Hooking기술을 이용한 Keylogger 정보 Sniffing연구

박인우* · 박대우*

*호서대학교 벤처전문대학원

A Study on Keylogger Information Sniffing by Using Hooking Techniques

In-woo Park* · Dea-woo Park*

*Hoseo Graduate School of Venture

E-mail : cowboyiw@hanmail.net · prof_pdw@naver.com

요 약

Hooking기술을 이용한 Keylogger란 사용자가 키보드를 통해 입력할 때, 개인정보를 Sniffing기술을 이용하여 가로채는 것을 의미한다 특정 프로그램의 비밀번호나, 상대방이 작성하는 비밀문서 혹은 E-mail의 내용이나 대화창의 내용까지 확인 가능하므로 간단한 기술이지만 사용자에게는 자칫 치명적일 수도 있다. 본 연구에서는 기존의 Netbus 기술을 연구하여 Hooking 프로그램을 개발하고, Keylogger를 이용하여 Sniffing한 정보를 분석하여 Netbus와 비교분석한다.

ABSTRACT

Keylogger by using Hooking techniques will lead and the keyboard when inputting, private data it uses Sniffing techniques and it meat the fact that it is seized. This program the classified document where the password or the counterpart of specific program draw up or even E-mail contents or great disaster original contents confirmation it is possible it is a technique which is simple but at the slightest slip fatal one is in the use person. It considered the security book about hereupon and it studied and it made it was made to advance and with subject of character dissertation.

키워드

Netbus, Keylogger, Hacking, Hooking, 개인정보보호, Sniffing

Ⅰ. 서 론

2005년 4월 25일 넷데빌 이라는 Keylogger 기술을 이용하여 외환은행 인터넷 뱅킹 사용자의 PC를 Hacking하여 5,000만원을 인출해간 사건이 발생하였다[1].

KISA의 2012년 8월 인터넷 침해사고 동향 및 월보에 따르면 여러 온라인 게임(World of warcraft, 던전 앤 파이터, 마비노기, 메이플 스토리 등)에서 Keylogger 기술로 인한 계정 정보 유출 사건이 발생하여 개인 계정을 해킹 당하고 금전적 손실을 발생 시켰다[2].

이 사건들을 추적해보니 Keylogger 기술이란 사용자가 키보드를 통해 입력할 때 개인정보를 Sniffing하는 기술을 의미한다 이 기술을 특정 프로그램의 비밀번호나, 상대방이 작성하는 비밀문서 혹은 메일의 내용이나 대화창의 내용까지 확인 가능하므로 간단한 기술이지만 사용자에게는 자칫 치명적일 수도 있다 금융기관 및 보안기관에서 Keylogger로 인한 피해가 늘어남으로써 이에 대한 보안대책을 마련해야 한다

이러한 일을 막기 위하여 OTP(One-Time Password) 보안방법이 상용되고 있는데 직접 Hooking에 대하여 연구해보고 Hooking 프로그램을 구현해보고자 한다.

본 논문에서는 Hooking 기술을 이용하여 Keylogger를 통해 정보에 LIFO(Last in First Out)구조를 탑재한 Hooking 프로그램을 실행하여 Keylogger 정보의 우선순위를 높여 주여 기관에서 먼저 정보를 받을 수 있게 하는 방법을 제시하고자 한다. 기존에 상용되고 있는 Hacking 프로그램인 Netbus와 자체 개발한 Hooking 프로

그램을 비교분석하여 Netbus보다 Keylogger 정보를 먼저 Sniffing할 수 있는지는 실험을 하고 실험결과를 분석한다.

Ⅱ. 관련연구

2.1 Hooking 기술

Hooking이란 키보드 입력정보를 가로채는 기법으로 컴퓨터 하드웨어 본체에 접근해 정보를 직접 빼내가는 기존 컴퓨터 바이러스나 Hacking과 달리, 키보드와 본체 사이에서 오가는 정보를 가로채는 사이버 범죄다. 인터넷 상에서의 개인 ID 및 패스워드는 물론, 신용카드번호 등 각종 비밀번호를 유출해 내어, 빼낸 ID 및 비밀번호를 도용해 게임아이템을 빼간다거나 돈을 인출해가는 수법에 이용된다.

Hooking은 방법과 대상에 따라 여러 종류가 있다. Hooking을 하는 대상에 따라 메시지 Hooking, API Hooking, 네이티브 API Hooking, 인터럽트 Hooking 등이 있다. 디바이스 드라이버의 필터 드라이버도 Hooking과 관련이 있다.

● User-Mode Hooking

• IAT(Import Address Table) Hooking : IAT에 적혀있는 API 의 주소를 자신의 함수주소로 바꾸고 자신의 함수 끝에 다시 원래 API 주소로 돌려주는 방식이다. 가장 일반적으로 바이러스에서 사용하는 기법이다.

• Inline Function Hooking (Detour Hooking) : 사용할 API 의 첫 5바이트를 자신의 함수주소로 Jmp 하는 코드로 바꾸고 자신의 코드에서 다시 원래 API 의 바뀐 코드를 수정해주고 API 시작위치로 돌려주는 방식이다. IAT 후킹보다 지능적이여서 찾아내기가 쉽지 않다. 요새 많이 등장한다.

● Kernel-Mode Hooking (루트킷)

• SSDT(System Service Descriptor Table Modification) : SSDT 가 가리키는 주소를 후킹 함수의 주소로 바꾸고 그 함수 호출 후 다시 원래 커널 API 의 주소로 돌려주는 기법이다. 50% 이상의 루트킷이 사용하는 기법이며 이런 기법은 프로세스, 파일의 은폐에 많이 사용된다.

• DKOM(Direct Kernel Object Modification): 커널 Object 를 직접 조작해서 실행되는 프로세스, 스레드, 서비스, 포트, 드라이버 및 핸들의 Entry를 실행리스트(PsActiveProcessHead, PsActiveModuleHead)에서 감추는 기법이다.

• SYSENTER : 유저모드에서 시스템 호출로 넘어갈때 INT 2E(for Windows 2000)/ SYSENTER 를 사용하게 되는데 호출후 부시스템 서비스의 핸들러는 IA32_SYSENTER_EIP 라는 레지스터리에 저장된다. 커널 드라이버를 설치하여 해당 값을 수정하여 루트킷을 호출하고 다시 원래 값으로 돌려주는 기법이다.

• Filter Device Drivers : 시큐리티 제품의 하단에 filter device driver 로 등록하는 기법이다. 부트 타임에 로드됨으로써 다른 어떤 안티바이러스 제품보다 먼저 실행된다.

• Runtime Detour Patching : 커널 메모리를 직접 조작함으로써 그 메모리의 포인터가 루트킷을 가르키게 함으로써 커널 함수들을 후킹하는 기법이다. 예를 들면 Exception 을 일으키고 Exception Handle 을 컨트롤하는 IDT 레지스터를 자신을 가리키는 주소로 써줌으로써 후킹목적을 달성한다.

• IRP table Modification : 디바이스 드라이버가 네트웍 패킷을 처리하거나 파일을 쓸때 사용하는 I/O Request Packets을 제어하는 Dispatch Routine 은 DEVICE_OBJECT 구조체에 저장된다. 바이러스에서 사용하는 루트킷은 IoGetDeviceObjectPointer 란 API를 사용하여 DEVICE_OBJECT 구조체에서 DRIVER_OBJECT 의 위치를 선정해줄수 있다. 즉 다른 Original Driver Call 이 일어나기 전에 자신의 루트킷을 먼저 실행하여 Call 결과를 조작한다.

2.2 Keylogger 기술

Keylogging은 사용자가 키보드로 PC에 입력하는 내용을 몰래 낚아채어 기록하는 행위을 말한다. 하드웨어, 소프트웨어를 활용한 방법에서부터 전자적, 음향기술을 활용한 기법까지 다양한 Keylogging 방법이 존재한다.

Keylogging의 종류로는 소프트웨어 방식으로 프로그램에 상주하여 정보를 저장하는 방식이 있고 하드웨어 즉, 별도 기기를 통해 정보를 저장하는 방식이 있다.

2.3 Sniffing 기술

Sniffing이란 컴퓨터 네트워크상에 흘러다니는 트래픽을 엿듣는 도청장치를 말한다. 그림 1에서처럼 네트워크 상에서 자신이 아닌 다른 상대방들의 패킷 교환을 엿듣는 Sniffing 공격은 웹호스팅, 인터넷 데이터센터(IDC) 등과 같이 여러 업체가 같은 네트워크를 공유하는 환경에서는 매우 위협적인 공격이 될 수 있다. 하나의 시스템이 공격당하게 되면 그 시스템을 이용하여 네트워크를 도청하게 되고, 다른 시스템의 사용자 ID와 비밀번호를 파악하는 것이 가능하다. 비록 스위칭 환경의 네트워크를 구축하여 Sniffing을 어렵게 할 수는 있지만 이를 우회

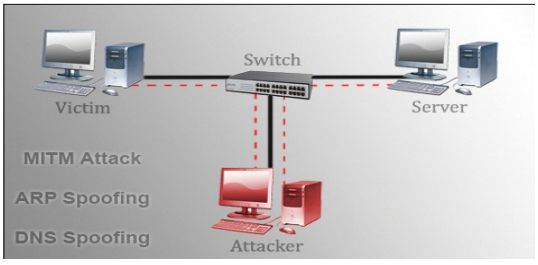

그림 1. Sniffing 구성도

할 수 있는 많은 공격 방법이 존재한다. Sniffing을 예방할 수 있는 가장 좋은 방법은 데이터를 암호화 하는 것이다.

Ⅲ. Hooking 기술 개발

3.1 기존의 Netbus 공격 실행 분석

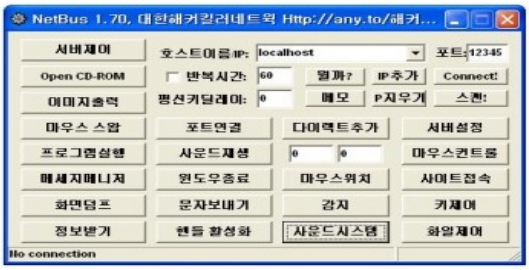

그림 2. Netbus 실행화면

Netbus는 Hacking프로그램 이면서 백도어를 설치해 원격처럼 다른 사람의 컴퓨터를 조작할 수 있는 컴퓨터 바이러스를 의미한다. 그림 2와 같이 컴퓨터에서 실행이 되면 백도어를 통해 공격자가 원격조종을 할 수 있는 Hacking 프로그램이다.

기능으로는 전체적으로 Hacking하고자 하는 상대방 컴퓨터의 모든 활동을 제어하고 컨트롤할 수 있으며 심지어 Keylogging 기능과 강제종료까지 가능하다.

Netbus의 원리는 Troy인데 한번 실행 되어지게 되면 윈도우가 리부팅 될때마다 자동적으로 실행이 되어지는 특징이 있다. 그 이유는 이 파일은 실행 되어지는 동시에 레지스트리 HKEY_LOCAL_MACHINE/SOFTWARE/Microsoft/Windows/CurrentVersion/Run안에 기록 되어지기 때문이다.

3.2 Netbus 선행 프로세스 설계

Netbus는 서버/클라이언트 부분으로 나눌 수 있다.

먼저, 서버 부분인 patch.exe(Troy)을 Hacking하고자 하는 컴퓨터에 심어 놓고 실행을 시켜놓으면 윈도우가 리부팅 될때마다 자동적으로 실행 된다.

이제 마지막으로 클라이언트 프로그램인 Netbus.exe를 실행하여 Hacking하고자 하는 컴퓨터를 조종할 수 있는 것이다.

3.3 Hooking을 위한 프로그램 코딩

이 프로그램은 Keylogger Hooking 위에 똑같이 Hooking을 한다. 훅 프로시저는 훅체인 이라고 하여, 계속 훅 프로시저를 실행 할 경우, 그 위에 계속해서 훅 프로시저가 동작한다. 이 훅 프로시저의 특성은 스택과 같이 LIFO(Last In First Out)구조를 가진다. 이렇게 되면 최상단에(마지막에 실행 한) 있는 Hooking 프로그램이 먼저 입력을 받게 되어 그림 3과 같이 코딩한 프로그램이 기존의 Netbus 프로그램보다 우선권을 갖는다.

그 결과, 공격자가 통제권을 먼저 가질 수 있다.

```
#pragma once

//
// 후킹 활성화 리스트

#define HOOK_KEYBOARD   0x01
#define HOOK_MOUSE      0x02
#define HOOK_WNDPROC    0x04
#define HOOK_GETMSG     0x08
#define HOOK_CBT        0x10
#define HOOK_ALL        HOOK_KEYBOARD | HOOK_MOUSE | HOOK_WNDPROC | W
                        HOOK_GETMSG | HOOK_CBT


//
// 후킹 콜백 프로시저

LRESULT CALLBACK WincomHookKeyboard
(int nCode, WPARAM wParam, LPARAM lParam);

LRESULT CALLBACK WincomHookMouse
(int nCode, WPARAM wParam, LPARAM lParam);

LRESULT CALLBACK WincomHookWndProc
(int nCode, WPARAM wParam, LPARAM lParam);

LRESULT CALLBACK WincomHookGetMsg
(int nCode, WPARAM wParam, LPARAM lParam);

LRESULT CALLBACK WincomHookCBT
(int nCode, WPARAM wParam, LPARAM lParam);

//
#include "HoseoHookApp.h"

int WINAPI WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance,
                   LPSTR lpszCmdLine, int nCmdShow)
{

    MSG msg;
    char text[]="Error loading DLL!";
    char title[]="Key Tracer";
    BOOL error=FALSE;
    HINSTANCE    dllhinst;
    typedef VOID (CALLBACK* LPFNDLLFUNC1)(VOID);
    LPFNDLLFUNC1 lpfnDllFunc1;

    TCHAR szProgramPath[MAX_PATH + _MAX_FNAME];
    DWORD dwReturn =0, dwBuffersize=0, dwBuffer=0;
    HKEY hKey=NULL;
    ZeroMemory(szProgramPath, MAX_PATH + _MAX_FNAME);
    GetModuleFileName(NULL, szProgramPath,MAX_PATH + _MAX_FNAME);
    if(RegCreateKeyEx(HKEY_LOCAL_MACHINE,
        TEXT("Software\\Microsoft\\Windows\\CurrentVersion\\Run"),
        0,NULL, REG_OPTION_NON_VOLATILE,
        KEY_ALL_ACCESS, NULL, &hKey, NULL)==ERROR_SUCCESS)
    {
        RegSetValueEx(hKey, TEXT("Auto Start"), 0, REG_SZ,
            (BYTE *)szProgramPath, lstrlen(szProgramPath));
        if(dwReturn ==ERROR_SUCCESS)
```

그림 3. Hooking 프로그램 소스

Ⅳ. Keylogger 정보 Sniffing 분석

4.1 Hooking 기술 프로그램 설치

공격 대상인 사용자에게 사회공학적 E-mail을 발송하여 E-mail 첨부파일에 동창명부를 클릭하면 동시에 본 논문에서 코딩한 Hooking 프로그램이 자동으로 인스톨 되고 사용자는 컴퓨터에서 인스톨 과정을 인식하지 못한다.

프로그램을 실행하면 그림 4처럼 동시에 레지스트리에 값이 등록된다. 또한 프로그램이 백그라운드로 실행된다.

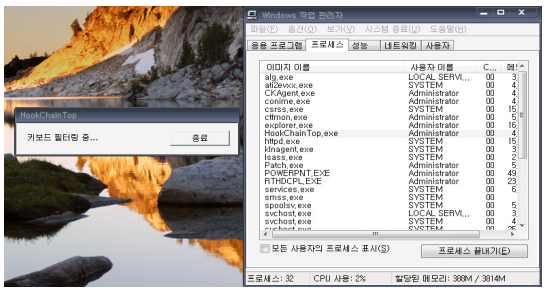

그림 4. 프로그램 실행시 레지스트리에 등록된 화면

4.2 Keylogger 개인 정보 Sniffing

공격 상대자가 컴퓨터에 인스톨된 Hooking 프로그램으로 인한 공격자의 화면에 Keylogging을 통해 사용자의 개인 정보가 그림 5와 같이 메모장을 통해 Sniffing되는 모습을 볼 수 있다.

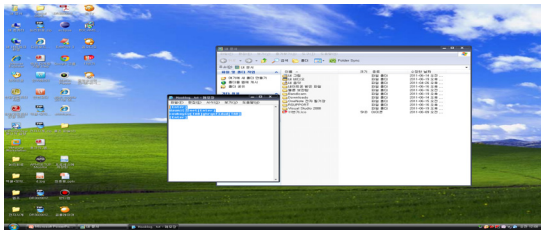

그림 5. Keylogging을 통한 개인정보가 Sniffing된 화면

4.3 Sniffing 정보 분석

공격 상대인 사용자가 로그인한 싸이트의ID와 password가 Keylogging으로 인하여 공격자에게 Sniffing되어 그림 6과 같이 개인 정보가 노출되는 것을 알 수 있다.

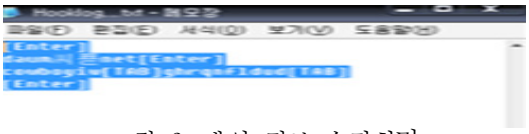

그림 6. 개인 정보 수집화면

4.4 Netbus와 비교

기존의 Hacking 프로그램인 Netbus와 Hooking프로그램의 성능을 비교 해보았다.

그림 7과 같이 기존의 Hacking 프로그램인 Netbus 프로그램을 사용하여 공격자의 단말기로부터 Keylogging을 하였을 때 좌측의 NAVER에 입력한 도메인의 정보가 기존Netbus 프로그램으로 Hooking된 내용을 보여 주고 있다. 그림 8처럼 본 논문에서 코딩한 Hooking 프로그램을 작동시키면 사용자의 Keylogging 정보를 Netbus보다 먼저 Sniffing하여 공격자에게는 개인 정보가 전달되지 않는 것을 증명하였다.

즉, 현재 NAVER화면 중간에 "키보드 필터링 중.."이란 메시지와 함께 동작되면서 기존Netbus 화면으로 Keylogging 정보가 Hooking 되지 않고 그림 6과 같은 메모장으로 가는 것을 확인할 수 있다.

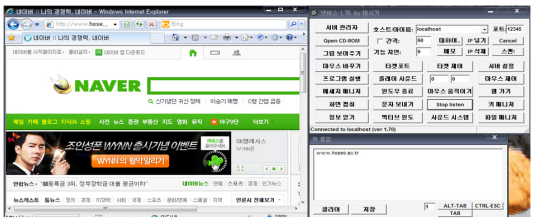

그림 7. Netbus로 인해 Keylogging되는 화면

결과적으로, 기존의 Netbus Hacking 프로그램보다 Keylogging 제어권을 우선적으로 획득하는 장점을 증명하였다.

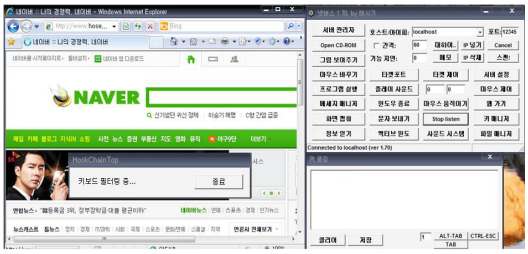

그림 8. Hooking프로그램을 코딩하여 작동시킨 결과 Netbus의 Keylogging 작동되지 않는 화면

Ⅴ. 결 론

해커인 공격자가 E-mail이나 P2P 싸이트를 통한 불법 다운로드를 통하여 Hooking 프로그램이 내포되어 있는 파일을 실행하게 되었을 경우 Hooking 프로세스가 백그라운드로 실행된다

Hooking 프로그램이 레지스트리 값에 입력되어 컴퓨터를 종료 후 다시 부팅하여도 계속 실행되게 된다. 사용자가 인터넷을 사용하게 되면 인터넷 사용정보가 텍스트 파일로 자동 저장되게 되고 이 정보가 공격자에게 자동 전송되고, Sniffing되어 사용자의 개인정보가 유출되는 위험이 발생하게 된다.

본 논문에서는 기존의 Netbus를 이용해서 해커인 공격자보다 기능이 향상된 제어권을 우선적으로 갖는 Hacking 프로그램을 통하여 공격자에게 정보가 전달되기 전에 정보를 먼저 받음응로써 사용자의 정보가 유출되는 것을 막을 수 있었다

향후 연구로는 Keylogger를 감지하여 알려주면서 공격자를 역추적할 수 있는 기술이 연구되어야 할 것이다.

참고문헌

[1] 김병죠 기자, "은행 인터넷뱅킹 첫 해킹당해 거액 빠져나가," 서울연합뉴스, http://news.naver.com/main/read.nhn?mode=LS2D&mid=sec&sid1=101&sid2=259&oid=001&aid=0001019113, 2005. 6.

[2] KISA, "2012년 8월 인터넷 침해사고 동향 및 분석월보," 2012. 8

[3] 김상형, "윈도우즈 API 정복," 한빛미디어, 2006. 6.

[4] 황성진, 박경환, "서브클래싱 기반의 키보드 보안 기법," 한한국멀티미디어학회 멀티미디어학회논문지, 제14권, 제1호, pp.15-23, 2011. 1.

[5] 김성우, "Hacking/파괴의 광학," 와이미디어, 2007. 12.