WAVScanner : 웹기반 안티 바이러스 스캐너 설계 및 구현

WAVScanner : Design and Implement of Web based Anti-Virus Scanner

이 상 훈* 김 원** 도 경 화*** 전 문 석****
Sang-Hun Lee Won Kim Kyoung-Hwa Do Moon-Seog Jun

요 약

컴퓨터 및 네트워크의 발달로 무한한 정보들을 쉽고 빠르게 사용할 수 있게 되었지만 이에 따른 부작용도 증가되었다. 이러한 부작용에는 해킹이나 크래킹, 개인정보 유출 등이 있으며 최근에는 컴퓨터 바이러스가 심각한 문제로 제기되고 있다. 컴퓨터 바이러스에 대한 해결책은 안티 바이러스이다. 안티 바이러스는 클라이언트 측에 설치되어 서버에서 바이러스의 시그니쳐를 내려받아 업그레이드 하는 형태로 구현되고 있으나 최근에는 서버와 연동하는 제품들도 생겨나게 되었다. 그러나 이러한 안티 바이러스 시스템들은 사용자의 무관심으로 적절히 시그니쳐가 갱신되지 않으면 안티 바이러스가 정상적으로 작동하지 않으며 원격관리가 되지 않는다는 단점이 있다. 따라서, 본 논문에서는 이러한 문제를 해결하기 위하여 인터넷 서버에 설치되어 원격으로 관리할 수 있는 웹기반 안티 바이러스 스캐너(WAVScanner)를 설계 및 구현하였다. 이를 통해 새로운 바이러스에 대한 효과적인 안티바이러스 대책 수립과 백신 개발을 가속화할 수 있을 것이다.

Abstract

It is easy to access to the infinity information and programs. but it gives rise to the side effect. There are many side effects(ex. Hacking, Cracking, expose the personal information, etc). Nowadays, the computer virus raise the serious problems. The making program called Vaccine is work out a count measure. The Anti-Virus programs install on the client side computer and upgrade by downloading on the server's signature. the latest date, the program bound both of them is shown. but these programs have the defect that they have no remote control and no signature update because user's unconcern. This paper reported the research of existing virus infecting technology and the development of Web based Anti-Virus Scanner using the remote control on the internet server. Through this paper, I want to set up the counter measure for new virus easily, and to make more fast the vaccine for virus.

Keyword : Virus, Anti-Virus, Scanner

1. 서 론

현재 네트워크를 통해 연결된 모든 컴퓨터들은 바이러스의 위협에 노출되어 있다. 과거에는 플로피 디스켓 등을 통해 파일에 감염되었으나 인터넷 등의 발전으로 인하여 언제 어디서든 네트워크를 통하여 바이러스에 감염될 수 있다. 또한 바이러스들은 점차 진화하여 과거의 바이러스와는 다른 새로운 형태의 바이러스가 급속도로 출현하면서 짧은 시간동안 큰 피해를 주고 있다[1-4]. 이러한 바이러스에 대응하기 위한 방법으로는 안티 바이러스 프로그램이 가장 좋은 대안이다[5].

안티 바이러스란 바이러스를 발견하고 제거하는 툴을 말한다. 그러나 안티 바이러스는 컴퓨터 바이러스를 분석한 후 분석한 자료를 바탕으로 바이러스 시그니쳐를 생성하여 프로그램에 삽입해야만 동작이 가능하다. 효과적인 안티 바이러스

* 정 회 원 : 숭실대학교 대학원 컴퓨터학과 박사과정
iam@leesanghun.pe.kr(제 1저자)
** 정 회 원 : 전주기전여자대학 실용예술학부 조교수
wkim@kijeon.ac.kr(공동저자)
*** 정 회 원 : 행정자치부 전자정부지원센터 전문위원
khdo0905@dreamwiz.com(공동저자)
**** 정 회 원 : 숭실대학교 정보과학대학 교수
mjun@computing.ssu.ac.kr(공동저자)

를 만들기 위해서는 바이러스에 대한 분석과 스캐닝 작업이 빠르고 정확하게 선결되어야 하며 바이러스의 시그니쳐 생성과 이러한 바이러스 시그니쳐로 바이러스를 분석해 내는 작업이 매우 중요하다[6]. 이러한 분석과정은 확산과 피해를 최소화하기 위하여 최단시간 내에 이루어져야 한다. 대부분의 안티 바이러스는 클라이언트 측에서 설치되어 서버에서 바이러스의 시그니쳐를 내려받아 업그레이드 하는 형태로 구현되고 있으나 최근에 일부 서버와 연동하는 제품들도 출시되고 있다. 그러나 이러한 유형의 안티 바이러스 시스템들은 적절한 시점에 시그니쳐가 갱신되지 않아 원하는 시점에 안티 바이러스가 제대로 작동하지 않을 수 있고 원격관리과 되지 않는다는 단점이 있다. 따라서 본 논문에서는 이러한 문제를 해결하기 위하여 웹상에서 바이러스 시그니쳐를 이용해 바이러스를 탐색해 내는 안티 바이러스 스캐너를 설계 및 구현하였다. 이러한 웹 기반 안티 바이러스 시스템은 안티 바이러스의 빠른 개발을 독려하며 원격 바이러스 검사를 통한 원격관리를 할 수 있다.

안티 바이러스 시스템을 구현하기 위해서는 다음 사항을 고려해야 한다. 첫째, 안티 바이러스의 중요한 엔진은 다양한 플랫폼에 맞게 컴파일과 실행 될 수 있어야 한다. 바이러스의 특성상 각 운영체제마다 바이러스가 동작하는 모양, 사이트, 내용들이 다르므로 플랫폼을 신중히 고려한 후 설계해야 한다. 둘째, 안티 바이러스 프로그램은 빠르게 검색하고 처리 되어야 하기 때문에 속도가 빠른 프로그래밍 언어가 좋다. 또한 다양한 시스템에 탑재될 수 있도록 여러 가지 경우의 수를 프로그래밍 할 수 있는 언어가 좋으며 이러한 선택은 플랫폼 구현에 직접적으로 영향을 미친다. 일반적으로 안티 바이러스 엔진은 모든 플랫폼에서 실행될 수 있는 언어가 좋으며 기타 인터페이스 등은 다른 프로그래밍 언어를 이용하는 경우가 많다. 셋째, 안티 바이러스의 핵심 엔진이 주변의 운영체제와 독립되도록 설계하는 것이 바람직한데 이렇게 하기 위해서는 엔진과 파일 시스템 사이에 가상 레이어를 두는 것이 좋다. 이러한 레이어는 조건에 따라서 컴파일 할 수 있으며, 가상 레이어를 사용자 인터페이스로 접근 할 수 있도록 해야 한다. 넷째, 모듈화는 대부분의 시스템에서 유용한 방법으로 바이러스 백신 제작에서도 대단히 중요하다. 그 외에도 압축된 파일 시스템의 검사를 위한 압축 알고리즘의 삽입과 각각의 파일 시스템을 검사할 때 파일 타입 스캐너등도 필요하다. 또한 제2세대 바이러스 형태인 암호화 바이러스에 대처하기 위해 메모리 스캐너와 코드 에뮬레이터등도 필요하다. 그리고 가장 중요한 온라인 업데이트 모듈을 반듯이 구현되어야 한다.

논문 구성은 5장으로 다음과 같다. 2장에서는 윈도우 시스템에서 바이러스의 형태를 분석하였다. 3장에서는 웹기반 안티 바이러스의 코어 및 웹 인터페이스를 설계한 내용을 보였으며 4장에서는 3장에서 설계한 내용을 바탕으로 구현한 결과 및 성능평가를 보였다. 마지막으로 5장에서는 결론 및 향후 연구방향에 대해서 기술하였다.

2. 컴퓨터 바이러스 분석

컴퓨터 바이러스란 컴퓨터 시스템의 부트 영역, 메모리, 실행 프로그램, 문서 등에 하나 또는 그 이상에 감염되어 자기 증식 및 복제를 하는 파괴성 컴퓨터 프로그램을 의미한다. 컴퓨터 바이러스 또한 프로그램이기 때문에 이러한 프로그램은 일관성 있는 프로그램 루틴을 가지게 된다. 이러한 루틴으로 인해 바이러스 제작자들은 바이러스를 인식할 수 있고, 또한 정의 및 분류를 해 낼 수 있다[7].

2.1 윈도우 파일 분석

대부분의 바이러스는 실행 파일에 기생하여 파일이 실행될 때 다른 파일들을 감염 시킨다. 윈도우 실행파일은 내부적으로 PE(Portable Execution)

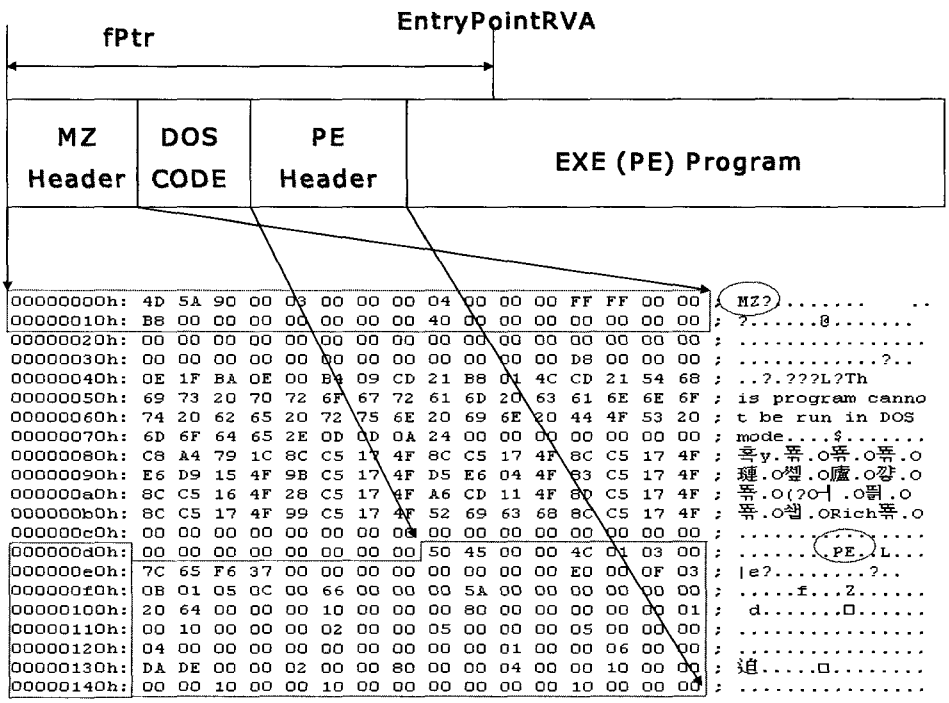
헤더, NE(New Execution) 헤더, LE(Liner Execution) 헤더, MZ(Magic) 헤더를 가지는 파일들을 말한다 [8]. 윈도우의 파일을 실행하기 위해서는 4가지의 헤더 중에서 한 가지 이상을 받듯이 포함하고 있어야하며, 현재 활동하고 있는 바이러스를 분석하기 위해서는 PE파일의 헤더를 반듯이 분석해야 한다. PE 파일은 마이크로소프트 32 bit 운영체제와 호환되는 실행 파일로서 윈도우 95, 98, ME, NT, 2000, XP등에서 실행되며, 모든 PE 파일이 실행 가능하지만, 모든 실행 가능한 파일이 호환되는 것은 아니다. PE파일의 형식은 COFF(Common Object File format) 형식을 보완한 것으로서 다양한 플랫폼에서 사용되기 때문에 높은 이식성을 가지고 있다.

(그림 1)은 NotePad.exe의 PE 헤더를 울트라에디터(편집기)로 분석한 것이다. PE 파일에서 제일 처음 나타나는 것은 MZ 헤더인데 이는 윈도우가 DOS(도스)기반으로 해서 작성이 되었기 때문에 호환성을 유지하기 위해서는 꼭 필요한 헤더이다. MZ 헤더는 도스 모드에서 실행이 되었을 때를 위한 MS-DOS Stub의 위치와 PE 헤더의 위치를 가지고 있다. 이러한 점 때문에 DOS에서 작동하는 바이러스들은 PE파일의 MS-DOS Stub등에 상주하여 감염시킬 수 있다.

MZ 헤더 가 위치를 가리키는 곳에 PE 헤더가 시작되며 그 주소에는 "PE\0\0" 가 위치하게 된다. 이때부터 윈도우는 파일의 안전성 및 타당성을 여러 가지 섹션과 PE파일의 정보로부터 얻게 되어 로더에 적재시키게 된다.

여기서는 주소 000000d08h에서부터 PE 헤더가 시작된다. PE 헤더를 분석하는 이유는 대부분의 윈도우 바이러스들이 자신의 위치를 숨기고 PE 헤더를 위조하여 프로그램의 순서 및 위치를 변경시키기 때문이다.

NotePad.exe 파일의 PE 헤더를 분석한 (표 1)에서처럼 일반적인 PE파일이라면 00004550h의 값을



〈그림 1〉 NotePad.exe의 파일분석

〈표 1〉 NotePad.exe의 PE File Header

주소	표기	값	의미
000000d8h	50450000	00004550h	Signature PE
000000dBh	4C01	014Ch	Machine (014C= i386)
000000dDh	0300	0003h	Number Of Sections
000000e0h	7C65F637	37F6657Ch	Time/Date Stamp
000000e4h	00000000	00000000h	Pointer to Symbol Table
000000e8h	00000000	00000000h	Number of Symbols
000000eBh	E000	00E0h	Optional Header Size
000000eDh	0F03	03E0h	Characteristics

〈표 2〉 NotePad.exe의 PE Optional Header

주소	표기	값	의미
000000f0h	0B01	010Bh	Magic (PE 32)
000000f2h	050C	0C05h	LinkerVersion
000000f4h	00660000	00006600h	SizeOfCode
000000f8h	005A0000	00005A00h	SizeOfInitializedData
000000fBh	00000000	00000000h	SizeOfUnitializedData
00000110h	20640000	00006420h	AddressOfEntryPoint
00000114h	00100000	00001000h	BaseOfCode
00000118h	00800000	00008000h	BaseOfData
0000011Bh	00000001	01000000h	ImageBase

갖는 "PE\0\0"으로 시작하여야 하며 Machine은 프로그램이 동작할 수 있는 시스템을 나타나게 된다. 현재 NotePad.exe 파일은 014Ch로 i386 인텔 계열을 나타나게 되는데 이 값을 변경하면 시스템이 파일을 구동시키지 못하게 된다.

PE 파일은 옵션 헤더와 여러 개의 섹션 헤더를 같은데 "Number Of Sections"에서 섹션의 수를 표기한다. 후위형 바이러스의 경우 흔히 이 수치를 변경하여 위장을 시도한다. 제일 마지막에 표기된 "Characteristics"는 파일의 속성 등을 나타내는 값으로서 바이러스에 감염되었을 때 파일의 속성이 변경되었다면 이 값을 확인해 보아야 한다.

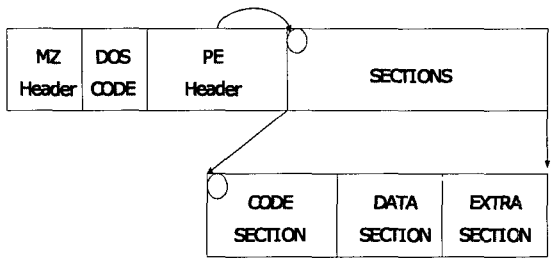
(표 2)는 NotePad.exe 의 PE 옵션 헤더를 나타낸 것이다. 옵션 헤더는 말처럼 없어도 상관이 없다는 뜻이 아님에 주의해야 한다. 이 헤더에서 가장 중요한 것은 "AddressOfEntryPoint"로서 PE 로더가 PE 파일을 로드하여 처음에 실행할 명령어를 가리키는 주소이다. 따라서 바이러스 제작자들이 이 값을 수정하여 바이러스를 가리키게 한다.

2.2 윈도우 바이러스 분석

윈도우 바이러스의 대부분은 PE 파일에 바이러스를 숨기고 PE 헤더를 수정하여 만들어 진다. 이러한 바이러스들은 PE 헤더를 수정하는 것만으로도 치료가 가능한 것도 있으나 겹쳐 쓰기 바이러스 같은 경우 헤더정보만으로는 복구가 불가능한 경우도 있다.

2.2.1 헤더 감염

PE 파일의 끝과 처음 섹션의 시작점 사이에 바이러스 코드를 삽입하고, PE 헤더에서 바이러스의 엔트리포인트를 대신 가리키도록 "Address OfEntryPoint"필드를 수정하여 바이러스를 감염시킬 수 있다.



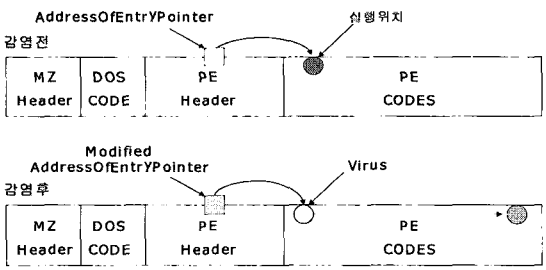
〈그림 2〉 바이러스의 헤더 감염

(그림 2)는 바이러스를 섹션의 처음에 위치하고 PE 헤더의 변조를 통하여 바이러스를 실행시키는 모습이다. 이러한 헤더 감염 바이러스의 코드는 매우 짧고 섹션들은 FileAlignment의 집합인 오프셋에서 시작해야 하기 때문에 최대한 덮어쓸 수 있는 공간은 FileAlignment 값보다 클 수 없다. 응용프로그램이 너무 많은 섹션과 FileAlignment를 포함할 때 그 크기는 512바이트가 되고 따라

서 그곳에는 바이러스 코드가 있을 수 없다. 그러나 바이러스는 섹션의 어디에도 위치할 수 없으므로, 실제 "AddressOfEntryPoint"필드의 RVA는 바이러스가 헤더에 위치하고 있는 파일에서 실제 오프셋이 된다. 따라서 엔트리 포인트가 코드 섹션의 어느 부분도 가리키지 않게 하여도 해당 프로그램은 실행되며, 이를 통해 윈도우 95의 로더는 감염된 프로그램을 자연스럽게 실행시킬 수 있다.

2.2.2 전위형 바이러스

PE 파일을 감염시키는 가장 쉬운 방법은 (그림 3)과 같이 PE파일의 "EntryPointer"를 수정하는 것이다. 그러나 이렇게 감염된 응용프로그램은 제대로 동작하지 않기 때문에 감염 즉시 발견될 수 있다.

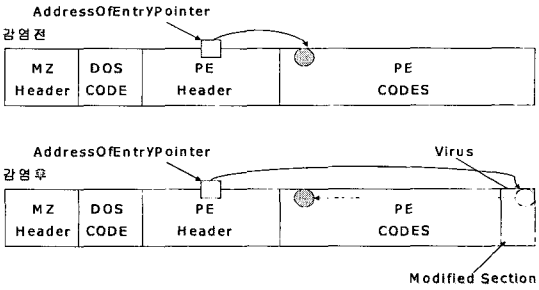


〈그림 3〉 전위형 바이러스 감염도

이러한 바이러스들은 보통 C와 델파이 같은 고급언어로 쓰여진다. 감염된 프로그램은 바이러스의 EXE헤더와 함께 시작된다. 컴퓨터 바이러스가 본래 프로그램 코드의 제어를 전송하기를 원할 때, 바이러스는 본래 코드를 임시 파일에 옮겨 놓고 거기에서 바이러스를 실행시킨다.

2.2.3 후위형 바이러스

섹션 테이블의 끝에 새로운 섹션 헤더를 추가하거나 추가하지 않고 바이러스에 알맞은 마지막 섹션을 섹션헤더에 고정되도록 수정하는 바이러스를 후위형 바이러스라 하는데 (그림 4)와 같다. 이 방법을 사용하면 쉽게 모든 PE파일을 감염시킬 수 있다.

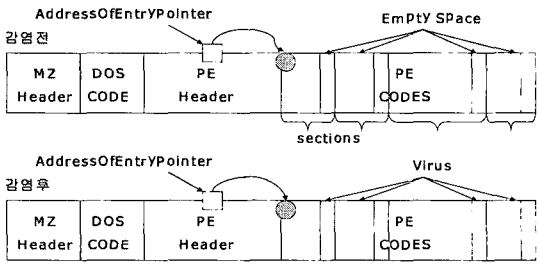


〈그림 4〉 후위형 바이러스 감염도

실제 섹션헤더가 섹션 테이블과 같지 않다는 것을 걱정할 필요는 없다. "VirtualSize"와 "SizeOfRawData"필드의 수정에 의해 바이러스 코드는 실행 부분의 끝에 위치할 수 있다. 따라서 PE헤더의 "NumberOfSection"필드를 수정할 필요가 없다. 다음으로 "AddressOfEntry Point" 필드는 바이러스 몸체를 가리키는 것으로 바뀌고, "SizeOfImage"는 프로그램의 새로운 크기를 나타내는 것으로 재계산된다. 마지막 섹션 헤더의 "characteristics" 필드는 쓰기 가능/실행 가능한 속성으로 바뀌게 된다. 쓰기 가능한 속성은 스스로 어떤 섹션으로부터 코드를 실행하게 할 수 있다.

2.2.4 기생 겹쳐쓰기형 바이러스

기생 겹쳐쓰기형 바이러스는 (그림 5)에서 처럼 링커에 의해 통상적으로는 0(또는 0x00)으로 채워진 대부분의 섹션들 사이의 여유공간(slack section)



〈그림 5〉 기생 겹쳐쓰기형 감염도

〈표 3〉 기존 안티바이러스 제품과의 비교 분석

구 분	기존의 보호 구조		본논문의 보호구조 (원격관리 서버형)
	클라이언트형	C/S형	
안티바이러스 설치 위치	클라이언트	클라이언트/ 서버(클라이언트 관리서버)	서버
편의성	낮음(각 시스템별로 시그니쳐 업데이트)	높음(서버가 강제로 업데이트)	높음(서버만 업데이트)
용도	클라이언트용	클라이언트용	서버용
검사범위	개인 컴퓨터	개인 컴퓨터/서버 시스템	서버 시스템
시그니쳐 업데이트 방법	콘솔에서 사용자 지정 (업데이트 서버이용)	서버 콘솔에서 사용자 지정 또는 원격관리 콘솔 사용 (업데이트 서버이용)	인터넷이 가능한 어떠한 PC에서도 가능

을 이용한다.

섹션은 PE헤더의 "FileAlignment" 필드에 기술되어 있는 파일 정렬 값에서부터 섹션이 시작되기 때문이다. 가상 크기를 갖는 각 섹션은 일반적으로 실제 데이터로 표시되는 값과는 다르다. 일반적으로 가상적인 크기가 작은 값을 가진다. 대부분의 마이크로소프트 링커 프로그램들이 이런 방식으로 PE파일들을 생성한다. 섹션의 실제 데이터 크기는 0으로 채워지고 프로그램이 갖는 주소영역을 통해 로드되지 않는 곳의 실제 정렬 영역과는 차이점을 갖는다.

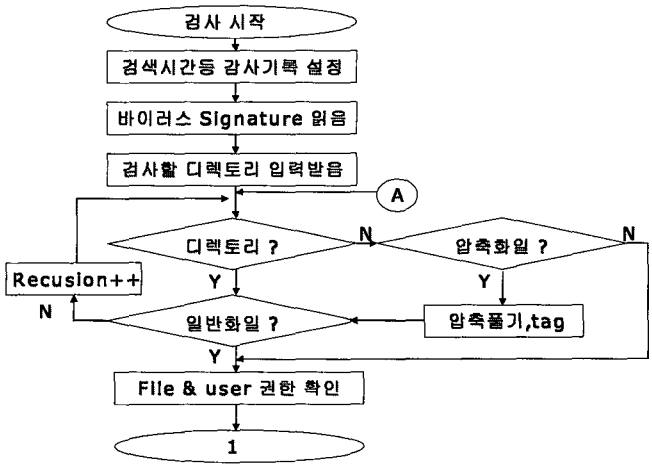
"FileAligment"의 기본값이 512바이트(일반적인 섹터의 크기)가 되면 일반적인 여유공간 크기는 512바이트 미만이 된다. 빈틈 감염을 위해서는 512바이트 보다 작은 크기를 가져야 하고, 이 크기라면 평균적인 PE감염 바이러스의 종류가 되기에는 크기가 너무 작다. 그러나 바이러스의 몸체를 여러 부분으로 나누어져 저장하게 되면 바이러스 코드를 삽입하는 것이 전혀 문제되지 않는다. 이 방법은 Win95/CIH 바이러스에서 정확하게 사용되었다. 이것은 컴퓨터 바이러스의 검색과 치료를 더욱 어렵게 만든다. 바이러스는 섹션의 가상 크기를 바꿔 각 섹션 헤더 안의 실제 데이터 크기와 같게 변화시킨다. 바이러스 몸체를 조각으로 나누어 감염될 프로그램에 침투한다. 이러한 바이러스의 특별한 특징은 일반적인 바이러스보다 훨씬 분석하기 어렵다. 왜냐하면, 바이러스의 몸체가 PE이미지의 첫 부분의 다른 영역으로부터 나눠어진채로 추출되기 때문이다.

2.3 기존 안티바이러스 제품과의 비교분석

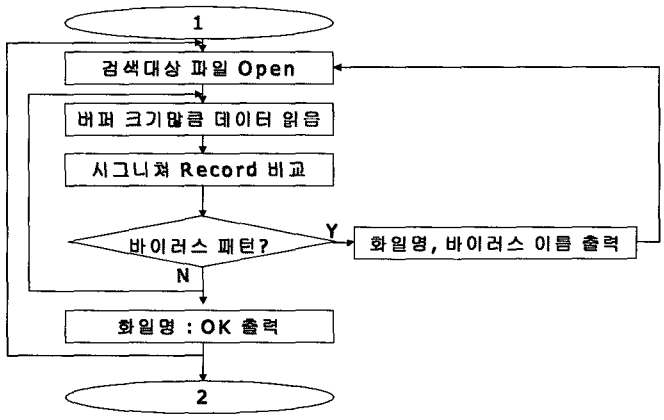
(표 3)은 기존의 안티바이러스 제품과 본 논문에서 제시하는 안티바이러스 시스템의 기능을 비교 분석한 것이다. 기존의 구조는 클라이언트 형이 대부분이며 서버는 관리용으로만 사용되었다는 것을 볼 수 있다. 따라서 시그니쳐 업데이트도 콘솔에서 사용자가 서버에서 업데이트 받아야 하나 본 논문에서 제안한 안티바이러스는 서버형으로 서버만 업데이트하면 등록되어 있는 클라이언트는 자동으로 업데이트 된다.

3. WAVScanner : 웹 기반 안티 바이러스 스캐너 설계

2장의 컴퓨터 바이러스 분석 내용을 바탕으로 웹 기반 안티 바이러스 스캐너(WAVScanner)를 설계하였다. WAVScanner는 서버 모듈로 동작하는 스캐너 모듈과 웹 인터페이스로 동작하는 외부 인터페이스 모듈로 나누어진다. 스캐너 모듈은



〈그림 6〉 WAVScanner 입력 서브모듈



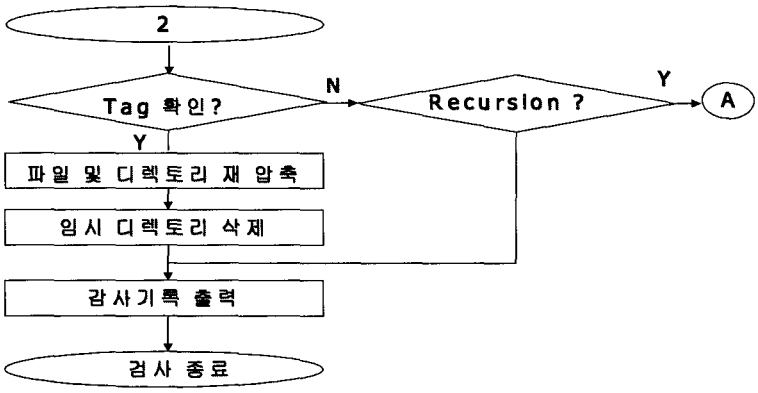
〈그림 7〉 WAVScanner 비교 서브모듈

C로 구현되어 있어 대부분의 Unix 및 Linux 시스템을 지원하며 웹 인터페이스는 어떤 웹 서버도 상관없으나 Apache, PhP, OpenSSL, Mod_ssl, X509v3 Cert(인증서) 등을 사용하였다.

3.1 스캐너 모듈 설계

WAVScanner 모듈은 입력 서브모듈, 비교 서브모듈, 출력 서브모듈로 구성되어 있다. (그림 6)은 파일의 내용과 바이러스의 시그니쳐를 읽는 입력 서브모듈로서 웹 인터페이스로부터 받은 디렉터리를 일반화일로 추출하는 역할을 한다. 제안한 스캐너는 하위 디렉토리를 검색할 수 있도록 Recursion을 사용하며 압축 파일 속의 데이터를 비교할 수 있도록 하였다. 이때 압축화일은 기존의 OS에 있는 압축 해제 프로그램을 사용한다.

(그림 7)은 입력 서브모듈에서 파일의 이름과 바이러스 시그니쳐를 받아와서 비교하는 비교 서브모듈이다. 이때 바이러스 검색에 사용되는 알고리즘은 Knuth-Morris-Pratt(KMP) 알고리즘을 사용했다. KMP string matching 알고리즘은 주어진 패턴을 가지고 긴 파일 안에서 동일한 패턴을 찾아내는 알고리즘으로서 시그니쳐와 읽은 파일 데이터의 구조를 TRIE 구조를 사용하여 저장, 비교하



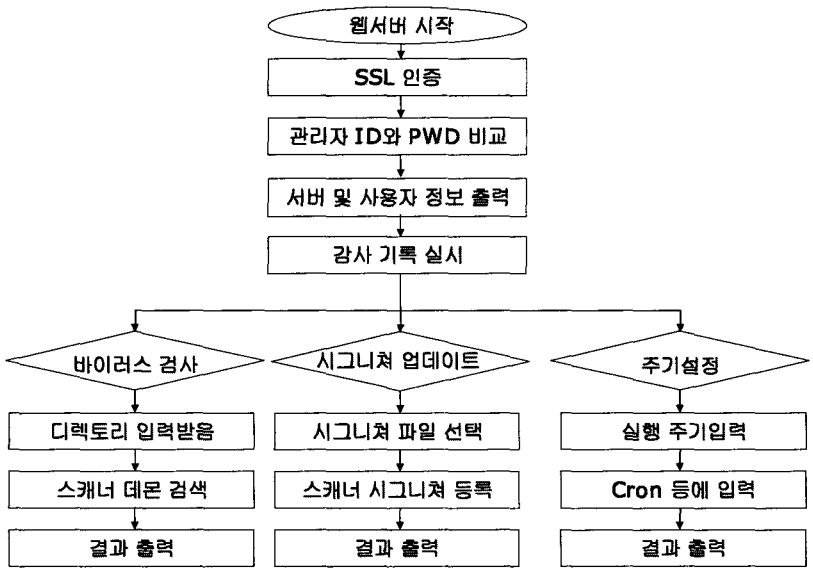
〈그림 8〉 WAVScanner 출력 서브모듈

여 바이러스를 검사 한다. 패턴이 매칭 되면 바이러스가 감염된 것이므로 파일명과 바이러스 명을 출력한 후 다른 파일 검색을 시작하고 매칭 되지 않으면 끝까지 읽은 뒤 다른 파일을 읽게 된다.

모든 파일의 검색이 끝난 후 (그림 8)과 같이 Recursion과 압축화일의 여부를 판단하여 압축되어 있던 파일은 재 압축을 실시하고 하위 디렉터리로 들어온 경우라면 상위 디렉터리로 가기위해 입력 서브모듈(A)로 다시 돌아가게 된다. 마지막 파일까지 검사가 끝난 경우 감사 기록 및 스캔 결과를 기록하고 웹 인터페이스에게 값을 넘겨주게 된다.

3.2 인터페이스 모듈 설계

(그림 9)의 웹 인터페이스 모듈은 바이러스 스캐너 모듈, 시그니쳐 모듈, 스케쥴 모듈로 구성되어 있다. 보안측면에서 스니핑 과 비 인가된 사용자의 접근을 막기 위해서 SSL 과 사용자 인증 모듈을 추가하였으며 초기화면에서는 접근에 대한



〈그림 9〉 웹 인터페이스 모듈

서버정보와 접속자의 정보를 출력하게 하였다. 각 부분은 로그를 남기게 되어 있으며, 추후에 서버 관리자가 감사 기록 메시지 등을 통하여 바이러스 스캐너의 사용 정보를 확인해 볼 수 있다.

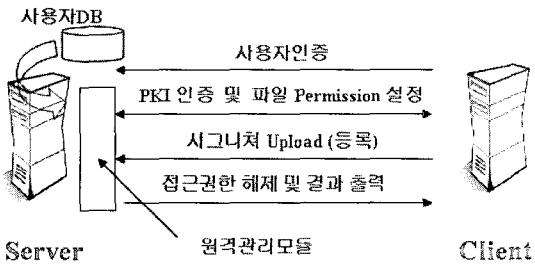
바이러스 시그니쳐 파일은 텍스트 파일로 저장된다. 제일 먼저 바이러스의 이름이 오게 되며 "="의 뒷부분에는 바이러스 시그니쳐가 오게 된다. 바이러스들은 이름으로 정렬되어 있으며 바이러스의 변종에 따라 시그니쳐가 달라진다. 바이러스 시그니쳐는 바이러스가 지니는 특징을 나타내는 스트링으로 검사 파일을 오픈할 때 바이너리 모드로 읽어서 16진수로 비교를 통한 스트링 매칭을 할 수 있도록 한다.

3.3 원격 관리 모듈의 정의와 설계

안티바이러스 모델은 대부분 클라이언트 형태의 모델과 클라이언트/서버형 모델이 있다. 여기서 클라이언트형 안티바이러스 모델은 바이러스 탐지 및 치료가 사용자 시스템에 설치되어 있다는 것을 말하는데, 보호받을 수 있는 시스템의 가장 마지막 부분이라고 할 수 있다. 이러한 모델의 안티바이러스 백신은 원격지 서버에 있는 바이러스 시그니쳐 업데이트 서버에 인터넷을 통하여 사용자가 접근함으로서 시그니쳐를 다운받아 사용하는 형태로 사용자의 주의가 필요하며, 서버에는 적합하지 않는 일반적인 형태의 바이러스 백신이다. 서버형 안티바이러스 모델은 기본적인 틀은 동일하게 유지하며 인트라넷에 바이러스 시그니쳐 서버의 미러링 서버를 운용하고 각 클라이언트들의 시그니쳐를 강제적으로 업데이트 혹은 검사하는 형태의 모델이다. 클라이언트 형태의 모델에 비해 관리가 용의하며, 바이러스 통계 등 보고용 및 관리용으로 회사 및 기관에서 많이 사용하는 안티 바이러스 시스템 유형이다.

기존의 클라이언트/서버형태의 모델은 정책 및 시그니쳐 업데이트 등 서버 등의 설정 및 변환을 위해서는 서버 시스템의 콘솔에서 직접 작업을 하거나 원격 관리 툴(R-admin 또는 터미널서비스)을 이용하여 처리한다. 그러나 본 논문에서 제안하는 웹을 기반으로 하는 안티바이러스 모델은 시스템에 들어있는 원격관리 모듈을 이용하여 웹에서 관리자가 언제 어디서든지 웹 브라우져 이외의 어떠한 원격 관리 툴 없이도 안티 바이러스 서버에 접근이 가능하도록 하며, 시그니쳐 업데이트를 사용자가 즉시 업로드 하도록 함으로서 중앙 시그니쳐 서버 없이 동작이 가능하도록 설계/구현 한다. 데이터의 보안을 위하여 PKI를 이용하여 송수신간의 데이터를 암호화 한다.

원격관리 모듈에서 가장 중요한 부분은 기존 모델들이 일반적으로 웹에서 nobody permission으로 동작하기 때문에 바이러스의 검색 엔진이 지정된 부분이외의 파일에 대한 접근 허가를 얻을 수 없다는 것이 문제였으나 본 논문에서는 사용자 인증을 통해 원격관리 모듈에서 검색되는 동안 파일시스템의 접근허가를 부여할 수 있도록 하여 이러한 부분을 해결하였다. 논문에서 제안한 원결관리모듈의 동작과정을 (그림 10)에 도식하였다.

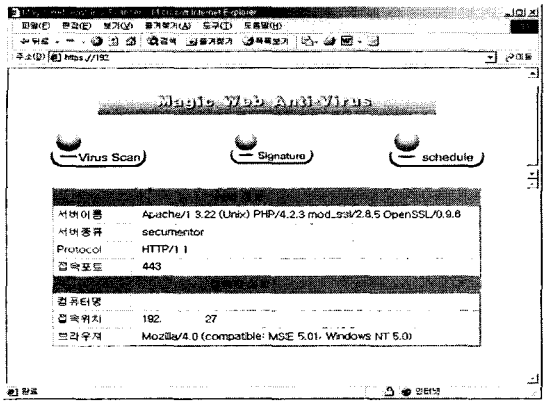


〈그림 10〉 WAVScanner 원격보안 구성도

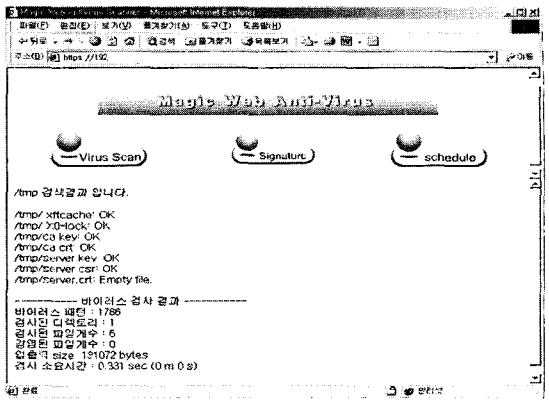
4. WAVScanner : 웹기반 안티 바이러스 스캐너 구현

4.1 웹기반 안티 바이러스 스캐너 구현

안티 바이러스 스캐너의 인터페이스 모듈이 각 내부 모듈을 실행하고 있으므로 관리자는 단지 아파치와 원격관리모듈을 실행시키는 것으로서



〈그림 11〉 WAVScanner의 주화면



〈그림 12〉 바이러스 검사 결과

웹 기반의 바이러스 서버를 동작시킬 수 있다. 관리자는 인터넷이 되는 곳에서 웹 브라우저를 기동하여 서버의 바이러스 점검 상태 등을 확인할 수 있다.

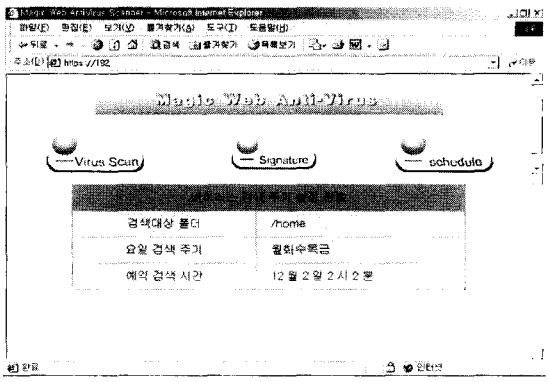
(그림 11)은 WAVScanner를 클라이언트(IP : 192.XXX.XXX.XXX)에서 MS 브라우저를 통해 실행시킨 것이다. WAVScanner은 인증확인을 끝낸 관리자에게 서버의 정보(이름, 종류, 프로토콜, 포트)등을 확인 시키고 접근 하고 있는 접속지의 정보(컴퓨터이름, 접속위치, 브라우저)등을 확인하게 하고 감사기록에 저장하여 시스템의 이상유무와 해킹 유무를 확인한다.

WAVScanner의 3가지 모듈 중에서 제일 핵심이 되는 것이 바이러스 검사이다. 바이러스 검사는 파일 및 디렉터리를 검사할 수 있으며 패스워드가 압축되어 있지 않은 기본적인 압축 파일도 검사가 가능하다. 또한 디렉터리는 하위 디렉터리를 검사하는 재귀적인 검사가 가능하다. 이때 주의해야할 것은 아파치를 실행시키는 권한과 바이러스를 구동시키는 권한 설정을 확인하는 일이다. 웹 기반의 바이러스 서버는 특별한 권한을 가지고 구동되어야 하기 때문에 안티 바이러스 전용 서버로만 사용되어야 하며 검색할 디렉토리는 바이러스 모듈에 대한 읽기 권한을 부여해야한다. (그림 12)은 임시 디렉터리(/tmp)의 바이러스 검사 결과이다.

바이러스 검사 결과는 검사 대상파일을 나열하고 바이러스가 발견되지 않았으면 파일 이름과 함께 "OK"를 출력하고 바이러스가 발견되었을 때에는 파일 이름과 더불어 발견 바이러스 이름 그리고 "FOUND"을 출력하도록 구현하였다. 모든 파일을 검사한 후에는 바이러스 검사에 대한 결과를 나타내는데 여기에는 검사된 디렉터리, 파일 개수, 감염된 파일 개수, 검사 소요시간 등이 출력된다.

WAVScanner에 시그니쳐를 갱신하기 위해서는 관리자가 시그니쳐를 직접 갱신해주어야 한다. 이는 안티 바이러스 모듈이 현 구현단계에서는 독립적일 수밖에 없기 때문이다. WAVScanner는 주기적으로 작동할 수 있도록 주기 설정 모듈을 구현하였다. Unix System에서는 Cron을 이용하고 Windows 시스템계열에서는 Services의 설정을 통해서 이루어진다. 바이러스 검사 주기설정은 검색의 대상이 되는 디렉터리 입력창과 검색 요일별의 검색 요일 창, 그리고 예약 검색을 할 수 있는 예약 검색 창으로 구성하였다.

(그림 13)는 매주 "월화수목금"에 해당되는 날에 검사를 실행하며 12월 2일 2시 2분에도 검사를 실행하도록 바이러스 검색 주기를 설정한 후 설정결과를 나타낸 것이다. 이 내용은 관리자가 Unix System 일 경우 Crontab을 이용, Windows System 계열일 경우 Services 파일들을 확인하여



〈그림 13〉 바이러스 검색주기 설정

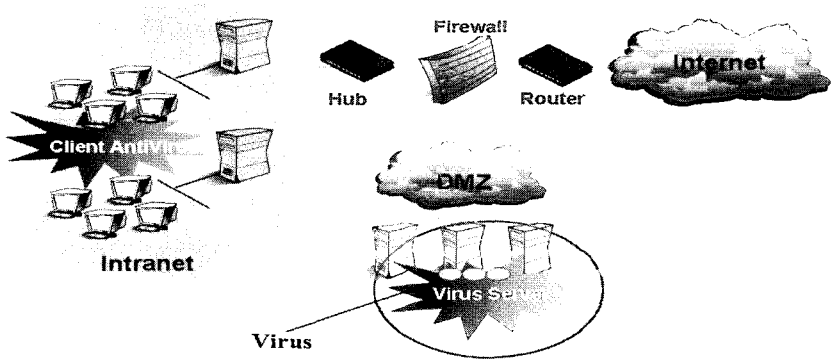
볼 수 있다.

4.2 성능평가

웹 기반의 안티 바이러스 시스템의 성능 평가를 위하여 (그림 14)와 같은 시스템을 구성하였다.

본 논문에서 제안한 방식은 다른 바이러스 시스템과 비슷한 성능을 보이고 있다. 그러나 본 논문은 바이러스를 검사 하고자 하는 네트워크 구조와 바이러스 및 웜에 대한 시그니쳐 발생 등 새로운 바이러스에 대항하는 관점에서 제안한 것이다. (표 4)는 A, B사의 기존 안티바이러스 제품과 본 논문이 제안하는 방법을 비교한 내용을 보이고 있다. 기존 제품들은 구동, 검사, 종료하기 위하여 원격 관리 콘솔 프로그램 등을 이용해야 하지만 본 시스템에서는 웹 브라우져를 통해 간단히 제어할 수 있으며 새로운 시그니쳐 등록을 사용자가 직접 할 수 있도록 하여 새로운 웜, 바이러스 등에 능동적으로 대응할 수 있다.

일반적인 서버 위치에는 Linux, Solaris 등 Unix 계열의 시스템이 설치되어 있으며, 동일한 버전의 운영체제가 탑재 되어있고 클라이언트 시스템의 PC는 Windows XP로 설치되어 있다.



〈그림 14〉 성능평가를 위한 네트워크 구성도

〈표 4〉 WAVScanner와 타제품의 검사구조 비교

구분	A	B	논문
바이러스 검출위치	server	server	server
바이러스 검사방법	원격 콘솔을 지원하는 프로그램설치 필요	원격 콘솔을 지원하는 프로그램설치 필요	웹 브라우져 필요
검출 확인방법	원격콘솔을 통한 모니터링	원격콘솔을 통한 모니터링	웹 브라우져를 통한 확인
기반구조	C/S 환경	C/S 환경	Web-Based 환경
시그니쳐업데이트	업데이트 서버에 의한 등록	업데이트 서버에 의한 등록	사용자 임의의 시그니쳐 즉시 등록 가능

<표 5> WAVScanner와 타제품의 성능 평가 비교

구분	A사	B사	제안한 모델
시그니쳐 개수	약 60,000	약 30,000	1,800
검사대상화일 개수	1,310	1,310	1,310
검사소요시간	2분	1분 42초	10초
바이러스검출률	100	100	100

따라서 성능평가는 바이러스 검출개수, 검출속도, 바이러스 검색위치 및 시스템 구성 등으로 측정한다. 본 논문에서 제안한 모델의 특수성 및 일반성을 검증하기 위해 (표 5)와 같이 기존 상업용 안티바이러스 제품 A 와 제품 B를 비교 측정하여 그 결과를 분석하였으며, 검출 오차 및 시스템 여건 등에 따른 성능 평가에 오차를 줄이기 위하여 3회 이상 반복 측정하여 결과를 평균값으로 산정하였다. 검사 대상이 될 바이러스 샘플은 총 100개의 바이러스를 파일에 내장하여 테스트 하였으며, 가장 중요한 시그니쳐의 개수는 검사 대상 프로그램에서 제시한 숫자를 그대로 인용하였다. (표 5)의 결과에서와 같이 다른 제품에 비해 본 논문에서 제안한 방법이 A, B사에 비해 빠른 검출 속도를 가지고 있다는 것을 볼 수 있다.

5. 결론 및 발전 방향

본 논문에서는 일반적인 클라이언트의 바이러스 백신이 아닌 웹기반 안티 바이러스 스캐너(WAVScanner)를 설계 및 구현 하였다. 구현된 웹기반 안티 바이러스 스캐너는 리눅스 및 Unix 상에서 동작 하도록 설계 되었으며, NT 및 Windows2000 서버 등에도 포팅이 가능하도록 설계되었다. 특히, 웹을 기반으로한 인터페이스이기 때문에 관리자는 어느 곳에서나 관리할 수 있으며, 새로운 바이러스 발견시 바로 시그니쳐를 업데이트를 할 수 있다는 장점과 함께 어디에서든 바이러스를 서버로 올려 검사, 분석할 수 있다는 장점도 가지고 있다.

그러나 지금도 새로운 바이러스의 출현은 계속되고 있으며 현재까지 제시된 모든 형태의 바이러스 시그니쳐를 가지고 있지 못하기 때문에 본 논문에서는 바이러스 백신의 새로운 개념을 제시하고 설계 및 구현하여 가능성을 보였으며 만약 향후에 출현하는 바이러스들에 대한 시그니쳐가 바이러스 백신 회사들과 공유된다면 좋은 결과를 얻을 수 있을 것이다.

바이러스의 백신 개발 자체가 많은 부분이 오픈 되지 않고 공유하지 않는 성격을 지니기 때문에 향후의 연구과제 또한 많은 것이 남겨져 있다. Unknown virus Detection은 바이러스 출현 시 바이러스를 분석, 정의하여 시그니쳐로 만드는 것이 아니라 백신이 파일들을 검사시 패턴 비교와 확률 등을 통하여 바이러스가 유포되기 전에 찾아내어 제거하는 형태를 말한다. 현재 많은 이론적인 형태의 연구가 진행되고 있다. 추후 Unknown Virus Detection방법의 연구와 함께 본 논문에서 제시한 웹에 기반을 둔 바이러스 백신이 결합되면 바이러스에 대한 좋은 해결방법이 될 것이다.

참 고 문 헌

[1] Edwards, J, "Next-generation viruses present new challenges", Computer , Volume : 34Issue : 5, May 2001. Page(s) : 16-18

[2] Cass, S. "Anatomy of malice[Computer Viruses]", IEEE Spectrum, Volume : 39Issue : 11, Nov. 2001, Page(s) : 56-60

[3] Schreiner, K. "New viruses up the stakes on

old triks", IEEE Internet Computing, Volume : 6Issue : 4, July-Aug. 2002, Page(s) : 9-10.

[4] Subramanya, S.R. ; Lakshminarasimhan, N. "Computer viruses", IEEE potentials, Volume : 20Issue : 4, Oct.-Nov. 2001, Page(s) : 16-19.

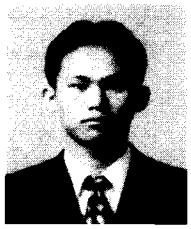
[5] Badhusha, A. ; Buhari, S. ; Junaidu, S. ; Saleem, M., "Automatic Signature files update in Antivirus software using Active Packets", Computer Systems and Applications, ACS/IEEE International Conference on. 2001 , 25-29 June 2001, pp. 457 -460

[6] Jeffeny O. Kephart and William C. Arnold, "Automatic Extraction of Computer Virus Signature", 4th conference, ford, edo, virus Bulletin Ltd, Abingdon, Englind, 1994, pp.179-194.

[7] Serpanos, D.N. ; Lipton, R.J. "Defense against man-in-the-middle attack in client-server systems", Computers and Communications, 2001. Proceedings. Sixth IEEE Symposium on, 2001, Page(s) : 9-14

[8] 황규범, 김광조, 안철수, "CIH 바이러스 분석 및 대책", 한국통신 정보보호학회 논문지, Vol.9, No.4, 1999.

◐ 저 자 소 개 ◑



이 상 훈
2001년 숭실대학교 컴퓨터학부 졸업(학사)
2003년 숭실대학교 대학원 컴퓨터학과 졸업(석사)
2003년~현재 : 숭실대학교 대학원 컴퓨터학과 박사과정
관심분야 : 침입 차단 · 탐지 시스템, 바이러스, 공개키 기반구조, etc.
E-mail : iam@leesanghun.pe.kr

김 원
1988년 숭실대학교 전자계산학과 졸업(학사)
1993년 숭실대학교 대학원 컴퓨터학과 졸업(석사)
1997년 숭실대학교 대학원 컴퓨터학과 졸업(박사)
1995~현재 : 전주기전여자대학 실용예술학부 조교수
관심분야 : 멀티미디어 통신, 멀티미디어 저작권 보호, 컴퓨터 알고리즘, etc.
E-mail : wkim@kijeon.ac.kr

도 경 화
1997년 건양대학교 컴퓨터공학과 졸업(학사)
1999년 숭실대학교 컴퓨터학과 졸업(석사)
2004년 숭실대학교 컴퓨터학과 졸업(박사)
2001년~2003년 2월 : 숭실대학교 생산기술연구소 연구원
2004년 4월~현재 : 행정자치부 전자정부지원센터 전문위원
관심분야 : 안티바이러스, 정보은닉, DRM, 네트워크보안, 데이터통신, 암호학, etc.
E-mail : khdo0905@dreamwiz.com

전 문 석
1980년 숭실대학교 컴퓨터공학과 졸업(학사)
1986년 University of Maryland 전산과 졸업(석사)
1989년 University of Maryland 전산과 졸업(박사)
1989년 Morgan State University 전산수학과 조교수
1989년~1991년 New Mexico State University부설 Physical Science Lab. 책임연구원
1991년~현재 : 숭실대학교 정보과학대학 정교수
관심분야 : 네트워크보안, 컴퓨터알고리즘, 병렬처리, VLSI 설계, 암호학, etc.
E-mail : mjun@computing.ssu.ac.kr