2001년 한국정보처리학회 춘계 학술발표논문집 제8권 제1호

네트워크상의 바이러스 탐지를 위한 방화벽 시스템 설계 및 구현

최준호*, 김두현*, 김판구*, 권석철**
*조선대학교 대학원 전자계산학과 멀티미디어 시스템 연구실
**(주) 하우리
e-mail : spica@hitel.net

A Design and Implementation of Firewall System for Virus Detection in Network

Jun-Ho Choi*, Doo-Hyun Kim*, Pan-Koo Kim*, Suck-chun Kwon**
*Dept. of Computer Science, Graduate School Chosun University
**Hauri Inc, Seoul Korea

요 약

인터넷 사용의 증가와 더불어 컴퓨터 바이러스의 제작기법의 지능화 및 그로 인한 피해가 날로 확산됨에 따라 시스템 사용자들의 바이러스 탐지에 대한 어려움은 증가되고 있으며, 또한 개인용 컴퓨터뿐만 아니라 서버급 컴퓨터를 겨냥한 신종 바이러스가 계속 증가함으로 이에 대한 대비책이 시급한 상황이다.

이에 본 논문에서는 네트워크 상의 바이러스 탐지를 위해 방화벽 시스템을 도입하여 네트워크 보안을 유지하며 네트워크 상에서 유입되는 파일의 바이러스 진단 및 치료의 과정을 효과적으로 수행 할 수 있는 시스템을 제안한다.

1. 서론

악성 소프트웨어란 컴퓨터 시스템의 데이터 파괴 및 시스템 정지 등의 악의적인 현상을 일으키는 프로그램들을 통틀어 일컫는 말이다. 악성 소프트웨어에는 컴퓨터 바이러스(Computer Virus), 트로이 목마(Trojan horse), 웜(Worm), 논리 폭탄(Logic Bomb), 그리고, 백 도어(Back door) 등이 있다. 1999년도부터 적지 않은 피해를 입혀온 러브레터 바이러스와 그 변종들은 최근 컴퓨터 바이러스의 대표적인 형태이다.

이런 바이러스는 인터넷 사용이 급속히 확산됨에 따라 그 피해 또한 기하급수적으로 늘어만 가고 있다. 이런 바이러스를 퇴치하기 위한 백신의 대부분은 윈도우즈 클라이언트용으로 제작, 배포되고 있지만 Linux나 Unix용 백신은 고가이면서 그 숫자 또한 극히 드문 형편이다. 또한, Linux나 Unix용 백신 소프트웨어는 단순히 파일시스템에 저장된 이후에 사용자나 관리자에 의해 검사되는 선저장-후검사 방법이므로 바이러스의 침입에 적극적으로 대응하지 못하고 있는 실정이다. 이에 본 논문은 네트워크 상의 바이러스 탐지를 위한 방화벽 시스템을 도입하여 상당 수준의 네트워크 보안을 유지하며 네트워크 상에서 유입되는 파일의 바이러스 진단 및 치료의 과정을 효과적으로 수행 할 수 있도록 하였다.

2. 바이러스 탐지를 위한 방화벽 시스템 설계

네트웍 상에서 바이러스를 차단하는 시스템의 목적은 내부 네트웍으로 들어오는 바이러스 감염 파일이 들어올 수 없도록 하는데 있다. 또한 네트웍 상에서 바이러스를 차단하기 위하여 패킷 처리하는 시간을 최소화하여야 하며, 네트웍 부하가 발생하지 않는 방향으로 설계되어야 한다.

따라서, 본 연구에서의 바이러스 탐지 및 차단 시스템은 패킷 필터링 방화벽과 어플리케이션 프록시 방화벽을 혼합한 형식의 방화벽 시스템으로 외부 Bastion host 사이에서 리눅스 머신을 사용하여 패킷 필터링이 가능한 라우터를 구성하여 외부의 호스트가 방화벽 내부의 IP로 위장하여 접근하지 못하도록 막아주는 기능을 하고, Bastion Host에 TIS FWTK(Firewall Tool Kit)을 기반으로 하는 어플리케이션 프록시 방화벽을 설치하였다. 내부 자원을 보호하기 위하여 명백히 허용되지 않는 서비스 이외에는 금지하도록 정책을 세웠으며, 허용되는 서비스는 가장 많이 사용하는 Telnet, FTP, WWW 이다. 내부의 호스트가 외부의 Telnet 또는 FTP 서비스를 사용할 경우에는 먼저 Bastion Host의 프록시 Telnet 또는 FTP Gateway에 접속을 한다. 이때 사용자 인증을 요구하게 된다. 사용자 ID와 패스워드를 입력하여 사용자 확인을 마친 후에 원하는 서비

스에 재 접속하는 방식으로 서비스를 사용할 수 있다. 따라서, FTP서비스를 통하여 파일을 전송 받게 될 경우 파일 전송 후 전송된 파일이 바이러스에 감염된 파일인지를 감시하여 이를 통보하여 주도록 수정하였다. 외부의 호스트가 내부에 호스트에 Telnet이나 FTP를 이용하려고 하는 경우에도 마찬가지로 Bastion Host에 먼저 접속을 한 후에 다시 원하는 호스트로 재 접속해야 한다.

본 논문에서는 FWTK라는 방화벽 도구를 기반으로 FTP-GW에 네트워크 상의 바이러스 탐지 및 차단을 위한 각각의 모듈 통합하여 바이러스 차단 시스템을 구현하였다. 주요 구성은 FWTK의 FTP Gateway 프로그램을 수정하여 바이러스가 감염된 파일을 전송 받을 경우 파일 전송 후 바이러스 감염 사실을 통보하여 사용자로 하여금 바이러스를 치료 후 사용하거나 삭제를 할 수 있도록 하였다.

3. 시스템의 구성 요소 설계

3.1. TIS Firewall Toolkit

TIS Firewall Toolkit은 방화벽 소프트웨어를 제작하는데 필요한 툴 패키지이다. 이것으로 구축이 가능한 방화벽 호스트는 프록시 방식의 방화벽 호스트인데 각각의 네트워크 서비스 별로 프록시를 두고 이 프록시들이 방화벽의 기능을 수행할 수 있게 되어 있다. 이 중 본 연구에서 구현하는 네트워크 상의 바이러스가 감염된 파일의 전송 시 진단하는 주체인 FTP 프록시 ftp-gw는 방화벽 호스트를 통과하는 사설 네트워크 또는, 공용 네트워크로의 ftp 트래픽을 허용하는데, telnet 프록시와 마찬가지로 방화벽으로 표준 ftp 포트를 경유하는 ftp 접근이 감지되면 프록시의 수행이 시작된다. 여기서 바이러스 감염 진단을 위한 방화벽 시스템은 ftp 서비스가 제공되어지는 시스템과 별도의 시스템에 설치하여야 한다. 이는 보다 효율적이고 안정적인 바이러스 진단을 위함이다.

3.2. 네트워크 접근 제어

TIS Firewall Toolkit 기반의 바이러스 차단 시스템의 네트워크 접근 허용 여부를 위해 netacl이라는 프로그램을 사용한다. netacl은 inetd 데몬에 의해 기동되며, 원격 사용자의 시스템으로부터의 서비스 요구를 허용하거나 거부하는 기능을 수행한다. 이에 본 연구에서는 이를 이용하여 바이러스를 유포한 사용자나 시스템의 접근을 효율적으로 제어 및 거부할 수 있게 한다. inetd.conf 파일에서 netacl을 설정하면 netacl은 오직 하나의 인수만을 취하는데 이 인수로는 시작하고자 하는 서비스의 이름이 입력된다. 또한 이외의 인수들은 netacl이 기동하는 서비스가 사용된다. inetd.conf 파일의 ftp 서비스와 관련된 부분은 다음과 같다.

```
ftp stream tcp nowait
root /usr/local/etc/netacl  /usr/sbin/in.ftpd
```

위의 경우는 ftp 서비스 접속 요청이 inetd에 의해 받아들여지게 되면 netacl 프로그램이 /usr/sbin/in.ftpd를 인수로 하여 동작을 시작하게 한다. ftpd 데몬이 시작되기 전 netacl은 해당 서비스 요구가 netperm-table 내 접속 규칙에 부합되는지를 검사하여 ftpd 데몬의 실행 여부를 판단하게 된다.

접속 요청 서비스이 수용과 거부는 syslog 데몬에 의해 다음과 같이 기록되며 이는 방화벽 시스템의 분석에 사용된다.

```
Jan 3 00:10:43 firewall netacl[339]: deny host=security.chosun.ac.kr/
203.237.110.75 service=in ftpd
Jan 3 00:13:28 firewall netacl[354]: deny host=security.chosun.ac.kr/
203.237.110.75 service=in.ftpd excute=/usr/sbin/in.ftpd
```

위 로그 레포트의 첫 번째 라인은 호스트 *security.chosun.ac.kr*이 요청한 FTP 서비스가 netacl에 의해 거부되었음을 나타내 주고 있고, 두 번째 라인은 ftp 접속 요청이 허가되었음을 나타낸다. 그러나 접속을 요청한 사용자에 대해 아무런 정보를 보여주지 못하므로 바이러스 유포 및 불법 침입자의 추적에 한계가 있다. 이를 위해 netacl 규칙을 참조하여 필요한 접근 규칙을 만들면 효과적으로 접근을 제어할 수 있다.

```
netacl-in.telnetd: permit-hosts 203.237.103.* -exec /usr/sbin/in.telnetd
netacl-in.ftpd: permit-hosts unknown -exec /usr/bin/cat noftp.txt
netacl-in.ftpd: permit-hosts 203.237.110.* -exec /usr/sbin/in.ftpd
netacl-in.ftpd: permit-hosts * -exec /usr/etc/ftpd
```

3.3 FTP 프록시 설정

리눅스 시스템에서의 서비스는 inetd daemon에 의해 기동된다. inetd은 시스템 부팅 시에 구동되는데 구동 시 /etc/inetd.conf 파일에서 서비스의 목록을 얻는다. 그러나 일반적인 데몬처럼 항상 실행되어 있는 것이 아니라, inetd 서버에서 일괄적으로 관찰하고 있다가 요청이 오는 서비스를 그때그때 실행시켜 시스템 리소스를 절약할 수 있게 돕고 있다.

따라서 다음과 같이 /etc/inetd.conf 파일을 수정하여 ftp 프록시 동작을 구현할 수 있다. ftp에 대한 접속요청이 있을 시에는 /usr/local/etc/ftp-gw를 실행하고, telnet에 대한 접속요청이 있을 경우에는 /usr/local/etc/tn-gw를 실행하여 연결을 수립하게 된다.

```
# fwtk setting
ftp stream tcp nowait root /usr/local/etc/ftp-gw ftp-gw
telnet stream tcp nowait root /usr/local/etc/tn-gw tn-gw
```

이러한 환경 구성에서 ftp 포트로의 접속 시도가 발생되면 ftp-gw가 동작하게 되며, ftp-gw는 요청 호스트가 프록시 접속이 허용된 호스트인지 검사하게 된다. ftp-gw는 netperm-table에 설정되어 있는 접근 규칙에 따라 접속 허용 여부를 판별하게 되는데 ftp-gw를 위한 접근 규칙은 다음과 같다.

옵션	설명
userid 사용자	숫자로 표시된 UID나 /etc/passwd 내에 기록된 사용자 이름
directory pathname	서비스 프로그램을 호출하기 위해 ftp-gw가 chroot(2) 명령어를 실행하는 디렉토리
prompt 문자열	명령어 모드에서의 ftp-gw를 위한 프롬프트
denial-msg 파일	프록시 사용이 거부되었을 때 원격 사용자에게 표시할 메시지 파일명
timeout 초	프록시의 연결을 끊을 대기 시간
welcome-msg 파일	프록시 사용이 허용되었을 때 원격 사용자에게 표시할 메시지 파일명
help-msg 파일	'help' 명령어에 대하여 원격 사용자에게 표시할 도움말 파일명
denydest-msg 파일	사용자 인증이 거부되었을 때 원격 사용자에게 표시할 메시지 파일명

표 3.1 ftp-gw의 접근 규칙

ftp-gw와 관련된 netperm-table에는 다음과 같이 접근 규칙이 설정되어 있다.

```
ftp-gw:  denial-msg     /usr/local/etc/ftp-deny.txt
ftp-gw:  welcome-msg    /usr/local/etc/ftp-welcome.txt
ftp-gw:  help-msg       /usr/local/etc/ftp-help.txt
ftp-gw:  denydest-msg   /usr/local/etc/ftp-baddest.txt
ftp-gw:  timeout        3600
```

ftp 프록시에 대한 접근 허용 및 거부 규칙은 몇 가지 추가 옵션에 의해 변경될 수 있다.

```
# telnet proxy
tn-gw: permit-hosts 203.237.*.*
tn-gw: deny-hosts unknown
# ftp-proxy
ftp-gw: permit-hosts 203.237.*.*
ftp-gw: deny-hosts unknown
```

이 규칙이 적용되면 도메인 이름을 DNS에서 발견할 수 없을 경우 접속이 거부되며, 203.237.*.* 네트워크로부터의 접근만을 허용하게 된다.

ftp 프록시를 통한 접근이 이루어지게 되고 허가된 호스트로 판단되면 접근 규칙에 따라 사용자 인증이 요구될 수 있다. 사용자 인증이 사용된 경우의 netperm-table의 내용은 다음과 같다.

```
ftp-gw: permit-hosts 203.237.*.* -authall -log { retr stor }
```

3.4 바이러스 검색 모듈

ftp의 get 명령을 이용하여 리모트 서버에서 로컬 클라이언트로 파일을 전송하면 ftp-gw.c의 copyout() 함수가 실행됨으로써 방화벽의 바이러스 진단 모듈이 동작을 시작하게 된다. copyout() 함수에 의해 파일을 다운로드 하면서 ScanDir(), CheckVirus() 함수로 바이러스 감염유무를 진단하게 되고 만약 바이러스에 감염된 파일이 발견되었을 때에는 사용자의 화면에 치료를 요구하는 메시지를 띄우고, 동시에 write_mail() 함수에 의해 감염된 파일의 이름과 바이러스 이름이 관리자(root@localhost)에게 메일로 발송된다. 네트워크 상의 바이러스 탐지 시스템에 사용된 주요 함수는 다음과 같다.

함수명	함수인자	리턴값
copyout	void	void
ScanDir	const char *MyFile	void
CheckVirus	unsigned char *pFilePath	int
write_mail	FILE *stream, char *VName	int

표 3.2 바이러스 진단 관련 함수

전송되는 모든 파일의 사본은 /tmp 아래에 같은 이름으로 생성되고 바이러스 검사 엔진으로 바이러스 감염여부를 진단하여 바이러스 발견 시 조치하도록 하고 있다.

네트워크상의 Unknown 바이러스 진단 함수의 상관도는 다음과 같다.

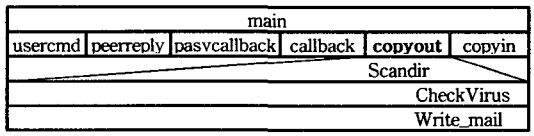

그림 3.1 바이러스 진단 함수의 상관도

3.5 바이러스 검색 시나리오

① 내부 사용자가 외부 인터넷으로부터의 파일 전송을 위해 방화벽 호스트에 접속.
② 간단한 인증 절차를 거친 후 원래 접속하고자 하는 외부 호스트로 접속.
③ 사용자는 필요한 파일을 전송 받는다.
④ 파일 전송시 FTP-GW는 내부 사용자의 호스트로 파일을 전송함과 동시에 방화벽 호스트(혹은 임의의 호스트)에 동일의 파일을 임의의 이름으로 저장.
⑤ 파일 전송이 끝나기 직전 해당 파일의 바이러스 감염 여부 조사.
⑥ 바이러스 감염시 경고문을 사용자에게 알려준다.
⑦ 파일 전송을 마친다. 임시 저장 파일은 삭제한다.
⑧ 사용자 요구 시 외부 호스트 접속 종료.
⑨ 사용자 요구 시 방화벽 호스트 접속 종료.

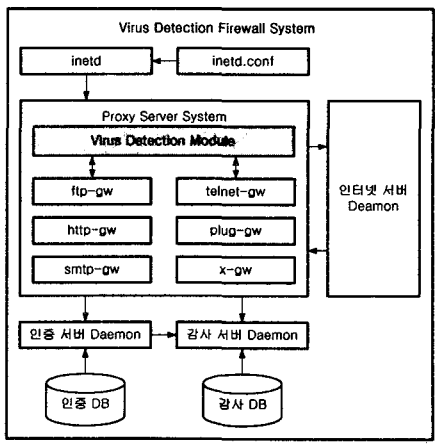

그림 3.2 바이러스 탐지 방화벽 시스템

4. 구현 결과

4.1 로그인

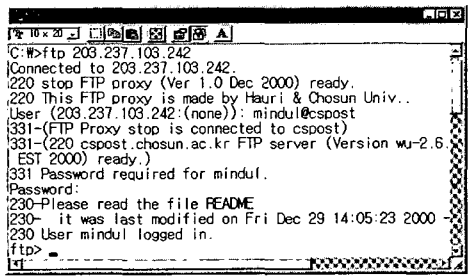

그림 4.1 로그인

username이라는 프롬프트를 받으면 **username@hostname** 형식으로 입력한다.

4.2 파일 전송

다음은 리모트 서버에서 로컬 클라이언트로 binary mode 상태에서 파일을 다운로드 하는 것을 보여주고 있다.

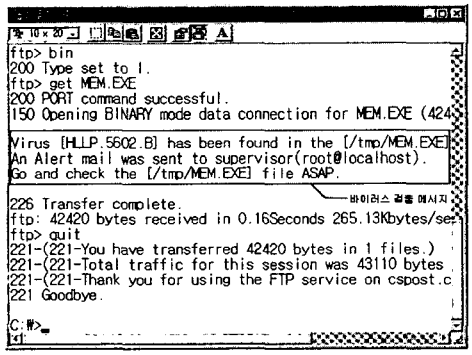

그림 4.2 파일전송과 네트워크상의 바이러스진단

4.3 전송 파일의 Proxy 서버 저장

전송된 파일(MEM.EXE)의 사본이 Proxy 서버에 저장되었는지 확인한다. Proxy 서버의 임시 디렉토리에서 바이러스 감염여부를 진단 및 치료한다.

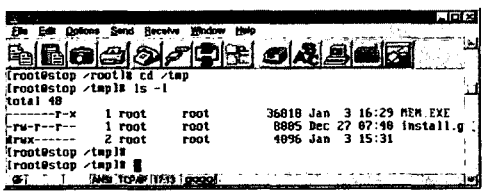

그림 4.3 전송 파일의 Proxy 서버 저장

4.4 바이러스 탐지 결과 전달

구축된 바이러스 차단 및 진단 시스템은 네트워크를 통해 들어오는 파일에 대한 바이러스 검사 결과 바이러스로 진단되는 경우에는 즉시 시스템 관리자와 해당 파일을 전송 받는 사용자에게 경고 메시지를 보내고, html 파일 형식으로 웹 상에서 감염된 파일, 파일 소유자, 바이러스 진단 날짜, 감염 바이러스 명칭 등이 저장되어 출력된다. 출력 결과는 다음과 같다.

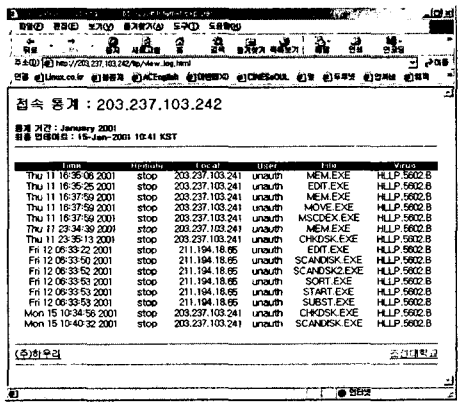

그림 4.4 바이러스 진단 결과의 웹 페이지 출력

5. 결론

본 논문에서 설계 및 구현된 바이러스 탐지 방화벽 시스템은 방화벽 고유의 기능을 유지하면서 지금까지 네트워크 상에서 유입되는 파일의 바이러스 진단 및 치료의 과정을 효과적으로 수행 할 수 있는 시스템을 설계 및 구현하였다. 이를 이용함으로써 날로 증대되고 있는 바이러스의 침입에 대해 시스템의 관리자 및 사용자는 수동적이고 정기적인 바이러스 검사 방법에서 탈피하여 내부 및 외부 네트워크 상에서 데이터가 유입되는 시기에 실시간으로 바이러스 검사를 행할 수 있음으로 보다 안전하고 신속한 바이러스 예방 효과를 얻을 수 있다.

참고문헌

[1] RFC 959 "File Transfer Protocol(FTP)"
[2] Keith Haviland, Dina Gray, Ben Salama "UNIX System Programming, 2/E" 1999
[3] Kurt Wall "Linux Programming by Example" QUE 2000
[4] W. Richard Stevens "Advanced Programming in the UNIX Environment" Addison Wesley 1992
[5] W. Richard Stevens "UNIX Network Programming Vol.1 2/E" 교보문고 1999
[6] Richard Stones "Beginning Linux Programming, 2E" 정보문화사 2000
[7] "UNIX System V/386 Release 4 프로그래머 지침서" UNIX PRESS 챌라 1992
[8] 권석철, 주영흠, 김판구 "컴퓨터 바이러스 완전 소탕" 크라운 출판사 1997
[9] 조선대학교 "컴퓨터 바이러스 진단 치료 프로그램 및 감염예방시스템 구현" 한국정보보호센터 1997