Winter Research Proposal

Preventing User Input Leaks Caused by Rootkits

Christopher Wood

Faculty Mentor: Professor R. K. Raj

Department of Computer Science, Rochester Institute of Technology

October 26, 2009

Abstract

A rootkit is a set of small programs used by attackers to gain permanent, undetectable access to the root of the target system. They are typically deployed in the kernel of the host operating system to achieve stealth and provide future functionality for the attacker. These two main objectives are usually accomplished by modifying the kernel memory and objects or host OS using hooking techniques, Direct Kernel Object Manipulation (DKOM), hardware manipulation, and covert channels. A rootkit that can successfully employ these techniques will not only be nearly invisible to the system administers, but if implemented correctly it can completely control the machine and host operating system as well.

The most realistic and useful rootkits are those that utilize covert channels for remote access and control. They are much more of a security threat because they open unwanted channels, or “backdoors”, so that the attacker may access the machine at will. Specifically, covert channels that are used to retrieve user input from an infected machine are a considerable security threat to both the target machine and the user whose information is being leaked. It is my goal in this project to research and develop both a local and networked device driver that can be used to detect a rootkit attempting to steal user input and potentially close the door to the open connection.

Problem:

One robust and reliable approach for an attacker to retrieve sensitive information from a Microsoft Windows PC is to implement a layered driver for a human interface device (HID). With the ability to intercept, modify, and analyze I/O requests between the hardware device and the software sending the request, layered HID drivers can do whatever they want with user input. Rootkits implemented as layered HID drivers for this purpose typically intercept information from the I/O Request Packet (IRP) that carries keyboard information to and from the kernel subsystem. This information can then be saved elsewhere on the local machine for future exportation or sent over a covert channel to the attacker in real time. "Backdoors" used by the attacker to send and receive data from the rootkit poses an enormous threat to the safety and security of the infected machine. Not only is the user's input data at risk of being used maliciously, but the machine itself is open to future attacks through the same backdoors used by the attacker. When the target machine is connected to a network, the other networked computers are also at risk. In this project, I will attempt to prevent user input leaks by monitoring for the behavior exhibited such rootkits. Specifically, I will try to match data being read in from the keyboard to data being saved to a file on the local machine's hard disk or being sent over a network. If patterns begin to emerge between the user input and data transfers, I will continue to analyze the circumstances causing such transfers to more successfully identify any malicious software on the machine. Finally, if the rootkit utilizes covert channels to send and receive data, I will try to pinpoint the open socket and close it to prevent future leaks.

Involvement:

This project expands upon earlier research on rootkits that I conducted this past summer [1]. I will now focus on networked and remote controlled rootkits. Under the guidance and supervision of Dr. Raj, I will implement the necessary deliverables for the project, which include a local and networked device driver, a remote controlled rootkit, and a simple user interface to control the device drivers. Every deliverable will target Microsoft Windows NT operating systems. Therefore, the device drivers and rootkit will be implemented in the C and C++ programming languages using the Windows Driver Model (WDM) and Winsock Kernel (WSK) Network Programming Interface (NPI) and the user interface will be implemented using the standard Win32 libraries. Developing the rootkit will be the most challenging task of the entire project so I will use the resources available on www.rootkit.com and the texts *Rootkits: Subverting the Windows Kernel* [2] and *Exploiting Software: How to Break Code* [3] for examples and further support. In parallel to these four deliverables, I will also work with Dr. Raj to write a paper that covers my research and findings that can be submitted to an appropriate regional or national conference. I plan on devoting approximately two to three weeks to each deliverable and spending two hours per day during the week. This should give me enough time to adequately cover the problem and implement working deliverables in the ten-week period.

Impact:

My work is an attempt to integrate a form of security designed to prevent user input leaks directly into the existing operating system. Typical anti- virus and malware programs are installed into the user mode of a computer and can access the kernel mode of the host operating system only up to a certain extent. By implementing a device driver that is fully trusted by the underlying kernel it will not be limited to the memory, functions, and objects it can access. This helps when attempting to monitor for system-wide malicious behavior because the only obstacle to overcome is algorithm efficiency and optimization. If the device drivers to be implemented are both successful and efficient then they could be the gateway to kernel-level security to protect against user input leaks. Also, this project will give me more knowledge and experience with the Windows operating system, device driver development, networking, and the Win32 library. Since I am aspiring to enter the field of low-level development and computer security in the future these skills will prove to be extremely useful throughout the course of my education and job hunt.

Timeline:

This project consists of five major deliverables: the local and networked device drivers, the remote controlled rootkit, the user interface for the device drivers, and the final paper. Since this project is a continuation of previous work I will start with the local device driver that is nearly complete; all that remains are the final portions of my detection algorithm and thorough testing. Upon completion of that driver I will begin to implement my own rootkit which focuses primarily on using covert channels as a medium for control. Since implementing the rootkit will be the most challenging part of the project I plan to spend at least three weeks to implement it correctly. Once that rootkit is working properly, I will move on to adding networking capabilities to my localized driver and test it with the rootkit implemented in the second stage. Depending on the amount of time remaining and the correctness of the previous three deliverables I will focus my remaining time on implementing the user interface for both the local and networked device drivers. I will complete the paper in parallel with the development of the other four deliverables so that it flows naturally with the progression of the project and covers it adequately.

References:

[1] C. Wood, *Layered Driver Rootkit Detection on Microsoft Windows PCs,* Poster Presentation, RIT Undergraduate Symposium, August 2009.

[2] J. Butler and G. Hoglund, *Rootkits: Subverting the Windows Kernel,* Addison-Wesley, 2006.

[2] G. Hoglund and G. McGraw, *Exploiting Software: How to Break Code,* Addison-Wesley, 2004.