정책에 따른 점검사항

보 안 정 책	점 검 항 목
• 허가되지 않은 외부 접속 불허 • 접속요구에 대한 기록 유지	◦ 외부접속자 root 로그인 불허 여부 ◦ 3회 이상 접속실패시 접속차단 및 관련정보기록 여부 ◦ 접근통제 도구 설치 및 운영 여부 ◦ 다이얼업 모뎀으로의 접근통제 여부 - 모뎀 사용자 제한 여부, 역호출 모뎀 사용 여부 - 모뎀의 분리사용 여부(수신/호출)
• 불필요한 인터넷 서비스 차단 • “r” 서비스 제한 • 승인되지 않은 E-MAIL relay 거부	◦ 보안이 취약한 서비스의 안전한 설치 및 불필요한 서비스 차단 여부 - sendmail의 최근버전으로 패치 여부 - tftp 서비스 제공 차단 여부 - 해킹시 이용될 수 있는 서비스의 제공 차단 여부 · finger, ping, rsh, rpc, rexec - ftp 서비스의 안전한 설치 - .netrc, .rhost, /etc/host/equive 파일의 사용제약 여부 ◦ 전자우편 보안 여부 - /etc/aliases에서 “decode” 제거 여부 - sendmail.cf에서 “wizard”, “debug” 사용금지 여부 ◦ 최근의 보안취약성 자료에 대한 조치 여부

〈표 2〉 네트워크 관리 보안 정책에 따른 점검사항

보 안 정 책	점 검 항 목
• “r” 서비스 거부 • 불필요한 인터넷 서비스 거부 • portmap관련 서비스 거부 • 허가되지 않은 E-MAIL 중계 거부(SPAM 차단) • 가상터미널에서의 관리자 권한 획득 차단 • E-MAIL 첨부 파일 악성코드 검사 • EXPN과 VRFY 기능억제	◦ /etc/inetd.conf 파일 권한 모드가 644 이며 소유주가 root인가? tftp, finger 서비스를 금지하고 있는가? “r” 시리즈 명령어(rlogin, rsh)들을 금지 하고있는가? rexd 서비스를 금지하고 있는가? uucp 서비스를 금지하고 있는가? 외부의 접근제어를 하고 있는가? ◦ “r” 명령관련 파일 /etc/hosts.equiv 파일을 없앴는가? $HOME/.rhosts 파일을 없앴는가? ◦ NFS/NIS /etc/exports나 /etc/dfs/dfstab에 꼭 필요한것만 export 하였는가? /etc/exports의 접근권한은 644 이며, 소유주는 root인가 exports 파일이 “localhost” 엔트리를 가지지 않았는지? exports 리스트가 256문자를 넘지 않도록 했는가? /etc/netgroup에 허용된 사용자 및 호스트 이름만 기재 되었는가? ◦ /etc/services 파일의 접근 권한이 644, 소유주는 root인가 ? 불필요한 서비스(tfpt, finger, login)는 없앴는가? ○ /etc/hosts.lpd 파일의 첫 문자가 ‘-’가 있는가? 파일의 접근권한은 600, 소유주는 root로 되어있는가? 파일내에 “!”나 “#”이 사용되고 있는가? ◦ Sendmail 최신의 sendmail(현재 9.1.x) patch를 하였는가? /etc/sendmail.cf 파일의 로그수준은 “9” 수준인가? /etc/sendmail.cf 파일에서 wizard, debug 명령어 사용을 금지하고 있는가? ◦ 기타 Console을 제외하고 root 로그인을 막았는가? /etc/ttytab 또는 /etc/default/login /etc/aliases 파일내 “decode” alias를 없앴는가?

〈표 3〉 사용자 인증 보안정책에 따른 점검사항

보 안 정 책	점 검 항 목
• 패스워드 파일/루트접근 • Vendor 계정 제거여부 • 추정 가능한 패스 워드 확인	◦ 불필요한 사용자 계정은 제거했는가? 패스워드 없는 계정, 특별 계정들은 없는가? root와 같은 UID를 갖는 계정은 없는가? ftp계정을 막았는가? UUCP 계정을 없앴는가? /etc/ftpusers 에 root 및 특수계정이 포함되어있는가? 실행하고자 하는 shell이 맞는가? 간단한 패스워드 검출이 되는가?(Crack 이용)
• 디렉토리 퍼미션 검사 • SUID/SGID 검사 • FTP에 Vendor 계정접속 거부	◦ 파일 접근 권한 모드 /etc/utmp, /etc/motd 모드는 644인가? /etc, /usr/etc, /bin, /usr/bin, /sbin, /usr/sbin, /tmp, /var/tmp는 root 소유 인가? /tmp와 /var/tmp는 스티키비트(sticky-bit)가 지정되어 있는가? ◦ /usr/lib/expreserve /usr/lib/expreserve이 SUID로 설정돼 있거나 1993년 7월 이전 버전인가? ◦ 기타 · netrc 파일을 없앴는가? · exrc, .forward 파일에 불필요한 내용은 없는가? 불필요하게 SUID, SGID로 설정된 파일은 없는가? 소유주가 없는 파일은 있는가? /etc/ftpusers에 root, uucp, daemon, news, nobody 특수 계정을 추가하여 특정 사용자에 대한 ftpd접근을 막고 있는가?
• 쉘 획득 거부 • passwd, group 접근 거부 • forwarding 거부	◦ 익명 FTP /etc/passwd엔트리에 ftp : * : UID : GID : Anony mous FTP : /home/ftp:/bin/false 지정 여부? ~ftp/etc/passwd가 실제 /etc/ passwd의 복사본을 갖고 있는가? ~ftp/etc/group에 실제 /etc/group의 복사본을 갖고 있는가? ~ftp/.rhost와 ~ftp/.forward가 있는가?

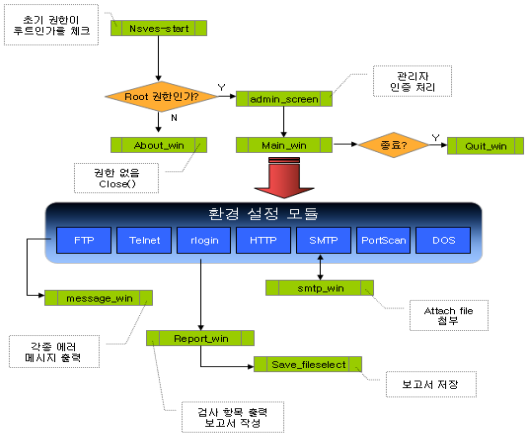
〈표 4〉 익명 FTP 보안 정책에 따른 점검사항

보 안 정 책	점 검 항 목
• 설정파일의 접근 거부 • 실행파일의 목록 표시 거부 • 공유 데이터의 접근 허용 • 익명 디렉토리 변경 허용	◦ 접근권한 ~ftp/etc/*의 모든 파일의 모드는 444이며, 소유주는 root로 되어있는가? ~ftp/etc/와 ~ftp/bin의 서브 디렉토리의 모든 파일은 111 모드와 root 소유인가? /usr/spool/mail/ftp의 소유주는 root, 모드는 400으로 되어있는가? ~ftp/pub의 모드가 1777이며, 소유주/그룹은 ftp로 되어있는가? ~ftp 디렉토리의 모드는 555이며, 소유주는 root인가?

4. 시스템 구현

4.1 시스템 구성

시스템은 (그림 2)와 같이 그래픽 사용자 인터페이스(GUI), 평가 기능 관리 모듈, 보안 시스템 정보 관리 모듈, 취약성 정보 데이터 베이스, 그리고 평가 보고서 모듈로 구성하고 초기 설정을 위해서 <표 5>와 같은 기능을 두어 시스템 사용자와 시스템간의 초기화 작업을 수행하도록 한다.



(그림 2) 시스템 모듈 구성도

〈표 5〉 시스템 설정 함수

함 수	기 능	반환값
int ftp_check(char * target, char * conn, char * timeout, char * userid, char * passwd, char * port)	- 사용자 로그인 검사 - Session Time out 검사 - ftp_expore_dir 함수 호출	입력이 부족하면 0을 반환
void ftp_expore_dir(int soc, struct ftp_dirs * dirs, int write)	- 디렉토리 권한 검사	
int ftp_get_pasv_address(int soc, struct sockaddr_in * addr)	디렉토리 위치 변경이 가능한지 확인	error가 발생하면 1을 반환
int ftp_login_check(const struct hostent * hostname, int port, char * username, char *passwd, int conn)	- 사용자 로그인 수 검사 - ftp_login 함수 호출	접속가능한 로그인 수를 반환
int ftp_timeout_check (const struct hostent * hostname, int port, char * username, char * passwd, int timeout)	- Session Time out 시간을 검사 - 연결시간과 종료시간을 비교	Sesson Timeout 시간 반환
int ftp_login(int soc, char * username, char *passwd)	- ftp서버에 로그인	

4.2 시스템 서비스 평가 모듈 구현

4.2.1 TELNET 서비스 평가 모듈

서버는 클라이언트와 서버간의 대화를 주도하며, 네트워크 가상 터미널에 서비스들을 제공한다. 클라이언트는 로컬 호스트에서 사용자 명령을 받아들여서 이것들을 네트워크 가상 터미널 명령들로 변환한다. 또한, 클라이언트는 서버가 네트워크 가상 터미널에 제공하는 응답을 받아들여서 그것들을 실제 로컬 호스트가 받아들일 수 있는 응답으로 변환하여 내보낸다.

이를 이용하여 <표 6>에 표시된 기능들처럼 원격호스트에 텔넷으로 접속하여 최대 접속시간을 평가하고, 텔넷으로 접속할 수 있는 최대 세션 수를 평가한다.

〈표 6〉 Telnet 서비스 평가 함수

함 수	기 능	반 환 값
int check_telnet (char * host, char * conn, char * tout, char * uid, char * passwd, char *port)	- telnet 기능을 체크함	입력이 부족하면 0을 반환하고 함수 종료
int check_telnet_session (char * hostname, char * port, char * conn, char * uid, char * passwd)	- 사용자 로그인 수 검사	접속가능한 로그인 수를 반환
int check_telnet_timeout (char * host, char * port, char * uid, char * pass, char * tout)	- Session Timeout 시간을 검사 - 연결시간과 종료시간을 비교	Session Timeout 시간 반환
int telnet_login (char * target, char * port, char * uid, char * passwd)	- telnet 서버에 로그인	error 발생시 -2를 종료시 1을 반환
int pty_open (char * command, int * childpid)	- 쉘 열기 - get_master_pty( ) 함수 호출	error가 발생하면 -1을 반환
int get_master_pty( )	- 터미널 쉘 획득	error 발생시 -1 반환
int telnet(char * host, char * port)	- 원격로그인 명령 실행	획득한 터미널 정보 반환
void telnet_receive_string (char * buf, int bytes)	- 문자열 반환	
int get_slave_pty( )	- 새로운 쉘얻기	error가 발생하면 -1을 반환

- 사용자 인증 : 특정 정보 시스템에 대한 서비스 허용 여부를 검사한다.
- Download 명령어 허용 : 시작 호스트에서 목적 호스트에 접속 후 X, Y, Zmodem, Kermit 등 파일 전송 프로토콜을 이용한 파일의 다운로드 제한 여부 등을 점검한다.
- Upload 명령어 허용 : 시작 호스트에서 목적 호스트에 접속 후 X, Y, Zmodem, Kermit등 파일 전송 프로토콜을 이용한 파일의 업로드 제한 여부 등을 점검한다.
- Telnet 명령어 제한 : 시작 호스트에서 목적 호스트에 접속 후 Telnet 명령어 사용을 제한하는지 여부를 점검한다.

3.4.3 RLOGIN 서비스 평가 모듈

rfc1282는 RLOGIN 프로토콜 사양을 정의하고 있는데, RLOGIN은 클라이언트와 서버 사이에 하나의 TCP 연결을 사용한다. 정상적으로 TCP연결 설정이 완료된 후 다음의 응용 프로토콜이 클라이언트와 서버 사이에 생긴다. RLOGIN 평가 모듈은 <표 7>의 기능들처럼 원격 호스트에 RLOGIN으로 접속하여 최대 연결 지속 시간을 평가하고 접속할 수 있는 최대 세션 수를 평가한다.

- 사용자 인증 : 특정 정보 시스템에 대한 서비스 허용 여부를 점검한다.
- 세션 연결 개수 제한 : 목적 호스트로의 연결 개수가 설정 개수 이상 연결 요청 시 제한 하는가를 점검한다.
- 세션 타임아웃 : RLOGIN을 이용하여 접속 후 아무 입력 없이 관리자에 의해서 설정된 시간이 경과되면 접속이 종료되는지를 점검한다.
- RLOGIN 명령어 제한 : 시작 호스트에서 목적 호스트에 접속 후 RLOGIN 명령어를 제한하는지 여부를 점검한다.

〈표 7〉 RLOGIN 서비스 평가 함수

함 수	기 능	반 환 값
int check_rlogin(char * host, char * conn, char * tout, char * uid, char * passwd, char * port)	- rlogin test 수행 - 호스트이름, 접속가능 숫자, 타임아웃 시간, 포트번호을 입력받아 처리	입력이 부족하면 0을 반환하고 함수 종료
int check_rlogin_session(char * host, char * conn, char * uid, char * passwd)	- rlogin 사용자 연결 세션 갯수를 검사한다.	사용자 연결된 세션 갯수를 반환
int check_rlogin_timeout(char * host, char * tout, char * uid, char * passwd, char * port)	- Session Timeout 시간을 검사 - 연결시간과 종료시간을 비교	Session Timeout 시간 반환
int rlogin_login(char * host, char * uid, char * passwd)	- rlogin 서버에 로그인	error 발생시 음수반환 함수 종료시 1을 반환
int r_pty_open(char * cmd, int * cpid)	- 터미널 화일 열기 - 쉘을 획득한다.	획득한 터미널 기술자 반환
int r_get_master_pty (void)	- 터미널 파일 (마스터) 열기	성공시 파일 기술자 반환 에러시 -1리턴
int r_get_slave_pty (void)	- 터미널 파일 (슬래브) 열기	성공시 파일 기술자 반환 에러시 -1 리턴

4.2.2 HTTP 서비스 평가 모듈

rfc2068은 HTTP 프로토콜 사양을 정의하고 있는데, HTTP는 클라이언트와 서버 사이에 TCP/IP 연결이 선정된 후 클라이언트가 보내는 각 HTTP 목적 URL의 다음에 오는 메쏘드로 시작한다.

HTTP 메쏘드는 클라이언트가 서버가 요청하는 목적을 지정하기 위해 사용하는 명령이다. 모든 HTTP 메쏘드는 URL에 의해 확인된 리소스에 상응한다. HTTP평가 모듈에서는 <표 8>의 기능처럼 응답 메시지 가운데 세 자리 상태 코드로서 서버의 리소스 상태를 파악하게 된다. HTTP 모듈에서는 이를 이용하여 URL필터링을 테스트하고, java

코드, Active-X를 목적 호스트로 전송하여 필터링이 되는지를 평가하게 된다.

〈표 8〉 HTTP 서비스 평가 함수

함 수	기 능	반 환 값
int http_connect (unsigned long proxy, char * url, char * port)	- 서버에 연결, http text 수행	에러시 0 성공시 양의수
void check_http (char * proxy, char * target, char * port)	- http서버와 연결 - 파일 필터링에 대한 검사 - JAVA 코드에 대한 필터링 검사 - Active-X에 대한 필터링 검사 - 특정 확장자에 대한 필터링 검사	

4.2.3 SMTP 서비스 평가모듈

FTP와 마찬가지로 SMTP 프록시 평가 방법은 rfc0821에 따라 클라이언트가 서버로 명령어를 보내게 되면 서버는 숫자로 표현된 응답코드와 선택적으로 사용자가 읽을 수 있는 문자열을 보내어 응답하게 된다.

이 모듈에서는 <표 9>의 기능들처럼 목적 호스트에 접속한 후 최대 세션 연결 수와 최대 지속시간을 평가하고 Sendmail명령을 이용하여 메시지를 전송 및 테스트를 수행한다.

- 첨부 파일 제한 : 내·외부망 사용자가 메일 전송 시 첨부한 파일이 통제되는지를 점검한다.
- 메일 내용 제한 : 관리자에 의해서 설정된 단어가 포함된 메일의 송수신이 정확하게 통제되는지를 점검한다.
- 특정 호스트 메일 송수신 개수 제한 : 메일 관리자에게 설정된 호스트에서 송수신되는 메일이 설정된 메일 개수/시간이 초과되었을 때 통제되는지를 점검한다.

〈표 9〉 SMTP 서비스 평가 함수

함 수	기 능	반 환 값
int check_smtp (struct _smtp * se)	- 전체 함수 제어 - 연결 측 프락시 서버에 sendmail 명령어 전송 및 테스트 - 결과 저장	정상적인 처리시 1 반환, 비정상 처리시 0 반환
int check_smtp_session (struct _smtp * se)	- count 수에 따른 서버측에 연속적인 접속 및 테스트	연결 완료 수
int smtp_connect (int soc, struct _smtp * se)	- 소켓 개방 및 select 함수를 이용하여 준비된 읽기 파일 지정자 체크.	파일 지정자 획득시 1 반환 비 획 득 시 0 반환
int check_smtp_timeout (struct _smtp * se)	- timeout에 따른 일정한 시간동안 프락시 서버와의 메시지 전달 단절 - 경과 시간 보고.	경과시간
unsigned int file_attach (int soc, char * fname)	- 파일을 서버측 연결 소켓을 통하여 전송.	전송한 파일 크기
void send_message (int soc)	- 서버측에 메시지 내용 전달.	
void send_quit (int sock)	- 서버측에 QUIT 문자 전달	
char * make_msg_id (char * target)	- 메시지 ID 생성	생성된 메시지 ID 포인터
int chk_localhost (char * host)	- 서버측 주소를 네트워크 주소체계로 변환.	주소가 정확하면 16진 코드 주소를 반환
int chk_digit_only (char * p)	- 10진 주소 체계인지 판별	참이면 1을 반환 거짓이면 0 반환
char * get_host_fqdn (char * host)	- 호스트 이름을 도메인 주소 구조체에 저장.	주소 구조체속의 호스트의 공식이름 및 알리아스 이름 반환.
char *basename (const char *pname)	- 파일 경로에서 루트 디렉토리 경로 표시를 삭제	경로 값

4.2.4 PORT SCAN 서비스 평가 모듈

서버는 일반적으로 잘 알려진(well know) 포트 번호를 이식한다. 예를 들어, 모든 TCP/IP 구현에서 FTP 서버의 서비스는 TCP포트 21에서 제공한다. 또, Telnet 서버의 포트 번호는 TCP포트 23, 모든 TFTP(Trivial File Transfer Protocol)구현은 UDP 포트 69에서 제공하는 것으로 되어있다. 또한 SYN Flooding 공격을 감지하고 차단하는지를 점검한다.

4.2.5 SYN FLOODING 서비스 평가 모듈

SYN FLOODING 공격은 클라이언트가 마지막 ACK를 보내지 않고 연속적으로 SYN 요청 신호만을 보냄으로써 Half-Open상태에서 큐(queue)의 부족으로 인해 서버의 시스템이 오작동 하도록 만드는데 이를 평가하도록 한다.

4.2.6 SMURFING 서비스 평가 모듈

SMURFING은 공격자가 ICMP echo request 패킷을 보내면 그 응답으로 ICMP echo reply 패킷을 보낸다. 이것을 이용하여 공격자가 echo request 패킷을 어떤 네트워크의 브로드캐스트 주소로 여러 차례 보내면 그 패킷을 받은 네트워크의 모든 호스트가 request 패킷을 응답하여 패킷을 보낸 호스트로 일시에 ICMP echo reply 패킷을 보내게 된다. 여기서 만약 패킷을 출발지 주소를 공격하고자 하는 서버 주소로 보내면 그 서버는 다운이 되게 된다. 이와 같은 공격을 강도를 평가한다.

5. 결 론

일반적으로 보안정책 구현을 위한 보안제품이 알려지지 않은 보안 취약점을 가지고 있을 경우 이러한 시스템을 사용하는 조직은 중요한 정보를 안전하게 보호하지 못하고 누출, 파괴, 위·변조와 같은 예상치 못한 보안문제가 발생할 수 있다. 특히 각종 네트워크를 통하여 정보가 교환되는 상황에서 취약점이 있는 정보보호시스템을 사용하는 것은 국가 사회적으로 심각한 문제가 야기될 우려가 있으며, 국가 공공망은 더욱 이러한 문제점에 대해서 철저한 보안관리가 있어야 할 것이다.

본 논문은 기존 보안성 평가방법을 고찰하고, 그 결과를 토대로 정보보호시스템 안전성 유지를 위한 자동화 보안 분석 시스템을 설계 구현하였다. 이는 향후 다양한 보안 분석 기준을 마련하는데 도움이 될 것이며, 국내 정보보호 산업과 정보시스템의 보안관리 체계를 수립하는데 기여하게 될 것이다.

참 고 문 헌

[1] Network and Distributed Systems Management, 1994, Moris Sloman.
[2] BSI, BS 7799, Code of practice for Information Security Management, 1997
[3] ISO/IEC JTC1/SC27, TR 13335-1, Guidelines for the Management of IT Security (GMITS) : Part 1 - Concepts and Models for IT Security, 1996
[4] IETF RFC 1244, Site Security handbook, 1995.
[5] 한국전산원, 인터넷/유닉스 관리자를 위한 기술 지침서, 1995.
[6] 한국전산원, 인터넷 보안지침서, 1995.
[7] NCA Ⅲ-RER-9557 통합망 관리 표준화 연구, 1995, 한국전산원.
[8] NCA Ⅴ-RER-95108 초고속국가정보통신망 구축 세부 추진방안 연구, 1995, 한국전산원.
[9] 국방전산통신망 관리지침, 1996, 국방부.
[10] 정보보호뉴스, 2002, 정보보호진흥원

김 점 구

광운대학교 전자계산학과 이학사
광운대학교 전자계산학과 이학석사
한남대학교 컴퓨터공학과 공학박사
(주)제성프로젝트 연구원
(주)시사컴퓨토피아 인터넷사업 본부장
현재 남서울대학교 컴퓨터학과 교수
관심분야 : 정보보호, 컴퓨터 네트워크, 무선통신

김 태 은

중앙대학교 전기공학과 공학사
중앙대학교 전자공학과 공학석사
중앙대학교 전자공학과 공학박사
한국재단참여연구원
삼성전자 휴먼테크 논문대상 은상 수상
현재 남서울대학교 멀티미디어학과 교수
관심분야 : 멀티미디어시스템, 영상인식, 증강현실, 웹 3D처리기술