정보보호

(5111041)

6장

악성소프트웨어

악성 소프트웨어의 유형

- 악성 소프트웨어 정의 [SOUP13]
  - 피해자의 데이터, 응용 프로그램 또는 운영체제의 기밀성, 무결성 또는 가용성을 손상시킬 의도로 은밀하게 시스템에 삽입된 프로그램

- 악성 소프트웨어와 콘텐츠는 시스템에 대한 가장 치명적인 위협

악성 소프트웨어 관련 용어

명칭	설명
지능형 지속 위협 (APT)	• 다양한 침입 기술과 악성 소프트웨어를 사용 • 상업적 또는 정치적 목적을 위해 특정 공격 대상에 지속적이며 효율적으로 적용함 • 조직의 기밀 정보 획득을 위하여 사용됨
애드웨어 (Adware)	• 소프트웨어에 통합된 광고, 팝업 광고를 포함 • 상업 사이트로 연결하는 경로 변경으로 사용자에게 부정적인 결과를 초래할 수 있음
공격 킷 (Attack kit)	• 다양한 전파와 페이로드 기법을 사용 • 새로운 악성 소프트웨어를 자동으로 발생시키는 도구의 모음
오토 루터 (Auto-rooter)	• 새로운 기계를 원격으로 손상시키기 위하여 사용되는 공격 툴
백도어 (Backdoor)	• 특정 지점에 우회하여 정상적인 방법으로 통과하는 기법 • 프로그램이나 시스템에 있는 기능을 인증 없이 접근 가능
다운로더	• 공격을 위해 시스템에 다른 서비스/어플등을 설치하는 코드 • 악성 소프트웨어 코드가 하나의 손상된 시스템에 우선적으로 삽입된 후 대량의 악성 소프트웨어 패키지에 주입됨

악성 소프트웨어 관련 용어

명칭	설명
다운로드에 의한 구동	• 클라이언트 시스템을 공격하기 위하여 브라우저의 취약점을 이용하는 손상된 웹사이트에 있는 코드를 사용하는 공격
익스플로잇(Exploit)	• 하나의 취약점 또는 다수의 취약점에 특화된 코드(취약점 공격)
플러더(Flooder)	• 네트워크에 연결된 컴퓨터를 공격하기 위해 대량의 데이터를 만듦 • 만들어진 데이터를 이용해 DoS 공격의 형태를 수행
키로거 (Key logger)	• 공격된 시스템으로부터 자판 입력을 수집하는 프로그램
논리 폭탄 (Logic bomb)	• 공격자에 의하여 악성 소프트웨어에 삽입된 코드 • 특정 조건이 만족될 때까지 잠복하고 조건이 만족되면 논리 폭탄이 트리거 되어 공격 동작을 실행
매크로 바이러스	• 전형적으로 문서에 포함된 매크로나 스크립팅 코드를 사용 • 문서를 열거나 편집할 때 자동으로 실행됨
모바일 코드	• 다양한 장치의 플랫폼에 변경 없이 설치되어 실행되는 소프트웨어
루트킷 (Rootkit)	• 공격자가 컴퓨터 시스템에 침입하여 루트 권한 접근을 얻은 후에 사용되는 크래킹 공격 도구의 모음

악성 소프트웨어 관련 용어

명칭	설명
스팸 프로그램	대량의 전자메일을 보내기 위하여 사용됨
스파이웨어 (Spyware)	키스트로크, 스크린 데이터, 네트워크 트래픽 등을 모니터링하는 것에 의하여 컴퓨터로부터 정보를 수집하여 다른 시스템에 전송하는 소프트웨어
트로이 목마 (Trojan)	유용한 함수를 갖는 것으로 보이지만 시스템의 합법적인 인증을 이용하여 안전한 기계를 공략하는 숨겨진 악성 함수를 수행하는 컴퓨터 프로그램
바이러스 (Virus)	자신을 다른 실행 가능한 기계에 복제하려고 시도하는 악성 소프트웨어. 성공하였을 경우 이를 감염되었다고 말함. 감염된 코드가 실행될 때 바이러스 또한 실행됨
웜 (Worm)	독립적으로 실행할 수 있고 목표 시스템에서 소프트웨어의 취약성을 이용하여 네트워크상의 다른 서버에게 자신의 복사본을 전파할 수 있는 컴퓨터 프로그램
좀비-봇 (Zombi-Bot)	다른 기계를 공격하도록 감염된 기계에서 활성화되는 프로그램

악성 소프트웨어의 유형

- 유형 분류 방법
  - 1. 목표에 도달하기 위한 확산과 전파하는 방법 기반
    - 기생하기 위한 호스트 프로그램을 필요
      - 예) 바이러스 등
    - 시스템에서 실행하는 독립적인 프로그램을 갖는 것
      - 웜, 트로이 목마 등
    - 복제되지 않는 콘텐츠
      - 트로이 목마, 스팸 메일 등
    - 복제되는 악성코드
      - 바이러스, 웜
  - 2. 목표에 도달하였을 때 수행되는 동작이나 페이로드 기반
    - 서비스 및 정보의 도난, 악성코드에 자신의 숨김 기능 포함

악성 소프트웨어의 유형

- 공격 킷(Attack kit)
  - 처음에는 악성 소프트웨어의 개발 및 보급 기술은 소프트웨어 작성자에 의한 상당한 기술이 요구되었음
    - 1990 년대 초반 바이러스 생성 툴킷의 개발에 의하여 변화
    - 2000 년대에는 악성코드의 개발 및 보급을 쉽게 지원할 수 있는 일반화된 공격 키트가 개발됨

- 크라임웨어 (Crimeware)로 발전되어 알려짐
  - 초보자가 결합 , 선택 , 보급할 수 있도록 다양한 전파 방법과 페이로드 모듈을 포함하여 제공
  - 시스템의 약점과 광범위한 패치관련 정보에서 발견된 최신 취약점을 이용하 여 쉽게 사용자가 공격 방법을 정의 할 수 있음
  - 툴킷으로 만들어진 악성소프트웨어가 전문 공격자로부터 설계된 악성 소프트웨어보다 정교하진 않지만 시스템 방어를 어렵게 하는 새로운 변종의 공격 방 법이 쉽고 다양하게 생성될 수도 있음
    - 예) 제우스, 블랙홀, 사쿠라, 피닉스 등

악성 소프트웨어의 유형

- 익스플로잇 킷(Exploit-kit) 이란 위에서 말한 취약점 공격을 할 수 있도록 해커가 만든 소프트웨어 도구입니다. 전문적인 지식 없이도 구입만 하면 악성 소프트웨어를 유포할 수 있도록 만들어져 있기 때문에 공격자들 사이에서 인기가 좋습니다. 현재 다크웹에서 비트코인을 통해서 500달러~10,000달러 사이에 거래되고 있습니다. 다음의 익스플로잇 킷 사례를 보고 감염되는 일이 없도록 주의하시기 바랍니다.

- MPack: 러시아 해커에 의해 개발된 멀웨어 킷입니다. 2006년에 처음으로 등장하였으며, 여태까지 160,000대의 컴퓨터를 감염시켰다고 알려집니다. 주 기술은 사용자의 키보드 활동을 추적하는 키로깅입니다.

- Blackhole: 2012년에 유행했던 익스플로잇 킷입니다. 스팸 이메일이나 수상한 링크를 통해 전파됩니다. 피해자가 소셜 엔지니어링에 속도록 하고, 블랙홀 익스플로잇 킷의 서버로 접속하도록 유도합니다. 그리고 피해자의 컴퓨터는 자바(Java) 언어로 만들어진 트로이 목마에 감염됩니다.

- RIG kit: 2014년에 처음 생겼지만 2023년 현재까지 사라지지 않고 30%의 공격 성공률을 기록하고 있습니다. 마찬가지로 소셜 엔지니어링 기법을 사용해서 악성 웹사이트로 사람들을 현혹합니다. 그 다음에는 브라우저가 디바이스에 멀웨어를 설치하도록 합니다. 국내 인터넷 뱅킹 공격을 위한 악성코드를 유포하고 있습니다.

- Angler exploit kit: 랜섬웨어와 트로이목마 유포의 주범입니다. 악성 광고를 통해서 악성 웹사이트로 연결이 되고, 시스템의 취약점을 통해서 잠입합니다. 랜섬웨어 등 멀웨어 설치는 물론 개인 정보를 탈취하여 빼돌립니다. 감염된 시스템을 봇넷으로 이용하기도 합니다.

지능형 지속 위협(APT)

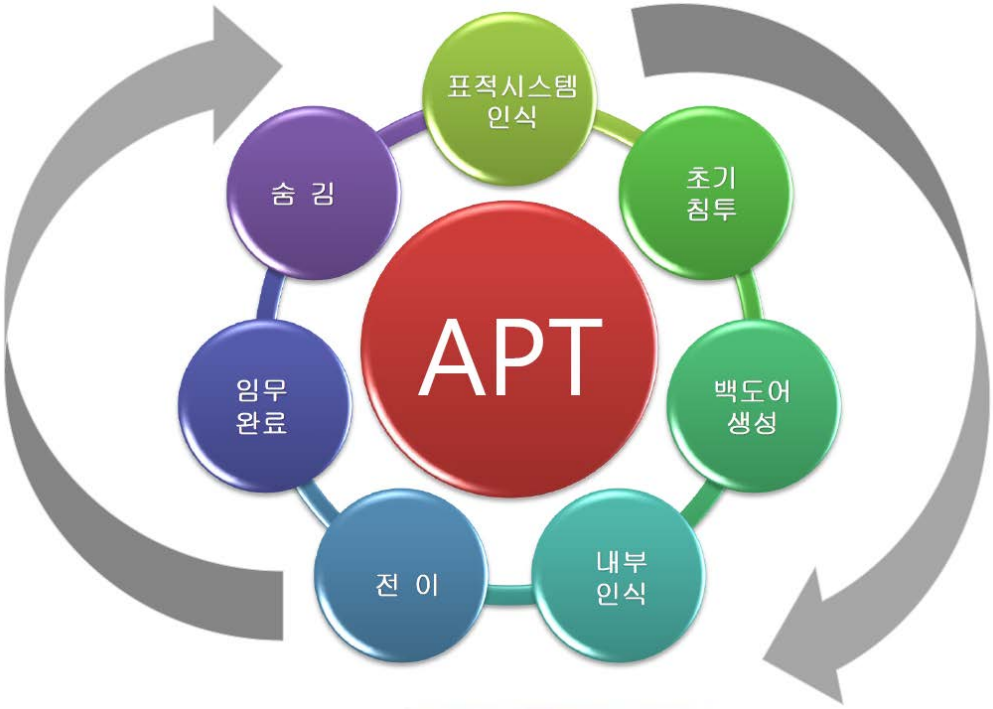
- APT 공격이란 ?
  - 지능형 지속 위협 (Advanced Persistent Threat) 을 의미함
  - 다양한 IT 기술과 방식들을 이용해 조직적으로 경제적인 목적을 위해 다양한 보안 위협들을 생산해 지속적으로 특정 대상에게 가하는 공격 기법
  - 2015년 이후...
    - 지능적으로 이메일을 이용해 악성코드 등을 심거나 강제로 암호화를 시켜 데이터를 날리는 것 또는 메일 내용을 통해 심리적 헛점을 파고드는 행위에 초점이 맞추어져 가는 추세
    - 기존의 보이스 피싱이나 전화사기와 구별됨
    - 첨단매체인 PC나 스마트폰 등에서 전자 네트워크와 이메일을 사용해 보안 취약점이나 심리적인 약점을 파고드는 형태를 통칭

지능형 지속 위협(APT)

- 1. 지능형 (Advanced)
  - 각양각색의 침략 기술과 특화된 악성코드를 이용
  - 여러가지 공격 요소들이 선정된 목표에 적합하도록 알맞은 공격 기술을 선택
- 2. 지속 (Persistent)
  - 성공률을 극대화하기 위하여 오랜 시간에 걸쳐서 선정된 목표에 대한 공격
  - 목표에게 목적이 달성 될 때까지 다양한 형태의 공격이 점진적으로 또는 은밀하게 이루어짐
- 3. 위협 (Threat)
  - 공격자는 보통 풍부한 자원 등을 가진 개인 /조직들을 목표로 삼음
  - 공격에 성공 할 경우 매우 큰 규모의 피해를 초래함
  - 컴퓨터 기술의 발전으로 자동화된 공격 도구와 향상된 공격기법은 위협의 수준을 증가시킴

지능형 지속 위협(APT)

- APT는 표적 지향의 공격이므로 방식을 규정하지는 않음
  - 하지만 다음의 7가지 공격 요소를 도출



지능형 지속 위협(APT)

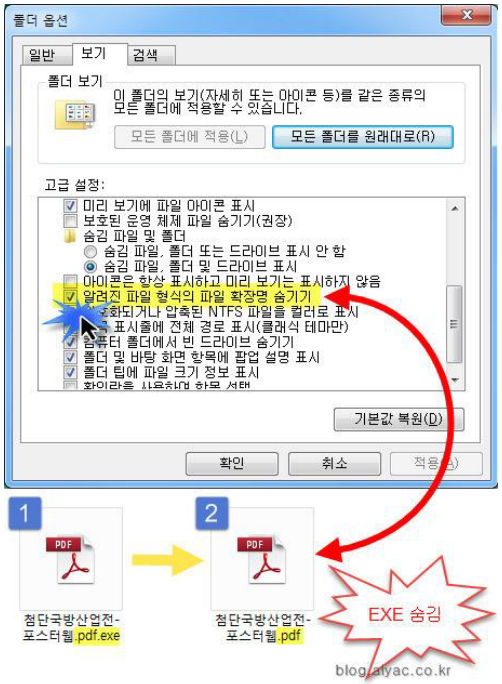
- 1. 표적시스템 인식 (Targeted System Recognition)
  - 침입 전 표적 시스템에 대한 정보 수집 활동
  - 기술적인 방법으로는 포트 스캔을 통해서 침입 가능한 포트 (Port) 를 검색
  - 비기술적인 방법으로는 Social engineering 을 통해서 보안 정보에 접근권한이 있는 담당자의 이메일 주소 등을 획득 후 Spear phishing, USB 감염 등을 위한 수단을 사용

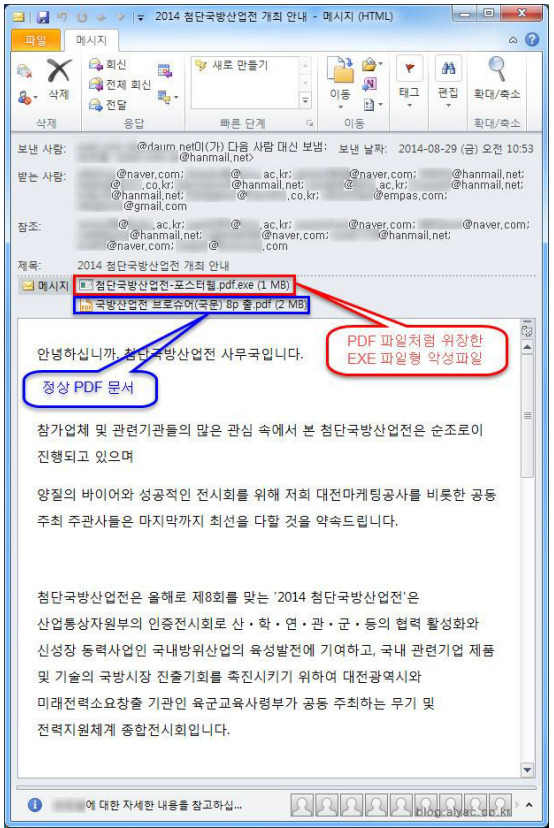
- 2. 초기 침투 (Initial Intrusion)
  - 실질적인 APT 공격이 이루어지는 단계로써 표적 시스템에 초기 침투를 수행
  - 수집한 정보를 토대로 Spear phishing Spear phishing Spear phishing Spear phishing Spear phishing 공격을 시도
  - 보안담당자 , 네트워크담당자와 같은 표적 시스템을 관리하는 담당자가 주요 표적
  - 공격자는 주로 악성코드에 감염된 웹사이트나 문서를 이메일에 첨부하여 표적에게 전송

- 3. 백도어 생성 (Backdoor Establishment)
  - 최초 침입 이후 공격자는 향후에 더욱 쉽게 시스템에 접근하기 위하여 백도어를 생성
  - 공격자는 자신이 생성한 C&C (Command and Control) Server와 표적 시스템이 직접적으로 정보를 교환
  - 정상적인 트래픽처럼 보이는 암호화된 데이터를 C&C 서버와 주고받아 Antivirus 제품의 감시망을 벗어나 정보를 쉽게 획득

스피어피싱

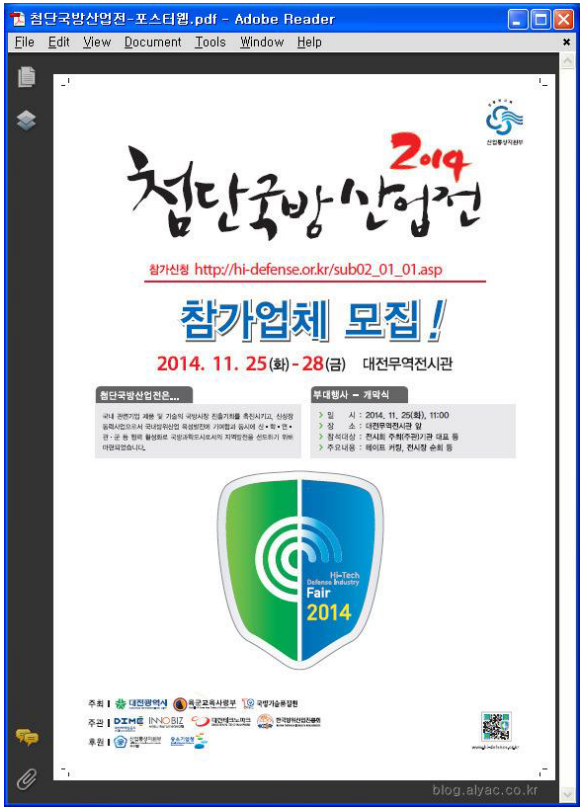
- 스피어피싱 (Spear Phishing) 용어
  - 특정다수가 아닌 특정기관이나 기업의 내부직원을 표적 삼아 집중적으로 공격하는 행위
- 2014 년 8월 29 일,2014 첨단국방산업전 사무국에서 발신한 것처럼 사칭한 스피어피싱 (Spear Phishing) 이메일
- 첨부파일의 2중 확장명은 99.9% 악성파일





스피어피싱

- 첨단국방산업전-포스터웹.pdf.exe 악성파일이 실행되면 동일 경로에 정상적인 첨단국방산업전 -포스터웹 .pdf 문서파일을 생성하고 실행
- 이용자 화면에는 실제로 정상적인 PDF 문서파일의 화면이 보여짐



지능형 지속 위협(APT)

- 4. 내부 인식 (Internal Recognition)
  - 신뢰된 컴퓨터를 통해서 네트워크 스캐닝을 시도
  - 내부네트워크를 통해서 정찰이 이루어지며 신뢰되는 관계를 통해 표적 시스템의 네트워크 정보를 획득
- 5. 전이 (Metastasis)
  - 내부 정찰을 통해서 얻은 정보를 이용하여 주요 정보가 있는 End -Point로 악성코드를 전이 동일 네트워크를 사용하는 모든 End -Point 를 감염
  - 공격자는 원하는 정보를 유출시킬 준비를 마침
- 6. 임무 완료 (Mission Complete)
  - 공격자의 목적에 맞는 행동을 성공적으로 수행하는 단계
  - 공격자의 목적은 대표적으로 시스템 파괴와 기밀 정보의 유출
  - Antivirus 에 탐지되지 않게 하기 위해서 정보는 암호화 , 파일압축 , 파일 분할 등의 기법을 사용
- 7. 숨김 (Hiding)
  - APT 공격은 지속적인 공격을 위해서 특별한 이벤트가 없는 동안 악성 소프트웨어를 숨김
  - 몇몇의 APT 공격은 Rootkit 을 포함하고 있으며 이를 통해 시스템의 루트 권한을 획득
  - Rootkit 을 이용하여 Malware 의 행동을 숨김

APT 공격 사례

구분	정의
IceFog	• 일본과 대한민국의 정부출연연구소, 방위산업체, 조선해양사, 통신사 및 고도의 기술을 보유한 회사 등을 타겟으로 활동하고 있는 APT공격 • 스피어피싱 이메일이 이용, 이메일에 첨부된 악성코드를 열거나 악성 웹사이트를 방문 • 시스템을 감염시킨 뒤 공격자는 감염 시스템의 특성을 파악하고 확인하기 위해 폴더 목록, 어댑터 목록, IP구성, 네트워크 정보 등을 윈도우 주소록, WAB파일, HWP, XLS, DOC등 문서파일 그리고 사용자 계정 자격 정보 등을 탈취
Net Traveler	• 항공 우주 관련 기술 및 에너지 생산 관련 데이터나 통신 관련 데이터를 탈취 • 이메일 주소를 확보한 후 악성코드가 포함된 첨부 파일의 클릭을 유도하는 메시지를 발송 • 이메일에 첨부되는 마이크로소프트 오피스 형식(확장자 doc, xls 등) 및 PDF 파일에 잠복 • 시스템에 침입할 경우 시스템 내 모든 정보에 대해 모니터링 실시 뿐만 아니라 데이터 추출, 사용자의 키 입력 기록 확인 및 각종 문서 파일의 탈취 등도 가능

APT 공격 사례

구분	정의
Stuxnet	• 시스템을 감염시켜 원자로의 PLC(programmable logic controller)에 악의적인 프로그램을 삽입 • 사용된 취약점은 Window shell LNK 취약점, Window server service 취약점, Window printer spooler 취약점, 공유 네트워크 서비스 취약점 • 장치인 USB를 공격에 이용하여 네트워크 망이 분리되어 있는 시스템에도 악성코드를 전파 • Stuxnet은 대상시스템이 아닌 경우에는 특별한 활동을 하지 않았으며 자신의 존재를 지우기 위해서 Rootkit을 사용
Duqu	• Duqu는 정보 유출을 목적으로 하여 표적 시스템의 정보를 수집 • 감염된 Micro Office 문서를 첨부하는 Spear phishing을 통해 초기 침투를 시도 • 성공적으로 시스템에 잠입하고 백도어를 설치함으로써 C&C (Control and Command) 서버와 통신 • 패스워드와 같은 주요 정보를 수집하며, 수집된 정보는 공격자가 네트워크상의 다른 시스템에 접근할 권한을 얻는데 사용

APT 공격 특징

- APT 공격 특징
  - 특정 조직에 최적화된 공격 수행
  - 충분한 시간과 비용을 투자
  - 조직 및 구성원 개인에 대한 충분한 정보 수집 (사회공학적인 방법 이용)
  - 탐지 회피 기법을 병행하여 공격 수행
    - low and slow 전략
    - 알려지지 않은 악성코드 (Zero Day Attack) 사용
    - 이상징후를 파악하지 못하도록 장기간에 걸쳐 은밀히 활동
  - 다양한 방향으로 공격 , 사용자 및 Endpoint에 집중

- APT 공격 대응 방안
  - 공격자가 원하는 정보에 접근하기 까지 소요 시간 지연
    - 네트워크 분리 (망 분리)
    - 내부 시스템 인증 강화
  - 공격자가 원하는 정보에 접근하기 이전 단계에서 탐지 /제거
    - 알려지지 않은 악성코드 (Zero Day Attack) 탐지 / 제거
    - 악성코드 /원격접속 /명령 및 제어 지점 접근 트래픽 탐지 /차단

바이러스

- 프로그램을 감염시키는 소프트웨어조각
  - 바이러스를 침투시키도록 소프트웨어 변형
  - 원본 코드가 바이러스 코드를 복제하기 위한 루틴을 갖도록 수정함
  - 바이러스 코드는 다른 내용을 감염시키는 데 계속 사용될 수 있음
  - 초기 감염 형태는 의심되지 않는 사용자들로부터 프로그램 , 디스크 파일 , USB 스틱을 상호 교환하는 것에 의하여 컴퓨터에서 컴퓨터로 전염
  - 인터넷의 보편화로 네트워크를 통해서도 전염됨
- 프로그램에 바이러스가 침투되면 프로그램은 제한되었던 행위들을 실행 할 수 있음
  - 호스트 프로그램이 운영될 시 몰래 실행
- 특유의 운영 체제와 하드웨어
  - 특유 운영체제와 하드웨어의 세부 사항과 약점이용

전파: 손상된 내용- 바이러스

- 바이러스의 구성요소
  - 1. 감염 메커니즘
    - 바이러스가 퍼지거나 전파하는 수단으로서 감염 벡터로 표현됨
  - 2. 트리거
    - logic bomb 공격
    - 페이로드가 활성화하거나 전달될 때를 결정하는 이벤트 /조건
  - 3. 페이로드
    - 바이러스를 수행하는 부분
    - 손상을 포함할 수 있음
- 많은 악성코드의 형태는 각 부분에 하나 이상의 변형을 포함

전파: 손상된 내용- 바이러스

- 바이러스의 수행 4단계
  - 1. 휴지 단계
    - 바이러스는 활성화되지 않음
    - 모든 바이러스가 휴지단계를 거치는 것은 아님
  - 2. 전파 단계
    - 바이러스는 자신을 복제하여 다른 프로그램 또는 디스크 상의 시스템 영역에 삽입
    - 바이러스는 탐지를 피하기 위하여 변형되어 복제된 바이러스는 복제 이전의 바이러스와 같지 않을 수 있음
    - 감염된 프로그램은 바이러스의 복제를 실행하고 다시 전파 단계로 들어감
  - 3. 트리거 단계
    - 바이러스는 의도한 기능을 수행하도록 설정
    - 휴지 단계에서처럼 다양한 시스템 사건(날짜 , 다른 프로그램이나 파일의 생성, 디스크 용량의 초과 등)에 의하여 활성화
    - 바이러스가 자기 자신을 복제하는 경우도 하나의 사건으로 포함
  - 4. 실행 단계
    - 바이러스내의 의도한 기능이 수행
    - 단순히 스크린에 메시지 (광고)를 실행하는 등 시스템에 무해할 수도 있고 데이터 파일과 프로그램 등을 파괴하는 것과 같이 손상을 입힐 수도 있음

전파: 손상된 내용- 바이러스

- 실행 가능한 바이러스 구조
  - 전통적인 기계 실행 코드 바이러스는 어떤 실행 프로그램의 열에 또는 뒤에 붙여 질 수 있거나 다른 형태로 끼워질 수 있음
  - 감염된 프로그램이 적용될 때, 첫 번째로 바이러스 코드가 실행되고 그 다음 원래 프로그램의 코드가 실행
  - 코드의 첫 번째 줄은 바이러스에 의해 잠재적인 피해 프로그램이 이미 그 바이러스 에 감염되었는지 안 되었는지 결정하기 위해 사용되는 특별한 표시
  - 바이러스는 우선적으로 감염되지 않은 실행 파일을 찾고 그 파일을 감염시킴
  - 요구되는 트리거 조건이 만족된다면 바이러스는 페이로드를 실행

전파: 손상된 내용- 바이러스

- 바이러스 로직 예

```
program V
1234567;

procedure attach-to-program;
begin
   repeat
        file := get-random-program;
   until first-program-line ≠ 1234567;
   prepend V to file;
end;

procedure execute-payload;
begin
   (* perform payload actions *)
end;

procedure trigger-condition;
begin
   (* return true if trigger condition is true *
end;

begin (* main action block *)
   attach-to-program;
   if trigger-condition then execute-payload;
   goto main;
end;
```

(a) 단순 바이러스

```
program CV
1234567;

procedure attach-to-program;
begin
   repeat
        file := get-random-program;
   until first-program-line ≠ 1234567;
   compress file;    (* t1 *)
   prepend CV to file;   (* t2 *)
end;

begin (* main action block *)
   attach-to-program;
   uncompress rest of this file into tempfile;   (* t3 *)
   execute tempfile;   (* t4 *)
end;
```

(b) 압축 바이러스

전파: 손상된 내용- 바이러스

- 프로그램은 시간 t0에서 시작
- P1은 바이러스 CV 에 감염된 프로그램이고 , P2는 CV 에 감염되지 않은 깨끗한 프로그램
- 감염된 P1이 실행될 때 다음의 단계를 수행
  - t1 : 감염되지 않은 파일 P2에 대하여 바이러스는 우선 원래의 프로 그램보다 바이러스 크기만큼 짧은 P2'파일을 만들기 위해 압축함
  - t2 :복사된 CV 를 압축된 프로그램 앞에 붙임
  - t3 : 감염된 원래의 프로그램의 압축 버전 (P1')은 압축하지 않음
  - t4 : 압축되지 않은 원래의 프로그램 P1이 실행됨

바이러스 분류

공격 대상에 의한 분류

- 부트 섹터 감염 바이러스
  - 마스터 부트 레코드나 부트 레코드를 감염시키고, 바이러스를 담고 있는 디스크의 시스템이 부팅되면 바이러스가 확산됨
- 파일 감염 바이러스
  - 운영체제나 쉘이 실행가능 하다고 여기는 파일 감염
- 매크로 바이러스
  - 매크로나 어플리케이션에 의해 해석되는 스크립트 코드가 있는 파일 감염
- 다중 부분 바이러스
  - 다양한 방식의 파일 감염

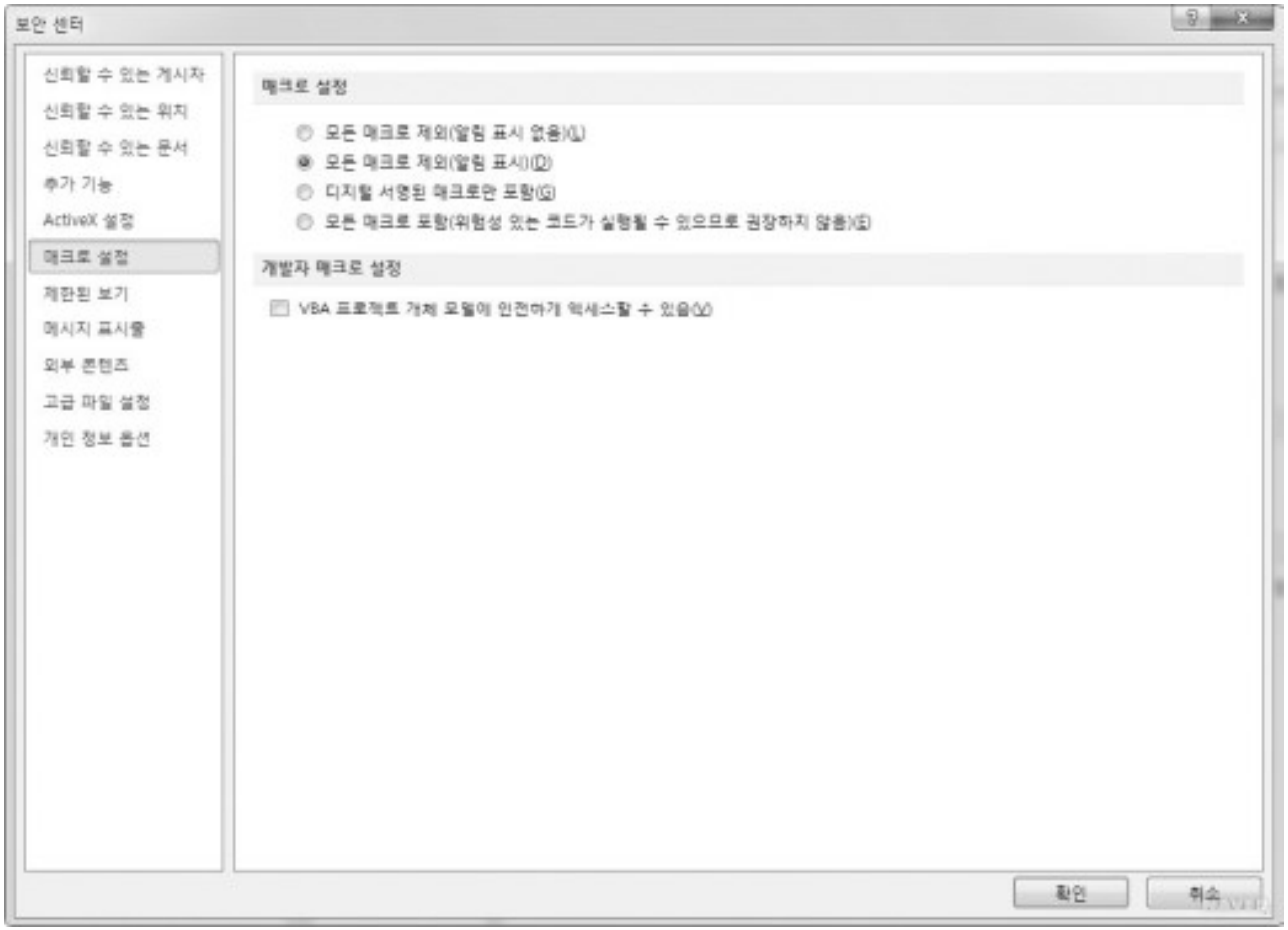
은닉 방법에 의한 분류

- 암호형 바이러스
  - 랜덤 암호화 키를 생성하고 잔류 바이러스를 암호화 시키는 바이러스의 일부
  - 탐지를 모호하게 하기 위해 키를 생성하여 바이러스 코드의 일부를 암호화
- 스텔스 바이러스
  - 안티 바이러스 소프트웨어에 의해 탐지로부터 자신을 은폐할 수 있도록 고안된 형태의 바이러스
- 다형성 바이러스
  - 바이러스 검사 프로그램을 격파시키기 위해서 기능적으로 동일
  - 탐지 알고리즘 기법을 피하기 위해 다른 비트 패턴을 가진 사본을 생성
- 변성 바이러스
  - 각 주기마다 자신을 변이시켜 완전히 다시 쓰는 바이러스로써 외관 뿐아니라 동작도 변경

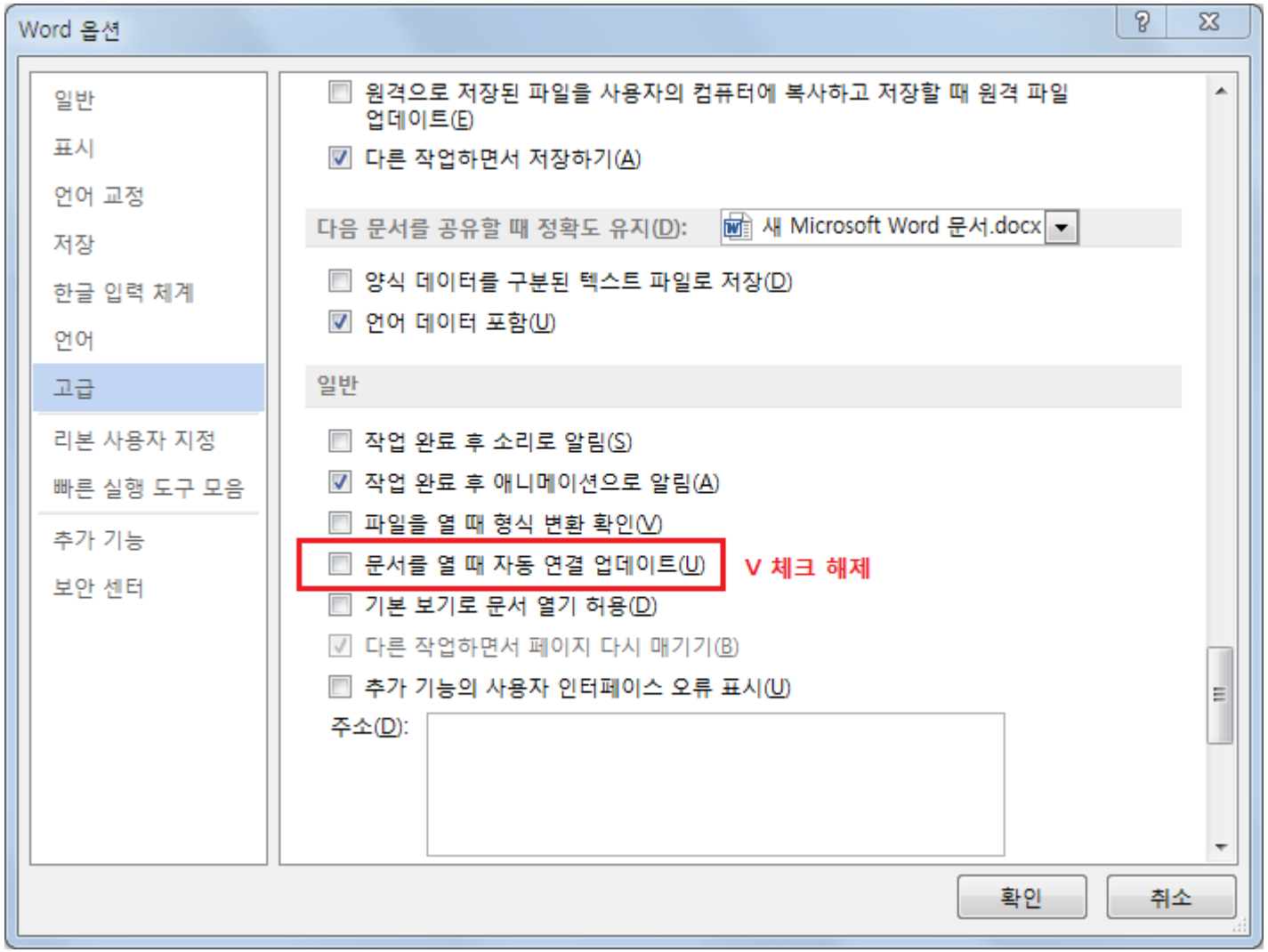
매크로/스크립팅 코드 바이러스

- 1990년대 중반에 매우 일반적이었던 바이러스
  - 플랫폼 독립성
  - 문서 감염 (코드의 일부가 실행되지 않음)
  - 확산의 용이성
- MS Office 응용프로그램의 매크로 기능 이용
  - 최신 판 제품은 보호기능 포함
- 다양한 안티-바이러스 프로그램이 개발되었으며, 두드러진 바이러스 위협 제거
- 컴퓨터 문서는 일반적으로 공유되어 사용되기 때문에 매크로 바이러스는 전자메일 등을 통해 쉽게 확산

매크로/스크립팅 코드 바이러스



매크로/스크립팅 코드 바이러스



매크로/스크립팅 코드 바이러스

- 멜리사 바이러스

1999년 3월 발견된 멜리사바이러스는 e-mail로 자동 발송되는 최초의 바이러스로 디스켓뿐 아니라 e-mail까지 감염통로가 확산되었음을 나타낸 바이러스다. 마이크로소프트(MS)사의 워드 프로그램 첨부파일 형식으로 전자우편으로 배달되며, 해당 컴퓨터 사용자에게 전자우편을 보낸 상대편 50명의 주소로 자동 전달되는 것이 특징이다. MS사 회장 빌 게이츠의 부인 이름(멜리사)을 따 명명됐다. 바이웨이(By-way) 확장자가 .com이나 .exe인 실행파일을 감염시키는 바이러스로, 사용하지 않는 특정 섹터에 위치해 파일을 실행시키지 않아도 저절로 감염되는 특성을 지녔다.

멜리사바이러스는 발생한지 불과 일주일 사이에 보잉사와 록히드 마틴사, 미 해병대 등 최소 300개 기관의 컴퓨터 10만 대를 감염시켰다. 멜리사는 사용자들이 첨부파일의 위험성을 알지 못했고, 윈도가 널리 퍼져 있었다는 것에서 피해가 컸으며 이로 인해 '사이버페스트'라는 악명을 얻기도 했다. 멜리사바이러스의 피해가 점차 커지고, 이후 전 세계적으로 확대되자 미국 FBI는 바이러스 제작자 검거에 나서, 그 해 4월초 멜리사바이러스 제작자인 데이비드 스미스(David Smith)가 체포됐다. 당시 31살의 프로그래머였던 스미스는 바이러스 개발혐의로 기소된 최초의 인물 중 한 사람이었다.

한편 1999년 멜리사바이러스가 포문을 연 이래 e-mail
빅 등이 대세를 이루고 있다. 처음에는 아웃룩에서만 자
이러스가 등장해 메일 프로그램이 없어도 바이러스 메일

```
보낸이: <감염 자료를 보낸 사람의 이름>
제목: Important message from <보낸 사람의 이름>
받는이: <50개의 이름들 가운데 받는 사람>
첨부 파일: LIST.DOC
본문: Here is that document you asked for ... don't show anyone else ;-)TCizzle
```

전파: 취약점 이용- 웜

- 컴퓨터들을 활발히 찾아 다니며 감염시키고, 감염된 컴퓨터들은 다른 컴퓨터를 공격
- 클라이언트 또는 서버 프로그램에 있는 소프트웨어의 취약점 이용
- 시스템들 간의 확산을 위해 네트워크를 연결할 수 있음
- 공유된 미디어를 통해 확산됨(USB 드라이버, CD, DVD 데이터 디스크)
- 전자 메일 웜은 매크로나 스크립트 코드에 포함된 첨부 파일 또는 인스턴트 메신저 파일 전송을 통해 확산됨
- 활성화 시 웜이 다시 복제되거나 증식 될 수 있음
- 일반적으로 페이로드의 일부 양식을 수반 함
- 최초로 알려진 구현은 1980년대 초기 제록스 팔로 알토(Palo Aloto) 연구소
  - 연산집약적 작업을 실행하기 위해 덜 바쁜 시스템을 찾기 위한 것. 악성적이지 않았음.

웜 복제

전자메일이나 인스턴트 메신저	• 자신을 다른 시스템에 복제한 웜 전자 메일 • 자기 자신을 첨부파일로 하여 인스턴트 메신저를 통해 전송
파일 공유	• 자신의 복제본을 생성하거나 이동식 미디어에 바이러스 같은 파일을 감염 시킴
원격 실행 기능	• 웜은 또 다른 시스템에 자신의 복제본을 실행
원격 파일 접근 또는 전송 기능	• 웜은 한 시스템에서 다른 시스템으로 자신을 복제하기 위해 파일 접근 서비스나 전송 서비스를 이용함
원격 로그인 기능	• 웜은 원격시스템에서 사용자로 로그인 한 다음 명령을 사용하여 자신을 다른 시스템에 복제함

전파: 취약점 이용- 웜

- **웜 전파 모델**

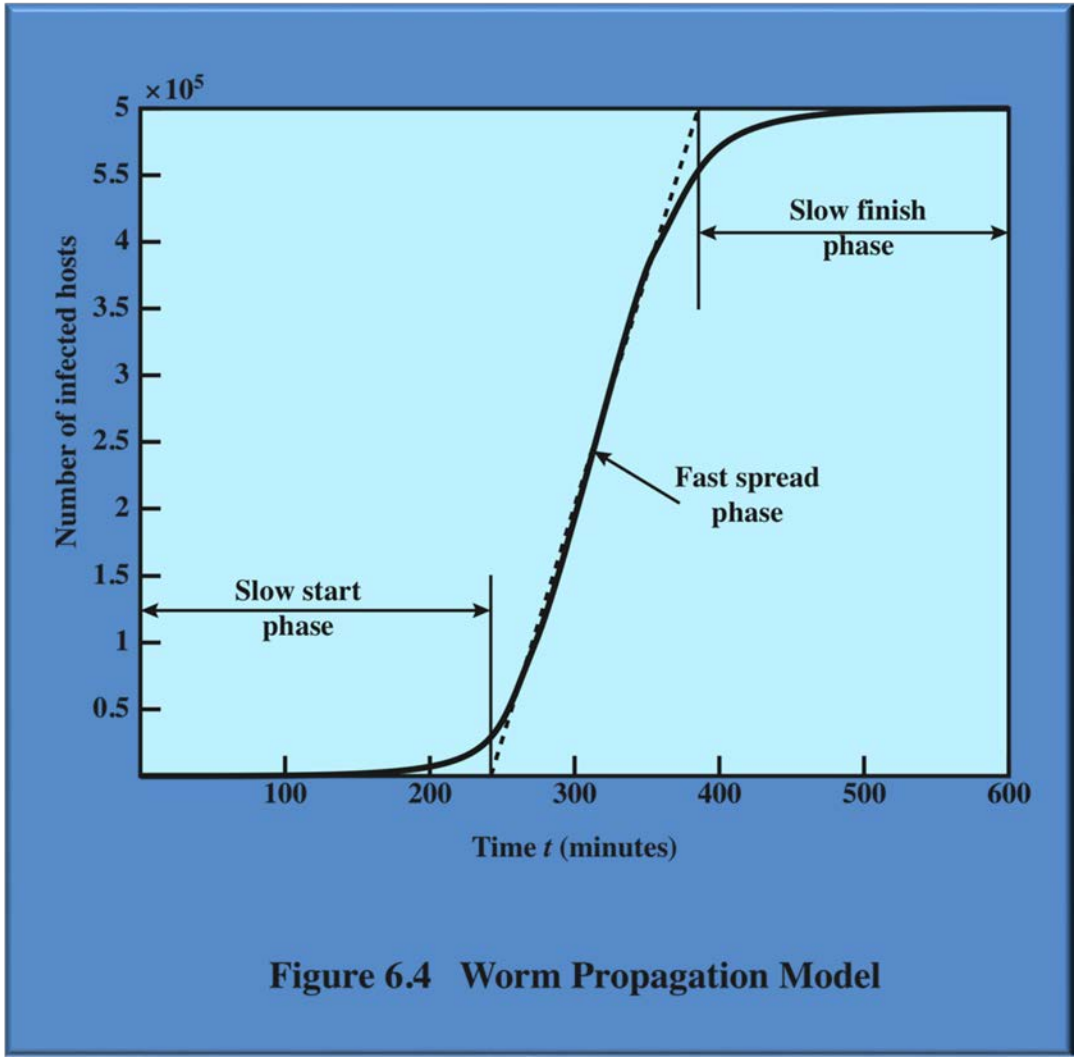


Figure 6.4 Worm Propagation Model

전파: 취약점 이용- 웜

- 모리스 웜
  - 최초의 심각한 주목을 이끈 웜 감염
  - 1988년 로버트 모리스(Robert Morris)에 의해 방출됨
  - 유닉스 시스템상에 확산되도록 고안됨
    - 로그인이나 패스워드로 다른 시스템에 로그온 해 로컬 패스워드 파일을 크래킹 하려고 시도
    - 원격 사용자의 행방을 보고하는 핑거 프로토콜(finger protocol)의 버그가 악용됨 (접속 요청 용도. 최근엔 이용 X)
    - 메일을 수신하고 전송하는 원격 프로세스의 디버그 옵션에 있는 트랩도어(trapdoor)가 악용됨
  - 공격 성공 시, 웜은 운영 체제 명령 인터프리터와의 통신을 달성하게 됨

웜 공격 예

코드레드 (Code Red)	2001년 7월	마이크로 소프트의 IIS(Internet Information Server) 버그 이용 다른 호스트에 퍼뜨리기 위해 랜덤 IP 주소 탐색 활성화 시 상당량의 인터넷 용량 소모
코드레드 II (Code Red II)	2001년 8월	마이크로 소프트의 IIS가 타겟이 됨 엑세스를 위한 백도어 설치
님다 (Nimda)	2001년 9월	웜, 바이러스 및 모바일 코드의 특징을 지님 이 메일, 윈도우 공유, 웹 서버, 웹 클라이언트, 백 도어를 통해 확산
SQL 슬래머 (SQL Slammer)	2003년 초반	SQL 서버에 버퍼 오버플로우의 취약점 이용 조밀하여 빠르게 확산 됨
소빅.F (Sobig.F)	2003년 후반	감염된 컴퓨터를 스팸 엔진으로 만들기 위해 개방형 프록시 서버 이용
마이둠 (Mydoom)	2004년	대량 메일 전자 메일 웜 감염된 컴퓨터에 백도어가 설치됨
와레조브 (Warezov)	2006년	시스템 디렉토리에 실행파일 생성 자신을 전자메일 첨부 파일로 전송 보안관련 제품들에 장애를 일으킬 수 있음
Conficker (Downadup)	2008년 11월	윈도우 버퍼 오버플로우의 취약점 악용 SQL슬래머 이래로 가장 널리 확산되어 감염
스턱스넷 (Stuxnet)	2010년	탐지율을 줄이기 위해 확산 속도에 제한이 있음 산업용 제어 시스템을 대상으로 함

전파: 취약점 이용- 웜

- 모바일 폰 웜
  - 2004년 카비르(Cabir)가 최초 발견됨
  - 2005년에는 라스코(Lasco)와 컴웨리어(CommWarrior)가 발견됨
  - 블루투스 무선 연결이나 MMS를 통해 통신 함
  - 스마트폰을 대상으로 함
  - 핸드폰을 완전히 못쓰게 하거나 기기상의 데이터 삭제 또는 유료 메시지를 전송하게 함
  - 컴웨리어는 블루투스를 통해 다른 핸드폰에 자신을 복제시켜, 스스로를 MMS파일 처럼 연락처에 전송하거나 문자메시지에 자동 답장 기능을 통해 전파시킴

전파: 사회공학 - 스팸전자메일, 트로이 목마

- 스팸 전자메일
  - 일부 스팸은 합법적인 메일 서버로부터 보내지는 반면 대부분의 최신 스팸은 사용자 시스템이 사용하는 봇넷을 이용
  - 피싱 공격과 악성코드를 전파하는 주요 수단
  - 사용자의 시스템상에 악성코드를 설치하기 위해 보내는 파일의 소프트웨어 취약성 이용
  - 사용자를 온라인 은행 사이트와 같은 합법적인 사이트로 가장하여 가짜 웹사이트로 연결시키고 , 이를 이용해여 사용자의 로그인 정보와 비밀번호 를 알아냄
  - 사용자에게 전자메일과 첨부 파일을 볼 것인지 또는 프로그램을 수행할 것인지에 대한 적극적인 선택을 요구

전파: 사회공학 - 스팸전자메일, 트로이 목마

- 트로이 목마(Trojan horse)
  - 원하지 않거나 해로운 기능을 행하는 숨겨진 코드를 포함한 프로그램이나 유틸리티
  - 공격자들이 직접적으로 수행하지 않고 간접적으로 기능을 수행하기 위하여 사용
  - 제작자는 게임이나 유용한 유틸리티 프로그램 , 배포되는 사이트, 앱 스토어 등 잘 알려진 소프트웨어를 이용하여 실행하도록 유도
  - 사용자의 도움 없이 자동으로 설치 및 실행되기 위하여 소프트웨어의 취약성을 이용함
  - 자기 복제를 하지 않음
  - 트로이 목마의 세 가지 공격 형태
    - 1. 원래 프로그램 기능의 수행을 계속하면서 분리된 악의적인 행동을 추가적으로 수행
    - 2. 원래 프로그램 기능 수행을 계속하지만 악의적인 행동을 수행하기 위하여 업데이트를 통해 기능을 수정 또는 다른 악의적인 행동을 추가
    - 3. 원래 프로그램의 기능을 완벽하게 대치하여 악의적인 기능 수행

전파: 사회공학 - 스팸전자메일, 트로이 목마

- 모바일 트로이 목마(Mobie Trojan horse)
  - Skuller 의 발견하였으며 2004 년에 처음 출현
  - 이동식 웜과 같이 주 목표 디바이스는 스마트폰
  - 휴대전화의 잠금 장치를 해체시키는 주된 수단으로 이용

페이로드: 시스템 파괴

- 소프트웨어가 목표 시스템상에서 활성화되면 다음 관심은 이 시스템상에서 하게 될 활동
- 몇몇 악성 소프트웨어는 비존재 또는 비활동적인 페이로드이고 , 이것의 목적은 감염 대상을 확산
- 많은 바이러스와 웜에서 보여준 초창기의 페이로드는 트리거의 조건이 만족 될 때 감염된 시스템의 데이터 파괴 수행
- 사용자 시스템 상에 원하지 않는 메시지나 내용을 보여주거나 더 심각하게는 다른 변종이 시스템상에서 대한 손상을 입힘
- 페이로드의 주 동작은 컴퓨터 시스템의 소프트웨어 , 하드웨어 또는 사용자 데이터의 무결성을 공격
- 페이로드는 바로 발생하지 않을 수도 있지만 특별한 트리거 조건이 부합될 때 활성화됨

페이로드: 시스템 파괴

- 데이터 파괴
  - 체르노빌 바이러스 (Chernobyl virus)
    - 윈도우 95, 98 95, 98 95, 98 의 메모리상에서 기생하면서 데이터를 파괴시키는 초창기의 예
    - 1998 년에 처음 발견되었으며 , 감염은 파일을 열 때 실행됨
    - 트리거 데이터가 도착할 때 감염된 시스템상의 하드디스크의 처음 수 megabyte 에 0의 데이터를 쓰는 것에 의하여 원래 데이터가 삭제되며 , 궁극적 으로는 전체 파일시스템의 파괴를 초래함
  - Klez mass -mailing warm
    - 윈도우 95 가 XP 시스템을 감염시키는 파괴적인 웜의 초창기 예
    - 2001 년 10 월에 처음 발생하였고 , 전자메일 주소를 주소록이나 시스템상의 파 일 등에서 복사하여 퍼뜨림
    - 안티 바이러스 프로그램의 실행을 멈추거나 삭제할 수도 있음
  - 랜섬 웨어 (Ransomware)
    - 사용자의 데이터를 암호화하여 이 정보를 복구하는 데 필요한 키를 받는 대가 로 돈을 요구함
    - 예) Cyborg Trojan (1989), Gpcode Trojan (2006)

페이로드: 시스템 파괴

- 랜섬웨어
  - 몸값 (Ransom) 과 소프트웨어 (Software)의 합성어
  - 시스템을 잠그거나 데이터를 암호화해 사용할 수 없도록 하고 이를 인질로 금전을 요구하는 악성 프로그램
  - 신뢰할 수 없는 사이트 , 스팸메일 , 파일공유 사이트, 네트워크망을 통해 유포

페이로드: 시스템 파괴

- 주요 랜섬웨어
  - 워너크라이 (WannaCry)
    - '17 년 5월 12 일(현지 시간 기준) 스페인 , 영국 , 러시아 등을 시작으로 세계에서 피해가 보고된 악성 코드로, 다양한 문서파일 문서파일 (doc, ppt, hwp 등) 외 다수의 파일을 암호화
  - 록키 (Locky Locky Locky)
    - '16 년 3월 이후 이메일을 통해 유포 , 수신인을 속이기 위해 InvoiceInvoice, Refund등의 제목 사용
  - 크립트 XXX(CryptXXX)
    - 지난 2016 년 5월, 해외 백신사의 복호화 툴 공개 이후에 취약한 암호화 방식을 보완한 크립트 XXX 3.0 버전이 유포
    - 비트코인 지불 안내 페이지에는 한글 번역 제공
  - 케르베르 (CERBER)
    - CERBER는 말하는 랜섬웨어로 유명
    - ※ 감염 시에"Attention! Attention! Attention! Your documents, photos, databases and other important files have been encrypted" 음성 메시지 출력
  - 크립토락커 (CryptoLocker)
    - '13 년 9월 최초 발견된 랜섬웨어의 한 종류로 자동실행 등록이름이 크립토락커 (CryptoLocker)로 되어있는 것이 특징
    - 웹사이트 방문 시 취약점을 통해 감염되거나 , E-Mail내 첨부파일을 통해 감염되며, 확장자를 encrypted, ccc 로 변경
  - 테슬라크립트 (TeslaCrypt)
    - '15 년 국내에 많이 유포된 랜섬웨어로 '16 년 5월경 종료로 인해 마스터키가 배포되었음

페이로드: 시스템 파괴

- 현실 세계 손상
  - 시스템 파괴 페이로드의 변종은 물질적인 장치의 손상을 목표로 하였음
  - 체르노빌 바이러스는 데이터를 파괴할 뿐만 아니라 컴퓨터를 부팅 하는 데 사용되는 BIOS 코드에 중복 쓰기를 시도함
  - BIOS 칩을 다시 프로그램하거나 교체될 때까지 시스템을 사용할 수 없게 됨
  - Stuxnet 웜은 특수 산업 제어 시스템 소프트웨어를 목표로 함
  - 제어 시스템이 감염되었다면 제어 시스템이 정상적으로 동작할 수 없도록 원래의 시스템 소프트웨어를 다른 코드로 변경시켜서 부착 된 장비의 고장을 초래함

페이로드: 시스템 파괴

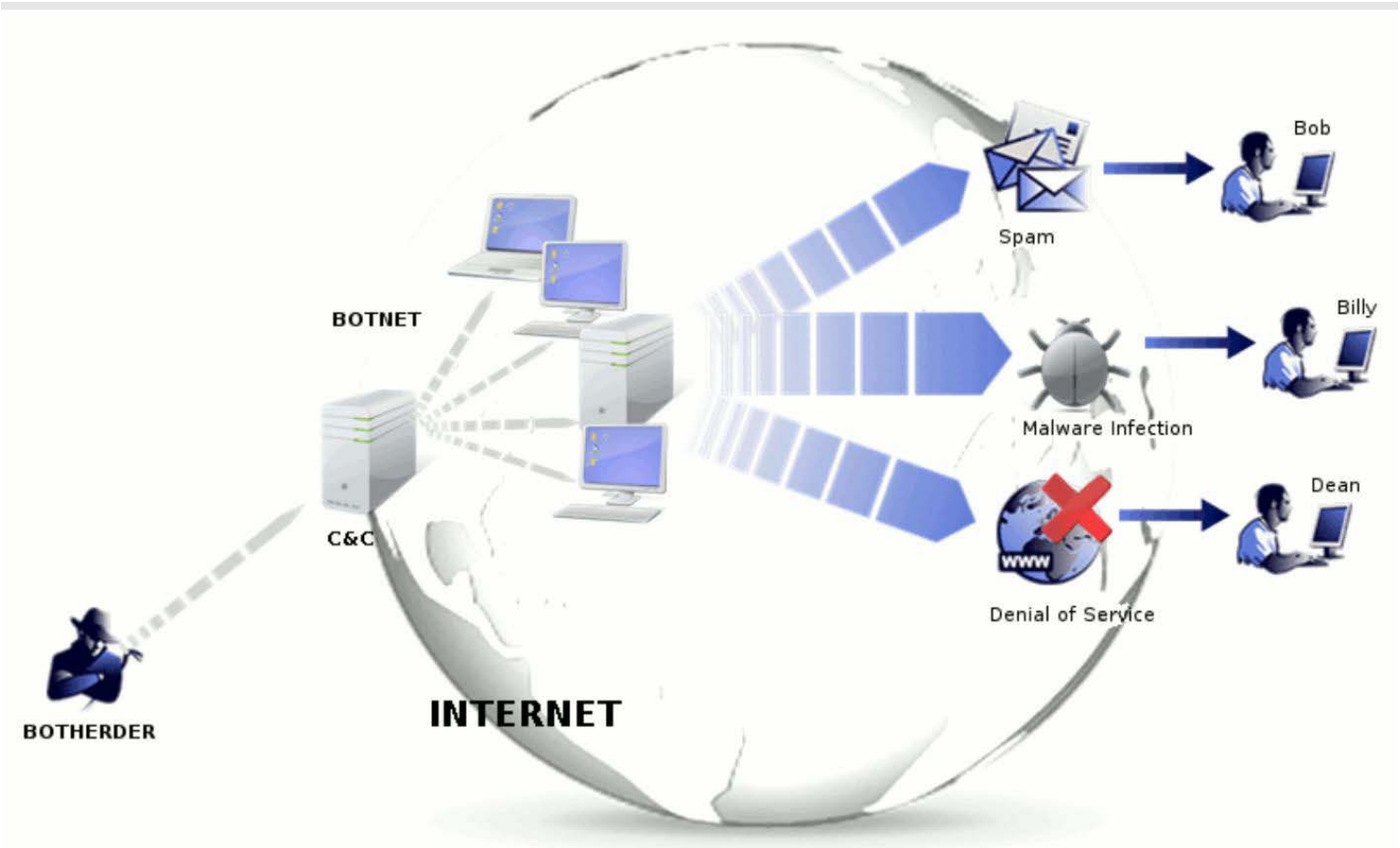
- 논리 폭탄
  - 데이터 파괴 악성 소프트웨어의 주요 구성 요소
  - 조건이 만족되었을 때 폭발하는 악성 소프트웨어에 내장되어 있는 코드
  - 논리 폭탄에 트리거로써 사용될 수 있는 조건의 예
    - 시스템 상에서 파일 또는 장치의 존재 또는 부재
    - 특정 날짜 혹은 업데이트 시기
    - 특정 소프트웨어의 구성 요소 실행
    - 응용프로그램을 실행하는 특정 사용자
  - 트리거가 되면 데이터나 파일 전체를 변경시키거나 삭제하여 기계를 다운시키거나 손상을 유발

페이로드: 공격에이전트-좀비, 봇

- 좀비, 봇은 페이로드의 한 종류로 공격자의 사용에 대하여 감염 시스템의 컴퓨팅이나 네트워크의 자원을 개조하는 악성 소프트웨어임
- 비밀리에 다른 인터넷이 부착된 컴퓨터를 인수한 뒤 그 컴퓨터를 다른 공격을 관리하거나 시작하는 데 사용
- 다른 매개체를 이용하므로 추적하기 어려움
- 로봇은 평범한 제 3자를 통해 수백 또는 수천 대의 컴퓨터에 이식
- 로봇 무리는 종종 상호작용 방식으로 행동하는데 이러한 집합을 봇넷 (botnet)이라고 함
- 감염된 시스템의 무결성과 가용성을 공격함

페이로드: 공격에이전트-좀비, 봇

- 좀비, 봇 공격 예시



페이로드: 공격에이전트-좀비, 봇

- 봇의 용도
  - 1. 분산 서비스 거부 (DDoS) 공격
    - 사용자에 대한 서비스의 손실을 야기하는 컴퓨터 시스템 또는 네트워크에 대한 공격
  - 2. 스팸
    - 봇 넷 또는 수천의 봇의 도움으로 공격자는 대량의 스팸 메일을 보낼 수 있음
  - 3. 스니핑 트래픽
    - 봇은 타협된 시스템에 의해 통과된 관심 텍스트 데이터를 보고 전송하기 위하여 패킷 스니퍼를 이용할 수 있음
    - 스니퍼는 이름이나 비밀번호 같은 민감한 정보를 검색하는 데 사용됨

페이로드: 공격에이전트-좀비, 봇

- 봇의 용도
  - 4. 로깅
    - 타협된 기계가 암호화된 통신 채널 (예, HTTPS, POP3S) 을 사용할 경우 , 희생 컴퓨터의 네트워크 패킷은 패킷을 해독하는 적절한 키가 없기 때문에 쓸모가 없음
    - 감염된 컴퓨터에 키 입력을 캡처하는 키 로거를 사용
    - 공격자는 공격대상의 기밀한 정보를 검색할 수 있음
  - 5. 새로운 악성 소프트웨어 전파
    - 봇넷은 새로운 봇을 전파하는 데 사용됨
    - 모든 봇들은 HTTP나 FTP 를 통해 파일을 다운받아 실행하는 메커니즘을 구현하기 때문에 전파가 매우 쉬움
    - 웜과 메일 바이러스의 전진 기지로서 동작하는 10,000 개의 호스트를 가진 봇넷은 더 빨리 전파하여 더 많은 해를 입힘

페이로드: 공격에이전트-좀비, 봇

- 봇의 용도
  - 6.광고 추가 및 BHO (Browser Helper Object) 설치
    - 가짜 웹사이트에 광고를 설정하는 것에 의하여 동작함
    - 웹사이트의 운영자는 광고 클릭에 대해 지불하도록 일부 호스팅 업체와 계약을 맺기도 함
    - 봇넷의 도움으로 이런 클릭이 수천 개의 봇이 팝업을 클릭하도록 자동화 될 수 있음
  - 7. IRC 채팅 네트워크 공격
    - 클론공격 : 봇넷은 인터넷 릴레이 채팅 (IRC)에 대한 공격에 사용
    - 피해자는 복제된 봇에 의해서 수천 개의 봇이나 수천 번의 채널 요구에 의하여 트래픽이 폭증
    - 피해 IRC 네트워크는 DDoS 공격과 유사하게 마비를 초래
  - 8. 온라인 설문 /게임 조작
    - 봇넷으로부터 조작하기 쉬움
    - 각각의 봇은 다른 IP 주소를 가지고 있기 대문에 모든 투표는 진짜 사람이 투표를 한 것과 같은 신뢰성을 가짐

페이로드: 정보도용-키로거, 피싱, 스파이웨어

- 자격 증명 도난 , 키로거 , 스파이웨어
  - 사용자는 네트워크 패킷의 모니터링에 의해 캡처되는 것을 보호하기 위해 암호화된 통신 채널 (예 HTTPS, POP3S) 을 이용
  - 금융 , 게임 및 이와 관련된 사이트에 자신의 로그인 및 비밀번호 자격 증 명을 보냄
  - 공격자는 감염된 컴퓨터에 중요한 정보를 모니터링할 수 있도록 감염된 기계에 키 누름을 캡처할 수 있는 키로거를 설치
  - 이것은 감염된 시스템에 입력된 모든 텍스트의 사본을 수신하기 때문에 키로거는 전형적으로 원하는 키워드에 가까운 정보 (예 :“ 로그인”, “ 비밀번호”, “paypal.com”) 만을 반환하도록 필터링 기법을 구현
  - 키로거 사용의 대처로 일부 은행 및 다른 사이트들은 패스워드와 같은 중요한 정보를 그래픽 애플릿을 사용하여 전환

페이로드: 은신-백도어, 루트킷

- 백도어(Back door)
  - 트랩도어라고 알려져 있으며 , 평소의 보안 접속 절차를 통하지 않고 접속 을 허용하는 지점
  - 유지 관리 후크 : 프로그래머는 수년 동안 프로그램을 디버그하고 시험하기 위하여 백도어를 사용
  - 백도어를 두는 이유 :프로그램을 신속하게 디버그 하기 위해 개발자가 특수한 특권을 얻거나 모든 설정과 인증을 회피하는 것을 선호
  - 부도덕적인 외부자가 인증되지 않은 접속을 위하여 백도어를 사용할 때 매우 위협적임
  - 최근 백도어는 공격자가 감염된 시스템에 연결하고 명령을 내릴 수 있는 비표준 포트의 네트워크 서비스를 도청하는 데 사용될 수 있도록 구현

페이로드: 은신-백도어, 루트킷

- 루트킷 (Root kit)
  - 관리자의 권한으로 해당 시스템 에 대한 은밀한 접속을 유지하려는 시스템에 설치된 프로그램의 집합
  - 악의적이고 은밀한 방법으로 호스트의 표준 기능을 변화
  - 공격자는 루트에 접속해서 시스템을 완전히 제어하고 프로그램과 파일 , 모니터 프로세스를 추가하거나 바꿀 수 있음
  - 네트워크 트래픽 위조 및 변조가 가능하며 백도어에 접속할 수 있음
  - 공격자는 자신의 존재를 숨기기 위하여 시스템에 변화를 줌으로써 사용자에게 루트킷의 존재와 루트킷이 무엇을 변화시켰는지 알기 어렵게 함
  - 숨기는 방법으로는 백신프로그램의 프로세스, 파일, 컴퓨터의 레지스트리 탐색 및 보고하는 것에 대한 기법을 파괴

대비책

- 악성코드에 대한 이상적인 해결책은 예방

네 가지 예방 요소:

- **정책**
- **인식**
- **취약점 보안**
- **위협 최소화**

- 예방의 실패할 경우, 기술적 매커니즘이 다음 위협완화 기능에 사용될 수 있음 :
  - 탐지
  - 식별
  - 제거

대비책

- 안티바이러스 소프트웨어의 구분

제 1세대: simple scanners

- 악성코드를 식별하기 위한 악성코드 시그니처가 요구됨
- 알려진 악성코드 탐지에만 제한됨

두 번째 단계: heuristic scanners

- 악성코드 의 가능성이 있는 인스턴스를 탐색하는 휴리스틱 법칙 사용
- 또 다른 접근으로는 무결성 검사가 있음

세 번째 단계: activity traps

- 감염된 프로그램의 구조 보다는 행동에 의해 악성코드를 식별하는 메모리-상주 프로그램

네 번째 단계: full-featured protection

- 함께 사용되는 다양한 안티-바이러스 기법으로 구성된 패키지
- 스캐닝 및 액티비티 트랩 요소들과 접근제어 기능을 포함

제너릭 암호해독
Generic Decryption (GD)

- 빠른 스캐닝 속도를 유지하면서 안티-바이러스 프로그램이 복잡하고 다양한 바이러스와 기타 악성 코드를 쉽게 탐지할 수 있게 함
- 실행 파일은 다음 요소를 포함하고 있는 GD스캐너를 통해 실행 됨:
  - CPU 에뮬레이터
  - 바이러스 시그니쳐 스캐너
  - 에뮬레이션 제어 모듈
- GD 스캐너에 관한 가장 어려운 설계 이슈는 각 해석(interpretation)을 실행 시간을 판단하는 것

경계 스캐닝 접근 (Perimeter Scanning Approaches)

- 이메일 및 한 기관의 방화벽이나 침입감시시스템(IDS)상에서 실행되는 웹프록시 서비스에 포함됨
- 침입감시시스템(IDS)의 트래픽 분석 요소에도 포함될 수 있음
- 침입 예방 조치, 의심스러운 트래픽 흐름 차단을 포함하고 있음
- 접근이 악성코드 내용을 스캐닝하는 것으로 제한됨

two types of monitoring software