情報保護學會論文誌
第18卷 第3號, 2008. 6

키보드 보안의 근본적인 취약점 분석

배 광 진, 임 강 빈†‡
순천향대학교

Analysis of an Intrinsic Vulnerability on Keyboard Security

Kwangjin Bae, Kangbin Yim†‡
Soonchunhyang University

요 약

본 논문은 인터넷 금융거래에서 공통적으로 이용하는 키보드 기반 사용자 아이디 및 패스워드 인증에서의 근본적인 취약점을 제시한다. 또한, 키보드 입력정보의 유출 방지를 위한 기존 기술이 동작하고 있는 실제의 상황에서 사용자 패스워드의 취득이 매우 간단하게 이루어질 수 있음을 보인다. 추가적으로 이러한 취약점을 해결하기 위한 방안으로서 장단기적으로 고려하여야 하는 하드웨어 및 소프트웨어 관점에서의 접근방법에 대하여 제시한다. 다양한 방안들을 구현하고 적용하여 보다 신속하게 대응책을 마련함으로써 기존의 인터넷 금융거래 환경의 보안 수준을 개선해야 할 것으로 사료된다.

ABSTRACT

This paper analyzes the intrinsic vulnerability problems of the authentication system for Internet commerce based on the ID and password strings gathered from the computer keyboard. Through the found vulnerability, it is easy to sniff user passwords as well as any other keyboard inputs even when each of the existing keyboard protection softwares is running. We propose several countermeasures against the possible attacks to the vulnerability at both points of the hardware and the software concerns. The more secure environment for Internet commerce is highly required by implementing the proposed countermeasures.

Keywords : keyboard sniff, volatile, password authentication, hardware vulnerability, access control

Ⅰ. 서 론

네트워크의 발전과 함께 인터넷을 이용한 뱅킹 서비스 및 전자지불 서비스 등이 일반화되었다. 이는 인터넷을 이용한 거래가 신뢰를 얻고 있음을 입증하는 증거로서 앞으로는 대개의 재화 교환이 인터넷을 이용한 전자상거래 형태로 이루어질 것으로 전망된다.

기술의 공급자 측면에서는 많은 연구의 결과로 인터넷을 통한 전자거래의 안전성 지원을 위한 기반구조가 이미 마련되었다. 이러한 기반 구조는 안전성에 대한 기술적 확인뿐만 아니라 많은 경험과 실험을 통하여 그 실효성 및 안전성이 확인되고 있으며 따라서 실용 가능한 기술로 정립되고 있다[1].

그러나 대개의 전자거래 서비스의 경우 문자열 형태의 아이디 및 패스워드를 기반으로 사용자 인증이 이루어지며 이 과정에서 사용자의 컴퓨터 키보드를 통하여 해당 정보를 수집하는 방식을 고수하고 있다. 따라서 아무리 안전한 기반 구조를 가지고 있다 할지라도 사용자 키보드의 감시를 완벽하게 해낸다면 최종 사용자의 패

접수일 : 2008년 3월 19일; 채택일 : 2008년 4월 16일
† 주저자, yim@sch.ac.kr
‡ 교신저자, yim@sch.ac.kr

스워드를 훔쳐냄으로써 인증시스템을 무력화할 수 있는 것이다.

이러한 문제에 대응하기 위하여 이미지 기반의 패스워드 인증방법[2] 등이 시도되고 있으나 그 실효성을 인정받지 못하고 있는 실정이며 대부분은 키보드 감시를 방지하기 위한 보안 소프트웨어가 주요사이트에서 제공되어 사용자에게 필수적으로 설치하여 운용하도록 요구하고 있다. 그러나 현재 컴퓨터의 구조적인 측면으로 볼 때 소프트웨어에 의한 키보드 감시의 완벽한 차단은 쉽지 않을 것으로 보인다. 이는 현재 컴퓨터에 내장된 키보드 처리를 위한 컨트롤러의 구조적인 문제에 더불어 운영체제에서의 사용자 특권수준 관리의 문제가 일조하고 있어 보다 근본적인 대책이 요구된다.

본 논문은 매우 간단한 기술로도 상기와 같은 키보드 감시를 통한 패스워드의 유출이 가능함을 보이고 이에 대한 근본적인 원인 분석을 제시하며 이러한 취약점의 해결을 위하여 요구되는 몇 가지 대응방안을 제안한다.

본 논문의 구성은 다음과 같다. 제2장에서는 현재 사용 가능한 키보드 보안 기술에 대하여 소개하고 제3장에서는 키보드 보안이 가지고 있는 근본적인 취약점에 대하여 서술하였다. 제4장에서는 3장에서 서술한 취약점을 이용하여 구성 가능한 키보드 스니핑 프로그램의 실례를 보였으며 제5장에서 키보드 보안과 더불어 하드웨어 보안 취약점과 관련한 앞으로의 해결과제를 논하는 것으로 결론을 내린다.

Ⅱ. 키보드 보안 기술

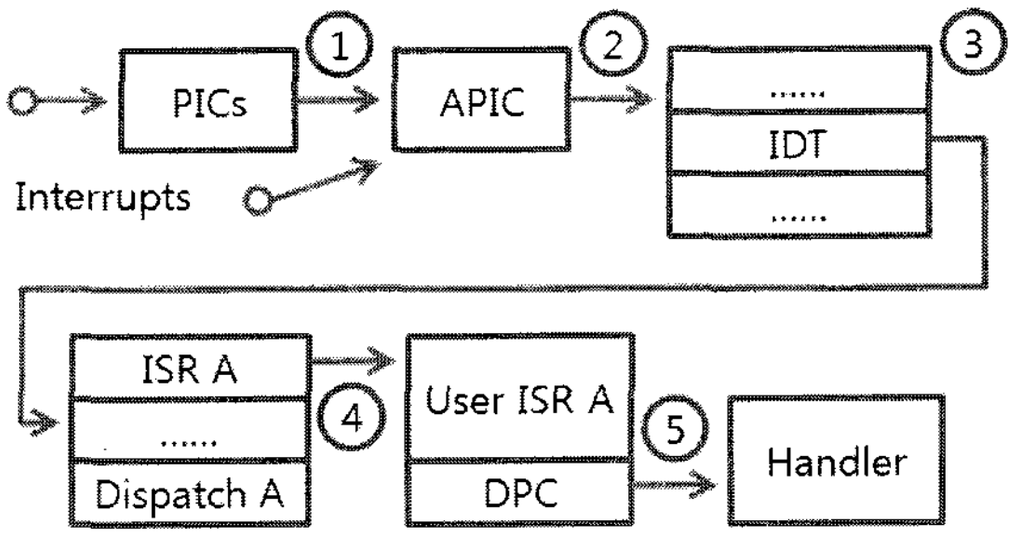
키보드 보안 기술은 근본적으로 키보드로부터 전달되는 키 값으로서의 스캔코드를 공격자보다 먼저 수집하고, 수집한 스캔코드를 공격자에게 공개되지 않도록 보호하는 데에 목적이 있다. 그러므로 플랫폼의 입장에서는 일반적으로 보안용 코드가 사용할 수 있는 기술은 공격용 코드에서 사용할 만한 기술이며 그 반대도 대개는 성립한다. 따라서 키보드 보안 기술을 구상하고자 하면 공격에 이용할 만한 기술을 생각해 볼 수 있다.

키보드로부터 입력된 패스워드가 응용 프로그램으로 전달되는 일련의 과정에서 공격자에게 탈취 당할 수 있는 다양한 포인트가 존재한다. 이러한 탈취 포인트를 크게 나누면 운영체제 커널 수준과 키보드컨트롤러 수준으로 구분할 수 있다.

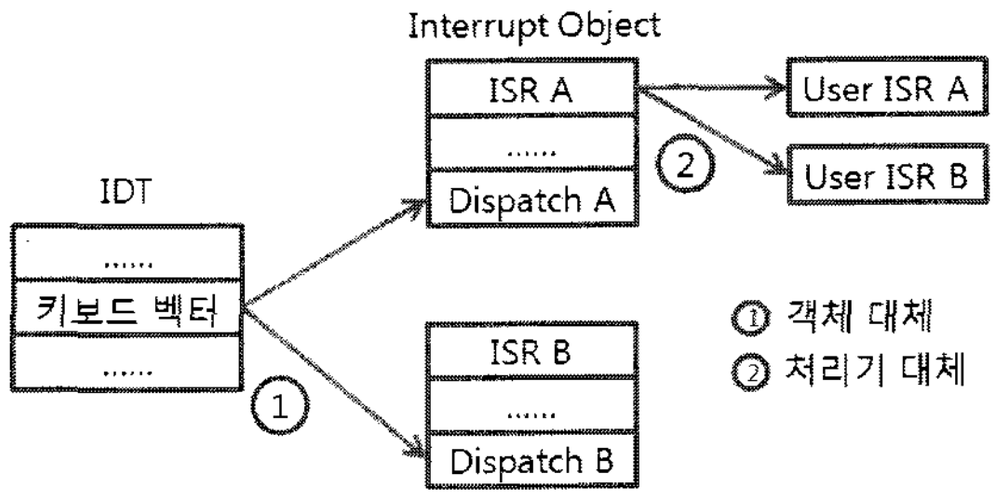
운영체제 커널 수준에서의 탈취와 관련하여서는, 그동안의 많은 선의적인 연구와 악의적인 공격을 경험함으로써 이러한 탈취 포인트와 이에 대한 공격 및 방어 방법들이 자주 노출되었다[3]. 여기서 사용하는 대개의 기술들은 인터럽트 및 예외처리 과정에서의 매우 상식적인 선점(preemption)을 노린 경우이며 탈취 포인트의 선택에 따라 사용자용 인터럽트 처리기를 대체하여 스캔코드를 얻는 방법, IDT[4] 내의 키보드 인터럽트 벡터에 존재하는 게이트 디스크립터[4]의 오프셋을 바꿔 운영체제의 인터럽트 객체를 대체하여 스캔코드를 얻는 방법 등이 포함된다. 이와 연관된 탈취 포인트를 [그림 1]에, 인터럽트 객체 또는 인터럽트 처리기를 대체하는 경우의 예를 [그림 2]에 보인다.

인터럽트 처리기를 대체하는 방법은 대개의 디바이스 드라이버에서 사용하고 있어 매우 일반적이다. 인터럽트 객체를 대체하는 방법에서는 키보드 인터럽트가 발생하면 운영체제에 등록된 인터럽트 처리기보다 교체된 인터럽트 처리기의 호출이 선행되어 키보드 스캔코드를 먼저 수집할 수 있다. 이러한 방법들은 선점 이력에 대한 검출이 아주 용이하므로 이를 방어할 수 있는 기술들이 이미 공개되어 공유되고 있어 현 시점에서 이를 이용하여 스캔코드 수집에서 선점할 수 있을 것으로 보이지는 않는다.

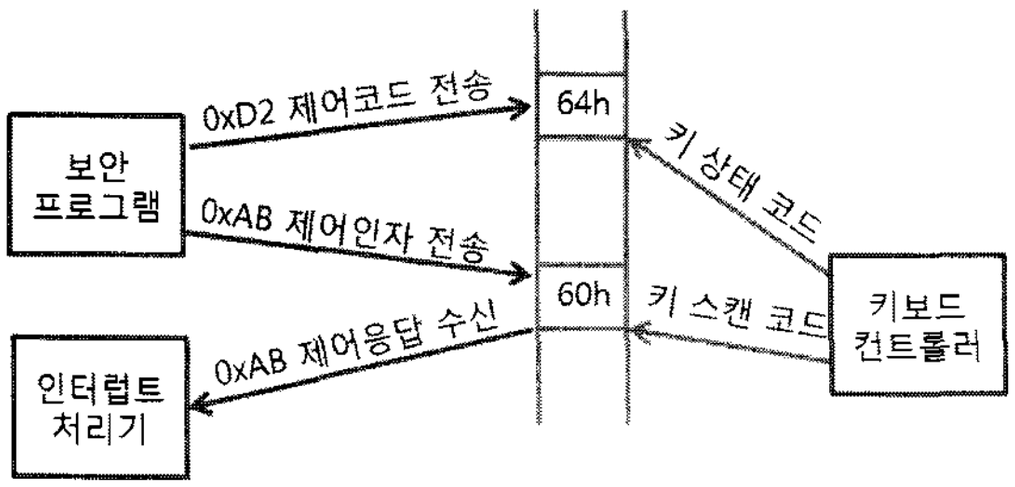
키보드컨트롤러 수준에서의 키보드 스니핑 프로그램



〔그림 1〕 운영체제 커널 수준에서의 탈취 포인트



〔그림 2〕 인터럽트 객체 및 처리기의 대체



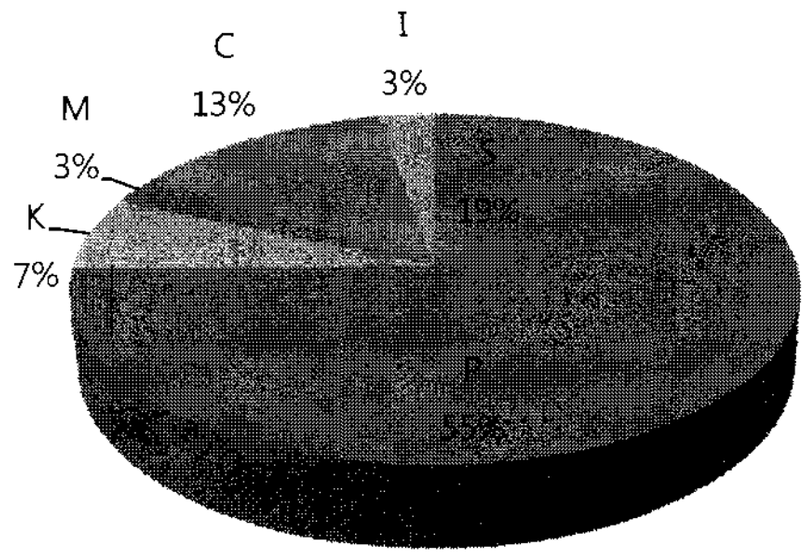
〔그림 3〕 임의의 스캔코드 전달 과정

이 사용할 수 있는 기술들을 고려하면 키보드컨트롤러의 입출력 포트에 직접 접근하여 스캔코드를 얻는 방법, 그리고 키보드컨트롤러의 제어코드를 활용하여 스캔코드를 교란시키는 방법 등이 있을 수 있다. 키보드컨트롤러의 입출력 포트에 직접 접근하는 방법은 운영체제가 스캔코드를 얻기 위하여 인터럽트 루틴의 역할을 빌리는 방식을 버리고 입출력 포트로부터 직접 스캔코드를 폴링하여 수집하는 방식으로서 많은 경우 방어자와 공격자가 경쟁상태(race condition)에 들어설 우려가 있으나 프로그래머가 하드웨어 제어 및 상호배제 메커니즘을 잘 활용할 경우 폴링에서 우선할 수 있다[5,6].

키보드컨트롤러의 제어코드를 이용한 동작방식은 보안 프로그램만이 사용할 수 있는 방어 방법이다. 이 방법에서는 공격자가 먼저 스캔코드를 가져가더라도 수집한 스캔코드가 사용자가 입력한 키로부터 만들어지는 스캔코드와 일치하지 않게 하기 위하여 의도한 시간에 의도한 데이터를 생성하도록 키보드컨트롤러의 가능한 제어코드를 활용하는 방법이다. [그림 3]은 이와 같은 제어코드의 활용 예를 보여주고 있다. 그림에서 0xD2는 뒤따르는 제어인자를 제어응답으로 반환하는 제어코드이며 이를 이용하면 임의의 스캔코드를 의도적으로 발생시킬 수 있다. 보안 프로그램이 상기의 방법을 이용할 경우, 키보드 입력 도중에 공격자에 의하여 완벽한 스니핑이 이루어지고 있다고 하더라도 보안 프로그램이 의도적으로 생성한 스캔코드의 값이 공격자가 스니핑하여 수집한 정보에 혼합된다.

Ⅲ. 키보드 보안의 근본적인 취약점

키보드 보안 문제가 심각해지기 시작할 무렵부터 키보드 보안 소프트웨어의 개발 및 보급이 급격히 증가하였다. 현재 30여개의 금융 및 기타 전자지불 서비스 기관에서 사용 중인 키보드 보안 프로그램은 M, K, C, S, P, I 등 여섯 종류가 존재한다. 다음 [그림 4]는 현재 20개사의 금융기관이 사용하는 키보드 보안 프로그램의 점유율 분포를 조사한 결과이다. 가장 많이 사용되는 프로그램으로 P 프로그램이 57%를 차지하고 있다. 상기에 분류한 키보드 보안 프로그램은 상용 프로그램이므로 내부 구현 방법을 확인할 수는 없으나 제2장에서 서술한 네 가지 또는 기타 다양한 변형 기술들을 고루 혼합하여 사용하고 있을 것으로 추정된다.



〔그림 4〕 키보드 보안 프로그램 통계

상기한 바와 같이 키보드 보안과 관련하여 다양한 단체에서 다양한 공격과 방어 경험을 통하여 키보드 스캔코드의 탈취를 방지하기 위한 기술이 보편화된 것으로 보인다. 따라서 많은 사람들이 키보드 보안이 상당한 기술적 발전을 이룬 것으로 알고 있으며 인터넷을 통한 금융거래 및 재화와 용역의 거래에서 패스워드 기반의 인증이 안전한 것으로 믿고 사용하고 있다. 그러나 현재 사용되고 있는 컴퓨터 하드웨어가 설계될 당시 정보보호에 대한 충분한 고려가 부재하였으며 이로부터 기인하는 취약점에 대해서는 지금까지의 해결방안이 이에 대응을 할 수 없다. 따라서 새로운 해결방안이 마련되기까지는 기존의 소프트웨어만의 힘을 빌려 안전한 정보교류를 보장받는 것은 쉽지 않을 것으로 판단된다.

특정 당사자 간에 보안이 요구되는 정보의 교환에 있어서 암호학적 기반에 의한 비화는 보안의 매우 중요한 요소이나 정보가 교환되는 행위 자체를 은폐하는 일이 무엇보다 중요하며 컴퓨터 하드웨어의 설계과정에서도 이는 반드시 고려해야 할 사항이다.

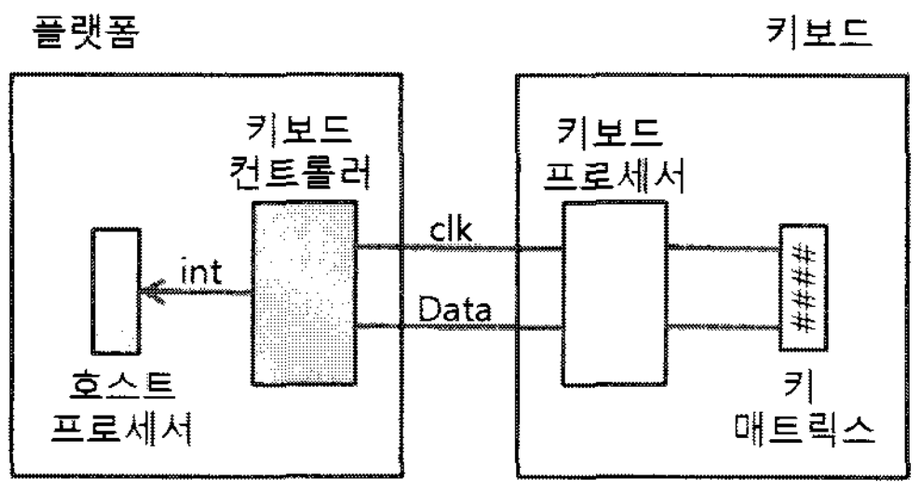
특히, 전달자 입장에서는 행위에 대한 로컬에서의 감시를 반드시 고려하여야 하며 이를 위해서는 전달되는 정보를 포함하여 전달행위에 대한 부가정보까지도 반드시 재사용 방지를 위한 휘발성(volatile) 도구를 이용할 것이 무엇보다 요구된다. 이러한 요구를 만족하지 못하는 설계의 경우는 로컬에서의 행위추적을 피할 수 없으

므로 많은 경우 보안 취약점을 제공하게 된다. 패스워드 인증에 있어서의 키보드의 역할행동을 살펴보면 비휘발성 도구 설계가 시스템의 보안에 얼마나 치명적인 취약점을 제공하는지를 확인할 수 있다.

현재의 PC 플랫폼에서 소프트웨어가 키보드 값 즉, 스캔코드를 수집하기 위해서는 스캔코드를 만들어 내는 키보드로부터 스캔코드를 수신하여 소프트웨어에게 전달하기 위한 키보드컨트롤러를 거쳐야 한다. 키보드컨트롤러는 [그림 5]와 같이 키보드와 호스트프로세서 사이에 연결되어 있으며 키보드 제어 및 호스트프로세서와의 통신을 위하여 내부에 [표 1]과 같은 레지스터를 가지고 있다. 상태레지스터 및 설정레지스터의 각 비트의 역할을 [표 2] 및 [표 3]에 나타낸다[7,8,9,10].

PC는 키보드컨트롤러 레지스터에의 접근뿐만 아니라 키보드컨트롤러로의 제어코드나 제어인자, 키보드로의 명령코드 등을 전달하거나 키보드컨트롤러로부터의 제어응답, 키보드로부터의 명령응답 및 스캔코드 등을 수수하기 위하여 읽기와 쓰기가 가능한 컨트롤 포트와 데이터 포트의 두 포트를 준비하여 각각 0x64 번지와 0x60번지에 할당하고 있다. 이 두 I/O 포트의 기능을 요약하여 정리하면 [표 4]와 같다.

제어코드는 키보드컨트롤러에 전달되고 해석되어 다양한 기능을 수행하며 추가적으로 인자를 요구하거나 응답을 제공하는 경우도 있다. 명령코드는 키보드컨트롤러를 통과하여 키보드로 전달되며 인자나 응답을 요구하기도 한다. 스캔코드는 키보드로부터 전달되어 키보드컨트롤러를 통과하여 키보드 문자를 생성[8]하는 데에 활용된다.



〔그림 5〕 키보드컨트롤러와 키보드 연결

〔표 1〕 키보드컨트롤러 레지스터

이름	동작	역할
상태레지스터	읽기	상태정보 판독
설정레지스터	읽기/쓰기	설정정보 저장

키보드컨트롤러는 호스트프로세서와 상기의 다양한 정보를 주고받기 위하여 두 포트를 대상으로 값을 읽고 쓰기 위한 버퍼로서 입력버퍼와 출력버퍼를 가지고 있다. 그러므로 소프트웨어는 입력버퍼를 통하여 키보드컨트롤러에게 정보를 써 넣고 출력버퍼를 통하여 키보드컨트롤러로부터 정보를 읽어낼 수 있다.

입력버퍼 및 출력버퍼는 호스트프로세서와 키보드컨트롤러 사이의 공유버퍼로서 이 공유버퍼를 통한 정보전달에서의 흐름제어를 위하여 키보드컨트롤러의 상태레지스터에는 [표 3]과 같이 OBF와 IBF 등의 플래그가 준비되어 있다. OBF와 IBF 비트는 각각 키보드의 출력버퍼 또는 입력버퍼에 데이터가 적재될 경우 1로 설정되며 해당 데이터를 읽어 낼 경우 0으로 지워진다. 단,

〔표 2〕 키보드 상태 레지스터의 구성

비트	필드명
Bit 0	OBF(Output Buffer Full)
Bit 1	IBF(Input Buffer Full)
Bit 2	System Flag
Bit 3	C/D(Control*/Data)
Bit 4	Inhibit Switch
Bit 5	Transmit Time-Out
Bit 6	Receive Time-Out
Bit 7	Parity Error

* IBM은 Command로 표기함.

〔표 3〕 키보드 설정 레지스터의 구성

비트	필드명
Bit 0	Enable keyboard interrupt
Bit 1	Enable mouse interrupt
Bit 2	System status flag
Bit 3	Unused
Bit 4	Enable keyboard
Bit 5	Enable mouse
Bit 6	Translation mode
Bit 7	Unused

〔표 4〕 키보드컨트롤러 포트의 역할

포트이름	읽기	쓰기	포트주소
컨트롤	상태레지스터	제어코드	0x64
데이터	스캔코드/ 제어응답/ 명령응답	명령코드/ 명령인자/ 제어인자	0x60

버퍼는 읽어 내더라도 기존에 적재된 정보가 그대로 남아 있어 이후에도 새로운 값이 적재되지 않는 한 재차 읽어낼 수 있다.

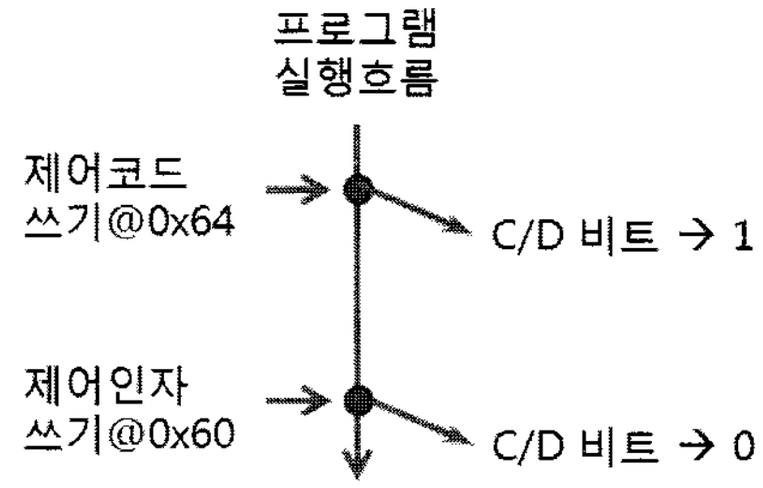
이 외에도 키보드컨트롤러의 상태레지스터에는 중요한 플래그들이 준비되어 있는데 이러한 비트들은 키보드컨트롤러가 설계될 당시 호스트프로세서와 키보드컨트롤러 사이의 인터페이스 자원을 절약하기 위하여 마련되었으며 상호 정보전송에서의 상태파악 등과 같은 흐름제어에 이용된다.

특히, 상태레지스터 내의 C/D 플래그는 키보드컨트롤러의 입장에서 매우 중요한 플래그로 동작한다. [표 2]에서 보인 바와 같이 호스트프로세서가 입력버퍼를 통하여 전달하는 정보의 용도는 다양하므로 키보드컨트롤러는 그 용도를 구분하기 위한 도구가 필요하며 이에 대한 최소한의 힌트로서 현재 입력된 정보가 어느 포트를 통하여 전달되었는지를 확인할 필요가 있다. C/D 비트는 이러한 정보를 반영하는 비트로, 호스트프로세서의 입장에서는 상호배제 메커니즘을 활용하면 이에 대한 정보를 제공받을 필요가 없다. 그러나 이것이 호스트프로세서에 공개되도록 설계됨으로써 해당 정보는 호스트프로세서 자신이 과거에 입력버퍼에 써 넣은 값에 대한 이력정보를 유추하는 데에 활용될 수 있다.

그 동안 보급된 키보드 보안 관련 소프트웨어들은 C/D 플래그의 이력정보 특성과 출력버퍼의 비휘발성 특성이 가지는 취약점을 고려하지 않고 설계되었다.[3] 이러한 취약점은 기존의 공격 및 대응 연구에 대비하여 전혀 새로운 취약점으로서 상기와 같은 플래그와 버퍼의 특성을 활용하고 기타 키보드컨트롤러의 기능을 정교히 제어하면 정보보호 측면에서 매우 치명적인 문제를 야기할 수 있다. 즉, 설정레지스터 및 제어코드를 이용하여 구현 가능한 다양한 방법 중의 하나로써 손쉽게 스캔코드를 탈취할 수 있으며 C/D 플래그의 상태를 추적함으로써 의도적으로 혼합된 스캔코드를 선별하여 제거하면 사용자의 입력 문자열을 정확히 확보할 수 있다.

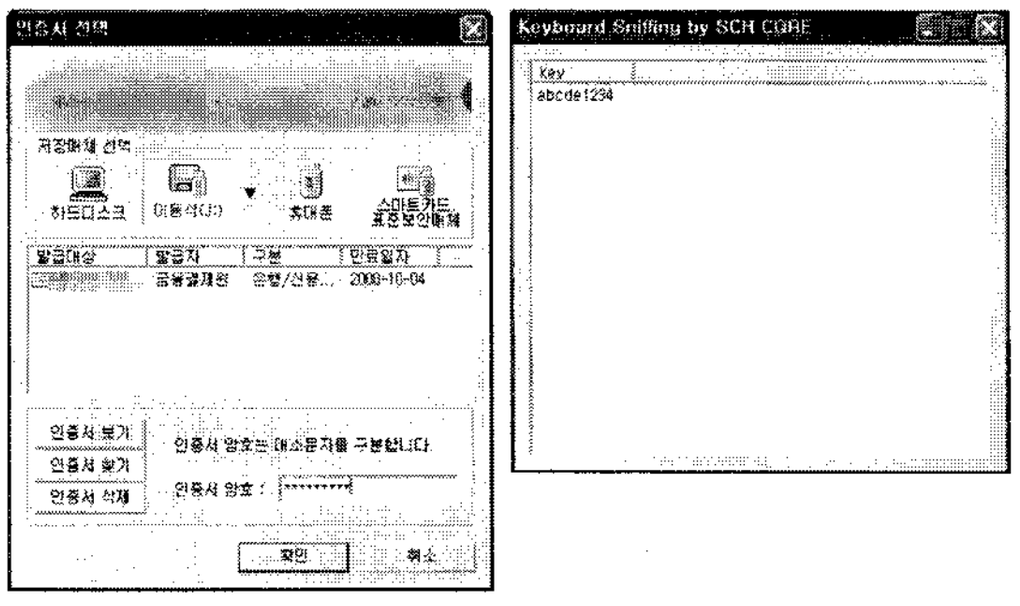
Ⅳ. 스니핑 프로그램의 실례

본 논문에서는 제3장에서 기술한 바와 같은 보안 취약점을 이용하여 실제의 스니핑 프로그램을 구성하고 그 결과를 분석하여 보았다. 스니핑 프로그램이 사용한 취약점은 C/D 비트로, 해당 비트는 [그림 6]과 같은 동작특성을 가지고 있으며 그 폴링 에지를 검출하면 제3장에서 서술한 스니핑이 가능하다. 즉, C/D 비트의 폴링에지가 검출된 상황에서 출력버퍼로부터 읽은 정보는 키보드가 보낸 스캔코드가 아닌 제어코드에 의하여 임의로 만들어진 정보이므로 이를 걸러냄으로써 실제의 키보드로부터 들어온 스캔코드만을 취할 수가 있다.



〔그림 6〕 제어코드와 C/D 비트와의 관계

〔그림 7〕 구현 프로그램 실행 환경 구성



〔그림 8〕 프로그램 실행결과의 일례

이를 이용하여 구현된 프로그램은 현재 인터넷을 통하여 서비스 중인 모든 키보드 보안 프로그램을 무력화하고 사용자가 입력한 모든 키에 대한 키보드의 스캔코드를 수집하는 것이 가능하였다. 구현 프로그램의 구성은 [그림 7]과 같으며 프로그램 구현의 세부사항은 본 논문에서는 생략하기로 한다.

또한 [그림 8]에는 프로그램 실행과 관련한 일례를 보였다. 실행 일례에 따른 오해의 여지가 있으나 본 논문과 관련하여 실험한 40여개 이상의 모든 사이트가 동일한 결과를 보였음을 밝힌다.

V. 결론 및 향후과제

본 논문에서는 키보드컨트롤러의 근본적인 하드웨어 취약점을 제시하고 이를 이용하여 실제의 스니핑 프로그램의 예를 구현하였다. 구현한 프로그램을 이용하여 현재 사용 중인 여섯 종류의 키보드 보안 프로그램이 실행 중인 상태에서 키보드로부터 입력되는 사용자의 패스워드를 모두 검출해 냄으로써 제시한 취약점이 매우 심각함을 보였다.

현재 금융 서비스를 제공하는 많은 기관에서 사용하는 소수의 키보드 보안 프로그램은 현재까지의 취약점에 대하여는 매우 훌륭하게 대응하고 있는 것으로 판단되며 다행히 본 논문에서 제시한 하드웨어 취약점에 대하여도 해당 보안 프로그램 개발자들은 스니핑을 회피하도록 다양한 제어코드를 혼용하여 활용하거나 키보드로 직접 전달되는 더미 명령코드를 활용하는 등의 방법을 모색함으로써 대응기술을 바로 시도할 것으로 기대된다. 다만, 본 연구에서도 이와 유사한 다양한 대응기술을 시도하고 있으나 발견된 하드웨어 취약점은 기존의 선점경쟁 형태의 취약점과는 달리 효과적인 소프트웨어적 해결책을 찾을 수 없었다. 따라서 보안 프로그램이 이에 효율적으로 대응하기 위해서는 당분간은 다양한 접근방법을 구상하여 시도해 볼 것이 요구된다.

물론 키보드 스니핑이 가능하다고 하더라도 심각한 사건으로 바로 이어지는 것은 아니다. 지역 공격을 위해서는 별도의 시스템 공격이 수반되어야 하고 원격 공격을 위해서는 네트워크를 건너가야 하므로 시스템의 관리와 네트워크의 감시를 철저히 한다면 사고를 방지할 수 있다. 다만, 논문에서 제시한 하드웨어의 근본적인 취약점이 당장 해결되지 않을 경우 문제의 하드웨어를 교체하거나 추가의 보안 하드웨어에 의한 대응 방안이 요구된다.

제3장에서 서술한 바와 같이 본 논문에서 제시한 취약점은 하드웨어 설계 당시 양자간 정보교환에서 비휘발성 정보를 통하여 흐름제어를 함으로써 발생하는 근본적인 취약점으로 볼 수 있다. 따라서 키보드뿐만 아니라 다양한 하드웨어에 동일한 특성의 취약성이 다수 존재할 것으로 사료된다. 다만, 현재까지 해당 하드웨어가 보안을 위한 자원으로 활용되고 있지 않거나 활용되고 있더라도 공격자가 발견하지 못하고 있을 뿐, 차후에는 심각한 문제로 떠오를 수 있다.

하드웨어 보안의 심각성은 하드웨어 자체의 취약성뿐만 아니라 현재의 운영체제 구조에 의하여도 확대된다. 현재 사용하고 있는 PC 플랫폼의 프로세서는 초창기로부터 20여 년 동안이나 하드웨어 자원에의 접근권한 제어를 위하여 4등급의 하드웨어 기반 특권수준을 제공하고 실행 프로세스마다 서로 다른 입출력 포트 접근권한을 부여할 수 있도록 설계 되었음에도 불구하고 운영체제는 여전히 전통적인 수퍼바이저/유저의 2등급 접근권한만을 구현하고 있다. 따라서 사용자 프로세스가 최고의 접근권한을 가지는 수퍼바이저 영역에서 자주 수행되나 취약점을 가진 하드웨어에 접근하는 것을 차단할 수 없으므로 보안 관리에 있어 매우 치명적인 허점을 제공한다.

궁극적으로, 운영체제는 PC 플랫폼의 프로세서가 제공하는 4등급의 특권수준을 통하여 보다 안전한 접근권한의 제어가 가능하도록 지원하여야 할 것이나 현재의 운영체제가 제공하는 접근권한 메커니즘 하에서 하드웨어 취약점에 대응하기 위한 최소한의 방법은 디바이스 드라이버의 구조에서 사용자의 코드를 사용자 권한에서만 수행되도록 변경하거나 디바이스 드라이버의 설치와 관련한 디바이스 드라이버의 인증에서 피인증자에 대한 관리를 철저히 하는 것이다.

참고문헌

[1] 최성욱, 김기태, “안전하고 신뢰성있는 전자상거래를 위한 키보드 입력 보안시스템의 설계 및 구현”, 한국정보처리학회 논문지, 제13-C권, 제1호, pp.55-62, 2006년 2월

[2] 정태영, 임강빈, “키보드 해킹에 대비한 이미지 기반의 새로운 패스워드 입력방식”, 순천향대학교 산업기술연구소 논문집 제13권 제1호, pp.75-80, 2007년 8월

[3] “키보드 해킹기법 및 대응기술 분석”, 금융 ISAC, 2005년 11월

[4] “Intel Architecture Software Developer’s Manual Vol.3 System Programming”, Intel Corporation, 1999

[5] Linda D. Paulson, “Key Snooping Technology Causes Controversy”, IEEE Computer, pp.27, Mar. 2002

[6] Daniel G. Treat, “Keyboard Encryption outlining ways to pad yourself with protection”,

IEEE Potential, p.40-42, Aug. 2002

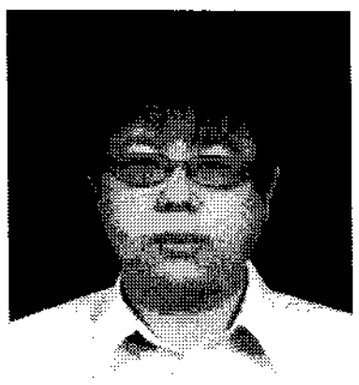
[7] Tom Shanley, ""ISA System Architecture", Mindshare Press, pp.407-414, 1993

[8] Michael Tischer, PC Intern : System Programming, pp.292, Abacus, 1995

[9] Frank V. Gilluwe, "The Undocumented PC", Addison Wesley, pp.261, 1994

[10] Sanchez, "IBM PC/AT Technical Reference Manual", IBM Corporation, 1985

〈著者紹介〉



배 광 진 (Kwangjin Bae) 학생회원

2005년 2월 : 순천향대학교 정보보호학과 졸업
2007년 2월 : 순천향대학교 정보보호학과 석사
2007년 3월~현재 : 순천향대학교 정보보호학과 박사과정
<관심분야> 시스템보안, 운영체제보안

임 강 빈 (Kangbin Yim) 종신회원

1992년 2월 : 아주대학교 전자공학과 졸업
1994년 2월 : 아주대학교 전자공학과 석사
2001년 2월 : 아주대학교 전자공학과 박사
1999년 3월~2000년 2월 : (미)아리조나주립대학교 연구원
2003년 3월~현재 : 순천향대학교 정보보호학과 교수
2005년 3월~현재 : 한국정보보호학회 이사
<관심분야> 시스템보안, 운영체제보안, 접근제어