최신 악성코드 기술동향 및 분석 방안 연구

한국인터넷진흥원 ▌임채태 · 오주형 · 정현철

1. 서 론

최근 10년 동안 CIH 바이러스, 슬래머 웜, 7.7 DDoS 등 대규모 사고를 포함한 사이버 공격은 지속적으로 증가하고 있으며, 그 피해는 점차 심각해지고 있다. 최근의 사이버 공격 성격을 공격기법, 피해범위, 해킹동기 측면에서 살펴보면, 초기의 수동적인 공격에서 자동화, 대규모화 되었으며, 최근에서는 은닉화, 지능화, 사회공학적 공격이 가미되는 특성을 보인다. 피해범위는 개별시스템에서 대규모 네트워크, 최근에서는 대규모 네트워크와 함께 특정 개인/기업/국가를 대상으로 정밀화되었다. 해킹동기는 과거 호기심, 자기과시형에서 최근에는 금전적 이득을 목적으로 한 범죄형 해킹, 기존 범죄조직과 연계되는 양상을 보이고 있다.

DDoS, 스팸발송, 정보불법탈취 등 사이버 공격에 악용되는 악성코드는 상당히 빠르게 진화하고 있으며, 공격대상과 공격기법 등을 주어진 조건에 적합하도록 통제하기 위하여 공격자가 쉽게 제어할 수 있는 봇넷이 출현하여, 봇넷을 구성하기 위한 악성봇이 크게 증가하고 있다. 최근 인터넷을 통해 악성코드 전문 제작툴(Malware Construction Kit)이 보급됨에 따라 누구나 지능화된 탐지 회피 기술이 적용된 악성코드를 제작할 수 있게 되었고, 그로 인해 신종 악성코드의 수가 급격히 증가하고 있다. 또한, 분석/회피 기술로써 루트킷, Anti-VM/디버거, 실행 압축 등 다양한 기술들이 사용되고 있으며, 정상 프로세스에 바이너리 코드 삽입, 가짜 gif 헤더를 이용한 원격 명령/제어 등 다양한 고도화된 기술들이 적용되고 있다.

악성코드의 등장과 함께 이에 대응하기 위한 연구 또한 활발하게 진행되고 있다. 대응기술 연구는 호스트 기반의 악성코드 분석 기술, 다양한 경로를 통해 유포되는 악성코드 수집 기술들이 개발되고 있으며, 최근 봇넷이 크게 증가하면서 봇넷을 탐지하기 위한 기술에 대하여 연구가 진행되고 있다. 결론적으로, 현재의 대응 기술은 악성코드의 진화속도에 뒤쳐져 있는 상황이다. 이는 기존의 자기 과시성 사이버 공격이 아닌 금전적/정치적 목적으로 악용된다는 점에서 들키지 않고 보다 오랫동안 사용하고, 원하는 공격을 즉시 수행할 수 있는 강력한 도구를 보유하고자 하는 점에서 기인하였다.

악성코드 또는 감염PC가 사이버 공격자의 주요 자산 또는 수단으로 받아들여짐에 따라 이를 보호하기 위하여 감염여부를 숨기고, 분석을 방해하고, 일시에 확산시키기 위한 다양한 기술들이 개발되어 적용됨에 따라, 그에 대응하여 악성코드의 능동적인 수집기법과 분석방해 무력화, 행위 분석 기술들이 최근 활발하게 연구되고 있으며, 네트워크 기반의 봇넷 탐지 기법 등 관련연구가 활발하게 진행되고 있다. 본 논문에서는 최근 악성코드의 기술동향과 주요 연구동향을 살펴보고, 악성코드로 인한 피해를 최소화하기 위한 대응방안을 제시한다.

2. 최근 악성코드 기술 동향

2.1 신종/변종 악성코드의 급격한 증가

AV-Test.org에서 발표한 내용에 따르면 2004년 9월부터 2009년 5월까지 수집한 악성코드는 시간이 지남에 따라 기하급수적으로 증가하는 것을 확인할 수 있다[1]. 그림 1을 보면 2004년 9월부터 2007년 12월까지 수집한 악성코드는 2008년 1월부터 2009년 5월까지 수집한 악성코드보다 약 2배 이상 많은 것을 알 수 있다. 또한 그림 2에 따르면 2008년 이후부터 기하급수적으로 늘어나는 것을 확인할 수 있으며 2009년 수집된 신규 악성코드의 수는 2007년 평균보다 약 4배가량 증가했다. 이를 반영하듯 Symantec 보고서에 따르면 급격하게 증가하는 악성코드에 대응하기 위한 시그니처의 수도 그림 3과 같이 2007년 대비 2008년에 약 150% 증가하였으며 2002년 대비 2008년 악성코드 시그니처 수는 약 1800% 증가한 것을 확인할 수 있다[14].

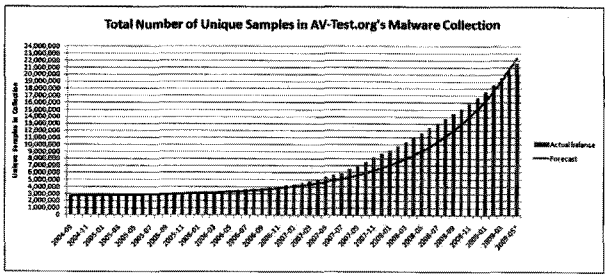

그림 1 수집된 악성코드 합계

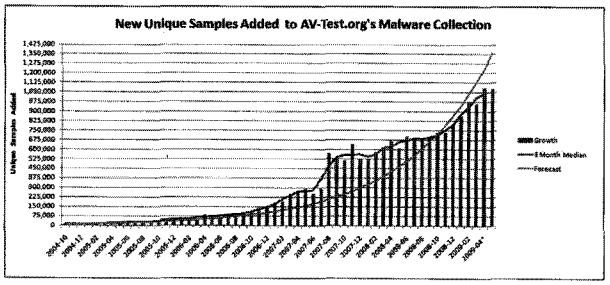

그림 2 신규 악성코드 합계

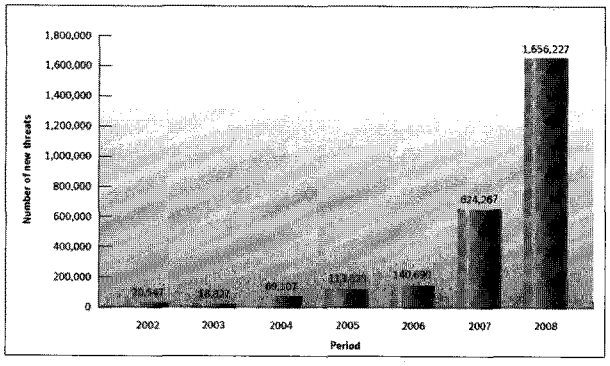

그림 3 연간 새로운 악성코드 시그니처 수

이와 같은 현상은 악성코드를 손쉽게 제작할 수 있는 툴들이 널리 유포되고 있기 때문이다. 툴들은 온라인에서 누구나 쉽게 구할 수 있는 것들도 있으며 전문적인 툴들이 제작되어 팔리고 있다는 것을 통해 유추할 수 있는 현상이다. 이와 같은 툴들은 전문 지식이 없는 초급 해커들도 실행압축, Anti-VM 등 최신 분석 방해 기술이 적용된 악성코드를 쉽게 제작할 수 있어 그 심각성이 증대되고 있다. 그림 4는 2008년부터 인터넷을 통해 쉽게 다운로드 받을 수 있는 대표적인 악성코드 제작 툴이며, 해당 툴은 단순한 버튼 클릭을 통해서 실행압축, 커널 루트킷 등 최신 분석 회피/지연 기능을 쉽게 추가할 수 있다. 그림 5는 2009년 7월 미국에서 3백만대 이상의 PC를 감염시킨 Zeus 악성코드 관리 툴로써 악성코드 생성, 감염 호스트 정보 확인, 원격 명령 제어가 가능하다[12,13].

또한, 러시아 암거래 시장에서 거래되고 있는 악성코드 제작 툴들을 보면 이 같은 현상을 확인할 수 있다. Barracuda Botnet은 각 기능들이 모듈화 되어 있어 필요한 모듈들은 구매하거나 또는 대여하는 것이 가능하며, ZeuS v1.1.2.2와 ElFiesta의 조합인 ZeuEsta Exploit Pack v5.0의 경우 PDF와 SWF와 같은 파일 형식의 취약점을 이용하거나 다른 취약점들을 이용하여 악성코드를 제작할 수 있다[10,11]. 이뿐만 아니라 Limbo Trojan Kit, Neon Exploit System, Elfiesta 등과 같은 수많은 악성코드 제작 툴들이 거래되고 있다.

이와 같이 급격하게 증가하는 악성코드에 대하여 처리량이 한정되어 있는 전문 분석가에 의한 대응은 하루 수집되는 악성코드의 양을 감안한다면 이제는 불가능한 상황이다. 따라서, 다양한 전파경로를 능동적으로 방문하여 악성코드를 수집하고, 대량의 악성코드를 분석하기 위하여 많은 분석 업무가 사람이 아닌 특화된 시스템이 필요하며, 수집, 분석, 대응 전과정의 자동화 기술이 필요하다.

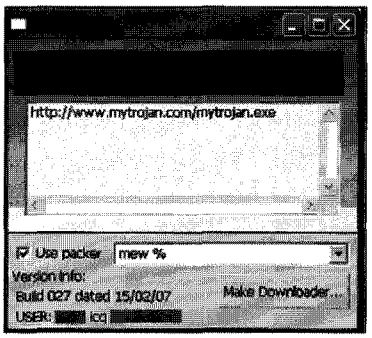

그림 4 악성코드 제작 툴

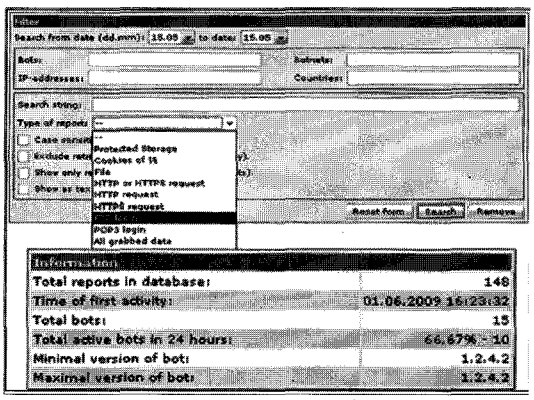

그림 5 좀비PC 제어 GUI

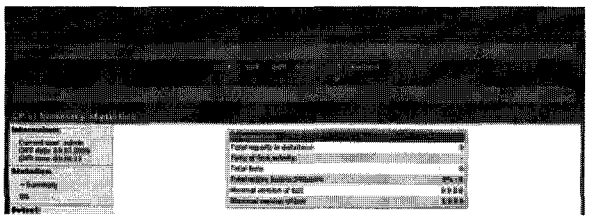

그림 6 ZeuEsta Exploit Pack v7.0

2.2 악성코드 감염 채널의 다양화

기존의 단순 유포 채널에서 보다 다양한 채널(ex. 웹, 이메일, Social Networking Site 등)을 기반으로 사회공학적 기법을 활용하여 악성코드를 유포하고 있다 공격자는 유명한 기업 또는 온라인에서 널리 알려진 사이트를 사칭하거나 사회적으로 이슈가 되는 것들로 위장하여 악성코드를 유포하고자 한다. 아래의 그림 7, 8에서 보는 것과 같이 사용자를 속이기 위해 이메일를 활용하는 것을 확인할 수 있으며 전 세계적으로

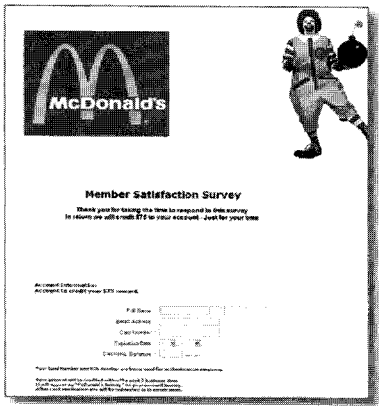

그림 7 맥도날드 피싱 메일

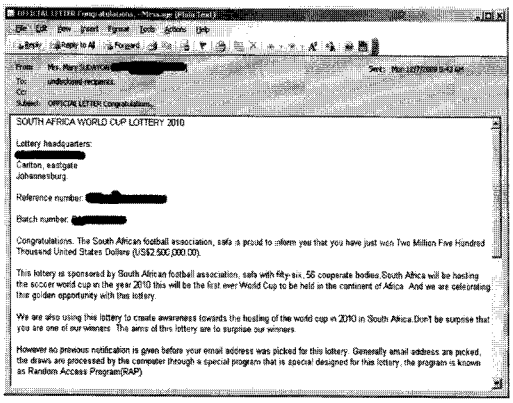

그림 8 남아공 월드컵 피싱 메일

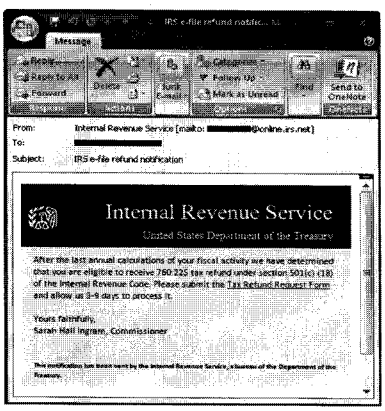

그림 9 IRS를 사칭한 ZeuS 메일

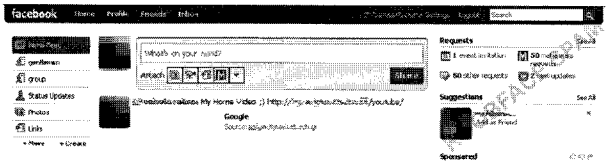

그림 10 Facebook의 Koobface 메시지

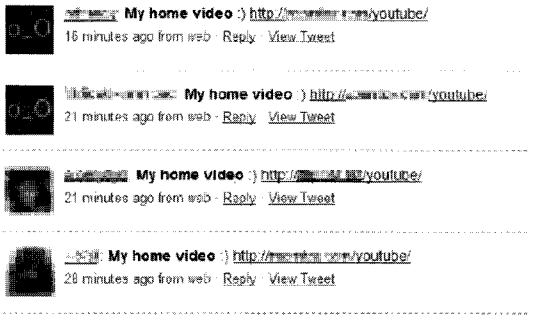

그림 11 Koobface Twitter 스팸

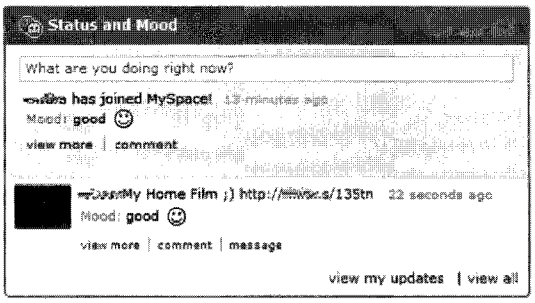

그림 12 Koobface Myspace 스팸

유명한 ZeuS 봇넷의 경우에도 그림 9와 같이 IRS(Internal Revenue Service)를 사칭한 이메일을 통해 전파하였다.

또한 TrendMicro 보고서에 따르면 그림 10~12에서는 보는 것과 같이 배포를 위해 SNS 사이트를 사용하고 있으며 최초에는 Facebook을 배포 채널로 사용하였으나 새로운 변종들에서는 Facebook을 포함한 Twitter, Myspace, Youtube와 같은 사이트를 배포채널로 사용하고 있는 것을 확인할 수 있다[15].

2.3 은닉화, 분석방해/지연 등 기술 적용의 보편화

최근 악성코드들은 발견되더라도 분석이 어렵도록 하기 위해 다양한 분석 방해/지연 기연 기술들을 적용하고 있다. 또한 좀비PC에 오랜 시간 상주할 수 있을수록 금전적인 이익을 취할 수 있기 때문에 최근 악성코드들은 Rootkit 기능을 비롯한 다양한 은닉화 기술이 적용되고 있다.

가장 대표적인 악성코드 분석 방해/지연 기술은 Packing이다. 일반적인 Binary 파일을 Packer로 Packing하게 되면 기존의 실행코드는 Packing 알고리즘에 의해

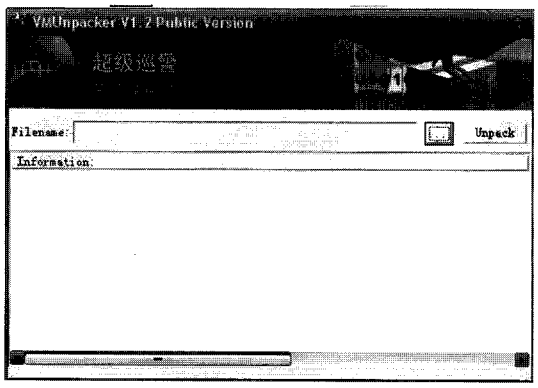

그림 13 VMUnpacker

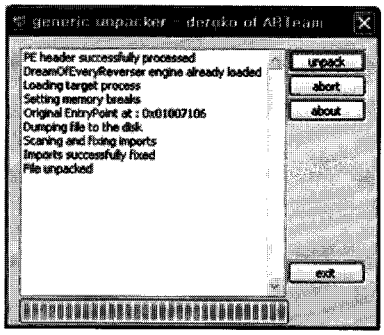

그림 14 Generic Unpacker

변조되기 때문에 Packing된 악성코드를 분석하기 위해서는 Packing되기 이전의 상태로 되돌려야 한다. 알려진 Packer로 Packing된 경우라면 현재 다양한 Unpacker(ex.VMUnpacker[8], Generic Unpacker[9] 등)들이 존재하기 때문에 쉽게 Unpacking할 수 있고 악성코드를 분석할 수 있을 것이다. 그러나 알려지지 않은 Packer로 Packing되었거나 다수의 Packer로 Packing되었다면 Packing 이전의 상태로 복원하는 것은 쉽지 않는 일이다.

또 다른 분석 방해/지연 기법으로는 Anti-VM, Anti-Debugging 기법 등이 있다. 악성코드를 분석하기 위해 분석가들은 가상환경과 Debugger를 많이 활용해왔고 지금도 사용하고 있다. Anit-VM과 Anti-Debugging들은 이러한 점들을 악성코드가 파악하여 이를 회피하기 위한 기술이다. Anti-VM은 악성코드를 분석하기 위해 많이 사용하는 가상 환경을 탐지하여 가상환경에서 악성코드를 실행시키며 다른 행위를 하거나 실행을 종료하는 기술이며 Anti-Debugging은 분석을 위하 악성코드를 Debuger에서 실행시키면 이를 탐지하여 다른 행위를 하거나 실행을 종료하는 기술이다. 이와 같은 분석 방해 기법이 적용된 악성코드들을 분석하기 위해서는 악성코드에서 악성행위를 하는 코드를 분석하기 위해 몇 가지 장벽을 지나서 본래의 악성행위를 유발하는 코드에 접근할 수 있기 때문에 분석이 어렵게 되고 분석이 지연되는 것이다.

이와 더불어 악성코드들은 좀비PC에 오래 상주하기 위해 Rootkit 기술들을 비롯한 다양한 은닉 기술들을 사용하고 있다. 이와 같은 은닉 기술의 대표적인 방식이 악성코드의 실행에 따라 생성되는 프로세스의 정보를 숨기는 방식이다. 프로세스 정보를 숨기기 위해 프로레스 정보를 조회하기 위한 Win 32 API를 Hooking하여 호출 결과 값에서 악성 프로세스 정보를 삭제하거나 EPROCESS와 같은 커널 레벨의 구조체의 정보를 수정하는 방식을 사용했다. 그러나 이와 같은 방식은 SSDT, Kernel 구조체 재생성 등의 기법을 통해 탐지가 가능하기 때문에 Windows의 정상 프로세스(ex. svchost.exe, winlogon.exe 등)에 악성 바이너리를 Injection하여 동작하는 악성코드가 증가하고 있다. 이의 대표적인 예로, 2009년 5월 전체 스팸 메일의 45.8%를 발송하고 있는 Cutwail은 윈도우 OS의 svchost.exe 프로세스에 실행 가능한 바이너리 코드(Executable Code)를 삽입하고, 이를 통해 명령/제어 메시지 수신, 스팸 발송 등의 기능을 수행하였다.

또한, 기존의 악성코드는 감염 호스트를 원격에서 제어하고 명령을 전달하기 위해 텍스트 기반의 명령어 전달 방식을 사용하였으나. 최근 보안솔루션들이 응용 레벨의 프로토콜 파싱, 패턴 매칭 등의 기능을 제공함에 따라 명령 메시지들이 쉽게 차단 될 수 있다. 2009년 3월 Secureworks에 의해 보고된 Cimbot은 이러한 단점을 극복하기 위해 gif파일로 위장한 명령 메시지 사용한다. 그림 15와 같이 HTTP 응답 메시지의 Content-Type을 Image/gif로 설정하였기 때문에 대부분의 보안 장비에서 정상적인 메시지로 처리하게 되지만, 실제로 HTTP Body의 16 바이트부터 복호화하기 위한 키와 암호화된 명령/제어 메시지, 스팸 발송을 위한 템플릿이 내장되어 있는 것을 확인할 수 있다.

이렇듯 최근 악성코드는 자신의 실행 정보를 삭제하는 방식을 통해 은닉하거나 발견되더라도 분석이 어

그림 15 gif파일 위장 Cimbot 명령/제어 메시지

렵도록 하기 위한 다양한 기술들을 적용하고 있으며 지금도 그 기술들은 발전하고 있다. 그러므로 급증하는 악성코드와 분석 방해 기법의 발전에 신속하게 대응하기 위해서는 앞서 설명한 다양한 분석 방해 기법 및 은닉화 기술들이 적용된 지능형 악성코드를 탐지하고 분석 방해기법들을 무력화하기 위한 기술 개발이 선행되어야 한다.

2.4 세부 기능 모듈들의 그룹화 및 분업화

과거 많은 악성코드들은 기능이 단순하거나 또는 기능이 복잡하더라도 하나의 파일로 구성되어 배포되어 왔다. 그러나 최근 악성코드들은 실행 정보의 은닉, 코드 난독화 등과 같은 기술과 함께 복잡한 기능을 포함하고 있어 하나의 파일로 구성하기에는 그 크기가 크고 배포하기 용이하지 않다. 그 결과 악성코드를 보다 빠르게 배포하고 쉽게 변경이 가능하도록 하기 위해 악성코드들은 바이너리 업데이트, 스팸 발송, DDoS 등 각각의 기능을 전담하는 모듈들로 나뉘고 각 모듈들이 유기적으로 협력하는 하나의 악성코드 그룹 형태를 보이고 있다. 대표적으로 Koobface를 살펴보면 그림 16에서 보는 것과 같이 다운로더, 감염 확산 등의 역할을 담당하는 파일들로 나누어져 있는 것을 확인할 수 있다. 또한 그림 17의 Cutwail 역시 5개 이상의 악성 파일들이 역할을 분담하는 것을 확인할 수 있다.

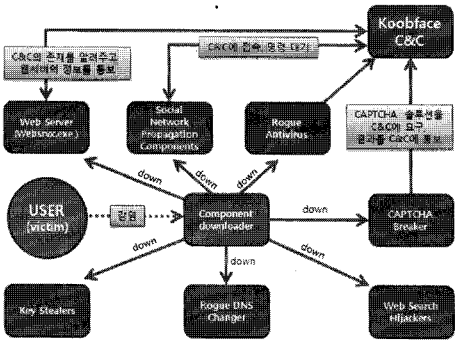

그림 16 Koobface 모듈 구성도

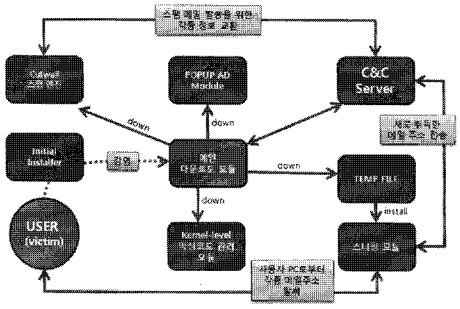

그림 17 Cutwail 모듈 구성도

이와 같이 악성코드가 하나의 파일이 아닌 기능별로 모듈화되어 각각 배포되고 서로 연관관계를 유지하고 있는 패밀리와 그룹형태로 발전하고 있으나 각각의 코드에 대한 분석도 쉽지 않는 상황에서 악성코드 전반에 걸친 파악은 더더욱 힘들어지고 있다.

그러므로 앞으로 하나의 파일을 대상으로 하는 악성코드 분석뿐만 아니라 분석한 결과를 토대로 여러 파일들의 상관관계를 파악할 수 있는 기술이 필요하다. 또한 각각의 악성코드들이 서로 다른 기능과 서로 다른 모습들로 배포되더라도 악성코드들에 대한 전체적인 분류와 대응이 가능하기 위한 기술 개발이 필요하다.

3. 악성코드 수집/분석 기술 동향

3.1 악성코드 수집 기술 동향

최근 악성코드는 감염 전파를 위해 사용자 PC 또는 서버 PC의 취약 서비스를 공격하는 방식에서 벗어나 사용자의 웹서핑 중 drive-by-download, 스팸 메일 본문 URL, 첨부파일을 통한 감염 등과 같은 사회 공학적 기법으로 변경되고 있는 추세이다[19]. Dirve-by download는 사용자의 동의 없이 임의의 코드가 다운로드 되는 것을 의미한다. 사용자 몰래 다운로드된 악성코드 파일은 개인정보 탈취, 스팸 발송 등의 다양한 악성행위를 수행하게 된다. 또한 많은 사용자가 방문하는 사이트에 Drive-by download를 유발하는 악성 스크립트가 링크되어 있을 경우, 짧은 시간에 많은 사용자가 감염될 수 있다. 따라서 이러한 Drive-by download를 유발하는 악성 웹 사이트를 조기에 탐지하고, 유포되는 악성코드를 수집할 수 있는 Client Honeypot 기술이 활발히 연구되고 있다.

Client Honeypot은 수동적으로 악성코드의 감염을 기다리는 Server Honeypot과 달리 웹 서핑 등을 통해 악성코드의 감염을 능동적으로 유도하여 악성코드를 수집할 수 있다. Client Honeypot은 공격자에게 매우 제한된 환경을 제공하는 Low-interaction과 실제적인 시스템 및 서비스를 포함하는 환경을 제공하는 High-interaction으로 나눌 수 있다. 대표적인 Client Honeypot으로 Honeymonkey, UWSpycrawler 등이 있으며, 그림 18에서 보는 것과 같이 대부분이 High-interaction Client Honeypot인 것을 확인할 수 있다. Low-interaction Client Honeypot은 High-interaction Client Honeypot에 비해 제한된 공격 환경을 제공하기 때문에 악성코드 감염을 위한 다양한 공격을 지원하지 못하므로, 악성코드 수집에 어려움이 있다. 또한, Drive-by download

를 위해 사용되는 악성스크립트는 사용자 브라우저에서 동적으로 생성되어 악성코드 감염을 유도하기 때문에 웹 클로러를 사용하는 Low-interaction Client Honeypot에서 수집하기 어려운 단점이 있다.

HoneyMonkey는 마이크로소프트에서 자사의 Virtual PC를 이용하여 구현한 High-interaction Client Honeypot으로, 인터넷 익스플로러를 통해 의심 URL을 직접 방문한 다음 생성되는 파일의 행위를 모니터링해서 악성코드를 수집할 수 있는 기능을 제공한다[20].

또한, 비영리 연구 그룹인 허니넷 프로젝트는 VMWare를 이용해서 High-interaction을 구현한 Capture-HPC를 개발하였다. Capture-HPC는 HoneyMonkey와 유사하게 의심 URL 방문 후 발생되는 시스템 상태를 모니터링해서 비정상적인 변화를 유발하는 악성코드를 탐지하고 수집할 수 있는 기능을 제공한다[21].

독일 Mannheim 대학교에서 개발한 Monkey-spider는 Heritrix 크롤러를 이용한 Low-interaction Client Honeypot으로 의심 URL을 방문한 다음 다운로드 받은 웹 컨텐트를 분석해서 악성코드 유무를 판단하게 된다. 그러나, Monkey-spider와 같은 Low-interaction Client Honeypot은 Drive-by-download를 성공적으로 수행되게 할 수 있는 사용자 환경이 미약하므로, 최근 다양한 공격기법을 통해 유포되는 악성코드 수집에 어려움이 있다[22].

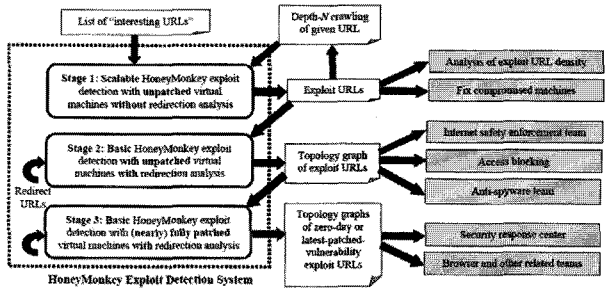

그림 18 HoneyMonkey 동작 구조 및 분석 방식

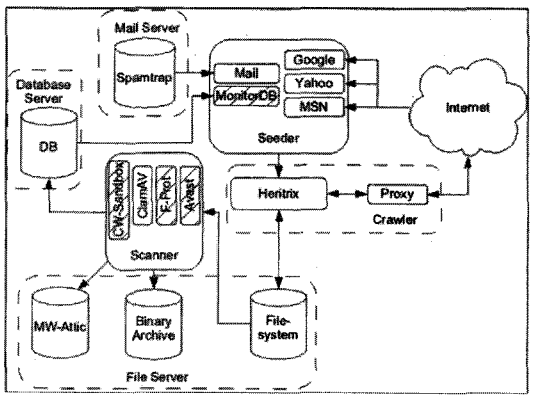

그림 19 Monkey-spider 아키텍처

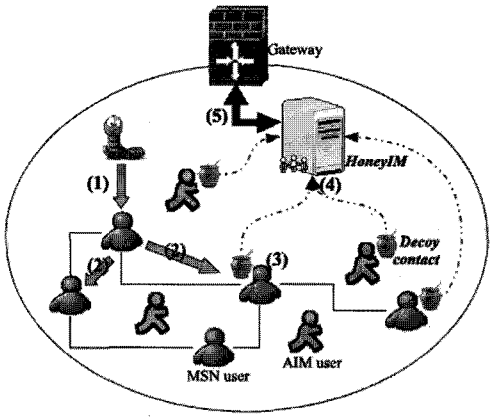

그림 20 HoneyIM 구성도

또한, 2009년부터 IM(Instant Messenger), SNS 사이트를 통한 악성코드 유포 사례가 발생함에 따라 이를 통한 악성코드 수집 기술이 꾸준히 연구되고 있다. 미국 William and Mary 대학에서는 다양한 IM 환경을 지원할 수 있는 메신저 프로그램을 이용하여, 기존 Honeypot과 가짜 메신저 계정을 사용해 악성코드를 수집할 수 있는 HoneyIM을 제안하였다.

웹을 통해 유포되는 악성코드 수집을 위해 다양한 Client Honeypot 기술이 제안되었으나, High-interaction Client Honeypot의 경우 시스템 구축 비용이 많이 들며, 악성코드 수집에 많은 시간이 소모되는 단점이 있으며, Low-interaction Client Honeypot은 동적 생성 스크립트를 통해 감염을 유도하는 악성코드 수집에 어려움이 있다. 또한, 새로운 배포채널로써 등장한 IM, SNS 등을 통해 유포되는 악성코드 수집 기술이 제안되었으나, 대부분이 초기 연구 단계이며 다양한 요구사항을 만족할 수 있는 기술은 미흡한 실정이다.

3.2 악성코드 분석 기술 동향

앞서 언급했듯이 악성코드의 수가 기하급수적으로 증가하고 있다. 이렇게 급증하는 악성코드를 하나하나 수집하는 것도 어렵지만 일일이 분석하는 것 역시 쉽지 않으며 현실적으로 불가능하다. 왜냐하면 악성코드를 분석할 수 있는 인력에 비해 시간대비 악성코드 증가량 너무 크기 때문이다. 그러므로 수동으로 악성코드를 분석하기 보다는 어떻게든 자동으로 악성코드를 분석하는 방법을 강구하게 되었다.

분석이 어려운 또 다른 이유는 바로 분석 방해 기법이다. 최근 대부분의 악성코드가 분석 방해 기법을 적용하고 있기 때문에 이러한 분석 방해 기법을 우회 또는 무력화해야 분석이 가능하다. 따라서 악성코드 자체 파일분석보다는 행위를 분석하기 위한 동적분석 기법에 대하여 많은 연구가 진행되어 왔으며, CWSand-

표 1 악성코드 동적분석 기술 및 제품 비교

분석도구 / 주요 특징	Anubis	ThreatExpert	Botwall	Norman Sandbox	CWSandbox
분석 플랫폼	에뮬레이터	가상머신	에뮬레이터	자체플랫폼	가상머신
판매 형태	무료	무료	상용	상용	무료/상용
분석 가능 파일 포맷	EXE, Javascript, 플래시	EXE	EXE	EXE	EXE, 오피스파일

box[3], Norman Sandbox[4], Anubis[2] 등과 같은 것들이 대표적인 기술에 해당된다. 이와 같은 분석 시스템들의 분석 방식은 악성코드를 실행시켜 악성코드의 프로세스, 파일, 네트워크, 레지스터 등과 관련된 행위 정보들을 모니터링 한다. 표 1은 대표적인 동적분석 기술 및 제품들의 특성에 대한 비교내용이다.

또한 최근에는 악성코드를 분석을 위한 환경에서 실행하여 그 행위를 모니터링 하는 것뿐만 아니라 정적 분석을 함께 병행하여 분석 결과를 보여주려는 연구가 진행되고 있다. 이와 같은 정적 분석의 병행과 함께 동적 분석의 효율성을 높이기 위해 단순히 실행하고 행위를 모니터링 하는 것이 아닌 실행하는 과정에 데이터의 흐름과 정확한 사용 목적 등을 파악하기 위해 실행코드 중간에 분석 코드를 삽입하는 Taint Analysis 방식을 사용하는 연구가 활발히 진행되고 있다[5,16-18].

이에 추가로 악성코드 분석 기술은 단순하게 악성코드를 분석하는 것에서 끝나는 것이 아니라 추가적인 주요 기능들을 제공하고 있다. 가장 대표적인 것이 Packer의 탐지 및 Unpacking이다. 앞서 언급했듯이 대부분의 악성코드들이 Packing 기법을 적용하고 있기 때문에 나타난 결과라 할 수 있다. 이렇게 제공되는 Packer 탐지 및 Unpacking 기술들은 악성코드 분석 기술 단계에서 악성코드를 시그니처 기반으로 어떤 Packer로 Packing되었는지 확인하고 Unpacking 가능한 Packer를 사용했을 경우 이들을 Unpacking하고 본래의 코드를 제공한다. 다음으로는 악성코드의 분류이다. 일반적으로 악성코드 Clustering이라 불리는데 악성코드를 분석하고 분석한 결과를 토대로 하여 현재 분석한 악성코드가 기존의 어떤 악성코드의 변종인지 어떤 악성코드와 가장 유사한지 알려주는 기능이다. 이와 같은 기능은 현재 Anubis에서 악성코드의 행위 모니터링 정보에서 호출되는 API의 정보를 기반으로 악성코드를 Clustering하고 이 결과를 그림 22와 같이 보여주고 있다.

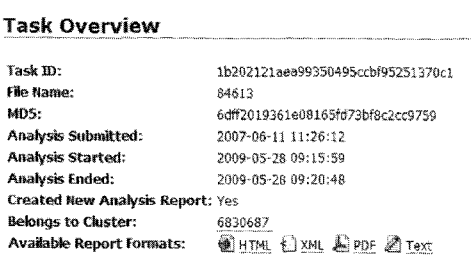

그림 21 Anubis Clustering 결과

마지막으로 악성코드는 이제 실행파일 형태로만 배포되는 것은 아니라 PDF, Word, PPT, Excel, SWF등 다양한 형태로 만들어져 배포되고 있다. 이와 같은 현상은 Adobe의 취약점이 공개되고 이를 대상으로 하는 악성코드가 많이 나타나는 것을 보면 확인할 수 있다. 그러므로 최근 악성코드 분석 기술에서는 기존에 윈도우 실행파일인 PE 파일뿐만 아니라 PDF 파일 등과 같이 다양한 타입의 악성 코드를 분석할 수 있는 기능을 일부 제공되거나 개발되고 있다[6,7].

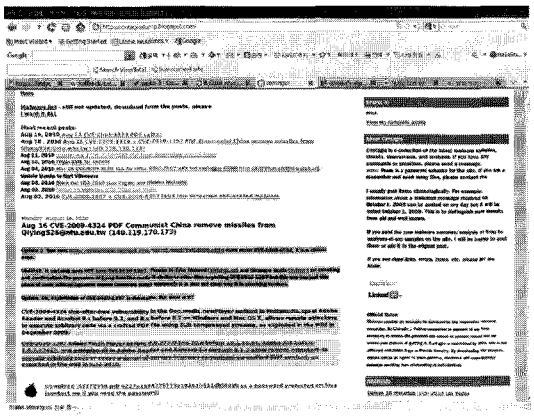

그림 22 PDF 파일의 취약점을 이용한 악성코드 분석

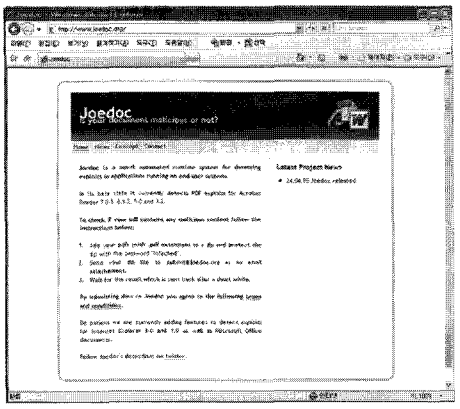

그림 23 PDF 파일 분석을 위한 사이트

이와 같이 최근 악성코드 분석 기술은 기존의 수동 방식을 자동화하는 방향으로 나아가고 있으며 자동화 하는데 있어 보다 정확한 분석 결과를 보여주기 위한 방향으로 나아가고 있다. 이와 더불어 분석에서 끝나는 것이 아니며 분석한 결과를 다시 정리하여 그 의미를 파악하기 위한 연구가 계속 되고 있다.

4. 대응방안

최근의 악성코드 추세에 따른 대응 방안으로는 기존의 악성코드 분석 기술 국한하여 초점을 맞추기보다는 악성코드 수집/분석 측면에서의 근원적 대응, 악성코드 또는 감염PC의 행위 및 전파 현황에 대한 모니터링, 효과적인 예방 및 대응을 위한 대외 공조로 구분하여 살펴보고자 한다. 본 논문에서는 자세한 대응 기술보다는 대응방안에 대한 방향을 제시하고자 한다.

4.1 악성코드 수집/분석 측면에서의 근원적 대응

신종/변종의 급격한 증가, 고도화된 은닉/분석방해 기법의 적용, 감염채널이 다양화되고 있는 악성코드를 수집/분석하기 위해서는 크게 세 가지 요소가 선결되어야 할 것이다. 첫째로 급속히 증가하는 악성코드를 적시에 분석하기 위한 자동화, 두 번째로 은닉/분석방해 기법 무력화 및 악성코드 구성모듈의 정확한 분석을 위한 행위 분석 기술이 적용되어야 한다. 마지막으로 악성코드가 어떠한 경로를 통해 배포되는지를 파악하여 감염경로를 차단함으로써, 추가적인 확산을 방지하여야 한다.

급격히 증가하고 있는 신종/변종 악성코드에 대응하기 위해 다양한 경로로 전파되는 악성코드의 수집, 분석, 시그니쳐 생성, 분류 등 일련의 과정을 자동화하여 대량의 악성코드를 적시에 분석/대응할 수 있는 시스템이 필요하다. 수집에 있어서 스팸트랩, 웹 크롤링, SNS 링크 URL 방문 등 다양한 채널을 능동적으로 수집할 수 있어야 하며, 분석에 있어서는 악성코드 변종 관계를 분석하고 정적인 정보를 추출하기 위한 정적 분석, 악성코드의 행위를 분석하기 위한 동적분석을 자동화하고, 일부 전문가의 수동 분석이 필요한 부분에 있어서는 자동화 시스템과 연계하여 운용할 수 있다.

정상 프로세스에 바이너리 코드를 삽입, Rootkit 기술 적용, 실행압축, Vmware 탐지 등과 같은 분석 분석방해 기법 적용 등 지능형 악성코드 분석은 초기 수집된 악성코드의 정적분석으로는 전체적인 파악이 어려울 수 있다. 따라서, 악성코드가 어떠한 행위를 하는지 분석하는 행위 분석 기술이 적용되어야 한다. 예를 들어, 악성코드를 실행하여 다운받는 실행파일을 추출하고 분석을 요청하고, 메모리 쓰기 행위를 추적해서 정상 프로세스 영역에 메모리 쓰기 이벤트가 발생할 경우 정상 프로세스를 분석 대상으로 추가해서 행위를 분석할 수 있다. 또한 분석플렛폼은 다양한 분석방해 기법을 회피하기 위하여 분석방해 무력화 기법이 적용되어야 한다.

수집되는 과정에서 파악된 관련 사이트 정보를 기반으로 악성코드 경유/유포지를 탐지하고, 탐지사이트들의 관계 및 악용취약점 분석을 통해 얼마나 많이 전파되고 있는지 분석하여, 추가적인 전파를 방지하기 위한 최적의 차단 사이트를 추출하는 기술이 적용되어야 한다.

4.2 악성코드 유포현황 및 행위 등에 대한 모니터링 측면에서의 조기경보

악성코드를 조기에 수집하여 분석하더라고, 사용자는 바이러스 백신 프로그램 등을 이용하여 악성코드 제거를 수행하여야 하지만, 사용자를 직접 제어할 수는 없는 상황이며, 지속적으로 감염PC는 증가하는 추세를 보이고 있는 부분에서 확인할 수 있듯이 악성코드에 대한 근원적인 대응과 더불어 악성코드로 인한 사이버공격을 모니터링하고 대응할 수 있는 수단이 필요하다.

최근 대부분의 심각한 사이버 공격이 좀비PC로 구성된 봇넷을 통해 수행됨에 따라, 악성코드 수집/분석, 네트워크 이상 트래픽 분석, 사고분석 등 다양한 경로를 통해 수집되는 정보를 기반으로 봇넷의 구성 및 분포를 탐지하고, 봇넷 구성/분포 정보를 기반으로 봇넷의 공격행위를 실시간으로 모니터링 하기 위한 기술이 필요하다. 봇넷을 실시간으로 모니터링 함으로써, 사이버 공격을 조기에 탐지가능하며 해당 공격에 대한 즉각적인 대응이 가능하다. 그 대응으로서 봇 C&C와 좀비간 트래픽을 차단하거나 기존 보안솔루션과 연동을 통해 봇넷을 와해하여 공격을 예방하거나 공격대상 및 방법 등에 대한 정보를 실시간으로 확보하여 공격에 즉각 대응 할 수 있다.

4.3 효과적인 대응측면에서의 국내외 공조

최근의 사이버 공격은 국경 구분 없이 넓게 분포되어 있는 악성코드에 의해 공격이 수행되고 있다. 즉 특정 망에서의 사용자는 공격에 의한 피해자가 될 수도 있지만 자신도 모르는 사이에 공격자가 될 수도 있다. 특히 봇넷은 인터넷에 연결된 전세계 어디라도 분

포되고 있으며, 그 범위가 점차 확대되고 있다.

사이버 공격에 대한 예방 측면에서 봇넷의 정확한 규모를 파악하고, 봇넷의 구성/분포를 와해시키기 위해서는 여러 도메인에서의 봇넷 관련 탐지정보가 통합되어야 전체 규모 및 구성을 파악할 수 있다. 봇넷을 통한 사이버 공격이 발생하였을 경우, 효과적인 대응을 위해서도 도메인간 공조가 필요하다. 프라이버시 침해 관련 법적인 제약사항, 망 사업자의 내부 정보를 공개하지 않으려는 성향, 구조적인 복잡도로 인한 공조의 어려움 등 풀어야할 문제도 많지만, 최근의 사이버 공격양상 및 국제적인 추세를 보면, 사이버공격에 대한 국내외 공조는 지속적으로 논의되고 있으며, 가까운 시일 안에 어떠한 모습으로든 구현될 것으로 예상된다.

참고문헌

[1] AVTEST, AV-Test.org

[2] Anubis, anubis.iseclab.org

[3] CWSandbox, www.sunbeltsandbox.com

[4] Norman Sandbox, www.norman.com/technology/norman_sandbox/

[5] BitBlaze, bitblaze.cs.berkeley.edu

[6] CVE-2009-4324 PDF Communist China remove missiles from Qiying526@ntu.edu.tw, contagiodump.blogspot.com

[7] Joedoc, www.joedoc.org

[8] VMUnpacker, http://sucop.com/

[9] Generic Unpacker, www.exetools.com/unpackers.htm

[10] Prices of Russian crimeware, http://evilfingers.blogspot.com/2009/03/russian-prices-of-crimware.html

[11] Prices of Russian crimeware. Part 2, http://evilfingers.blogspot.com/2009/08/prices-of-russian-crimeware-part-2.html

[12] ZeuEsta: ZeuS cybercrime hosting with SPack, http://www.abuse.ch/?p=1662

[13] ZeuS and power Botnet zombie recruitment, http://evilfingers.blogspot.com/2009/10/zeus-and-power-botnet-zombie.html

[14] Symantec Global Internet Security Threat Report Trends for 2008, http://www.symantec.com/connect/downloads/symantec-global-internet-security-threat-report-trends-200, 2009

[15] TrendMicro The Real Face of KOOBFACE : The Largest Web 2.0 Botnet Explained, http://us.trendmicro.com/imperia/md/content/us/trendwatch/researchandanalysis/the_real_face_of_koobface_jul2009.pdf, 2009

[16] Dwan Dong, Bavid Brumley, BitBlaze : A New Approach to Computer Security via Binary Analysis, ICISS 2008, pp.1-25, 2008

[17] Anh M. Nguyen, Nabil Schear, MAVMM : Leightweight and Purpose Built VMM Malware Analysis, Computer Security Applications Conference 2009, pp 441.450, 2009

[18] Zhiqiang Lin,p Xiangyu Zhang, Automatic Reverse Engineering of Data Structures from Binary Execution, NDSS 2010, 2010

[19] Kelly Jackson Higgins, Senior Editor, Dark Reading, "The World's Biggest Botnets", http://www.darkreading.com/document.asp?doc_id=138610&WT.svl=news1_1

[20] Yi-Min Wang, Doug Beck, Xuxian Jiang, and Roussi Roussev, "Automated Web Patrol with Strider HoneyMonkeys: Finding Web Sites That Exploit Browser Vulnerabilities", NDSS(ANNUAL SYMPOSIUM ON NETWORK AND DISTRIBUTED SYSTEM SECURITY)06, August 2006

[21] Radek Hes, Peter Komisarczuk, Ramon Steenson, Christian Seifert, "The Capture-HPC client architecture", 2009, http://ecs.victoria.ac.nz/twiki/pub/Main/TechnicalReportSeries/ECSTR09-11.pdf

[22] Ali Ikinci, Thorsten Holz, Felix Freiling, "MonkeySpider: Detecting Malicious Websites with Low-Interaction Honeyclients", Proceedings of Sicherheit 2008, April, 2008

약 력

임 채 태

2000 충남대학교 컴퓨터과학과 졸업(학사)
2003 포항공과대학교 컴퓨터공학과 졸업(석사)
2009~ 전남대학교 정보보호협동과정(박사과정)
2003~ 한국인터넷진흥원 선임연구원
관심분야 : 봇넷, 악성코드, VoIP 보안
E-mail : chtim@kisa.or.kr

오 주 형

2005 인재대학교 컴퓨터과학과 졸업(학사)
2008 성균관대학교 전자전기 및 컴퓨터공학과 졸업(석사)
2008~ 한국인터넷진흥원 주임연구원
관심분야 : 악성코드 분석, 네트워크 보안, 침해사고 분석
E-mail : jhoh@kisa.or.kr

정 현 철

1996 서울시립대학교 전산통계 학사
1998 광운대학교 전자계산 석사
1996~현재 한국인터넷진흥원 팀장
관심분야 : 악성코드 분석, VoIP보안, 디지 털포렌식, 침해사고대응, 웹 보안
E-mail : hcjung@kisa.or.kr