기 고 3

컴퓨터 바이러스(Computer Virus)의 실체와 예방

대한산업보건협회
전 산 실

1. 바이러스?

1980년대 초, 컴퓨터 바이러스에 대한 인식은 일반 사용자나 심지어 컴퓨터 전문가들에게 조차도 그리 명확하게 판단되지 못했었다. 일반 사용자들이 그것의 존재에 대해 인식하기 시작했을 때, 대부분의 사용자들은 바이러스에 걸린 디스켓을 접촉시키기만 하면 바이러스에 감염된다는 식으로 컴퓨터 바이러스를 생물학적 바이러스로 생각했었다. 어떤 회사에서는 컴퓨터 바이러스에 감염된 사실이 보고되자 사장님이 하시는 말씀이 "컴퓨터 소독을 안하니까 그렇지. DDT를 뿌려 소독해!"라고 하셨다고 한다. 그러나 그러한 인식은 "컴퓨터 바이러스"라는 이름에서도 알 수 있듯이 올바른 것이 아니다. 컴퓨터 바이러스는 컴퓨터 내에서 작용하는 하나의 프로그램에 불과한 것이다. 그리하여 감염 조건에 맞는 특정 프로그램(예를 들면 모든 .exe화일이나 .com화일)을 공격하여 그 프로그램을 조작 변경시키는 수행을 한다.

컴퓨터 바이러스는 컴퓨터에 기생하는 병균이 아니다. 컴퓨터가 지금처럼 대중화되지 않은 몇년 전에는 위의 일화처럼 컴퓨터 바이러스가 일종의 병균처럼 오해되기도 했다.

현재는 인터넷이라는 거대한 통신망으로 전 세계의 자료가 하나로 공유됐고, 만일 여기에 알려지지 않은 바이러스가 감염된다면 곧바로 전 세계적으로 퍼진다. 따라서 이제 바이러스도 국산이니 외산이니 구별할 것이 없이 곧바로 사용자의 데이터를 공격하게 되는 것이다.

2. 탄생

바이러스의 기원설은 여러 가지가 있지만, 일반적으로는 컴퓨터 프로그래머들이 재미로 만든 깜찍한 프로그램들에서 비롯됐다.

화면에 재미있는 글자가 나왔다가 사라져 상대방을 놀라게 하는 프로그램들이 그 예다. 이외에도 해커들이 모험심으로 바이러스를 제작하거나 불평 불만자들의 까닭 없는 복수심, 고의적인 업무 방해 등이 바이러스를 만드는 주요 목적으로 지적된다.

최초의 컴퓨터 바이러스는 파키스탄에서 탄생했다. 1985년 파키스탄 출신의 프로그래머가 자신이 만든 프로그램이 불법 복제되는 것을 보고 불법 복제 사용자들을 응징하기 위해 바이러스를 만들어 유포시켰는데, 이것이 세계 최초의 컴퓨터 바이러스가 됐다. 이름은 브레인. 브레인

바이러스는 이후 1988년 국내에 유입되어 여러 가지 피해를 일으켰었고, 이어 발견된 것이 요즘도 여전히 악명을 떨치고 있는 예루살렘 바이러스이다.

이 바이러스는 13일의 금요일 바이러스라는 이름으로 알려져 있으며, 이름처럼 13일의 금요일마다 감염된 컴퓨터를 괴롭히며 그 후 수많은 변종 바이러스의 원조가 됐다.

3. 바이러스 정의

컴퓨터 바이러스는 컴퓨터의 운영체제나 소프트웨어에 몰래 들어가 시스템이나 사용자의 프로그램에 자신을 복제하고 그 컴퓨터 시스템과 화일들을 파괴하는 프로그램을 말한다. 그런데 그 증상이 마치 살아 움직이는 바이러스와 유사하다고 하여 컴퓨터에서의 '바이러스'라고 부르는 것이다.

다른 프로그램에 그 자신을 실행 가능한 형태로 복제하여, 복제된 다른 프로그램도 성질을 가지게 하는 프로그램을 컴퓨터 바이러스라고 정의하기도 하지만 그리 적절하지는 못하다. 그러나 지금까지 소개된 어떠한 정의도 만족스러운 결과를 주지는 못한다.

『기계어 M으로 쓰여진 A라는 프로그램이 있다고 하자. 만약 이 프로그램이 어떠한 입력도 가지지 않으면 그것의 기계어 코드를 출력하거나 그 프로그램을 메인 메모리로 복사한다면 그 과정은 '자기복제(self-reproduction)'라고 할 수 있다.』(1981, J.Kraus)

4. 바이러스 감염

바이러스 프로그램이 시작되면 그것은 현재의 디스크 드라이브 내에 있는 프로그램을 찾아서 그 프로그램의 바이러스 감염여부를 검사한다. 감염여부 검사를 위해 바이러스 프로그램은 바이러스 식별자를 사용한다. 여기서 바이러스 식별자는 프로그램의 감염 상태를 나타내는 바이러스 고유의 정보이다. 이미 감염된 프로그램은 다시 감염시킬 필요가 없으므로 바이러스는 감염되어 있지 않은 사용자 프로그램, 즉 바이러스 식별자를 갖고 있지 않은 프로그램이 발견될 때까지 탐색을 계속한다. 첫 번째로 감염된 프로그램은 그림과 같이 바이러스 식별자를 포함하고 있다.

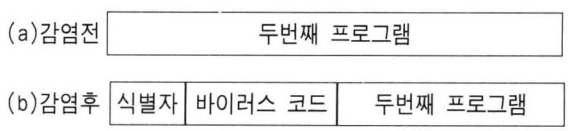
· 바이러스 식별자 : 감염 상태 판단을 가능하

게 하며 감염된 프로그램이 다시 감염되는 것을 방지한다.

· 바이러스 코드 : 바이러스의 재생산을 위해 필요한 루틴과 함수를 가진다.

바이러스는 탐색을 계속하면서 이미 감염된 프로그램은 무시하고, 감염되지 않은 두 번째 프로그램을 찾는다. 이 때 바이러스는 그 자신을 이 프로그램의 처음 부분에 오버라이트(overwrite)한다.

만약, 지금 감염된 두 번째 프로그램이 실행된다면 이 프로그램의 처음 부분에 오버라이트(overwrite)된 바이러스 프로그램이 실행되어 세 번째 프로그램을 감염시킨다. 세 번째 프로그램에 대한 감염이 끝난 후 두 번째 프로그램에는 에러가 발생되는데, 이것은 두 번째 프로그램의 처음 부분이 바이러스 코드로 오버라이트(overwrite)되어 파괴되었기 때문이다.



바이러스가 컴퓨터를 감염시키는 방법에 관해서 살펴보자. 바이러스 프로그램이 그 대상을 감염시키는 방법에는 크게 두 가지가 있다. 하나는 네트워크에서 감염시키는 것이고 다른 하나는 감염된 디스크나 화일에 감염시키는 것이다.

네트워크에서는 일단 슈퍼바이저가 감염되면 연결된 모든 노드들이 순식간에 감염된다.

디스크나 화일을 감염시키는 형태의 바이러스가 현재 가장 많은데, 이것은 다음의 두 가지 종류로 나누어 볼 수 있다.

하나는 감염된 화일이 실행될 때만 다른 화일이나 디스크를 감염시키는 것이고, 다른 하나는 메모리에 상주해 있다가 특정 명령이나 인터럽트를 가로채서 감염시키는 것이다.

화일 실행시 감염시키는 바이러스는 실행 화일 속에 숨어 있다가 그 화일이 실행되면 바이러스 코드로 점프하여 바이러스 프로그램을 실행시킴으로써 다른 화일이나 디스크를 감염시킨다. 프로그램의 로드 속도가 떨어지게 되는데, 그것은 실제 로드 시간이 길어지는 것이 아니라 바이러스 프로그램이 실행되기 때문이다. 평소보다 실행하는 속도가 느려졌다면 일단 그 프로그램을 의심해 보는 것이 좋을 것이다.

5. 바이러스 일반적 성격

· 자기복제 : 컴퓨터 바이러스는 살아 있는 바이러스처럼 다른 시스템이나 소프트웨어를 감염시키기 위해 자신을 복제할 수 있는 코드를 갖고 있으며, 기회가 오면 언제든지 자신을 감염시킬 만반의 준비를 갖추고 있다.

· 저수준 언어 사용

· 다양한 변종 : 새로운 컴퓨터 바이러스가 나오게 되면 그것을 모방한 잡종 변형 바이러스가 다양하게 나오게 된다.

· 지능화 및 악성화 : 바이러스 프로그램들의 양성은 계속해서 날이 갈수록 더욱 지능화되어 가고 있음과 동시에 초창기만 하더라도 거의 피해를 주지 않던 양성 바이러스에서 벗어나 심각한 피해를 끼치는 악성 바이러스로 점차 발전하고 있다.

6. 생물학적 바이러스와 컴퓨터 바이러스의 차이

생물학적 바이러스가 단백질 분자와 그 안의 핵으로 이루어진 하나의 살아 있는 생명체로서 나가는 것과 마찬가지로, 컴퓨터 바이러스도 유전인자(프로그램 코드)를 갖고 있으면서 컴퓨터 시스템이나 다른 프로그램에 자신을 복제하도록

〈 생물학적 바이러스와 컴퓨터 바이러스의 비교 〉

생물학적 바이러스	컴퓨터 바이러스
특정한 세포에 침투	프로그램에 침입(*.exe, *.com)
본래의 세포조직을 변형	프로그램 조작:목적을 수행
새로운 바이러스가 세포 내에서 증식	감염된 바이러스 프로그램이 새로운 바이러스 프로그램을 만듬
한 세포는 같은 바이러스에 대해 한번 이상 감염되지 않음	감염된 프로그램이 다른 바이러스에 감염될 수 있음
오랫동안 아무 증상 없이 있을 수 없음	오랫동안 아무 에러없이 작업할 수 있음
모든 세포가 침입하는 바이러스에 감염되지 않은(항체 반응)	특정 바이러스에 대항하는 백신 프로그램을 만들 수 있음
바이러스 자신 스스로 변형할 수 없으며, 완전하게 제거할 수 없음	바이러스 프로그램은 자신 스스로 변형할 수 있어 백신 프로그램에 대항할 수 있음

작성된 프로그램인 것이다.

7. 컴퓨터 바이러스의 작동원리

1) 바이러스 코드를 정상적인 코드보다 먼저 처리한다.

감염된 프로그램을 실행시키면 먼저 바이러스 프로그램이 실행되어 다른 프로그램을 감염시키는 등의 역할을 수행한 뒤 원래의 프로그램을 실행시키게 된다. 사용자는 속도가 약간 느려졌다고 느낄 뿐, 다른 점은 감지할 수 없다. 결국 사용자가 바이러스의 존재를 깨닫게 될 때는 파괴적인 루틴이 실행되고 난 뒤인 경우가 대부분이다. 이런 바이러스는 보통 프로그램의 뒤에 붙기 때문에 프로그램의 크기를 증가시켜서 사용자가 쉽게 알아볼 수 있는 약점을 가지고 있었다.

2) 메모리 영역에 바이러스 코드를 숨겨둔다.

3) 정상적인 코드 사이에 숨어 작동

8. 발전

안철수연구소의 안철수 소장 견해에 따르면 컴퓨터 바이러스는 크게 다섯 세대를 거쳐 발전해 왔다. 초기 1세대는 원시형 바이러스로 프로그래머 구조가 단순해 분석하기 쉬운 바이러스였다. 2세대 바이러스는 암호화시켜 저장하는 것이었다. 그러나 암호 해독 부분이 항상 일정하기 때문에 역시 쉽게 퇴치할 수 있는 바이러스다. 불가리아에서 만들어지기 시작한 3세대 바이러스인 은폐형 바이러스는 바이러스 스스로를 은폐하는 것이 특징, 파일의 길이를 속이고 감염된 부분을 감추면서 백신 프로그램이 알아 챌 수 없도록 만든 것이다. 그러나 메모리를 먼저 검사해 은폐 기능을 찾아낼 수 있으므로 치료가 그렇게 어려운 편은 아니었다.

그러나 이후 등장한 4세대 바이러스부터는 백신 제작 과정이 점차 까다로와지기 시작했다. 이른바 갑옷형 바이러스로 알려진 4세대 바이러스는 감염될 때마다 암호로 만든 부분이 달라져 바이러스를 분석하기가 쉽지 않게 되었다. 특히 이러한 바이러스는 여러 명의 전문가가 동원된 것으로 알려져 충격을 주기도 했다. 백신을 만들기 어려워서 그렇지 한 번 만들면 퇴치가 가능하다. 안철수 소장은 제5세대 바이러스로 매크로 바이러스를 꼽았다. 마이크로소프트의 오피스 프로그램에 들어 있는 매크로 기능을 이용한 바이러스로 1997년에 최초의 매크로 바이러스가 만들어진 이래 2,000여종 이상이 발견되었다. 주로 워드와 엑셀 매크로 바이러스이며 시스템 오동작을 유발하는 원인으로 꼽힌다.

9. 예방

다음과 같은 몇 가지 기본 절차를 따르면 비교적 쉽게 감염 위험을 최소화할 수도 있다.

1) 플로피디스크를 사용하기 전에 항상 검사한다.

2) 소프트웨어 설치, 또는 소프트웨어 복제품 설치를 위해 디스크를 공유하지 않는다. 시스템간의 바이러스를 이동시키는 감염 형태중 하나다.

3) 신뢰할 수 없는 사이트에서 소프트웨어를 다운로드하지 않는다.

4) 모든 컴퓨터는 처음에 C 드라이브, 다음에 A 드라이브에서 부팅되도록 구성하여야 한다. 대부분의 컴퓨터들은 부팅하는 동안 'CMOS구성'에서 이런 세팅 변경을 허용하고 있다. 이렇게 하면 모든 기본 부트 바이러스 감염을 없애줄 것이다.

5) 모든 서버와 데스크탑 컴퓨터들은 데스크탑 안티바이러스 소프트웨어의 실시간 보호기능으로 무장되어야 한다.

6) 모든 그룹웨어 시스템들은 실시간 안티바이러스 보호로 무장되어야 한다.

7) 안티바이러스 엔진 업데이트는 공급 후, 테스트 되자마자 전 사이트 차원으로 제공되어야 한다.

8) PC에서 실시간 안티바이러스 소프트웨어의 작동을 중지시키지 않는다.

9) 외부에 디스켓이나 E메일로 파일을 보낼 경우, 바이러스 감염 여부를 다시 한 번 확인한다.

10) 모든 E메일 첨부 파일들은 반드시 검사한 다음에 사용한다. 이때 안티바이러스 소프트웨어의 실시간 검사 기능이 유용하다.

10. 실전에서의 바이러스

〈 걸프전 〉

90년 걸프전에서 미국이 승리하는 데는 정보전 기술이 크게 기여했다. 전운이 감돌자 이라크는 컴퓨터와 주변기기를 수입했고, 미국은 이라크로 수출하는 프린터 안에 컴퓨터 바이러스를 집어넣었다. 프린터 안에 숨어든 바이러스는 미국을 비롯해 서방 국가의 전폭기들이 출격하는 날 활동을 개시했다. 잠에서 깨어난 컴퓨터 바이러스는 이라크 전산망 안에서 엄청난 속도로 복제되었고, 연합군 전폭기가 바그다드 상공에 도착했을 때 이라크 방공망은 마비되기 시작했다. 컴퓨터 지시에 따라 발사되어야 할 이라크 대공 화기는 손으로 조작되어 정확성이 매우 떨어졌다. 미국 공군을 비롯해 연합군 비행기의 손실이 적은 것은 그 때문이었다. 미국은 이에 그치지 않고 이라크 군용 전화망을 교란하고 전파를 방해해 사담 후세인이 군대를 지휘할 수 없게 만들었다.

〈 논리 폭탄 〉

컴퓨터 바이러스와 비슷한 것이 논리 폭탄이다. 논리 폭탄은 적국 시스템에 일시적으로 오류가 발생하도록 시스템 내부 코드를 바꾸는 기능을 수행한다.

전 세계 컴퓨터 프로그램의 70~80%를 생산하는 미국이 쉽게 이용할 수 있는 방법이 트랩 도어이다. 트랩 도어는 시스템을 설계할 때부터 프로그램 내부에 실수나 고의로 장치된 침입로를 일컫는다. 개발자만이 알 수 있는 이 트랩 도어를 이용하면 언제든지 쉽게 시스템 내부에 침투해 시스템을 마비시킬 수 있다.