Keystroke Logging (Keylogging)

Tom Olzak
April 2008

Introduction

Cybercriminals have devised many methods to obtain sensitive information from your endpoint devices. However, few of them are as effective as keystroke logging. Keystroke logging, also known as keylogging, is the capture of typed characters. The data captured can include document content, passwords, user ID's, and other potentially sensitive bits of information. Using this approach, an attacker can obtain valuable data without cracking into a hardened database or file server.

Keylogging presents a special challenge to security managers. Unlike traditional worms and viruses, certain types of keyloggers are all but impossible to detect.

In this paper, I examine how keyloggers work. I look at the various types of keyloggers and how they differ. Finally, I explore ways to prevent keylogging and how to respond if a keylogger is discovered.

Before jumping into the mysteries of keylogging, we should understand how keyboards work and how they interface with systems. The next section is a review of keyboard operation. You can skip it if you understand keyboard technology.

How Keyboards Work

A keyboard consists of a matrix of circuits overlaid with keys. This matrix of circuits, known as a key matrix, can differ between keyboard manufacturers. See Figure 1. However, the key codes that are sent through the keyboard interface to a specific operating system are always the same.

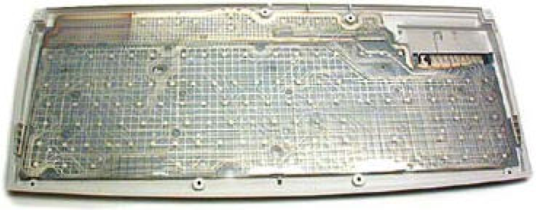


Figure 1: Key Matrix
(Wilson and Tyson, 2008)

Let's step through Figure 2 to trace the path between keystrokes and operating system (OS) or application.

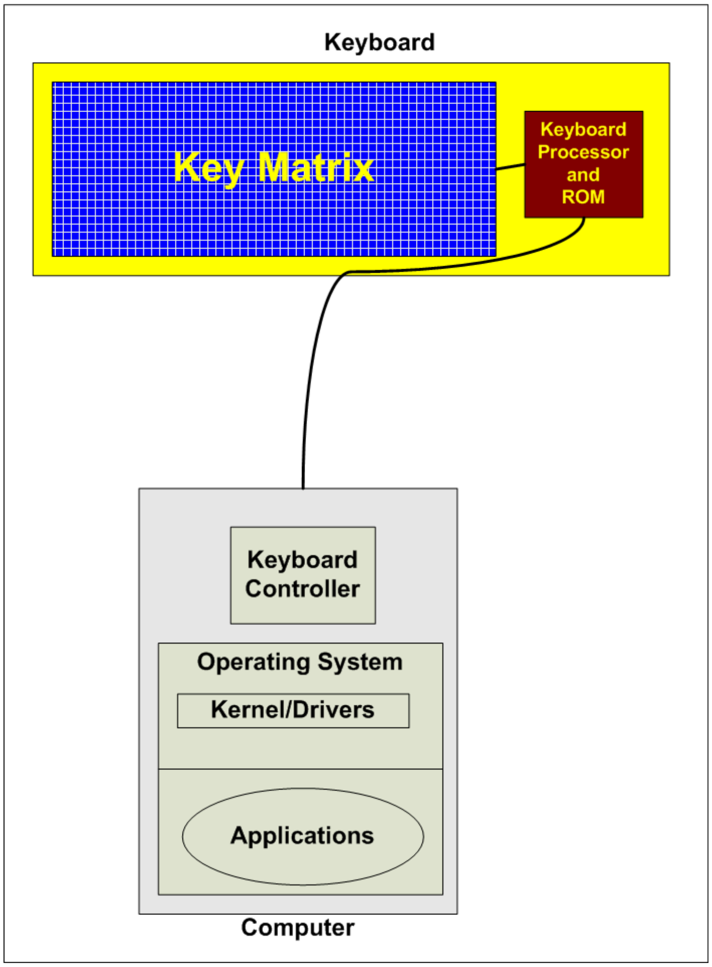


Figure 2: Keyboard/PC Layout

When the user presses a key, a circuit closes in the Key Matrix. The Keyboard Processor detects this event and captures the circuit location. Using a table stored in keyboard ROM, the processor translates the circuit location to a character or control code. Control codes are typically CTRL- or ALT- combinations.

The keyboard's memory buffer temporarily stores the translated character or control code and then sends it to the computer's keyboard interface. The computer's keyboard controller receives the incoming keyboard data and forwards it to the operating system. A keyboard driver is typically used to manage this part of the process. The operating system processes the keyboard input based on the current state of the OS and applications.

A keyboard interfaces with a computer via either a cable or a wireless connection. Common cable connections include the old PS2 standard and today's more common USB connector.

A popular wireless connection uses a 27 MHz signal with a range of about six feet. This type of connection is found in Microsoft and Logitech wireless keyboards. For solutions that require greater range, more robust wireless connections are available. These long range connections can reach about 100. One example is the wireless USB RF keyboard from Fentek Industries, Inc (http://www.fentek-ind.com/rf-wireless-keyboard.htm#kbrftb100).

With this high-level understanding of keyboard operation, let's move to a general discussion of how keystroke loggers work.

How Keyloggers Work

Keyloggers are hardware or software tools that capture characters sent from the keyboard to an attached computer. They have both lawful/ethical and unlawful/unethical applications. Lawful applications include:

- Quality assurance testers analyzing sources of system errors;
- Developers and analysts studying user interaction with systems;
- Employee monitoring; and
- Law enforcement or private investigators looking for evidence of an ongoing crime or inappropriate behavior.

On the other side of the line between lawful and unlawful use, cybercriminals use keylogging technology to capture identities, confidential intellectual property, passwords, and any other marketable information.

Keyloggers fall into four categories: software, hardware, wireless intercept, and acoustic. Although they differ in how they are implemented and how information is captured, these four keystroke logging technologies have one thing in common. They store capture information in a log file. When software or hardware keyloggers are used, the log files are stored on the compromised machine. Remote capture technologies (i.e., wireless intercept and acoustic) typically store keystroke data on the collection device.

Software Keyloggers

Software keyloggers capture keystroke information as it passes between the computer keyboard interface and the OS. They are implemented as traditional applications or kernel-based. In almost all malicious instances of this type of keylogger, users participated in some way in the software's installation.

Keylogging applications use a hooking mechanism (e.g., SetWindowsHookEx()) to capture keyboard data. Vendors often package solutions, like Perfect Keylogger, as an executable or a DLL (Shetty, 2005).

Most kernel-based keyloggers are replacement keyboard device drivers. A portion of the logger resides in the OS kernel and receives data directly from the keyboard interface. It replaces the kernel component that interprets keystrokes (Shetty, 2005). The red area in Figure 3 shows the location of a kernel-based keylogger in the keystroke-to-OS path.

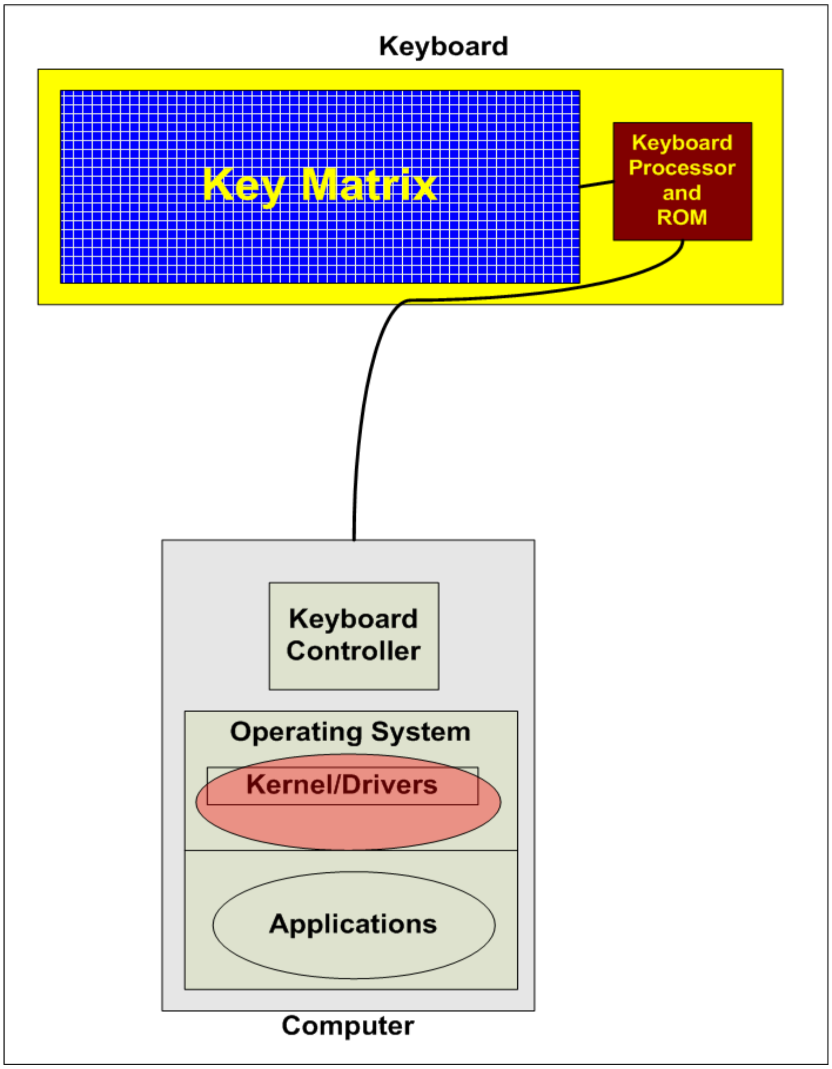


Figure 3: Kernel-based Keylogger

Both types of software keyloggers intercept keyboard data, write a copy to a local—often encrypted—log file, and then forward the information to the operating system. To the unsuspecting user, everything looks normal.

Anti-malware, personal firewall, and host-based intrusion prevention (HIPS) solutions detect and remove application keyloggers. Kernel-based solutions are not so easy to find, although prevention controls like HIPS can prevent their implementation.

Installed as part of a rootkit package, kernel-level loggers elude anti-malware detection. Further, they don't show up in the list of running processes. Only rootkit detection software (e.g., RootkitRevealer) can detect, report, and help remove them. However, rootkits once installed are almost impossible to eradicate. If detected, remediate with a complete reinstall of the infected system.

Other detection methods include:

- Scan local drives for log.txt or other log file names associated with known keyloggers;
- Implement solutions that detect unauthorized file transfers via FTP or other protocols;
- Scan content sent via email or other authorized means looking for sensitive information;
- Detect encrypted files transmitted to questionable destinations.

Software keyloggers can be detected using software tools. For this reason, users of keyloggers often prefer hardware solutions.

Hardware Keyloggers

A hardware keylogger is essentially a circuit located somewhere between the keyboard and the computer (en.wikipedia.org/wiki/Hardware_keylogger). Devices placed inline with the keyboard cable are the most popular means of deployment. Figure 4 shows two variations of PS/2 keylogger and Figure 5 a USB type.

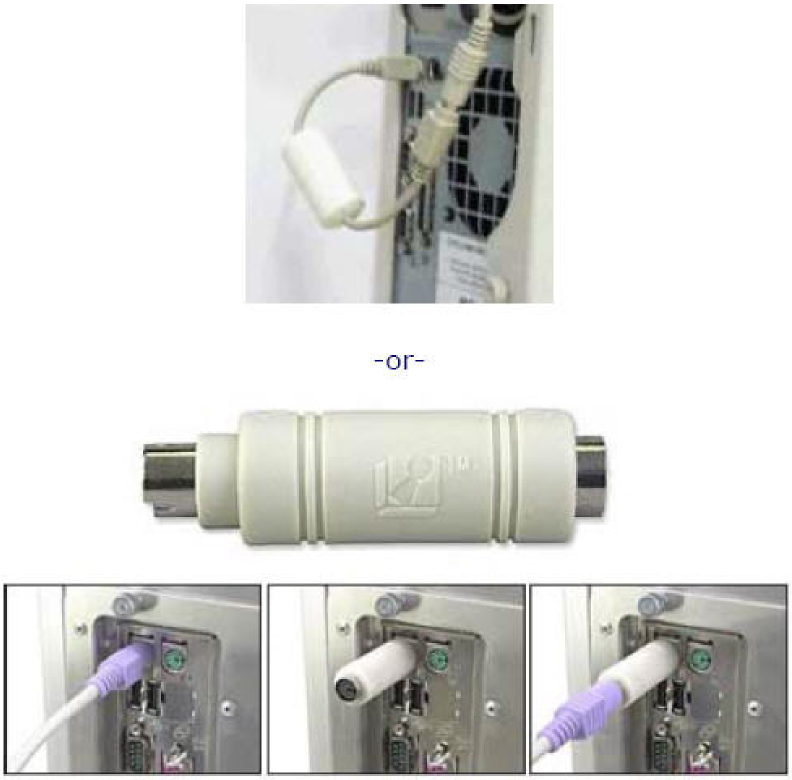


Figure 4: PS2 Keyloggers
(SpyCop)

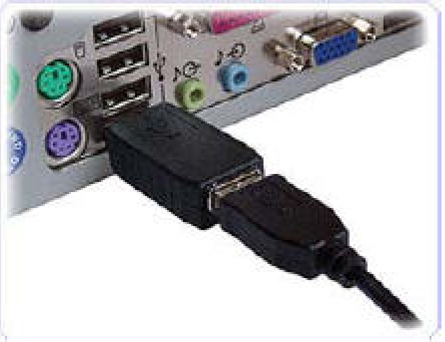


Figure 5: USB Keylogger
(Keelogger)

In both cases, the keylogger is connected directly to the PC and the keyboard to the keylogger. Another method is to install a keylogger circuit into a standard keyboard. This has the advantage of no physical evidence of user monitoring.

Laptops present a special challenge. External keyloggers are not an option unless the portable computer never leaves its docking station, and an external keyboard is used. So devices must be installed in the laptop. Figure 6 is an example of a mini-PCI hardware keylogger.

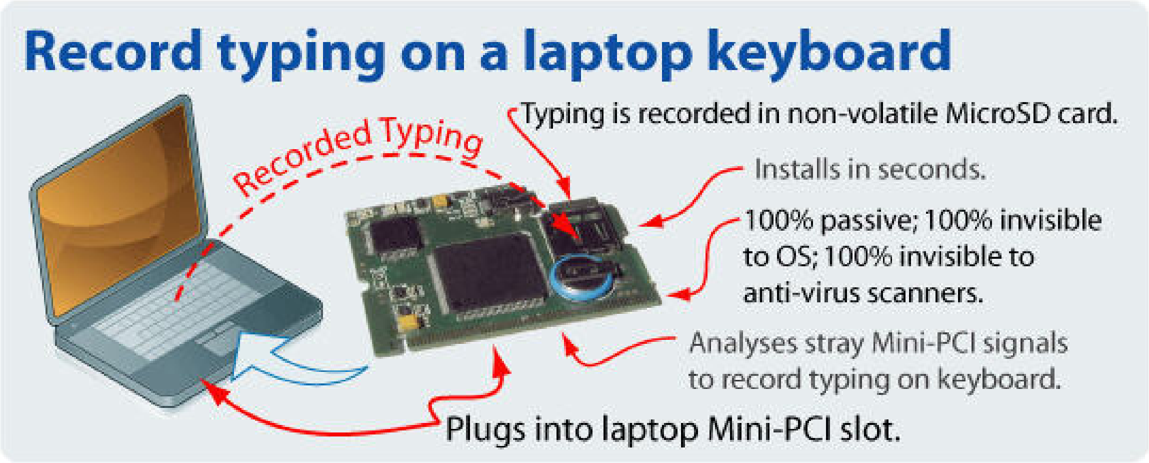
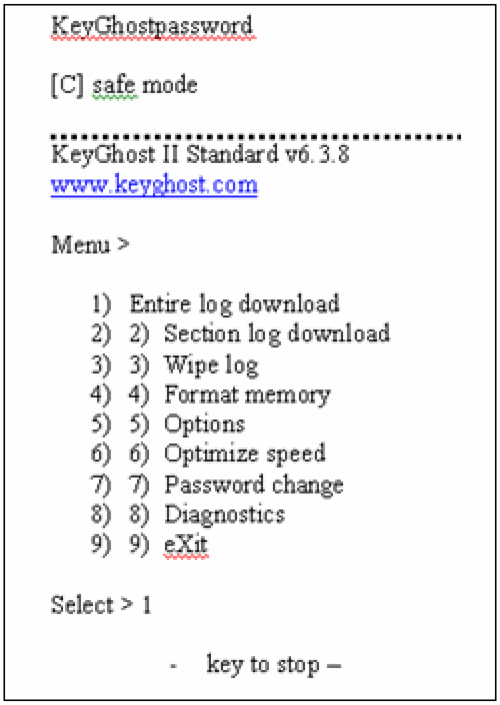

Record typing on a laptop keyboard
Recorded Typing
Typing is recorded in non-volatile MicroSD card.
Installs in seconds.
100% passive; 100% invisible to OS; 100% invisible to anti-virus scanners.
Analyses stray Mini-PCI signals to record typing on keyboard.
Plugs into laptop Mini-PCI slot.

Figure 6: Laptop Keylogger
(BitForensics)

Physical access or proximity is required when using a hardware keylogger, for installation and to extract captured data. Let's step through the process.

Once the keylogger is connected, it immediately begins keystroke collection, powered by the PC connector. A processor on the logger captures character and control code data and writes them to onboard memory. Memory capacity often exceeds 4 GB, enough to store up to two years of typing. This process is invisible to the user and impossible to detect. The keylogger stores no files on the target system nor does it require tell-tale software.

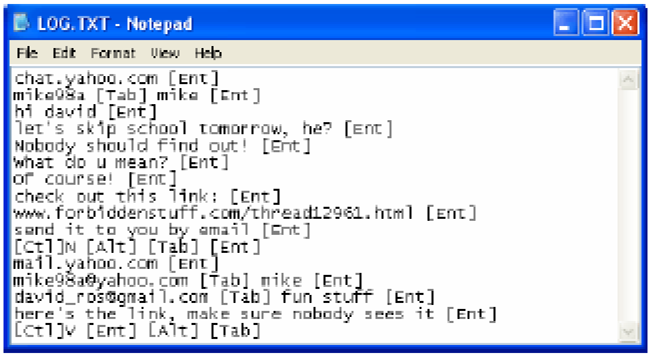
Data is extracted from keylogger memory in one of two ways. In the first method, a keystroke combination on the target system's keyboard loads and executes a menu stored on the keylogger. See Figure 7.



```
KeyGhostpassword

[C] safe mode

..............................................
KeyGhost II Standard v6.3.8
www.keyghost.com

Menu >

    1)  Entire log download
    2)  2)  Section log download
    3)  3)  Wipe log
    4)  4)  Format memory
    5)  5)  Options
    6)  6)  Optimize speed
    7)  7)  Password change
    8)  8)  Diagnostics
    9)  9)  eXit

Select > 1

               -    key to stop –
```

Figure 7: Sample Hardware Keylogger Menu
(KeyGhost)

As shown, the keylogger's password (key combination) and log are managed from the machine to which it's attached, either the target system or an offsite analysis device. The log can be downloaded to any attached storage device. Figure 8 shows sample log content.



```
chat.yahoo.com [Ent]
mike98a [Tab] mike [Ent]
hi david [Ent]
let's skip school tomorrow, he? [Ent]
Nobody should find out! [Ent]
what do u mean? [Ent]
of course! [Ent]
check out this link: [Ent]
www.forbiddenstuff.com/thread12961.html [Ent]
send it to you by email [Ent]
[Ctl]N [Alt] [Tab] [Ent]
mail.yahoo.com [Ent]
mike98a@yahoo.com [Tab] mike [Ent]
david_ros@gmail.com [Tab] fun stuff [Ent]
here's the link, make sure nobody sees it [Ent]
[Ctl]v [Ent] [Alt] [Tab]
```

Figure 8: Sample Log File Content
(Keelogger)

The previous retrieval method requires actual physical contact with the target system or the keylogger. However, there is another way. See Figure 9.

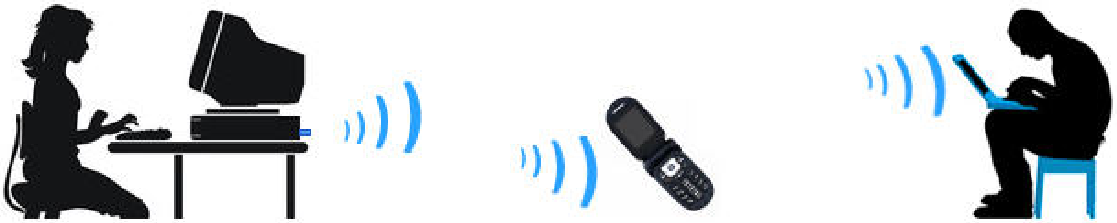


Figure 9: Bluetooth-accessible Keylogger
(Wirelesskeylogger.com)

Wirelesskeylogger.com offers a Bluetooth-accessible keylogger, capable of transmitting up to 300 feet, through walls and other physical structures (wirelesskeylogger.com). Although only available for PS/2 connections, the site states that USB support is in the works. It also comes in a wireless keylogger keyboard.

The log stored in the keylogger's memory is accessed via any of the following:

- All laptops and desktops running Windows 98, 2000, XP, Vista;
- All laptops and desktops running MAC OS 8/9, OSX;
- All mobile phones/PDA's running Windows Mobile; and
- The iPhone.

The advantage of using a hardware keylogger is its invisibility to anti-malware software; although security aware users can easily see them. A disadvantage, at least for non-Bluetooth-accessible devices, is the need for physical access to retrieve information.

Bluetooth keyloggers are visible to Bluetooth detection solutions. However, wirelesskeylogger.com claims it can help.

Physical access and AV software detection challenges are also addressed by the last two keylogger types. The first is wireless keyboard intercept.

Wireless Keyboard Intercept

For the purpose of this paper, wireless keyboards are devices that use a 27 MHz RF connection. The good news is that transmission range is limited to six feet. The bad news is that there is an RF transmission radius of six feet. And although wireless keyboard manufacturers encrypt RF transported keystroke characters, the encryption, at least on Microsoft keyboards, is very weak (Moser and Schrodel, 2008). Do not rely on it to protect sensitive data.

What makes this a serious problem, even with a six foot limitation, is an attacker's capability to capture packets from all wireless keyboards within range, at the same time. Each packet is flagged so the keyboard's receiver knows to process it. This also allows an RF device to sort captured wireless keyboard packets into appropriate character streams.

The one big disadvantage of using wireless intercept keyloggers is the need for a receiver/antenna relatively close to the target system. However, it's easy to hide small antennas in most office environments. The point to take away? If a workstation is processing highly sensitive information, don't use 27 MHz wireless keyboards. And physical security is always a good control.

Acoustic Keyloggers

The final keylogger type is like something out of a James Bond movie. It requires special equipment that "listens" to a user typing and special software that performs statistical analysis on captured data. Acoustic logging technology is more experimental than practical, based on work done at the University of California, Berkeley (Zhuang, Zhou and Tygar, 2005). Let's see how it works.

The same devices used to remotely listen to conversations are used to record typing sounds. Such microphones can be placed in the target work area or long distance solutions can be used. Parabolic microphones are an example of a long distance device. See Figure 10.

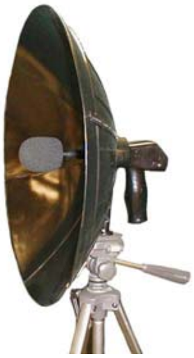


Figure 10: Parabolic Microphone

These microphones can pick up keyboard sounds from hundreds of feet away. Attached equipment records the sounds, which are then passed to audio-to-character translation software.

The software uses the statistical constraints of English, i.e. "the limited number of English words limiting the possible temporal combinations of keys and English Grammar limiting the word combinations" (Zhuang, Zhou and Tygar, 2005). Using a 10 minute sound recording of a user typing English text, it takes about 30 minutes to derive character results. The process correctly translates 75 to 90 percent of words typed and 90 to 96 percent of characters. Results assume that a key sounds exactly the same each time it's pressed and that a standard keyboard is used.

In addition to good physical security, systems processing sensitive information should probably not be placed in front of windows with good line of sight to adjacent structures. This also protects against long distance shoulder surfing.

Defending Against Keyloggers

Controls to defend against keyloggers are similar to those used to protect systems from other malware—particularly rootkits—including,

- Lock systems when not in use;
- Implement and enforce physical security controls;
- Enable safe-surfing
  - Use Web filtering to block access to known or suspected malicious sites;
  - Do not allow users local administrator access;
  - Deploy endpoint software policy controls (e.g., WebSense CPM);
- Maintain a regularly updated and monitored anti-malware solution;
- Apply security patches as soon as reasonably possible;
- Purchase and use keylogger detection software to spot-check sensitive systems (e.g., SnoopFree Privacy Shield); and
- Allow only necessary protocols on endpoint devices, and block unauthorized sessions between endpoints and external sites.

These controls are reasonable and appropriate for most environments. However, security managers responsible for systems processing highly sensitive information should also consider the following:

- Screen-based virtual keyboards—Instead of entering data at the physical keyboard, users press keys displayed on their monitors. This bypasses the normal path taken by keyloggers, making it impossible for them to capture keystrokes.
- Automatic form filler programs.
- Encrypting keyboard input—Software solutions like GuardedID from StrikeForce encrypt keyboard input so keyloggers can't use it. See Figure 11. According to the vendor, the encryption solution protects against 95 to 96 percent of software related keylogger attacks. The downside is that this only works within a supported browser. StrikeForce is working on a version that works at the OS level.

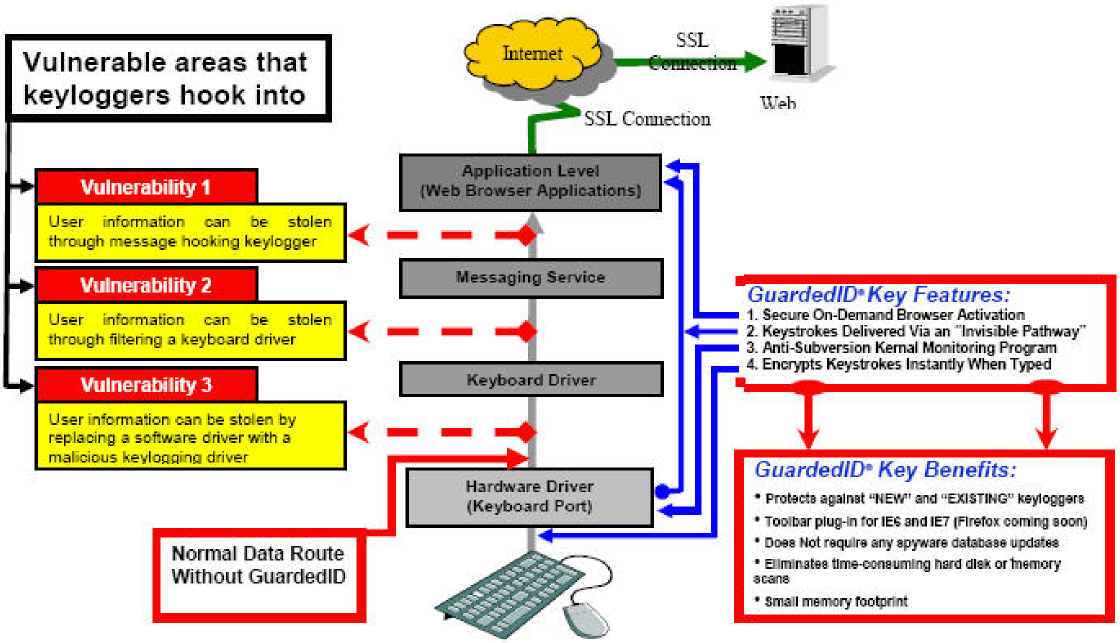

Vulnerable areas that keyloggers hook into
Vulnerability 1
User information can be stolen through message hooking keylogger
Vulnerability 2
User information can be stolen through filtering a keyboard driver
Vulnerability 3
User information can be stolen by replacing a software driver with a malicious keylogging driver
Normal Data Route Without GuardedID
Internet
SSL Connection
Web
SSL Connection
Application Level (Web Browser Applications)
Messaging Service
Keyboard Driver
Hardware Driver (Keyboard Port)
GuardedID® Key Features:
1. Secure On-Demand Browser Activation
2. Keystrokes Delivered Via an "Invisible Pathway"
3. Anti-Subversion Kernal Monitoring Program
4. Encrypts Keystrokes Instantly When Typed
GuardedID® Key Benefits:
* Protects against "NEW" and "EXISTING" keyloggers
* Toolbar plug-in for IE6 and IE7 (Firefox coming soon)
* Does Not require any spyware database updates
* Eliminates time-consuming hard disk or memory scans
* Small memory footprint

Figure 11: StrikeForce GuardedID
(StrikeForce)

If you believe one or more of your systems is compromised with a keylogger,

- Disconnect the system from the network and isolate it from physical access;
- If a software keylogger, locate the log file and retain it to identify potentially compromised information, re-image the system;
- If a hardware keylogger, remove it from the system and retain it to identify potentially compromised information;
- Change all passwords/PINS used by the users of the compromised system, including,
  - Local;
  - Network;
  - Web; and
- Notify management and recommend notification of affected,
  - Financial institutions;
  - Business partners;
  - Employees or customers if PII or ePHI might have been captured, in accordance with state or federal notification laws;

Conclusion

Keystroke logging attacks bypass all other controls. They are easy to implement and manage, providing attackers with useful account, identity, and intellectual property information. On the other hand, they are useful investigative tools.

Controlling keylogging technology within your organization is no different than managing other threats and tools, requiring common sense and a layered defense. The key is to be aware they exist, understand how they're used, and implement ways to detect them, with keylogger detection and containment part of your incident response plan.

© 2008 Thomas W. Olzak.

Tom Olzak, MBA, CISSP, MCSE, is President and CEO of Erudio Security, LLC.
He can be reached at tom.olzak@erudiosecurity.com.

Check out Tom's book, Just Enough Security

Additional security management resources are available at http://adventuresinsecurity.com

Free security training available at http://adventuresinsecurity.com/SCourses

Works Cited

Mosel, M. & Schrodel, P. (2008). *27MHz wireless keyboard analysis report aka "we know what you typed last summer."* Retrieved 3 April 2008 from http://www.dreamlab.net/download/articles/27_Mhz_keyboard_insecurities.pdf

Shetty, S. (2005, April). *Introduction to spyware keyloggers.* SecurityFocus. Retrieved 27 March 2008 from http://www.securityfocus.com/infocus/1829

StrikeForce (unknown). *GuardedID whitepaper*. Retrieved 4 April 2008 from http://www.guardedid.com/PDF/GuardedID_white_paper.pdf

Wilson, T. V. & Tyson, J. (2008). *How computer keyboards work.* HowStuffWorks.com. Retrieved 25 March 2008 from http://computer.howstuffworks.com/keyboard.htm

Zhuang, L, Zhou, F. & Tygar, J.D. (2005, November). *Keyboard acoustic emanations revisited.* Retrieved 3 April 2008 from http://www.cs.berkeley.edu/~zf/papers/keyboard-ccs05.pdf