컴퓨터 바이러스 감염 예방을 위한 연구

김경수°, 노태상, 정민영, 김판구
조선대학교 전자계산학과

A Study to Prevent Computer Virus Infection

KyungSu Kim°, TaeSang No, MinYoung Chung, PanKoo Kim
Dept. of Computer Science, Chosun Univ.

요 약

요즘 새로운 컴퓨터 바이러스가 날로 늘어나고 있고, 또한 지능화되어 가고 있다. 기존의 시스템들은 그들을 막기 위해 화일의 내용에 대해 바이러스의 문자열을 비교하거나 바이러스의 행동을 모니터하는 방법으로 바이러스에 대처하고 있다. 그러나 이러한 방법들은 알려지지 않은 바이러스를 감지할 수 없고 일단 바이러스가 먼저 실행되면 예방 시스템이 감지할 수 없게 된다. 따라서, 바이러스 감지 시스템은 반드시 시스템이 바이러스에 감염되기 전에 수행되어야 한다. 본 논문에서는 시스템의 오버헤드를 증가시키지 않고 부팅시에 부트 바이러스가 실행되기 전에 안전한 부팅을 보장하고, 감염되었다 하더라도 부팅시에 안전하게 복구할 수 있고 아울러 화일 바이러스의 피해를 최소화할 수 있는 새로운 예방 기법인 VIDS(Virus Intrusion Detection System)을 설계, 구현하였다.

1. 서론

1971년 마이크로 프로세서가 등장하면서 가속화된 컴퓨터 기술의 발전으로 오늘날 개인용 컴퓨터를 누구나 소유하게 되었고 컴퓨터를 개인의 작업 처리 및 정보 활용의 수단으로 이용하고 있다. 그러나, 최근 들어 많은 컴퓨터 이용자들이 개인의 중요한 정보를 잃어 버렸다든지 또는 시스템 자체가 동작 불능이 되어 버린 경험을 한 사례가 증가하고 있다. 1980년대 초부터 최근 몇년동안 일반 사용자 뿐만 아니라 컴퓨터 전문가들에게조차 컴퓨터 바이러스의 피해는 매우 심각하여 컴퓨터 바이러스에 대한 정확한 규명과 예방 내지는 치료에 대한 방법이 절실한 상태이다.최근에는 많은 외국산 바이러스들이 네트워크를 통해 대량으로 유입되고 국산 바이러스들이 만들어질 뿐만 아니라 심지어는 바이러스를 정확히 모르더라도 바이러스 프로그램을 만들 수 있는 라이브러리들이 제공되고 있어 1994년에는 국내 컴퓨터 사용자의 96.7%가 바이러스에 감염된 사실이 있는 것으로 보고되고 있다. 또한 전세계적으로 보면 1995년에 7,000여종 이상의 바이러스 및 변종과 함께 고도의 알고리즘을 이용한 악성 바이러스들이 계속 등장하고 있는 것으로 보고되고 있다[9][10][13][15][19].

컴퓨터 바이러스에 대응하는 기존의 방법은 컴퓨터 바이러스의 진단 및 치료 프로그램인 백신 프로그램이 있다[1]. 그러나, 이 방법은 컴퓨터 바이러스가 발견된 후 백신 프로그램이 나오기 전까지는 어느 정도 시간이 소요되며 사용자는 백신 프로그램이 개발되기 전까지 아무런 대책없이 계속 피해를 입을 수밖에 없다. 그래서 보다 적극적으로 바이러스에 대처하기 위해서는 컴퓨터 바이러스의 발생과정과 종류를 면밀히 분석하고 알려지지 않은 컴퓨터 바이러스를 미연에 방지할 수 있는 소프트웨어적, 하드웨어적 접근 방법이 필요하다. 지금까지는 주로 컴퓨터 바이러스 백신 프로그램을 개발하는데 주력하고 있어 알려지지 않은 바이러스를 미연에 방지하는 기능을 가진 시스템을 개발한 사례는 거의 없지만 컴퓨터 바이러스의 행동을 감시하여 감염이 되기 전에 발견할 수 있도록 하는 메모리 상주형 시스템이 개발되었다[14][20]. 이러한 시스템들은 바이러스의 행동이라 할 수 있는 화일의 속성을 바꾼다든지 혹은 디스크의 입출력 등을 잘 감시하고 있다가 바이러스를 감지한다. 그러나, 이러한 메커니즘은 MS-DOS의 특성상 안전한 시스템이 되지 못하고 바이러스가 미리 감염된 상태에서의 이

러한 시스템의 작동은 아무 의미가 없게 된다.

본 논문에서는 컴퓨터 바이러스 예방을 위해 새로 구축한 안전 영역으로부터 부트 바이러스의 감염을 예방할 수 있도록 시스템을 안전하게 부팅시키고 만일 감염되었을 경우 부트바이러스의 감염을 감지하고 안전 영역으로부터 원래의 부분을 복구하여 시스템을 다시 안전하게 부팅하는 시스템을 설계 구현하였다.

2. 제안된 예방방법

IBM 호환 PC의 MS-DOS상에서의 보호 메커니즘, 예를 들면 메모리 소유권에 대한 보호 메커니즘, 하드웨어 접근에 대한 보호 메커니즘등이 거의 전무한 상태이므로 같은 DOS상에서의 프로그램들은 서로 안전하게 수행되며 유지된다고 보장하기가 어렵다. 특히, 부팅시와 디스크 화일 시스템의 구조상 바이러스로부터의 취약성을 그림2.1과 2.2에 나타내었다.

프로그램 종류	바이러스 공격 가능성	감염 여부	비고
ROM BIOS	×	×	
CMOS Memory	×	×	
Disk Boot Sector(FDD)	○	○	
Partition Boot Sector(HDD)	○	○	
Disk Boot Sector(FDD)	○	○	
IO.SYS, MSDOS.SYS	○	○	
CONFIG.SYS	○	×	Trojan horses
COMMAND.COM	○	○	
AUTOEXEC.BAT	○	×	Trojan horses
EXE, COM	○	○	
BAT	○	×	Trojan horses
OVL,OV1,OV2	○	○	
MACRO FILES	○	○	MS Word macro

그림 2.1. 수행가능 화일들의 바이러스 감염 가능성 및 여부

번호	영 역 명	내 용	바이러스 공격 가능성
1	부트섹터	부트스트랩로더 디스크운영에 필요한 정보 DOS 종류,버전	○
2	FAT1	디스크 섹터의 사용여부 (불량, 사용중, 유휴)	○
3	FAT2	FAT1의 복사본	○
4	루트디렉토리	루트 디렉토리 화일 정보	○
5	데이타영역	화일,프로그램등의 데이타 저장공간	○

그림 2.2. 플로피 디스크 정보 저장 구조 및 감염 가능성

일반적으로, 바이러스에 대처할 수 있는 방법으로는 시스템에 다른 운영체제를 유지하든지, DOS 자체의 보안 메커니즘을 고치던지, 에뮬레이션시키든지 등의 방법이 필요하다. 즉, 바이러스 검사및 감시를 위해서는 다음과 같은 접근 방법이 취해져야 한다고 볼 수 있다.

1. DOS에 화일의 소유권을 갖도록 하는 보호 메커니즘 기능을 추가하는 방법
2. DOS가 메모리 소유권에 대한 보호 메커니즘을 갖도록 하는 방법
3. DOS가 메모리 보호에 대한 추가된 보호 메커니즘을 갖도록 하는 방법
4. DOS의 운영체제 외에 다른 운영체제를 유지하면서 다른 운영체제에서 바이러스를 감시하며 검사하도록 하는 방법
5. 현재의 DOS가 수행할 때마다 다른 프로세서의 수행방식을 흉내내는(에뮬레이션) 기법을 취하는 방법

위의 경우들에서 1-3까지는 기존의 DOS를 수정하는 방법인데, DOS를 변경하면 지금까지 개발된 DOS용 프로그램들의 효용성이 없어져 버릴 가능성이 있어 시스템을 바꾼다는 것보다는 또다른 PC 운영체제를 사용하는 방안이 더 나을 것 같다. 또한 5번의 경우는 이미 실험적으로 만들어진 시스템으로 VIDES라는 시스템에서 취하고 있는 방식이다. 하지만 이 방법에서는 8086 프로세서에 대한 소프트웨어 에뮬레이션이므로 속도가 기존의 PC/XT급 속도밖에 얻을 수 없어 현실성이 없으나 바이러스를 방지하고자 하는 시스템이 같은 DOS상에서 똑같이 수행된다면 보장받을 길이 없다. 또한 4의 경우는 DOS외의 다른 운영체제를 유지하면서 DOS내의 바이러스를 검사하거나 감시하는 시스템을 사용할 수 있으나 시스템 설계및 구현이 매우 어렵다.

본 연구에서는 앞의 방법에서와는 달리, DOS를 전혀 수정하지 않고, 또한 다른 운영체제를 이용하여 DOS 영역의 바이러스를 치료하는 것보다 DOS 화일 시스템과는 다른 화일 시스템을 갖도록 하는 즉, 비밀의 디스크 화일 시스템을 설계하고 만들어두어 DOS영역의 바이러스 프로그램들의 접근은 어렵게 하고, 여기에 시스템 무결성 검사를 위한 정보(안전정보)를 유지 관리하는 방법을 취한다. 이를 위해서는 DOS하의 제어권(영향력)을 미치지 않는 형태에서 바이러스 검사및 감시 메커니즘이 동작하도록 해야만 한다. DOS가 부팅되는 순서중에서 DOS영향력을 받지 않고 수행하는 부분은 주부트로더이기때문에 이 부분을 새로 만들어 DOS의 부팅된 상태가 바로 안전한 상태(virus-free)로 만들어만 준다면 즉, 안전한 상태에서 출발하여 다른 프로그램이 수행되도록 만 한다면, DOS 시스템을 안전하게(virus-free) 유지시킬 수 있는 가능성이 있게 된다. 또한 DOS상에서 바이러스 퇴치를 위해 수행되는 DOS 프로그램이 있다면 부팅이 안전하게 끝나기 전에 메모리에 상주시켜 놓으면 그 상태로부터 출발하여 수행되는 모든 프로그램의 안전을 보장할 수 있을 것이다. 즉, DOS 수행 중에 바이러스 감시및 검사를 위해 필요한 프로그램이 DOS가 수행되기 전에 먼저 수행되도록 하여 안전을 보장받을 수 있다.

2.1 디스크 화일 시스템 구조와 분할(partition)

먼저 DOS를 설치하기 전에 무료 소프트웨어인 FIPS(the First nondestructive Interactive

Partition Splition program)을 이용하여 디스크를 분할한 후, 포맷한다. 각 분할된 디스크 영역중 하나를 선택하여 DOS를 설치한다. 이때 설치된 주 부트 섹터의 내용은 새로이 변경된 프로그램이 들어간다. 이렇게 새롭게 구성된 로더는 DOS를 안전하게 로드시키고, 메모리에 바이러스 예방을 위한 DOS 프로그램을 메모리에 안전하게 상주시키는 일까지 수행한다. 그림 2.3은 분할 후 새로운 디스크 화일 시스템의 구조를 보이고 있다.

번호	영 역 명	내 용	비 고
1	변경된 주부트섹터	변경된 주부트스트랩 로더 분할 테이블들	안전상태
2	부트섹터	부트스트랩로더 디스크운영에 필요한 정보	"
3	FAT1	디스크 섹터의 사용여부 (불량, 사용중, 유휴)	.
4	FAT2	FAT1의 복사본	.
5	루트디렉토리	루트 디렉토리 화일 정보	.
6	데이타영역	화일,프로그램등의 데이타 저장공간	.
7	DOS와 다른 화일 시스템	보호 정보, 백업용 화일	안전상태

그림 2.3. 분할된 새로운 디스크 화일 시스템

2.2 주부트 로더 기능

새로이 변경된 주부트 로더의 기능은 다음 그림 2.4와 같다. 수정된 주부트 로더의 기능에서 안전성을 보장하기 위해서 이 로더는 그림 2.3의 분할 II에 여러 가지 정보를 DOS 화일 시스템과는 다른 시스템 형태로 저장해두고 유지 관리한다. 분할 II에는 분할 I의 1, 2영역에 대한 백업본을 넣어 둔다. 또한 DOS 부트 섹터의 내용이 변경되었는지를 알기 위해 Check-sum 정보를 유지해 두고, 부팅하기 전에 부트 바이러스에 감염되었는지를 검사하게 한다. 그 외에도 DOS시스템 화일인 IO.SYS, MSDOS.SYS, COMMAND.COM등과 그 외에 중요한 화일들을 선택적으로 안전 영역에 두어 무결성 검사를 할 수 있다. 이 분할 II 영역은 DOS 프로그램에서 화일 구조및 디스크 화일 시스템 구조를 모르기 때문에 쉽게 정보를 유추할 수가 없다. 이것은 ROM BIOS가 바이러스로부터 공격 가능하지 않기 때문에 ROM BIOS와 연계해서 주 부트 로더가 절대로 바이러스가 변경하지 못하도록 하면 이와 같은 안전 상태로 부팅이 되도록 하는 것이 가능하다.

2.3 부팅후 메모리 구조

부팅이 완료된 후 메모리 구조는 그림 2.5와 같고, 여기서 1,2,3,4는 시스템 영역이고 안전한 상태로 유지되는 부분이 된다. 번호 5, 7은 사용자 프로그램(명령어)이 수행될 부분이다. 6은 일반 화일 바이러스 및 램상주 화일 바이러스를 감시하기 위한 램상주 프로그램들이다. 이 부분도 안전 상태인 부분이다.

```
알고리즘 : 주부트 로더
 /* 시스템 영역에 속하는 모든 화일의 무결성 검사 */
 /* 램상주 프로그램을 메모리로 로드시킴        */
 /* DOS 부트 로더를 메모리로 읽고 수행시킴             */
 {
        디스크의 분할 II에 저장된 정보를 메모리로 읽음;
        if ( DOS 부트 레코드의 내용 존재)
        {
                if ( DOS 부트 레코드 내용 != 저장된 정보 )
                { /* DOS 부트 섹터가 부트 바이러스에 의해 감염됨 */
                        DOS부트 레코드가 변경됨을 알림;
                        exit();
                }
                else
                { /* 각 화일에 대해 루트 디렉토리를 탐색하여 분할 II에
                     저장된 정보와 비교하여 무결성 조사 */
                        IO.SYS 화일의 변경 여부 검사;
                        MSDOS.SYS 화일의 변경 여부 검사;
                        COMMAND.COM 화일의 변경 여부 검사;
                        시스템 설치자의 변경여부 조사;
                        램상주 프로그램 메모리로 로드;
                        DOS 부트 로더를 메모리로 로드함;
                        제어권을 DOS 부트 로더로 넘김;
                }
        }
        else
        { /* 바이러스로 파괴된 DOS 부트 영역을 복구함 */
                디스크 분할 II의 DOS 시스템 이미지를 디스크에 복사;
        }
 }
```

그림 2.4 주부트 로더 알고리즘

번호	메모리 내용
1	인터럽트 벡터 테이블
2	ROM BIOS Data 영역
3	IO.SYS, MSDOS.SYS
4	COMMAND.COM
5	사용자 프로그램
6	램상주 프로그램
7	사용자 프로그램

그림 2.5 부팅후 메모리 구조

3. 실험 및 평가

3.1 평가 항목

이 절에서는 기존의 예방 시스템인 VPS[2], VIDES[14]과 본 시스템인 VIDS에 대한 특성및 우수성 여부를 파악하기 위해 각 방법에 대해 다음과 같은 평가 항목을 설정하였다. 각 항목들을 보면 다음과 같다.

1. 방어력 평가 - 각종 컴퓨터 바이러스들이 시스템에 침입했을 때 견딜 수 있는 정도 평가
2. 시스템 크기에 따른 평가 - 각 예방 시스템을 구축함으로서 늘어나는 수행시간, 디스크 영역 차지 정도 평가
3. 복구 정도에 따른 평가 - 바이러스 피해를 입었을 경우 감염 부위 복구 가능 정도
4. 예방성에 따른 평가 - 예방 시스템이 바이러스가 감염시킨 후에 감지되는가 그 전에 감지되는 가를 보이는 항목
5. 최대 취약점에 따른 평가 - 각 예방 시스템별로 최대 결점들에 대한 내용을 평가

3.2 예방 방법들의 비교 및 평가

각 예방 방법들과 평가 항목별로 해당 사항을 다음 표 3.1에 제시한다. 표 3.1의 방어력 평가 부분에서 △는 방어가 안되는 경우도 있음을 나타낸다. ×표시는 완전히 방어력이 없음을 나타낸다. 시스템 크기에 따른 평가부분에서 시스템 속도는 기존의 DOS시스템과 비교하여 느려지는 정도를 나타낸 것으로, VIDES는 에뮬레이션 방식을 취함으로 매우 느리고, 그 외의 시스템은 기존의 DOS에서 몇 가지의 루틴을 추가한 정도이므로 속도가 크게 느려지지 않을 것 같다.

방어력 측면에서 보면 VIDES보다는 VIDS가 약간 더 우수한 방어력을 가지고 있다고 볼 수 있다. 특히 시스템 속도측면에서 본다면 VIDS와 VPS가 기존의 DOS와 거의 같은 속도를 가지고 수행되는 것으로 보인다. 감염 부위 복구 가능성을 보면 VIDS가 매우 우수하다는 것으로 나타난다. 대체적으로 본 연구에서 제안한 VIDS 시스템의 평가 내용이 다른 시스템의 경우보다 여러 가지 면으로 좋은 결과를 보이는 것으로 평가할 수 있다. 각 시스템의 취약점및 결점을 보면 VPS는 예방의 완벽성에 있어서 신뢰성이 떨어진다고 볼 수 있고, 기존의 DOS 사용자에게 부담을 줄 수 있는 시스템이라고 볼 수 있다. 그리고 VIDES는 어느 정도 완벽한 예방방법이지만 속도가 매우 느리다는 것이 결점이라고 볼 수 있다.

4. 결론

본 연구에서는 컴퓨터 바이러스를 예방하기 위한 시스템을 설계, 구현하였다. 이를 위해 기존의 바이러스 내용및 예방 시스템을 분석한 결과, 보호 기능 시스템이 전혀 고려되지 않은 DOS같은 단일 사용자 시스템에서는 바이러스가 먼저 수행되면 기존의 바이러스를 막는 프로그램으로는 바이러스의 감염을 원천적으로 막을 수는 없다는 것을 알았다. 따라서 본 연구에서는 DOS를 전혀 수정하지 않고, 또한 다른 운영체제를 이용하여 DOS 영역의 바이러스를 치료하는 것보다 DOS와 다른 간단한 구

표 3.1 예방 방법들의 비교 평가

평가항목 \ 예방시스템		VPS	VIDES	VIDS
방어력 정도에 따른 평가	-부트 바이러스 방어력	△	△	○
	-일반 화일 바이러스 방어력	△	△	△
	-램상주형 화일 바이러스 방어력	△	×	○
	-A:로부팅,바이러스 감염가능성여부	×	×	×
	-unknown 바이러스에 견디는 힘	△	△	△
	-예방 시스템 자체 방어 여부	△	○	○
시스템 크기에 따른 평가	-시스템 구축시 드는 작업량 정도	多	多	多
	-시스템 가동시 시스템 속도의 정도	비슷	느림	비슷
복구정도에 따른 평가	-부트 부분 복구	△	?	○
	-감염 화일 복구	△	△	△
	-예방 시스템 자체 감염 부위 복구	×	?	○
예방성에 따른 평가	-부트부분감염	감염후	감염후	감염후
	-화일 감염	감염전	감염전	감염전
	-램상주 감염	감염전	?	감염전
최대 결점	-최대 결점들에 따른 평가	예방 완벽성 신뢰도 떨어짐	속도 느림	작업량 과다

성 요소를 갖는 화일 시스템을 갖도록 하는 즉, 알려지지 않는 디스크 화일 시스템을 설계하고 만들어 두어 DOS영역의 바이러스 프로그램들의 접근은 어렵게 하고, 여기에 시스템 무결성 검사를 위한 정보(안전정보)를 유지 관리하는 방법을 취하였다. 이를 위해서 DOS 영향력을 받지 않고 바이러스 검사및 방지가 가능하도록 하기 위해 DOS 부팅 순서 중에서 DOS영향력을 받지 않고 수행하는 부분인 주부트 섹터의 주부트 로더를 수정하여 이 프로그램으로 하여금 DOS가 부팅된 상태가 바로 안전한 상태(virus-free)가 되도록 만든다. 몇 가지의 예방 시스템과 비교하여 방어력 측면, 시스템 속도측면, 감염 복구 가능성 측면등을 보면, 대체적으로 본 연구에서 제안한 VIDS시스템의 평가내용이 다른 시스템의 경우보다 좋은 결과를 보이는 것으로 평가할 수 있다.

참고 문헌

1. 이서로, 파워해킹 테크닉, 파워북, 1995
2. 박명순, 컴퓨터바이러스 - 분석, 제작 및 예·처방, 기한재, 1992
3. 양성무, 컴퓨터 바이러스(달갑지 않은 프로그램(UP)), 태성 출판사, 1993
4. 정윤기, 컴퓨터 바이러스! 그것이 알고 싶다, 크라운 출판사, 1995

5. 정윤기, 컴퓨터 바이러스의 모든것, 크라운 출판사, 1992
6. 안철수, 바이러스 분석과 백신 제작, (주) 정보시대, 1995
7. 황희융, 바이러스 퇴치, 교학사, 1993
8. 한성국, IBM PC 기술사전, 집문당, 1991
9. 한국전산원, 정보화 역기능 현황 및 분석, 한국전산원 보고서 NCA Ⅱ-AER-9495, 1994
10. 한국전산원, 컴퓨터 바이러스 감염 예방 시스템에 관한 연구, 한국전산원 보고서 NCA Ⅵ-RER-9567, 1995
11. D. Russell & G.T. Gangemi Sr, Computer Security Basics, O'Reilly & Associates Inc, 1992.
12. S. Garfinkel & G. Spafford, Practical UNIX Security, O'Reilly & Associates Inc, 1994
13. J. Hruska, Computer Viruses and Anti-Virus Warfare, Ellis Horwood, 1990.
14. Morton Swimmer, "A Virus Intrusion Detection Expert System", Proceeding of the eicar Conference '95, Zuerich, Nov. 1995.
15. Igor G. Muttick, "Virus against Antivirus World", Proceeding of the eicar Conference '95, Zuerich, Nov. 1995.
16. Fridrik Skulason, "The Evolution of Polymorphic Viruses", Proceeding of the eicar Conference '95, Zuerich, Nov. 1995.
17. Igor Daniloff, "New Polymorphic Random Decoding Algorithm in Viruses", Proceeding of the eicar Conference '95, Zuerich, Nov. 1995.
18. Dmitry O. Gryaznov, "Scanners of The Year 2000: Heuristics", Proceeding of the eicar Conference '95, Zuerich, Nov. 1995.
19. Urs E. Gattiker, "Computer Viruses in the Wild: What is the Threat for Canada?", Proceeding of the eicar Conference '95, Zuerich, Nov. 1995.
20. Mikko Hypponen, "Virus Activation Routines", Proceeding of the eicar Conference '95, Zuerich, Nov. 1995.
21. Thunderbyte, TBAV User Manual, Thunderbyte B.V., 1995.