특집

컴퓨터 바이러스와 인공 생명

安 哲 秀
安哲秀컴퓨터바이러스硏究所長

컴퓨터 바이러스'는 이미 일반인들도 잘 알고 있을 정도로 대중화된 용어이다. 그러나 신문이나 TV 등의 각종 매스컴을 통해서 알려지다보니 여러 가지 잘못된 개념들이 통용되고 있는 실정이다. 또한 컴퓨터 바이러스가 인공 생명의 한 가지 형태가 될 수 있지 않은가에 대해 관심을 표명하는 사람들도 많다.

본고에서는 먼저 컴퓨터 바이러스 자체에 대한 기본적인 지식, 즉 정의, 동작 원리, 분류, 발전단계에 대해서 설명한 다음에, 컴퓨터 바이러스가 인공 생명의 정의와 어떠한 관련이 있는가에 대해 다루어 보고자 한다.

I. 컴퓨터 바이러스의 정의

컴퓨터 바이러스란 컴퓨터를 다루는 사용자나 컴퓨터 자체에 감염되는 바이러스가 아니라, 컴퓨터에서 실행되는 프로그램의 일종이다. 그러나 이 프로그램은 다른 프로그램들과는 달리 사용자 몰래 자신을 다른 곳에 복사하는 명령어들을 가지고 있다. 바이러스라는 이름이 붙은 이유는 생물학적인 바이러스가 자신을 복제하는 유전인자를 가지고 있는 것처럼 컴퓨터 바이러스도 자신을 복사하는 명령어들을 가지고 있기 때문이다. 따라서 컴퓨터 바이러스라는 말보다는 바이러스 프로그램(virus program)이라는 말이 더 정확한 표현이라고 할 수 있다.

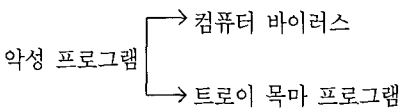
또한 컴퓨터 바이러스는 자기 복사 능력 이외에도 실제 바이러스와 비슷하게 부작용(side effect)을 가지고 있는 경우가 많다. 즉, 감기 바이러스가 인체내에서 증식만 하는 것이 아니라 감기를 일으키듯이, 컴퓨터 바이러스도 자신을 복사하는 명령어들의 조합만을 가지고 있지 않고 하드 디스크에 저장된 귀중한 자료들을 지워버리는 등의 일을 수행하는 명령어들을 포함하는 경우가 많다. 컴퓨터 바이러스가 사람에게 경계의 대상이 되는 이유는 사용자 몰래 자신을 복사하는 데 있는 것이 아니라 그 부작용 때문이다. 그러나 모든 컴퓨터 바이

러스가 이 부작용을 가지고 있는 것은 아니고, 자기 복사 능력만 가지고 있는 것들도 많다.

결론적으로 컴퓨터 바이러스는 '사용자 몰래 다른 프로그램에 자신을 복사하는 프로그램'이라고 정의할 수 있다. 더 정확하게는 '컴퓨터 프로그램이나 실행 가능한 부분을 변형해, 여기에 자신 또는 자신의 변형을 복사하는 명령어들의 조합'이다. 실행 가능한 부분의 예로는 오버레이(overlay), 장치 구동기(device driver), 부트 레코드, 운영체제 등을 들 수 있다. 또한 자신의 변형이란 용어를 사용한 이유는 컴퓨터 바이러스에 따라 자신을 그대로 다른 곳에 복사하는 것이 아니라 자신을 일부 변형시켜 다른 곳에 복사하는 컴퓨터 바이러스도 있기 때문이다.

II. 트로이 목마 프로그램

컴퓨터 바이러스는 여러가지 나쁜 영향을 미치는 악성 프로그램(malicious program)이지만, 악성 프로그램이 모두 컴퓨터 바이러스는 아니다. 개인용 컴퓨터에서 문제가 되는 악성 프로그램은 크게 컴퓨터 바이러스와 트로이 목마 프로그램으로 나눌 수 있다(그림 1). 대형 컴퓨터는 이들 외에 벌레(worm) 프로그램이 존재하고 있지만, 개인용 컴퓨터는 문제가 되지 않는다.



〈그림 1〉 악성 프로그램의 종류

트로이 목마 프로그램은 자기 복사 능력없이 고의적인 부작용만 가지며, 프로그래머의 실수로 포함된 프로그램의 버그(bug)와는 다르다. 또한 자기가 속해 있는 프로그램 내에서만 존재하고 다른 곳으로 자신을 복사하지 않는 점이 컴퓨터 바이러스와 다르다. 따라서 어떤 프로그램을 실행시켰을 때, 하드 디스크의 파일들을 지우지만 다른 프로그램에 복사되지 않으면, 이는 트로이 목마 프로그램이다.

컴퓨터 바이러스와 트로이 목마 프로그램, 자기 복사와 부작용간의 관계는 〈그림 2〉와 같다. 부작용의 유무에 관계없이 자기 복사 능력이 있으면 컴퓨터 바이러스이며, 자기 복사 능력없이 부작용만을 가지고 있으면 트로이 목마 프로그램이라는 것을 알 수 있다.

자기 복사	부작용	악성 프로그램
○	○	컴퓨터 바이러스
○	×	컴퓨터 바이러스
×	○	트로이 목마 프로그램

〈그림 2〉 컴퓨터 바이러스와 트로이 목마 프로그램 간의 구별

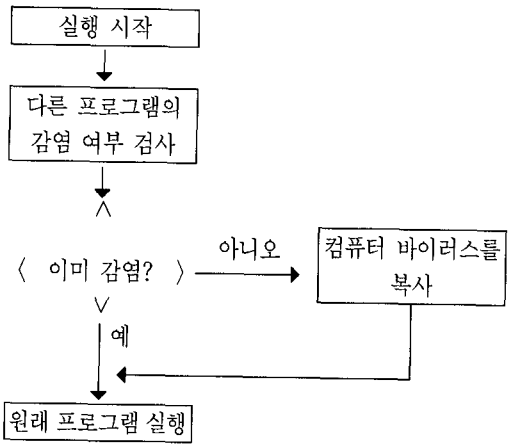
컴퓨터 바이러스와 트로이 목마 프로그램을 구별해야 하는 이유는 사용자 입장에서 대처 방법이 전혀 달라지기 때문이다. 컴퓨터 바이러스는 다른 프로그램에도 감염될 수 있기 때문에 한 프로그램에서 컴퓨터 바이러스가 발견되면 다른 프로그램들도 모두 검사해 봐야만 한다. 그러나 트로이 목마 프로그램은 한 프로그램 내에서만 존재하므로 그 프로그램만 지워버리면 문제가 해결된다. 따라서 컴퓨터에서 이상한 일이 발생했을 때는 컴퓨터 바이러스와 트로이 목마 프로그램을 반드시 구별할 필요가 있다.

트로이 목마 프로그램의 공통적인 특징은 사람들의 호기심을 자극하는 내용이거나 또는 기존의 유명 프로그램에 대한 새로운 소식이나 샘플 프로그램(sample program)의 가면을 쓰고 있는 점이다. 그 중 유명한 것이 RCKVIDEO.EXE다. 이 프로그램을 실행시키면 유명한 가수인 마돈나가 노래 부르는 모습을 보여준 뒤에 디스크의 모든 파일들을 지우고 "You are stupid to download a video about rock stars."(너는 가수에 대한 비디오 파일을 가져올 정도로 멍청하다)와 같은 내용을 담은 파일만을 남겨둔다. 또 다른 것으로 SEX-SHOW.EXE가 있다. 이것 또한 음란한 장면들을 보여 주면서 모든 파일을 지우는 것이다.

결론적으로 트로이 목마 프로그램의 정의는 '자기 복사 능력없이 부작용만을 일으킬 목적으로 만들어진 프로그램'이라고 할 수 있다. 더 정확하게는 '컴퓨터 프로그램 내에 사용자 몰래 고의로 포함된, 자신을 복사하지 않는 명령어들의 조합'이라고 할 수 있다.

III. 컴퓨터 바이러스의 동작 원리

컴퓨터 바이러스의 개념에 대한 이해를 돕기 위해 컴퓨터 바이러스의 기본적인 동작 원리를 설명하도록 하겠다(그림 3)



〈그림 3〉 컴퓨터 바이러스 동작 원리

컴퓨터 바이러스에 감염된 프로그램을 실행시키면 컴퓨터 바이러스도 일반 프로그램과 같이 컴퓨터의 기억장소에서 실행된다. 컴퓨터 바이러스는 사용자 몰래 다른 프로그램이 감염되었는지를 확인한 후, 프로그램이 이미 감염되었으면 그대로 놓아 두고 그렇지 않으면 자신을 복사한다. 컴퓨터 바이러스의 실행이 끝나면 원래 프로그램을 실행시킨다. 따라서 사용자는 원래 프로그램이 정상적으로 실행되므로 컴퓨터 바이러스에 감염된 것을 모르고 지나치기 쉽다. 이로 인해 새로 감염된 프로그램은 다음에 실행되면 같은 과정을 거쳐 다른

```
A)dir

 Volume in drive A has no label
 Directory of  A:\

JERUSAL  COM   1815  1-01-90   9:14p
                예루살렘 바이러스에 감염된 프로그램

TEST     COM      1   5-01-90   6:57p
      2 File(s)      324096 bytes free

A)jerusal      ←감염된 프로그램을 실행시킴
               ←당장은 아무런 일도 발생하지 않음
A)test         ←감염되지 않은 프로그램을 실행시킴

A)dir

 Volume in drive A has no label
 Directory of  A:\

JERUSAL  COM   1815  1-01-90   9:14p
TEST     COM   1814  5-01-90   6:57p
                예루살렘 바이러스에 감염됨
      2 File(s)      322560 bytes free
```

〈그림 4〉 프로그램이 예루살렘 바이러스에 감염되는 과정

프로그램을 감염시킨다. 따라서 하나의 프로그램이 감염되었다면 사용자가 모르는 사이에 컴퓨터 내의 거의 모든 프로그램에 컴퓨터 바이러스가 확산되는 것이다.

〈그림 4〉는 정상적인 프로그램이 예루살렘 바이러스(Jerusalem virus)에 감염되는 실제 예이다. 예루살렘 바이러스는 일명 '13일의 금요일 바이러스'라고 알려진 것이며, 평소에는 실행되는 프로그램들을 감염시키다가 13일과 금요일이 겹치는 날만 되면 실행되는 프로그램들을 지워버리는 특징적인 증상을 가지고 있다.

예루살렘 바이러스에 감염된 프로그램을 실행시켜도 감염되지 않았을 때와 동일하게 프로그램이 실행되며, 당장은 아무 일도 일어나지 않고 다른 프로그램이 감염되는 일도 없다. 그러나 이때 예루살렘 바이러스는 이미 기억장소에 그대로 존재하

면서 다른 프로그램들을 감염시킬 만반의 준비를 마친 상태가 된다. 그 다음부터는 프로그램을 실행시키면 실행시킨 프로그램들에 1813바이트 길이의 예루살렘 바이러스가 붙는다. DIR 명령으로 증가된 파일 길이를 살펴보면 이 사실을 확인할 수 있다.

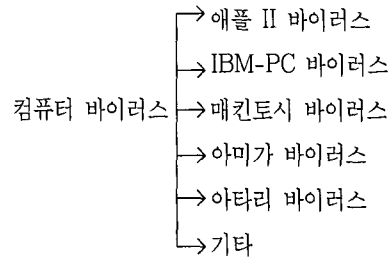
IV. 컴퓨터 바이러스의 분류

'사용자 몰래 다른 프로그램에 자신을 복사하는 프로그램'인 컴퓨터 바이러스는 지금까지 많은 수가 알려져 있다. 이들 중에서 공통적인 특징을 가진 것들을 모아 분류하는 작업은 학문적으로 필요할 뿐만 아니라 사용자들에게도 아주 필요한 일이다. 왜냐하면 컴퓨터 바이러스의 종류에 따라 사용자 입장에서 대처 방법이 다르기 때문이다. 그러나 컴퓨터 바이러스의 분류 방법은 통일되어 있지 않아서 많은 불편을 가져다 주고 있는 실정이다. 본고에서는 필자가 사용하고 있는 컴퓨터 바이러스의 분류 방법을 중심으로 설명하도록 하겠다.

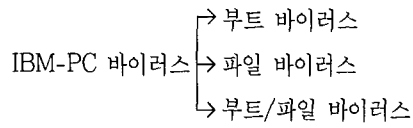
컴퓨터 바이러스를 나누는 가장 일반적인 분류 방법은 컴퓨터 바이러스가 감염되는 컴퓨터의 종류에 따른 분류다. 컴퓨터 바이러스는 그 컴퓨터가 가지고 있는 특정한 기능들을 이용하기 때문에 여러 기종에 감염되는 컴퓨터 바이러스란 존재하지 않기 때문이다. 따라서 컴퓨터 바이러스는 그 기종에 따라 애플(Apple) Ⅱ 바이러스, IBM-PC 바이러스, 매킨토시(Macintosh) 바이러스, 아미가(Amiga) 바이러스, 아타리(Atari) 바이러스 등으로 나눌 수 있다(그림 5).

개인용 컴퓨터 중에서는 IBM-PC가 대부분을 차지하고 있으며, 가장 문제가 되는 것도 IBM-PC 바이러스다. 본고에서는 IBM-PC 바이러스에 국한해 설명하도록 하겠다.

IBM-PC 바이러스는 감염되는 부위에 따라 부트 바이러스(boot virus), 파일 바이러스(file virus), 부트/파일 바이러스(boot/file virus)의 세 가지로 나눌 수 있다(그림 6).



〈그림 5〉 컴퓨터 기종에 따른 분류



〈그림 6〉 IBM-PC 바이러스의 분류

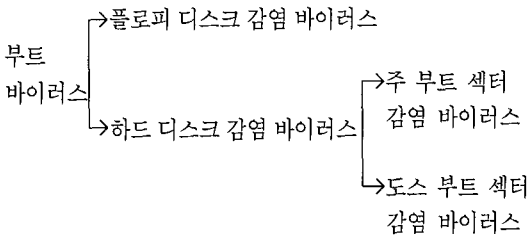
부트 바이러스는 디스크의 가장 처음 부분인 부트 섹터(boot sector)에 감염되는 것이며, 파일 바이러스는 일반적인 프로그램에 감염되는 바이러스다. 반면에 부트/파일 바이러스는 부트 섹터와 프로그램 모두에 감염되는 것을 말한다.

1. 부트 바이러스

컴퓨터를 처음 켰을 때는 디스크의 가장 처음 부분인 부트 섹터에 위치하는 프로그램이 제일 먼저 실행되는데, 여기에 자리잡는 컴퓨터 바이러스를 '부트 바이러스'라고 한다. 국내에서 최초로 발견된 컴퓨터 바이러스인 브레인 바이러스(Brain virus), 3월 6일만 되면 하드 디스크의 모든 내용을 지워버리는 것으로 악명을 떨쳤던 미켈란젤로 바이러스(Michelangelo virus) 등이 여기에 속한다.

부트 바이러스도 감염되는 디스크 종류에 따라 플로피 디스크에만 감염되는 것과 하드 디스크에도 감염되는 것으로 나눌 수 있다. 또한 하드 디스크에 감염되는 것들도 하드 디스크의 주 부트 섹터(master boot sector)에 감염되는 것과 도스 부트 섹터(DOS boot sector)에 감염되는 것으로 나눌 수 있다(그림 7).

플로피 디스크 감염 바이러스의 예로는 브레인 바이러스(Brain virus), LBC. Ⅱ 바이러스 등이 있

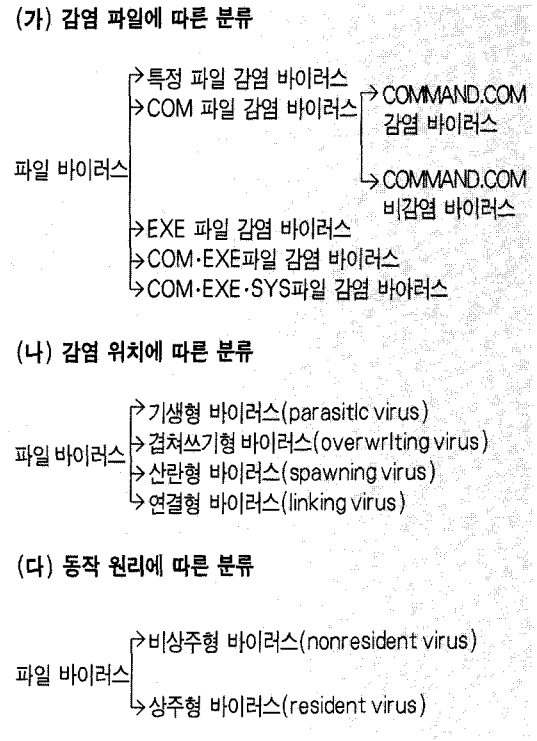


〈그림 7〉 부트 바이러스의 분류

다. 또한 주 부트 섹터 감염 바이러스에는 LBC 바이러스, 돌 바이러스(Stoned virus) 등이 있으며, 도스 부트 섹터 감염 바이러스에는 탁구 바이러스(Pingpong virus), 디스크 살해 바이러스(Disk-Killer virus) 등이 있다.

2. 파일 바이러스

'파일 바이러스'란 일반적인 프로그램에 감염되는 컴퓨터 바이러스를 말한다. 이때 감염되는 프로그램들은 COM 파일, EXE 파일 등의 실행 파일(executable file), 오버레이 파일(overlay file), 주변기기 구동 프로그램 (device driver) 등이다. 13일의 금요일 바이러스로 더 잘 알려진 예루살렘 바이러스(Jerusalem virus), 90년에 한창 위세를 떨쳤던 어둠의 복수자 바이러스(Dark-Avenger virus) 등이 여기에 속한다.



〈그림 8〉 파일 바이러스의 분류

파일 바이러스는 파일에 감염되기 위해 여러가지 다양한 전략을 사용하기 때문에, 어떤 한가지 방법으로 분류하기가 곤란하다. 따라서 여러 가지 분류 방법을 동시에 사용하고 있다(그림 8).

(1) 감염 파일에 따른 분류

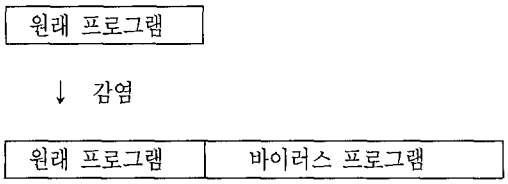
가장 일반적인 분류 방법은 감염되는 프로그램의 종류에 따른 것이다(그림 8 (가)). 파일 바이러스는 감염되는 프로그램에 따라 특정 파일 감염 바이러스, COM 파일 감염 바이러스, EXE 파일 감염 바이러스, COM EXE 파일 감염 바이러스와 COM EXE SYS 파일 감염 바이러스의 5가지로 분류할 수 있다. 이 중에서 COM 파일 감염 바이러스는 COMMAND.COM에 감염되는 것과 감염되지 않는 것이 있다. COM EXE SYS 감염 바이러스는 일반적인 실행 프로그램인 COM 파일과 EXE 파일뿐만 아니라 주변기기 구동 프로그램인 SYS 파일까지도 감염시키는 것이다.

대표적인 특정 파일 감염 바이러스로는 COMMAND.COM에만 감염되는 르하이 바이러스(Lehigh virus)가 있다. COM 파일 감염 바이러스로는 대만 바이러스(Taiwan virus), 비엔나 바이러스(Vienna virus) 등이 있으며, EXE 파일 감염 바이러스에는 맥가이버 바이러스(McGyver virus), 로즈버드 바이러스(Rosebud virus) 등이 있다. COM EXE 파일 감염 바이러스로는 예루살렘 바이러스, 어둠의 복수자 바이러스 등이 있으며, COM EXE SYS 파일 감염 바이러스에는 새해 인사 바이러스(Happy-New-Year virus)가 대표적이다.

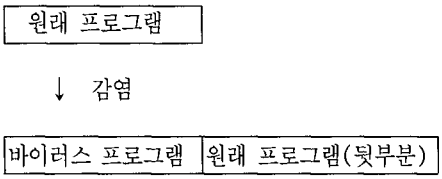
(2) 감염 위치에 따른 분류

파일 바이러스에 대한 두 번째 분류 방법은 파일 바이러스가 프로그램 내의 어디에 위치하느냐에 따르는 것이다(그림 8 (나)). 이에 따른 분류

(가) 기생형 바이러스



(나) 겹쳐쓰기형 바이러스



〈그림 9〉 기생형 바이러스와 겹쳐쓰기형 바이러스의 비교

로 파일 바이러스는 기생형 바이러스(parasitic virus), 겹쳐쓰기형 바이러스(over writing virus), 산란형 바이러스(spawning virus), 연결형 바이러스(linking virus)로 구분된다.

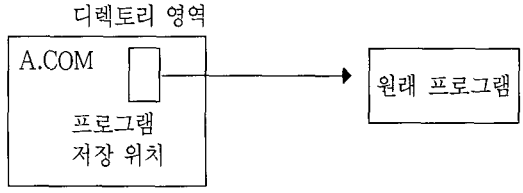
원래 프로그램을 파괴하지 않고 프로그램의 앞이나 뒤에 바이러스 프로그램이 붙는 것을 기생형 바이러스라고 한다(그림 9 (가)). 기생형 바이러스에 감염된 파일에는 원래 프로그램과 바이러스 프로그램이 같이 존재하기 때문에 필연적으로 길이가 증가하게 되지만, 바이러스 프로그램이 실행된 다음에 원래 프로그램을 실행시키기 때문에 사용자가 컴퓨터 바이러스에 감염된 사실을 알지 못하는 경우가 많다. 예루살렘 바이러스, 어둠의 복수자 바이러스 등 대부분의 파일 바이러스들이 여기에 속한다.

원래 프로그램이 있는 곳에 바이러스 프로그램이 겹쳐 위치하는 것을 겹쳐쓰기형 바이러스라고 한다(그림 9 (나)). 일반적인 겹쳐쓰기형 바이러스는 파일의 앞부분에 겹쳐 위치한다. 따라서 감염된 파일을 실행시키면 원래 프로그램은 실행되지 않고 바이러스 프로그램이 실행되며, 원래 프로그램이 파괴되었기 때문에 백신 프로그램으로도 복구할 수 없다. 단, 겹쳐쓰기형 바이러스가 프로그램에서 사용하지 않는 영역을 찾아 들어갈 경우에는 원래 프로그램의 수행에는 전혀 영향을 미치지 않게 되며, 백신 프로그램으로 복구가 가능하다.

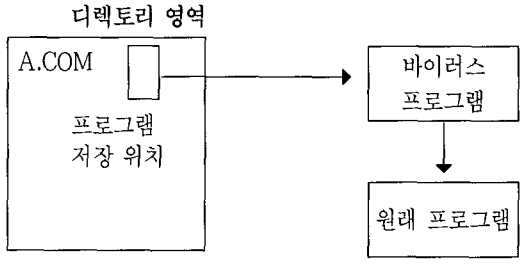
또한 기존 프로그램의 길이가 겹쳐쓰기형 바이러스보다 짧을 경우, 감염된 파일의 길이가 바이러스 프로그램의 길이와 동일하게 되지만, 원래 프로그램의 길이가 바이러스 프로그램보다 길 경우, 파일 길이가 달라지지 않아 사용자가 눈치채기 어렵다. 복구가 불가능한 겹쳐쓰기형 바이러스로는 문둥이 바이러스(Leprosy virus)가 있고, 복구가 가능한 겹쳐쓰기형 바이러스로는 르하이 바이러스가 있다.

산란형 바이러스는 EXE 파일에 직접 감염되지 않고 같은 이름의 COM 파일을 만들어 여기에 바이러스 프로그램을 넣어두는 것이다. 같은 이름의 COM 파일과 EXE 파일이 같은 디렉토리 내에 존재할 때 파일 이름을 입력하면 COM 파일이 우선적으로 실행되기 때문에, 파일 바이러스가 직접 감염된 경우와 같은 효과를 나타내게 된다. 산란형 바이러스의 대표적인 예로는 AIDS Ⅱ 바이러스를 들 수 있다.

(가) 연결형 바이러스에 감염되기 전



(나) 연결형 바이러스에 감염된 후



〈그림 10〉 연결형 바이러스의 동작 원리

연결형 바이러스도 프로그램에 직접 감염되는 것이 아니라, 디렉토리 영역에 저장된 프로그램의 시작 위치를 바이러스 프로그램의 시작 위치로 바꾸어 줌으로써 컴퓨터 바이러스로서의 동작을 수행하는 것이다(그림 10). 따라서 프로그램을 실행시키면 원래 프로그램 대신에 바이러스 프로그램이 먼저 실행되고, 바이러스 프로그램의 실행이 끝나면 원래 프로그램을 실행시켜 사용자가 눈치채지 못하게 만드는 것이다. 연결형 바이러스의 대표적인 예는 Dir Ⅱ 바이러스를 들 수 있다.

(3) 동작 원리에 따른 분류

파일 바이러스에 대한 세번째 분류 방법은 동작 원리에 따른 것이다(그림 8 (다)). 일반적인 프로그램들처럼 한번 실행된 후에 기억장소에서 사라지는 것을 비상주형 바이러스(nonresident virus)라고 하며, 기억장소 상주 프로그램(RAM resident program)들처럼 기억장소에서 계속 존재하는 것을 상주형 바이러스(resident virus)라고 한다.

비상주형 바이러스의 경우에는 감염된 프로그램이 실행될 때만 바이러스 프로그램이 동작해 다른 프로그램을 감염시키지만, 상주형 바이러스의 경우에는 감염된 프로그램이 한번 실행된 다음에는 실행되는 프로그램들을 계속 감염시킬 수 있다.

비상주형 바이러스의 예로는 대만 바이러스, 비엔나 바이러스 등을 들 수 있으며, 상주형 바이러스의 예로는 예루살렘 바이러스, 어둠의 복수자 바이러스 등을 들 수 있다.

3. 부트/파일 바이러스

부트/파일 바이러스는 부트 섹터와 파일 모두에 감염되는 바이러스이며, 부트 부분에 대해서는 부트 바이러스의 분류 방법, 파일 부분에 대해서는 파일 바이러스의 분류 방법을 사용하면 된다. 부트/파일 바이러스의 예로는 침입자 바이러스(Invader virus), 테킬라 바이러스(Tequila virus) 등을 들 수 있다.

V. 컴퓨터 바이러스의 발전 단계

컴퓨터 바이러스의 발전 단계는 연구자들에 따라서 여러 이론이 있지만, 다음의 네 단계로 나누는 사람들이 많다(그림 11)).

발전 단계	바이러스 종류
제 1세대	원시형 바이러스
제 2세대	암호화 바이러스
제 3세대	은폐형 바이러스
제 4세대	갑옷형 바이러스

〈그림 11〉 컴퓨터 바이러스의 발전 단계

처음 출현한 제 1세대 컴퓨터 바이러스는 원시형 바이러스(primitive virus)다. 이것은 실력이 뛰어나지 않은 아마추어 프로그래머들에 의해 만들어졌으며, 컴퓨터 바이러스라는 것이 실제로 존재할 수 있다는 가능성을 증명하는 수준이었다. 원시 바이러스는 분석하기도 쉬우며, 백신 프로그램을 제작하기도 쉽다. 돌 바이러스(Stoned virus), 예루살렘 바이러스 등 대부분의 컴퓨터 바이러스들이 여기에 속한다.

다음에 출현한 제 2세대 바이러스는 암호화 바이러스(encryption virus)다. 이것은 어느 정도 실력을 갖춘 프로그래머들에 의해 만들어졌으며, 백신 프로그램이 진단하기 어렵게 하기 위해 프로그램의 일부 또는 대부분을 암호화시켜 저장한다. 그러나 실행이 시작되는 부분에 존재하는 암호를 푸는 부분은 항상 일정하므로 실효를 거두지 못했다. 폭포 바이러스(Cascade virus), 느림보 바이러스(Slow virus) 등이 좋은 예다.

제 3세대 바이러스는 은폐형 바이러스(stealth virus)다. 여기서 'stealth'라는 말은 스텔스 비행기에서 유래된 것이다. 즉, 스텔스 비행기가 레이다에 걸리지 않게 자신을 은폐하는 방법을 사용하는 것과 마찬가지로, 은폐형 바이러스도 사용자나 백신 프로그램의 진단 과정을 방해하는 기법을 사용하기 때문에 붙여진 이름이다.

은폐형 바이러스는 어느 정도 실력을 갖춘 프로

그래머들에 의해 만들어졌으며, 자신을 은폐할 뿐만 아니라 사용자나 백신 프로그램에게 거짓 정보를 제공하기 위해 다양한 기법을 사용하고 있다. 즉, 은폐형 바이러스는 기억장소에 존재하면서, 감염된 파일의 길이가 증가하지 않은 것처럼 보이게 하고, 백신 프로그램에서 감염된 부분을 읽으려고 하면 감염되기 전의 내용을 대신 보여줌으로써 컴퓨터 바이러스가 존재하지 않는 것처럼 백신 프로그래머나 사용자를 속이는 것이다. 브레인 바이러스, 조쉬 바이러스(Joshi virus), 512 바이러스, 4096 바이러스 등이 좋은 예다.

그러나 은폐형 바이러스라도 기억장소를 먼저 검사해 은폐기능을 무력화시키면 쉽게 진단할 수 있다. 일부 매스컴이나 광고에서 은폐형 바이러스는 대부분의 백신 프로그램으로 진단되지 않는 최첨단 기술을 사용한다고 주장하지만, 이것은 잘못된 것이다. 또한 최첨단 기술을 사용하는 일부 백신 프로그램만 이들을 진단할 수 있다는 말도 잘못된 것이다.

제 4세대 바이러스는 갑옷형 바이러스(armour virus)다. 제 2세대 암호화 바이러스와 제 3세대 은폐형 바이러스들은 백신 프로그램 자체가 공격목표였다. 즉, 백신 프로그램이 컴퓨터 바이러스들을 진단하기 어렵게 하기 위해 암호화나 은폐 기법을 사용했다. 그러나 이 시도가 무력화되자, 컴퓨터 바이러스 제작자들은 백신 프로그래머에게 공격의 화살을 돌리기 시작했다. 즉, 백신 프로그램으로부터 숨으려는 노력보다는 여러 단계의 암호화와 고도의 자체수정 기법 등을 동원함으로써, 백신 프로그래머가 컴퓨터 바이러스를 분석하고 백신 프로그램을 만들기가 까다롭게 만듦으로써 백신 프로그램 개발을 지연시키는 방법을 사용한 것이다. 이것이 갑옷형 바이러스다.

갑옷형 바이러스의 일종으로 다형성 바이러스(polymorphic virus)가 있다. 이것은 암호화 바이러스의 일종이지만, 암호를 푸는 부분이 항상 일정한 단순 암호화 바이러스와는 달리 암호를 푸는 부분조차도 감염될 때마다 달라진다. 다형성 바이러스 중에는 한 바이러스가 100만 가지 이상의 변형을 만드는 경우도 있다. 따라서 백신 프로그램에서 단순히 문자열을 비교하는 방법만으로는 진단하기가 불가능하며, 바이러스의 암호화 알고리즘(algorithm)을 내장하면 진단할 수 있지만 백신 프로그램 자체의 속도를 현저하게 저하시키는 점이 문제점으로 지적된다.

그러나 은폐형 바이러스와 마찬가지로 다형성 바이러스들도 진단이나 치료가 불가능한 것은 아니며, 실제로 대부분의 백신 프로그램들을 사용하면 진단과 치료가 가능하다. 따라서 이 경우에도 역시 갑옷형 바이러스 또는 다형성 바이러스가 대부분의 백신 프로그램으로 진단되지 않는 최첨단 기술을 사용하고 있다는 말은 잘못된 것이며, 최첨단 기술을 사용하는 일부의 백신 프로그램에서만 이들을 진단할 수 있다고 선전하는 것도 잘못된 것임을 알아두어야 한다.

지금까지 설명한 구분들은 다분히 인위적인 것이다. 바뀐 인터럽트 주소만을 은폐시키는 바이러스를 원시형 바이러스라고 분류해야 할지, 은폐형 바이러스로 분류해야 할지도 사람에 따라 의견이 다를 수가 있다. 복잡한 암호화 기법을 사용한 바이러스를 암호화 바이러스라고 해야 할지, 갑옷형 바이러스라고 해야 할지도 미묘한 문제다. 즉, 컴퓨터 바이러스의 특성에 따라 확실하게 분류할 수 없는 경우들이 존재한다.

또한 제 1세대~제 4세대의 구분은 컴퓨터 바이러스의 발전 단계이지, 시기적으로 순서대로 발견된 것을 뜻하는 것은 아니다. 한가지 예로, 브레인 바이러스는 아주 초기에 발견된 컴퓨터 바이러스이지만, 은폐기법을 사용하고 있기 때문에 제 3세대 바이러스로 분류할 수 있다. 또한 최근에 발견되는 컴퓨터 바이러스들이라도 제 1세대 바이러스에 속하는 것이 많다. 따라서 지금은 제 1세대 바이러스부터 제 4세대 바이러스까지 공존하는 상태라고 말할 수 있다.

VI. 인공 생명으로서의 컴퓨터 바이러스

컴퓨터 바이러스가 인공 생명의 범주에 들어갈

수 있는지를 알기 위해서는 먼저 생명이란 무엇인지에 대한 정의를 알 필요가 있다. J.Doyne Farmer와 Jon A. Rochlis에 의하면 생명이란 다음과 같은 특징을 가진다고 한다.

- 생명은 특정한 물질이라기보다는 공간-시간상에 존재하는 하나의 패턴이다.
- 스스로 또는 관련된 물질내에서 자기 복제를 한다.
- 자신을 표현할 수 있는 정보를 저장한다.
- 물질이나 에너지를 이용한 대사기능이 있다.
- 주위 환경과 기능적인 상호작용을 한다.
- 구성 부분들이 상호의존적인 관계를 가진다.
- 주위 환경이 바뀌더라도 안정성을 유지한다.
- 진화할 수 있는 능력을 가진다.
- 자라거나 확장한다.

Eugene H. Spafford는 'computer viruses as artificial life'라는 논문에서 이러한 각각의 특징들이 컴퓨터 바이러스에 어떻게 적용되는지를 심도있게 논의한 바 있다. 본고에서는 그의 생각을 바탕으로 하고, 여기에 필자의 의견을 추가하도록 하겠다.

1. 공간-시간 상의 패턴

컴퓨터 바이러스는 소프트웨어의 일종이다. 소프트웨어는 컴퓨터 명령어의 조합이며, 컴퓨터 시스템에서 시간에 따라 순차적으로 실행되는 패턴이라고 볼 수 있다. 그러나 소프트웨어가 공간상으로 존재하는지에 대해서는 논란의 여지가 있다. 소프트웨어는 컴퓨터의 기억장치나 저장장치에서 전기나 자기 필드의 특정 변화의 조합으로 존재하기 때문에, 에너지의 패턴으로밖에는 볼 수 없기 때문이다.

2. 자기 복제 능력

컴퓨터 바이러스의 자기 복제 능력에 대해서는 의문의 여지가 없다. Eugene H. Spafford는 복제의 주체가 컴퓨터 바이러스가 아니라 컴퓨터이기 때문에 컴퓨터 바이러스의 자기 복제 능력에 대해서 의문을 표시했다. 그러나 필자의 견해로는 생물학적인 바이러스의 경우에도 스스로 복제를 하는 것이 아니라 숙주 세포의 기능을 빌려서 복제를 하기 때문에, 컴퓨터 바이러스는 자기 복제 능력이 있다고 보는 것이 옳다고 생각한다.

3. 자신을 표현하는 정보 저장

이 사항에 대해서는 이론의 여지가 없다. 컴퓨터 바이러스도 자신에 대한 모든 정보를 가지고 있으며 복제할 때 이 정보를 사용하고 있다.

4. 대사

컴퓨터 바이러스는 컴퓨터에서 실행이 되고 다른 프로그램을 감염시키는 과정에서 전기 에너지를 사용하고 있으며, 이러한 관점에서는 대사를 한다고 볼 수 있다. Eugene H. Spafford는 자기 복제 능력의 경우와 마찬가지로 에너지 사용의 주체가 컴퓨터 바이러스가 아니라 컴퓨터 자체라는 데 문제를 제기하고 있다. 그러나 필자의 견해로는 생물학적인 바이러스도 숙주의 대사 기능에 의존한다는 관점에서 볼 때는 무리가 없다고 생각한다.

5. 환경과의 기능적인 상호작용

컴퓨터 바이러스도 작동하기 위해서 컴퓨터의 운영체제를 변형한다. 즉, 컴퓨터 시스템, 기억장소, 파일들을 검사하고 기억장소내의 인터럽트 핸들러나 파일을 변형하는 일을 한다. 따라서 컴퓨터 바이러스도 환경과 기능적인 상호작용을 한다고 볼 수 있다.

6. 구성 부분간의 상호의존성

생명체의 각 부분을 분리해놓으면 생명체가 살아 있을 수 없듯이, 컴퓨터 바이러스도 각 부분을 분리해놓으면 작동을 하지 못한다. 단, 생명체의 경우에는 분리한 것을 다시 붙여놓아도 소용없지만, 컴퓨터 바이러스의 경우에는 붙여놓으면 다시 원래대로 작동한다는 점이 다르다.

7. 환경 변화에 대한 안정성 유지

컴퓨터 바이러스는 소프트웨어적인 구성이 다른 여러 컴퓨터에 감염될 수 있으며, 제한적이기는 하지만 운영체제가 달라도 감염될 수 있다. 즉, MS

-DOS에서 작동하는 컴퓨터 바이러스는 윈도우 3.1이나 윈도우 95에서도 작동할 수 있다. 또한 마이크로소프트 워드에서 작동하는 매크로 바이러스의 경우에는 컴퓨터 기종에 관계없이 마이크로소프트 워드 프로그램이 실행되는 환경이라면 성공적으로 다른 파일을 감염을 시킬 수 있다.

8. 진화

기존의 컴퓨터 바이러스가 조금 변형되는 경우는 흔히 볼 수 있으며, 기능이 추가된 새로운 형태로 만들어지는 경우도 많다. 단, 이러한 변화의 주체는 컴퓨터 바이러스가 아니라 컴퓨터 바이러스를 만든 프로그래머이다. 물론 컴퓨터 바이러스에 진화하는 기능을 넣어서 만들 수도 있지만, 이 경우에도 변화의 주체는 프로그래머라고 보는 것이 옳을 것이다.

단, 가끔 복제되는 과정에서 프로그램의 오동작이나 자기 매체의 잘못으로 저절로 변형이 생기는 경우도 있지만, 이것이 성공적으로 동작할 확률은 극히 미미한 정도이다.

9. 성장

주어진 환경내에서 일정 시간에 많은 수를 만든다는 의미에서 컴퓨터 바이러스도 성장을 한다고 볼 수 있다.

10. 기타 특징들

컴퓨터 바이러스 중에는 자기 치료 능력을 가진 것도 있다. 양키두들 바이러스들중의 일부는 자기 자신이 복사되는 과정에서 또는 다른 프로그래머에 의해서 일부가 변형되더라도, 자기 복제 과정에서 원래의 패턴을 복구한 다음에 다른 파일을 감염시킨다.

대부분의 컴퓨터 바이러스는 같은 종이 이미 감염되어 있을 때는 거기에 다시 감염되지 않는 기능을 기본적으로 가지고 있다. 생물계처럼 같은 종이 이미 차지하고 있는 능력을 인정해준다고 볼 수 있다.

또한 컴퓨터 바이러스 중에는 다른 종류의 바이러스를 파괴하는 능력을 가진 것도 있으며, 특정한 컴퓨터 바이러스를 변형시키거나 새로운 기능을 추가해주도록 프로그램된 것들도 있다. 즉, 생물 세계처럼 잡아먹거나 공생 관계가 존재하는 셈이다.

VII. 결 론

컴퓨터 바이러스는 기본적으로 인공 생명의 정의에 부합되는 많은 유사점들을 가지고 있지만, 그에 못지않게 많은 차이점도 존재하는 것이 사실이다. 따라서 생명의 정의 자체를 혁신적으로 바꾸기 전에는, 컴퓨터 바이러스를 살아있다고 보기는 어렵다고 생각된다.

그러나 인공 생명에 대한 정의를 만들 때 컴퓨터 바이러스는 많은 도움을 줄 수 있을 것이다. 컴퓨터 바이러스를 인공 생명의 범주에 포함시키느냐 또는 제외하느냐에 따라 인공 생명의 정의는 현저하게 달라질 것이며, 인공 생명의 정의를 새로 만들었을 때 컴퓨터 바이러스를 대상으로 시험해 볼 수 있을 것이다.

또한 컴퓨터 바이러스를 생명을 모델링하는 도구로 사용할 수도 있을 것이다. 컴퓨터 바이러스에 특정한 행동 양식을 모델링해서 복잡한 시스템에서 행동하는 양식을 관찰함으로서 그 시스템에 대한 이해도를 높일 수 있을 것이다.

저 자 소 개

安 哲 秀

1962年 2月 26日生
1986年 2月 부산고등학교 졸업
1986年 2月 서울대 의대 졸업
1988年 2月 서울대 대학원 의학 석사 학위 취득
1991年 2月 서울대 대학원 의학 박사 학위 취득
1995年 3月 현재 미국 University of Pennsylvania 공학 석사 과정 재학중

1986年 3月～1989年 9月 서울대학교 의과대학 조교
1989年 4月～1991年 2月 단국대학교 의과대학 전임강사
1991年 2月～1994年 4月 군의관으로 군복무
1986年 3月～현재 대학의학협회 회원
1993年 7月～1995年 2月 대학의학협회 의학정보연구위원회 위원
1995年 2月～현재 대한의학협회 전산 발전위원회 위원
1990年 12月～현재 월간 마이크로소프트웨어 편집 자문위원
1995年 2月～현재 (주)안철수컴퓨터바이러스연구소 연구소장
1994年 7月～현재 한국컴퓨터기자클럽 자문위원
1995年 4月～현재 서울지검 정보범죄수사센너 자문위원