情 報 保 護 學 會 誌
第17卷 第2號, 2007. 4

윈도우 비스타 보안 기술 분석 : 시만텍 연구를 중심으로

전용희*, 오진태**, 장종수**

요 약

수년간의 작업과 많은 투자의 결과로 윈도우비스타가 탄생되었다. 윈도우비스타는 마이크로소프트 윈도우 운영체제 중에서 가장 안전한 버전이라고 알려져 있다. 본 논문에서는 시만텍사의 윈도우비스타에 대한 연구를 중심으로 윈도우비스타가 채택하고 있는 보안 기술에 대하여 논의하고, 그리고 이런 보안기술이 특정한 종류의 보안 위협을 어떻게 경감시킬 수 있는지 알아본다. 윈도우비스타가 제공하는 향상된 보안기능에도 불구하고 어떠한 위협이 여전히 존재하는지에 대하여도 제시하고자 한다.

Ⅰ. 서 론

마이크로소프트의 새로운 운영체제인 윈도우비스타가 최근 공개됨에 따라, 본 논문에서는 윈도우비스타의 보안 기능에 대하여 살펴보고자 한다. 윈도우비스타를 설계할 때 마이크로소프트(MS)사가 최우선으로 고려한 것이 바로 보안 기능이다. 이를 위하여 윈도우비스타에는 방화벽, 안티-스파이웨어 도구, 안티-바이러스 경보가 내장되어 있다.

컴퓨터와 인터넷이 기업 활동 및 우리 일상생활에서 점차적으로 중요한 역할을 차지하고 보편화함에 따라, 바이러스 감염, 스파이웨어 확산, 스팸 분배 그리고 악성 공격 등에 대하여 쉽게 노출되게 되었고, 그 피해가 증가하고 있는 실정이다. MS사에서는 이런 경향을 예측하고 신뢰 컴퓨팅(Trusted Computing)에 대한 중요성을 더욱 인식하게 되었다. 이에 따라 새롭게 공개된 윈도우비스타는 지속적으로 증가되고 있는 보안 위협 환경에서, 종단 사용자로 하여금 더욱 안전하게 자신의 정보를 보호할 수 있도록 하고 또한 데이터 기밀성(confidentiality), 무결성(integrity) 및 가용성(availability)을 확보하면서 네트워크가 공격에 대하여 더욱더 강건해지는 방법을 제공하고자 한다[1].

보안, 신뢰성 및 관리의 신뢰 수준을 증가시키기 위하여, 윈도우비스타 설계 및 개발에 반영된 원칙은 아래와 같다[1].

- 신뢰 에코시스템(trust ecosystem) 구축: 장비 제조업자와 코드 작성자들이 적절히 식별되고 그들의 조치에 대하여 책임성을 보유함을 의미한다.
- 보안 공학(Engineering for security): 가장 좋은 기량, 보안 진단 도구 및 보안-특정 시험 방법을 확립하고 출판하고 공유함을 뜻한다.
- 보안 단순화(Simplifying security): 산업 표준, 통상적인 개발 도구와 플랫폼, 제품 및 서비스에 통합된 방법의 결합을 통하여 보안을 단순화 한다.
- 기본적으로 안전한 플랫폼: 고립(isolation), 신뢰-기반 복수인자 인증(multifactor authentication), 정책-기반 접근 통제 및 응용들 사이에 통합된 감사를 가능하게 하는 보호 기술.

이런 원칙들이 MS 버전의 신뢰컴퓨팅을 얻기 위한 중요한 이정표가 되도록 보안에 대하여 전체적인 접근을 수용하고 있다. 이처럼 윈도우비스타에서는 보안을 개발 초기부터 최우선으로 하는 정책을 사용하였다. 본 논문에서는 윈도우비스타에 의하여 제공되는 보안 기술

* 대구가톨릭대학교 공과대학 컴퓨터정보통신공학부 (yhjeon@cu.ac.kr)
** 한국전자통신연구원 정보보호연구단 보안응용그룹(showme@etri.re.kr, jsjang@etri.re.kr)

에 대하여, 시만텍(Symantec)사의 윈도우비스타에 대한 보안 기술 연구를 중심으로 기술하며, 윈도우비스타에 의하여 이루어진 보안 기술 개선뿐만 아니라, 아울러 이 새로운 운영체제가 가질 수 있는 위협에 대하여도 제시하고자 한다[2].

Ⅱ. 윈도우비스타 보안기술

윈도우비스타는 MSDL(Microsoft's Security Development Lifecycle)을 처음으로 사용한 윈도우 버전이다. 모든 개발자가 반드시 지켜야 할 반복적인 엔지니어링 프로세스를 정의하고 제품 출시 전에 해당 프로세스를 검증함으로써 개발 초기부터 보안을 최우선으로 한다. SWIAT(Secure Windows Initiative Attack Team)에 의하여 제품의 코드나 설계가 공격에 대하여 만족할 만한 수준의 저항을 얻을 수 있는지를 조사하기 위하여 윈도우비스타에 대하여 광범위한 설계 검토와 침투 시험을 수행하였다. 특별히 집중적인 테스팅을 위한 리스크(risk) 식별을 위하여 1,400 개 이상의 위협 모델이 개발되었다. 또한 개발 과정에서 윈도우 XP에서 발견된 취약성에 대하여 조사하였다.

윈도우비스타에서 채택된 주요 신규 전략을 요약하면 아래와 같다. 보다 자세한 내용은 본 학회지의 다른 논문이나 [1]을 참조할 수 있다.

- 윈도우 서비스 강화(WSD: Windows Service Hardening): 시스템 서비스에 대한 악성 공격 위협을 경감시키기 위하여, 윈도우비스타는 "한정 서비스(restricted services)" 개념을 도입하였다. 이것은 사용자 머신에 대한 무제한적인 손상을 끼칠 수 있는 서비스들의 수를 대폭 감소시키기 위하여, 로컬 머신이나 네트워크에 대한 특권을 가능한 낮게 하여 운영하고 활동을 제한시키는 것이다. 이와 밀접한 관련성이 있는 것으로 개인방화벽이 있으며, 네트워킹 운영을 위한 양방향 및 프로토콜 제한을 시행한다. 사용자와 시스템 관리자를 위하여 관리 복잡성이 도입되지 않도록 하는 것이 특정 목표이다.
- 사용자 계정 제어(UAC : User Account Control): 지금까지의 윈도우 버전에서는 대부분의 사용자 계정이 로컬 관리자 그룹의 한 구성원으로써 구성되어, 사용자에게 여러 응용들을 설치하고 구성하는데 필요한 모든 시스템 특권 및 능력을 주고, 몇 몇 백그라운드 태스크와 디바이스 드라이버를 운영하며, 시스템 구성 변경 및 많은 기본적인 유지보수 업무를 수행하게 하였다. 이 방법이 사용자에게 편의성을 제공하지만, 파일 손상, 구성 변경 등과 같은 특권을 남용할 수 있는 악성 소프트웨어에 대하여 취약하게 만든다. 따라서 UAC는 표준 사용자 특권과 관리자 접근을 필요로 하는 활동을 분리함으로써, 보통의 사용자들에게는 일상적으로 필요한 대부분의 능력을 제공하면서 운영체제 공격을 위한 "표면 구역(surface area)"을 감소시키고 있다.
- 네트워크 접근 보호(NAP: Network Access Protection): 네트워크 관리자로 하여금 침해된 머신을 네트워크로부터 격리할 수 있도록 하는 도구를 제공하여 조직의 네트워크 보안을 유지하도록 한다. 윈도우비스타 안의 NAP 클라이언트가 네트워크 건강 정책의 시행을 단순하게 하고 악성 네트워크 공격으로부터 보호를 한다.

윈도우 비스타에 대한 네트워크 공격 표면 분석에 대한 자세한 내용은 [3]을 참조할 수 있다.

- 악성 소프트웨어와 침입으로 부터의 보호: 윈도우비스타에 "윈도우 방어자(Window Defender)"라는 안티-스파이웨어 솔루션을 통합하였다. 윈도우 방어자는 스파이웨어, 애드웨어, 루트킷트, 봇(bot), 키 스트로커 로거(key stroke logger), 제어 유틸리티 및 다른 형태의 멀웨어(malware)에 대하여 보호하거나 제거하는 것을 돕는다. 윈도우 방화벽은 기본으로 켜지며 윈도우가 시작되면 사용자의 컴퓨터 보호를 시작한다. 양방향 필터링을 통하여 멀웨어의 유입과 확산을 예방하며, 피어-대-피어 공유나 인스턴트 메시징 응용이 다른 컴퓨터에 접촉하거나 응답하는 것을 차단할 수도 있다. 악성 소프트웨어 제거 도구를 매달 갱신하게 하고, 사용자가 안티-바이러스 소프트웨어를 운용하게 함으로써 바이러스를 지속적으로 탐지하고 제거하게 해준다.
- 데이터 보호 능력: 컴퓨터가 권한이 부여되지 않은 사람의 손에 들어갈 때 PC 상의 자료를 보호하기 위하여 하드웨어-실행 데이터 보호 설비인 비트 잠금 드라이브 암호화(BitLocker Drive Encryption)를 가지고 있다. 또한 사용자가 PIN 코드를 제공

하거나 적절한 복호화 키를 가지고 있는 USB 플래시 드라이브를 넣을 때까지 정상 부트 프로세스를 잠그도록 하는 선택사항을 제공한다. 이러한 부가적인 보안 대책으로 다중인자(multifactor) 인증 및 보증을 제공한다. 그리고 중요한 정보의 보안과 무결성을 보호하기 위하여 단지 권한이 부여된 사용자에게만 문서 접근이 허용되고, 이러한 사용자에 의하여 전달하고, 출력하고 공유하는데 특정 정책을 시행하는 저작권 관리 서비스(RMS: Rights Management Service)를 사용한다. 데이터를 다른 사용자나 외부 공격자의 접근으로부터 보호하기 위하여 파일 시스템 암호화(EFS: Encrypting File System)라는 도구를 제공한다. USB 드라이브와 같은 이동 저장 접속을 관리하기 위하여, 그룹 정책을 설정하여 지원되지 않거나 권한이 부여되지 않는 디바이스의 설치를 차단할 수 있도록 하고 있다.

- 진보된 브라우저 보안: 인터넷 익스플로어 7은 브라우저 보안과 보호에 주요한 전진을 하였다고 제시하고 있다. 새로운 브라우저 구조는 사용자들의 브라우징 활동의 보안에서 더욱 안정성을 주기 위하여 설계되고 피싱(phishing) 공격과 부정 웹 사이트로부터 데이터를 보호하도록 해준다. 이를 위하여 견고한 브라우징을 위한 "보호 모드(Protected Mode)", 다양한 인터넷 응용에 따라 적절한 보안 수준을 설정할 수 있도록 하는 "Fix My Settings", "보안 상태 바", "피싱 필터" 등이 제공된다.
- 미래의 위협에 대처하는 능력: 자동 업데이트 분배를 통하여 새로운 멀웨어와 잠재적으로 불필요한 소프트웨어 정의가 윈도우 방어자를 위하여 필요한데로 나올 것이고, 인터넷 익스플로어는 사용자들에게 최근 피싱 사이트들을 경고하게 된다.

Ⅲ. 윈도우비스타 보안기술 분석[2]

시만텍은 2005년 윈도우비스타에 대한 연구를 시작으로 개발과정을 주의 깊게 감시하여 왔다[3-7]. 윈도우비스타 보안에서 채택하고 있는 주요 핵심 기술을 분류하면 아래와 같다.

- 일반적인 익스플로잇 경감
- 커널 무결성
- 시스템 무결성 및 사용자-모드 방어

3.1 일반적인 익스플로잇 경감

이 범주는 특정한 클래스의 코드-레벨 취약성을 가지고 있는 응용을 공격자가 이용하지 못하도록 설계된 것이다. 이것을 위하여 두 개의 주요한 기법이 채택되었다: 개발자-제어와 운영체제 개선. 이런 기법들을 결합하여 메모리 붕괴와 메모리 조작 취약성 이용을 성공적으로 막을 수 있다. 이런 소프트웨어 결점으로는 아래와 같은 것이 있다:

- 스택 버퍼 오버플로우 취약성
- 스택 함수 포인터 덮어쓰기
- 구조적 예외 처리자(SEH: Structured Exception Handler) 덮어쓰기
- 힙 오버플로우 및 구조 조작

일반적인 익스플로잇 경감을 위하여 윈도우비스타에서 도입된 보안기술은 다음과 같다:

- 주소 공간 배치 임의화(ASLR: Address Space Layout Randomization)
- pointer obfuscation
- GS(Buffer Security Check)
- 데이터 실행 방지
- 안전한 힙(heap) 관리자
- SafeSEH

윈도우비스타에서 도입된 기술들이 컴파일 된 응용뿐만 아니라 코어 윈도우 운영체제를 잘 보호하고 있다고 분석되었다. 전통적인 취약성 이용뿐만 아니라, 지금까지 잘 알려진 웜 확산도 하지 못하도록 한다. 결과적으로 코드-레벨 결점으로 인한 전반적인 영향은 크게 감소되었다고 분석하였다.

3.1.1 개발자-제어 기술 분석

개발자-제어 기술은 소프트웨어 엔지니어들이 애플리케이션들을 더욱 강건하게 만들기 위하여 사용될 수 있다. 이 기술은 컴파일러 선택사항을 실행가능하게 하거나 명시된 코드 변경을 통하여 반영될 수 있다. 이 범주에 속하는 기술들로는 아래와 같다: pointer obfuscation, GS, SafeSEH, ASLR, 힙 붕괴 종료(Terminate on Heap Corruption).

이 기술의 성공에 대한 하나의 장벽은 제 3 자인 소프트웨어 공급자들로 하여금 이들 기술을 도입하도록

하는 요구사항에 있다. 소프트웨어 개발자들은 MS사의 개발 도구들의 최신 버전을 특정한 방법으로 이용해야 한다. 이렇게 함으로써 만이 다른 불법이용 기법의 영향을 막거나 최소화하도록 설계된 기능성을 가능하게 할 수 있다. 개발자가 그들의 애플리케이션들을 재 컴파일 하거나, pointer obfuscation과 같은 경우에는 애플리케이션 소스 코드에 대한 변경을 할 때만 이런 개선사항으로부터 혜택을 볼 수 있을 것이다.

새로운 대다수의 MS 애플리케이션들은 이 기술을 사용할 것으로 기대하지만, 예전 소프트웨어와 제 3자에 의하여 작성된 소프트웨어는 그렇지 못한 것이 문제가 된다. 결과적으로 구 MS사 혹은 제 3자 애플리케이션들과 드라이버는 계속해서 위험에 처하게 되고 대부분 보호되지 않은 채로 남게 된다.

시만텍 연구는 어떤 경우에는 코어 윈도우비스타 컴포넌트도 이 기술을 적절히 이용하지 못하고 있다고 지적한다. 구체적으로 윈도우비스타 32-비트의 작은 부분은 GS 기술로 컴파일 되지 않았다. 이런 컴포넌트들이 보호되는 것보다 훨씬 더 큰 위험을 준다고 인식되었다. GS는 비주얼 스튜디오 컴파일러가 지원하는 버퍼 보안 체크 옵션을 의미하며, 윈도우비스타의 GS 보호 분석에 대한 보다 자세한 내용은 [6]을 참조할 수 있다.

결론적으로 이런 윈도우비스타 컴포넌트들이 앞에서 기술한 메모리 붕괴와 메모리 조작 취약성 클래스에 대하여 보호되지 않는다. 이런 상황에 노출될 위험은 낮지만, 잠재적인 공격 표면을 증가시키는 역할을 한다. 시만텍은 공격자들이 이런 취약성을 식별하고 가능성을 조사할 것으로 판단하고 있다.

3.1.2 운영 체제 개선 분석

운영 체제 개선은 코어 운영 체제에 native 한 기술이다. 전반적인 효과에서 개발자-제어 기술과 비슷하지만 이 기능은 코어 운영 체제 안의 컴포넌트에 의하여 궁극적으로 구현된다. 이를 위한 기술로는 다음과 같은 것이 있다: 힙 관리자 개선, 데이터 실행 방지(DEP), SafeSEH, ASLR, 힙 붕괴 종료. 앞 절에서 논의한 바와 같이, 이 범주에 속하는 대부분의 기술들도 소프트웨어 엔지니어들이 그들의 애플리케이션에서 먼저 실행 가능하게 함을 필요로 한다. 위의 기술들 중에서 단지 힙 관리자만이 운영체제 전반에 기본으로 적용된다. DEP는 코어 운영체제 컴포넌트를 위하여서만 실행 가능하여지며, 인터넷 익스플로어와 같은 어떤 통상적인 애플리케이션을 위하여 그렇지 못하다. 나머지 세 개는 개발자가 애플리케이션 개발동안 세부적으로 지원하도록 하는 것을 요구한다.

결과적으로, 제 3자 애플리케이션과 코어 운영체제의 일부로 고려되지 않고 개발된 것들은 이 기술이 도입된다고 해도 동등한 보호를 제공받지 못한다.

3.1.3 DEP의 제한된 범위

윈도우비스타의 기본 설치에서, DEP 만이 코어 운영체제에 적용된다. 이 제한으로 윈도우비스타 상의 제 3자 애플리케이션은 코어 운영체제와 서비스보다 보호를 덜 받게 된다. 이러한 사실이 이런 응용들에 존재하는 취약성을 성공적으로 이용할 수 있는 가능성을 증가시킨다. 인터넷 익스플로어와 같은 통상적인 애플리케이션도 DEP의 혜택을 이용하지 못한다.

3.1.4 ASLR

시만텍은 ASLR의 유효성에 대하여 심층 분석을 수행하였다[5]. 이 기술의 목적은 프로그램들을 메모리에 무작위로 위치시킴으로써 보안을 증가시키기 위한 방법이다. 공격자가 취약 프로그램의 공격동안 정확히 무엇을 목표로 할지 모르도록 하는 것이다. 올바르게 구현된다면 이 기술은 메모리 붕괴와 메모리 조작 취약성 공격을 경감시키는데 매우 효과적이다. 시만텍에서는 메모리 사용을 보기 위하여 시험 장치의 11,500 번 실행을 통한 주소 선택 분포를 조사하였다. 분석 결과 메모리가 완전하게 무작위로 배치되지 않는다는 것을 발견하였다. 이런 임의성(randomness)의 감소가 공격자가 목표로 할 정확한 주소를 추측할 수 있도록 하는 가능성을 증가시킬 수 있는 것으로 분석되었다. MS 사에서는 이 문제를 확인하고 해결 한 것으로 되어있다.

3.2 커널 보안

3.2.1 개요

커널은 시스템 보안이 구축되는 중요한 핵심 컴포넌트이다. 만약 커널이 어떤 방법으로든 침해되거나 파괴되면, 시스템 안전이 위협받게 된다. 최근에 공격적인 루트킷(rootkit) 기술의 진화로 커널 무결성과 보안이 중요한 주제가 되었다. 공격자는 이런 기술들을 이용하여 시스템 내부에 잠재적인 백도어를 설치하는 동안 존재를 숨긴다. 더구나 디지털 저작권(DRM)의 진화가 오

디오 및 비디오 콘텐츠의 불법적인 가로채기를 방지하기 위하여 커널 안전을 위하여 훨씬 강력하다.

이런 이유로, MS사는 윈도우비스타 커널의 신뢰성과 보안을 증진할 수 있는 기술에 많은 힘을 쏟아 왔다. MS에 의하여 채택된 세 가지 기술은 다음과 같다:

- 드라이버 서명(Driver Signing)
- 코드 무결성
- 패치가드(PatchGuard)

드라이버 서명은 시스템에 의하여 적재되는 모든 커널 드라이브들이 신뢰된 당국에 의하여 서명되었다는 것을 보증하기 위하여 설계되었다. 이렇게 하는 것은 MS에 의하여 시험되었거나 혹은 신뢰되는 개발자에 의하여 서명된 코드만이 커널에 적재되게 함으로써, 악성 코드가 운영체제로 적재되는 것을 막기 위함이다.

코드 무결성은 코어 운영체제가 사고로 혹은 악의적으로 손상되지 않도록 하기 위함이다. 이런 손상을 탐지하기 위하여 특정 커널 컴포넌트내의 코어 운영체제 바이너리 상의 디지털 서명서나 관련 해시(hash)를 검증한다.

PatchGuard는 주요 운영체제 구조가 커널 메모리 안에서 패치되거나 확장되는 것으로부터 보호하는 것인데, 논쟁의 여지가 있는 기술이다. 시만텍과 같은 회사에는 이런 패칭 기술을 이용하여 루트킷과 같은 악성 코드에 대하여 최대한 보호를 하기 위하여 가능한 최저 수준에서 보호를 제공하여 왔다. 그러나 같은 기법들이 비밀스런 작업을 하기 위하여 루트킷 작성자에 의하여도 사용되고 있다는 점이다.

3.2.3 커널 무결성 기술 분석

이 기술의 혜택을 받을 수 있는 것은 윈도우비스타 64-비트 버전뿐이다. 당분간 많이 사용될 32-비트 버전은 혜택을 받을 수 없다. 윈도우비스타 개발과정에서도 드러났듯이 PatchGuard는 해커에 의하여 공격을 받을 수 있으며, 64-비트 버전에 있는 커널 무결성 보호 메커니즘도 공격을 지연시킬 수는 있으나 완전하게 막을 수는 없다. 시만텍 연구에서는 세 개 모두의 커널 무결성 기술을 무력화시킬 수 있는 가능성을 조사하였다. 한 사람이 일주일 정도의 작업 후에 모든 세 개의 기술들이 윈도우비스타로부터 무력화되고 제거될 수 있다는 결과를 보여 준다. 이런 경우 이 모든 새로운 보안 기술들이 윈도우비스타에서 통째로 없어질 수 있다는 지적이다.

3.3 시스템 무결성과 사용자-모드 방어

시스템 무결성과 사용자-모드 방어를 위한 MS의 전략은 최소한의 필요한 권한 집합으로 소프트웨어를 수행하는 것이고, 가능하면 애플리케이션을 별도의 구획된 환경에서 수행하는 것이다. 이 방법에다가 소프트웨어 제작자의 신원에 대한 보증을 위하여 서명에 의존함으로써 한층 더 강화시키는 것이다. 이렇게 함으로써 사용자로 하여금 어떤 응용 수행에 대한 정보화된 결정을 내릴 수 있도록 하고 또한 프롬트 될 때 호스트 상에서 행동을 수행하도록 허용한다.

이 기술들의 목적은 관리자로써 모든 것을 운영하는 것보다는 감소된 권한에서 프로그램을 수행하도록 사용자들에게 권장하고, 자신의 행동의 결과를 생각해보도록 하는 것이다. 또한 전체 시스템을 자동적으로 침해하기 위한 악성코드의 능력을 감소시키기 위함이다.

3.3.1 사용자-모드 방어 분석

이 보호 기술의 구현으로 MS가 구상한 보안 목적의 많은 부분이 달성되었지만, 여러 가지 위험이 그대로 남아 있다. 첫 번째는 보통의 운영체제 사용동안 나타나는 많은 대화 상자(dialogue box)와 프롬트에 의하여 제공되는 정보가 부족하다는 것이다. 이 정보의 부족으로 이런 프롬트들이 제시될 때 사용자 측의 무관심을 불러올 수 있다. 따라서 사용자들이 어떤 행동을 일단 허용하기만 한다면, 되돌릴 수 있는 방법이 쉽지 않게 된다.

시만텍이 발견한 또 다른 문제는 디지털 서명서에 대한 UAC의 의존을 해칠 수 있는 윈도우비스타와 함께 출하되는 어떤 실행물(executable)이다. MS 승인 코드라는 것을 의미하는 대화 상자가 사용자에게 제시됨에도 불구하고 비서명 임의의 라이브러리 실행이 가능하다는 것을 시만텍에서는 증명하였다. 따라서 사용자들에게 제시되는 정보가 더 이상 신뢰할 수 없기 때문에 사용자 통지가 훼손될 수 있다는 분석을 내놓고 있다.

마지막으로 더욱 염려스러운 문제는 사용자가 이런 보안 기능들을 궁극적으로 불능화 시킬 수도 있다는 데에 있다. 물론 기업 환경에서는 이런 위험이 관리되겠지만, 가정환경에서는 보안 관리 문제가 거의 불가능하게 될 수도 있기 때문이다. 시만텍 연구에서는 사용자 계정 제어(UAC)가 지역 보안 정책 도구를 통하여 수동으로 쉽게 불능화 될 수 있다는 것을 조사하였다.

Ⅳ. 악성 코드 영향 분석

시만택은 MS 윈도우의 이전 버전 상으로 전파된 위협에 대한 윈도우비스타의 노출을 연구하였다. 이 연구의 목적은 예전의 악성 코드가 비록 윈도우비스타 상에서 돌아가기 위하여 작성되지 않았고 새로운 보안 모델에 적합하지 않을 지라도, 윈도우비스타의 새로운 보안 기술이 이런 악성 코드에 의한 위협을 경감시킬 수 있는지를 알기 위해서이다.

시만텍에서는 악성 코드 실행을 자동화하기 위하여 프레임워크를 개발하였으며, 아래와 같은 세 개의 컴포넌트로 이루어져 있다[4].

- MonServ: 악성코드의 수명동안 시스템 행위를 감시한다.
- Sovmain: 프레임워크의 코어로써, 타겟 악성 코드를 적재하고 실행한다.
- Sovexamine: 데이터를 받아드려 분석에 필요하지 않는 것은 여과하고 HTML 표로 구성한다. 프레임워크 실행의 결과를 분석하기 위하여, 아래와 같은 세 가지의 주요한 질문 부류를 사용하였다.
- 악성코드의 실행이 성공적으로 시작되는가?
- 악성코드가 시스템 재시작에서 생존하는가?
- 왜 악성 코드가 실패하는가?

연구결과는 3%의 백도어가 수정 없이 윈도우비스타 상에서 시스템 재시작을 성공적으로 실행하고 생존할 수 있다는 것을 보여주었다. 아래 표 1에 위협 종류별로 결과 통계를 보여준다[3].

〔표 1〕 결과 통계

분 류	실행 시도	실행	실패율	Reboot율
백도어	197	143(72%)	24(12%)	6(3%)
keylogger	118	60(51%)	11(9%)	5(4%)
루트킷	17	3 (17%)	7 (41%)	0(0%)
대량 메일	113	81(71%)	7 (6%)	4(4%)
트로이목마	210	145(69%)	24(9%)	4(2%)
스파이웨어	260	150(58%)	24(9%)	4(2%)
애드웨어	118	74(62%)	9 (8%)	2(34%)
비분류	728	439(60%)	103(14%)	34(5%)

만약 이런 위협들이 윈도우비스타를 인지하도록 하기위하여 약간의 코드 변경을 하게 되면, 새로운 윈도우비스타 보안 모델 내부에서 성공적으로 수행되게 되고, 그렇게 되면 이 성공률이 엄청나게 증가할 것이라고 판단하고 있다.

커널-기반 루트킷은 성공적으로 설치될 수 없었으며, 이것은 기본적으로 사용자 애플리케이션을 수행하기 위하여 감소된 권한 집합이 사용되는데 있다. 32-비트 윈도우비스타상에서는 만약 어떤 위협이 완전한 관리자의 권한 수준으로 향상시킬 수 있다면, 방해받지 않고 윈도우비스타를 침투할 수 있다. 그렇게 하기위하여 물론 윈도우비스타의 사용자 계정 제어(UAC)를 우회해야 한다. MS사가 처음 생각한 것보다는 훨씬 쉽게 공격프로그램 제작자가 이용할 수 있는 여러 가지 기법들이 존재한다고 지적하고 있다. 이 연구를 통하여 윈도우비스타가 취약성 공격을 제한하고 전체 시스템에 대한 침해 가능성을 감소시키도록 개선이 되었지만, 여전히 기존의 위협이 그대로 존재하며 운영체제의 개선에 의하여 방해받지 않는다는 것을 보여주었다. 따라서 기존 위협 제작자들은 윈도우비스타에 맞추기 위하여 위협 코드를 조금만 변경하면 되고 그 범위 안에서 지속적으로 수행할 수 있게 됨을 나타낸다. 그러므로 윈도우비스타에 도입된 새로운 보안 기술이 기존의 위협에 대하여 완전한 해결책은 되지 못한다는 지적이다.

Ⅴ. 맺음말

본 논문에서는 시만텍에서 수행된 윈도우비스타 보안 기술에 대한 분석을 제시하였다. 시만텍의 연구는 윈도우비스타가 새로운 보안 기술의 도입에도 불구하고 여전히 존재하게 될 위협 노출에 대하여 제시를 하고자 하였다[2].

시만텍은 윈도우비스타의 새로운 보안 특징으로 코어 윈도우 운영체제 취약성을 목표로 하는 광범위한 웜의 사례가 줄어들 것이라고 예측한다. 금세기 초에 발생했던 대부분의 웜들은 이런 종류이다. 웜은 계속 존재할 것이지만 전파 방법이 변화될 것으로 예측하였다. 이러한 경향이 윈도우 XP SP2의 공개 이후 이미 관측되었으며 계속될 것으로 내다보았다. 시만텍은 윈도우비스타 보안 개선으로 윈도우 운영체제를 그동안 목표로 하였든 다른 종류의 악성 코드를 없앨 수는 없다고 믿고 있다.

참고문헌

[1] Microsoft, *Microsoft Windows Vista Security Advancement*, June 2006.

[2] Symantec, Security *Implications of Microsoft Windows* Vista, 2006.

[3] James Hoagland, Matt Conover, Tim Newsham, and Ollie Whitehouse, *Window Vista Network Attack Surface Analysis*, Symantec, 2007.

[4] Symantec, *The Impact of Malicious Code on Windows Vista*, 2007.

[5] Symantec, *An Analysis of Address Space Layout Randomization on Windows Vista*, 2007.

[6] Ollie Whitehouse, *Analysis of GS protections in Windows Vista*, Symantec, 2007.

[7] James Hoagland, *The Teredo Protocol: Tunneling Past Network Security and Other Security Implications*, Symantec, 2007.

〈著者紹介〉

전 용 희 (Yong-Hee Jeon)

1971.3~1978.2 : 고려대학교 전기전자전파공학부

1985.8~1987.8 : 미국 플로리다공대 대학원 컴퓨터공학과

1987.8~1992.12 : 미국 노스캐롤라이나주립대 대학원 Elec. and Comp. Eng. 석사, 박사

1978. 1~1978.11 : 삼성중공업(주)

1978.11~1985.7 : 한국전력기술(주)

1979.6~1980.6 : 벨기에 벨가톰사 연수

1989.1~1989.6 : 미국 노스캐롤라이나주립대 Dept of Elec. and Comp. Eng. TA

1989.7~1992.9 : 미국 노스캐롤라이나주립대 부설 CCSP (Center For Comm. & Signal Processing) RA

1992.10~1994.2 : 한국전자통신연구원 광대역통신망연구부 선임연구원

1994.3~현재 : 대구가톨릭대학교 컴퓨터·정보통신공학부 교수

2001.3~2003.2 : 대구가톨릭대학교 공과대학장 역임

2004.2~2005.2 : 한국전자통신연구원 정보보호연구단 초빙연구원

관심분야 : 네트워크 보안, BcN QoS & Security, 웜 모델링 및 대응 기술, 통신망 성능분석

오 진 태(Jintae Oh)

1990.2 : 경북대학교 전자공학과 공학사

1992.2 : 경북대학교 전자공학과 석사

1992.2~1998.2 : 한국전자통신연구원 선임연구원

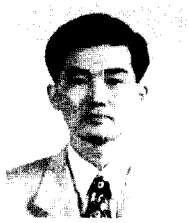
1998.3~1999.1 : 미국 MinMax Tech. 연구원

1999.2~2001.10 : 미국 Engedi Networks. Director

2001.10~2003.1 : 미국 Winnow Tech. Co-founder, CTO 부사장

2003.3~현재 : 한국전자통신연구원 선임연구원, 보안게이트웨이연구팀 팀장

관심분야 : 네트워크보안, 비정상행위탐지 기술, 공격 시그니처 자동생성기술, 보안 하드웨어기술



장 종 수(Jong-Soo Jang)

1984년 : 경북대학교 전자공학과 학사,

1986년 : 경북대학교 대학원 전자공학과 석사,

2000년 : 충북대학교 대학원 컴퓨터공학과 박사

1989년 7월~현재 : 한국전자통신연구원 정보보호연구단 보안응용그룹 그룹장/책임연구원

관심분야 : Network Security, 정책기반보안관리, 비정상트래픽탐지, 유해정보차단