2003 정보 및 제어 학술회의 (2003.11.21-22)

UNIX상에서 Shell 스크립트를 이용한 보안 시스템 구현

Implementing of Security System using Shell Script in UNIX

이 민 교
(Min Gyo Lee)

경북대학교 대학원 정보보호학과(전화:(053)940-8657, E-mail : freesem@hanmail.net)

Abstract : In this paper, I have implemented the security system using shell script that periodically checking the security elements of unix system for security, it transmit to a monitoring server and monitoring many clients. Agent of cleint executing by crontab scheduler, Environment of server to receive data use restricted TFTP in SunOS. And then, Because of using shell script, apply shell to system with flexible, control performance, and can meet on a sudden situation.

Keywords : UNIX, SHELL, SECURITY

Ⅰ. 서론

인터넷이 지역별 기술 차이를 갖고 빠른 속도와 광범한 범위로 확산되고 있는 정보화 시대에서 보안은 중요한 이슈로 부각되고 있다. 저장 및 전송 등의 모든 단계서의 보안은 전산화로 인한 비용절감, 관리의 효율성, 정확한 의사결정, 기업이미지 제고, 경쟁력 확보, 개인 생활의 안전 및 편의를 제공하는 것이다.

다양하고 많은 수의 컴퓨터가 상호 연동하는 네트워크를 통해 인터넷에 연결되어 시간과 공간을 초월하여 전 세계의 모든 사용자에게 항상 개방되어 있고 접근이 가능한 상태이며, 이로 인해 하나의 보안사고는 사고 시스템은 물론 동시에 전 세계 인터넷으로 연결된 시스템에 빠른 속도로 전파되어, 파생되는 영향력은 매우 크다고 할 수 있다. 또한 각 기관에 따라 보안정책, 시스템 운영 및 관리기술등의 보유 격차가 크므로 정보보호를 위해 투자비용과 직·간접적인 피해 정도에 따른 적절한 균형을 유지시키고, 최선의 보안대책을 수립하고 이에 따른 가이드라인이 있어야 하며, 사용자의 보안교육 및 보안전문가의 육성과 관리 및 법령상의 문제점과 대비책이 필요하다.

이를 위해 비인가자, 불법사용자로부터 자료의 유출방지, 변경, 삭제, 생성, 파괴 등으로부터 자료를 보호하여 무결성 유지, 적시적소에서의 정보 접근이 가능한 상태 유지하여 가용성을 확보하고, 조직 내부 혹은 전산시스템을 사용하는 환경 등에 내재된 약점(weakness)이나 위협에 의해서 자산이나 조직의 업무환경에 피해를 가할 수 있는 요소를 미리 제거하고, 시스템에 수립된 사용정책 및 보안정책을 준수하여 안전하게 운용되고 있는지를 확인하기 위한 전산시스템 내에 기록(log) 및 저장되어 있는 각종 사용자 행위에 대한 상세 내용을 조사 분석하는 감사가 있어야 하고, 인가된 사용자, 프로그램, 프로세스, 시스템 등의 주체만이 전산시스템의 자원에 접근할 수 있도록 제한해야 한다.

따라서 본 보고서는 인터넷에서 서버로 많이 사용되고 있는 Unix 시스템의 보안을 위해 주기적으로 보안요소를 점검하고, 이를 전송시켜 한 곳에서 여러 대의 서버를 모니터링 할 수 있는 시스템을 쉘 스크립트로 구현하였다.

Ⅱ. UNIX 시스템의 보안요소

1. Patch

Unix 시스템은 개방된 개발자 환경과 인터넷으로 인해 많은 운영 및 보안상 허점이 내재되어 있으므로 밴더로부터 패치 목록을 구한 다음 시스템에 패치 및 Upgrade 하여 운영, 성능 및 보안 등의 허점을 보완해야 한다.

2. 계정

시스템에 해커가 침입할 때는, 네트워크로 직접 루트 권한을 얻거나, 일단 시스템의 일반 계정을 얻어낸 후 버그를 이용하여 루트 권한을 얻는 두 가지 방법이 있다. 보통의 나중의 방법으로 공격이 많이 이루어지나 처음부터 공격의 가능성을 줄이는 방법은 계정의 노출을 방지하는 것이다.

3. NETWORK

r 명령은 한 사용자가 여러 호스트를 사용 할 경우, 각 각의 시스템에 접근하기 위해 매번 로그인 명과 패스워드를 입력해야하는 불편을 덜어 주기위해 사용하는데, 기본적으로 신뢰된 시스템에 한하여 이용하는 서비스 방식이다. r 명령으로는 rsh, rlogin, rcp 등이 있으며 보안상의 문제점을 안고 있기 때문에 가능한 사용하지 않는 것이 좋다. 이를 사용하기 위해서는

$HOME/.rhosts 나 /etc/hosts.equiv 파일의 설정이 되 있어야 한다.

UUCP는 Unix-to-Unix Copy를 줄인 말로서, 보통 리모트 프린팅이나 application으로써 전자우편이 통용되고 있었다. UUCP는 대역폭이 적다는 단점이 있지만 기본적으로 하나의 호스트에서 다 른 호스트로 파일을 복사한고 그것은 또한 리모트 호스트상에서 확실하게 작업을 수행시킬 수 있다. 또한 guest 사용자로 로긴해서, 그 사이트로 접근할 수 있으며, 공개적으로 접근 가능한 아카이브 지역에서 파일들을 전송받을 수도 있다.

Inetd는 Client로부터 서비스 사용 요청될 때 통합된 daemon으로 일단 먼저 요청을 받은 뒤 해당 daemon을 실행시켜 주는 것이다. 이렇게 실행하기 위한 inetd의 환경 설정파일이 /etc/inetd.conf 이다.

FTP는 파일서버와 같이 대용량의 저장 공간을 가진 서버에 응용 프로그램, 소스 코드, 문서 등의 많은 정보를 담고 있어서 인터넷 사용자가 그 서버에 접근하여 원하는 자료를 다운로드 하거나 반대로 업로드 할 수 있는 서비스를 제공해 주는 서비스라 할 수 있다. Anonymous FTP는 사용자의 제한을 주지 않으므로 특별히 주의하여야 한다.

4. File System

NFS는 네트워크를 통하여 서로 다른 컴퓨터들이 파일 시스템들을 공유할 수 있도록 한다. NFS를 사용할 때 일반적으로 마운트된 파일들을 관리해주는 NFS 서버의 보안을 전적으로 신뢰하는 경향이 있다.

NIS는 유닉스 시스템을 셋업(Setup)하고 그 네트워크를 관리하는데 필요로하는 노력을 최소화 시키려는 목적으로 만들어졌다. 이런 주요 설정 파일들의 갱신은 이 NIS 서버 시스템에서만 이루어지면 된다. 이는 매우 강력한 네트워크 관리 도구로 알려져 있다. 그러나, 보안에 관한한은 강력하지 못하다.

Sticky bit는 일반 사용자가 패스워드를 바꾸거나 mail 보낼 때에는 root 권한이 필요한데, 일반사용자에게 는 권한이 없으므로 UNIX 시스템은 사용자가 아닌 프로그램에게 특권을 주어서 가능하도록 하였다. 프로그램이 SUID나 SGID의 속성을 가졌다면 ls -l 명령의 출력에서 실행할 수 있음을 나타내는 x 퍼미션이 s로 보인다. SGID와 sticky bit는 원래 프로그램에 적용하는 것이 원칙이지만, Berkeley UNIX, SunOS, Solaris와 다른 운영체제들은 디렉토리의 특성을 바꾸기 위해 그 bit를 사용하기도 한다.

5. 로그

로그 파일은 시스템 프로그램에 의해 시스템의 움직임을 기록하므로 해킹의 흔적과 시스템에 생길 수 있는 문제점들을 미리 파악 할 수 있는 확률이 높으므로 주기적으로 로그 파일을 정기적으로 백업하고 체크해야 한다.

III. 보안 시스템 구현

보안 시스템의 구성은 client의 에이전트가 client의 상태를 주기적으로 체크하여 로그를 생성하고 그 로그를 간추려 레코드 형대로 보안 서버로 통신을 통하여 보내고 이를 점검하여 주의를 표시하는 형태로 구성하였다. 전체 구성도는 아래 그림 1과 같이 계정, 네크워크, 파일시스템,로그 검사 순으로 구성하였다. 하지만 이를 다 적용할 경우 부피가 너무 커 주로 환경 설정부분을 점검하는 방향으로 shell 스크립트를

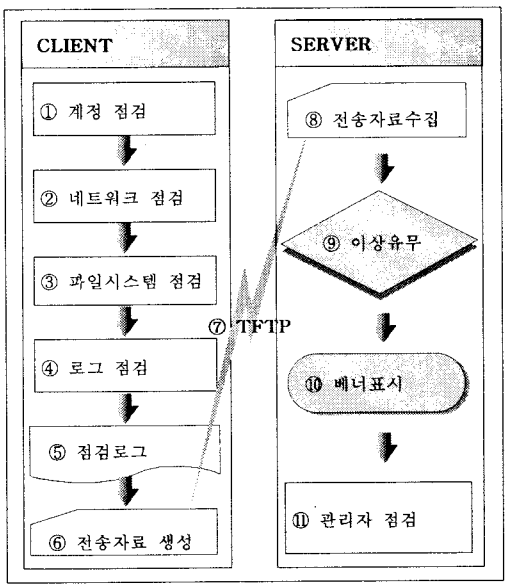

그림 1. 보안 시스템의 구성도

Fig 1. Organization diagram of security system

작성하였다. 통신은 제한된 tftp로 구성하였다.

각 client에서 실행된 에이젼트 shell 스크립트는 아래 그림 2와 같이 UNIX 시스템의 job scheduler라 할 수 있는 crontab에 등록되어 주기적으로 실행 되며, 필요시 이 스크립트 실행으로 생성된 로그를 직

```
crontab -l
00 17  * /monitor/sarmonitor.sh
00 17  * /monitor/iostatmonitor.sh
0,10,20,30,40,50 * * * * /security/ch_sec.sh
```

그림 2. crontab 예제

Fig 2. Sample of contab

접 볼 수 있다.

Client들로부터 받은 자료를 읽어서 경고성 메시지를 표현하고, 자료를 받기위해 Solaris의 제한된 tftp를 이용하였다. 그림 3은 이를 위한 그 환경설정 파일인 /etc/inetd.conf파일의 내용을 나타내고 있다.

```
/etc/inetd.conf
telnet .. /usr/sbin/in.telnetd in.telnetd
tftp   .. /usr/sbin/in.tftpd in.tftpd -s /monitor
#shell .. /usr/sbin/tcpd      in.rshd
```

그림 3. Solaris /etc/inetd.conf의 내용

Fig 3. Contents of /etc/inetd.conf in the Solaris

Ⅳ. 실행결과

Client에서 보안 체크를 실행하였을 경우, 그림 6과 같은 로그를 생성하였다. 이로그의 마지막 줄을 보면 /etc/profile의 owner나 퍼미션이 잘 못 되었음을 그림 4에서 확인 할 수 있다.

```
Passwd guest : good
Passwd shadow : good
Passwd root : good
TCP Wapper : good
netd port : good
passwd permision : good
wtmp permision : good
/.rhost : warn
showmount everyone : good
hosts equiv : good
default umask : good
path in profile : good
inetd.conf own/per : good
/etc/profile own/per : warn
```

그림 4. Client 보안체크의 로그

Fig 4. Log of security check in Cleint

서버에서는 각 client들로부터 받은 자료를 분석하여 이를 관리자에게 알리는 역할을 한다. 간단히 한 줄을 추가하여 메일 발송이 가능하나 제외시키고 단지 메시지 표현이 가능하도록 구현하였다. 아래의 그림 5의 나열된 그림은 실제 서버에서 표현된 메시지이며 현재 rhosts 파일 설정에 이상이 있음을 나타낸다.

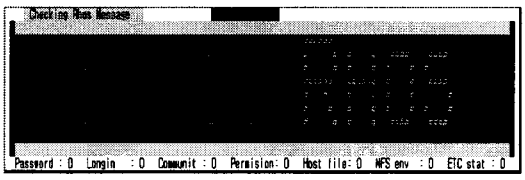

그림 5. first호스트의 .rhost 파일 비정상 메시지

Fig 5. No good message of the .rhost file in the first host

그림 6은 현제 모니터링 하고 있는 전체 client에 대한 통계를 하나의 화면으로 구성하여 보안 상 문제 있는 호스트와 서비스를 항목별로 나타나고 있다.

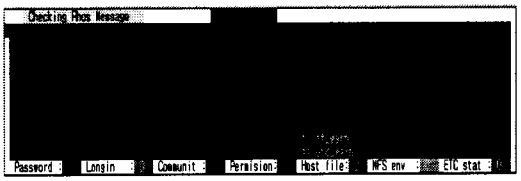

그림 6. 모든 호스트의 비정상 요약 화면

Fig 6. A summary screen of no good message in all host

Ⅴ. 결론

위에서 shell 스크립트를 이용하여 각각의 시스템에 최적화 시킬 수 있는 보안 시스템을 구현해 보았다. 이는 보다 다양한 방법의 침입에 대해 유연하게 적응할 수 있고, 시스템 수행에 관해 많은 포퍼먼스를 필요로 하는 명령을 직접 제어할 수 있다. 하지만 모든 침입에 대해 방어 할 수 없다. 이를 위해 보다 다양한 방법을 모색해야 한다. 이를테면 실제 시스템을 관리하는 데에 있어서 가장 기본적인 백업, 시스템 기본정보 유출의 최소화, 시스템 업무와 환경에 적절한 최신의 패치의 여러 보안툴을 적용하는 것이다. 본 논문은 보안을 위해 사람의 점검사항이나 기존에 있는 제품보다 즉시 또는 유연하게 대처할 수 있는 보안 시스템을 구현 한다는 데 의의가 있다고 할 수 있다.

참고문헌

[1] Matt Bishop, COMPUTER SECURITY Art and Science, PART 8: PRACTICUM, 2003 Page(s): 771 - 920

[2] Farrow, Rik, UNIX System Security, Addison Wesley, 1993.

[3] Peek, O'Reilly, and Loukids, UNIX POWER TOOL 한빛 미디어, 2000

[4] D. Brent Chapman, Elizabeth D. Zwicky, Building Internet Firewalls, O'Reilly & Associates, 1995

[5] tp://info.cert.org/pub/tech_tips/intruder_detection_checklist (CERT에서 제공하는 침입자 detection checklist)

[6] ftp://ftp.auscert.org.au/pub/auscert/papers/unix_security_checklist(UNIX Computer Security hecklist, 19-Dec-1995)