Secure Software Project Report On

Key Logger

Mihir Dhanani
College of Computer Science,
University of Canterbury.
Email: - msd54@uclive.ac.nz

Abstract— The main objective of this project was to make a user aware of how a simple software can be a reason for an attack and also to develop an attacking software demonstrating an attack and measure of avoidance. The purpose of this report is to present all the preparations and work done on the project, this report consist of 5 sections; Introduction, System Models, Project Description, Discussion and Conclusion.

Keywords— Key Logger, Anti Key Logging, Secure Key Logger, Key Logger threat.

I. INTRODUCTION

Keystroke logging, often referred to as key logging or Keyboard Capturing, is the action of recording (or logging) the keys struck on a keyboard, typically in a covert manner so that the person using the keyboard is unaware that their actions are being monitored. It also has very legitimate uses in studies of human-computer interaction. There are numerous key logging methods, ranging from hardware and software-based approaches to acoustic analysis. This project refers to a software based key logger.

There are two main goals of this project,

1. To develop the key logger.

   Implementing the software with the tasks involved.

2. To avoid an attack (Security Awareness).

   Measures implementing the attack from occurring.

[3][4] Key Loggers are computer programs designed to work on the target computer's operating system. From a technical perspective there are six categories:

1. Hypervisor-based: The key logger can theoretically reside in a malware hypervisor running underneath the operating system, which remains untouched. It effectively becomes a virtual machine. Blue Pill is a conceptual example.
2. Kernel-based: This method is difficult both to write and to combat. Such key loggers reside at the kernel level and are thus difficult to detect, especially for user-mode applications. They are frequently implemented as rootkits that subvert the operating system kernel and gain unauthorized access to the hardware, making them very powerful. A key logger using this method can act as a keyboard device driver for example, and thus gain access to any information typed on the keyboard as it goes to the operating system.
3. API-based: These key loggers hook keyboard APIs; the operating system then notifies the key logger each time a key is pressed and the key logger simply records it. Windows APIs such as GetAsyncKeyState(), GetForegroundWindow(), etc. are used to poll the state of the keyboard or to subscribe to keyboard events. These types of key loggers are the easiest to write, but where constant polling of each key is required, they

can cause a noticeable increase in CPU usage, and can also miss the occasional key. A more recent example simply polls the BIOS for pre-boot authentication PINs that have not been cleared from memory.

4. Form grabbing based: Form grabbing-based key loggers log web form submissions by recording the web browsing on submit event functions. This records form data before it is passed over the Internet and bypasses HTTPS encryption.
5. Memory injection based: Memory Injection (MitB)-based key loggers alter memory tables associated with the browser and other system functions to perform their logging functions. By patching the memory tables or injecting directly into memory, this technique can be used by malware authors who are looking to bypass Windows UAC (User Account Control). The Zeus and Spyeye Trojans use this method exclusively.
6. Packet analyzers: This involves capturing network traffic associated with HTTP POST events to retrieve unencrypted passwords.

Remote access software key loggers are local software key loggers with an added feature that allows access to the locally recorded data from a remote location. Remote communication may be achieved using one of these methods:

1. Data is uploaded to a website, database or an FTP server.
2. Data is periodically emailed to a pre-defined email address.
3. Data is wirelessly transmitted by means of an attached hardware system.
4. The software enables a remote login to the local machine from the Internet or the local network, for data logs stored on the target machine to be accessed.

There are various steps involved in the development of the software such as basic planning which comprises of: -

a. Understanding the user.

Initial understanding is to think of the software development from both an attacker and a user’s point of view as the usability of a general user varies. Though there are just two types of users in general intelligent and starter users.

b. Why is it necessary to understand the user?

In general no one thinks for an attack to occur so the negligence is one of the reasons for these attacks. Before realizing the effect of this negligence harm is already been done.

c. Act as an user.

To understand these issues I checked other similar software’s available online which helped me understand how an attacker thinks and how I as a general user just installed the software for my use unaware of the harm it could cause.

d. Expected outcomes of this projects and project contributions.

The aim of all user should be able to control the software they install in this case the key logger should be achieved.

e. Project management plan

The different tasks to be evaluated are mentioned in %:

1. Designing (10%) [Form Designs]
2. Planning Stage (20%) [Gathering requirements and understanding it.]
3. Actual Implementation (40%) [Designing the form and coding]
4. Coding/Testing (20%) [Checking the code for error and testing the software]
5. Report (10%) [Maintaining report for the stages and testing]

II. SYSTEM MODELS

A. System Model

As mentioned above about understanding the user, hence considering the user having connected to the internet and using the key logging software unaware of its hidden functionality is prone to suffer from an attack.

Consider the user model below:

As shown above the user is using the software but doesn't know the outcome of it as is unaware of its other functionality as it is a third party software which clearly states that all users are responsible for its use.

Hence the user's details in this case are accessible by the attacker, though the behavior of the software depends on the type of attack it resembles. This details could be the user identification such as email id, password, and credit card details etc.

The log represents the key strokes which is every input from the keyboard.

The outcome of this could be unauthorized access to email or bank details or the attacker acts as the user i.e. taking his/her identity and using it for his own profit etc.

B. Software

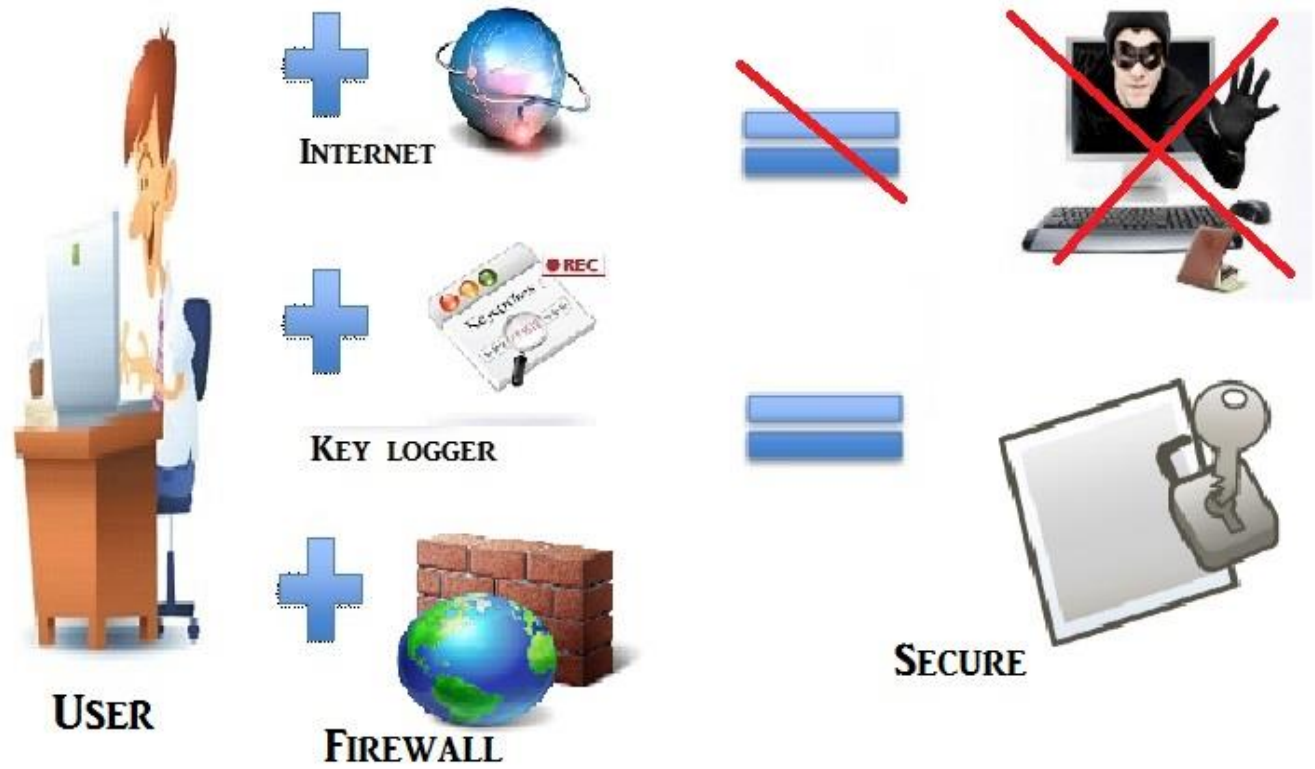
As stated in the proposal the software requirement for this project are

1. Visual Studio 2012
2. .Net Framework

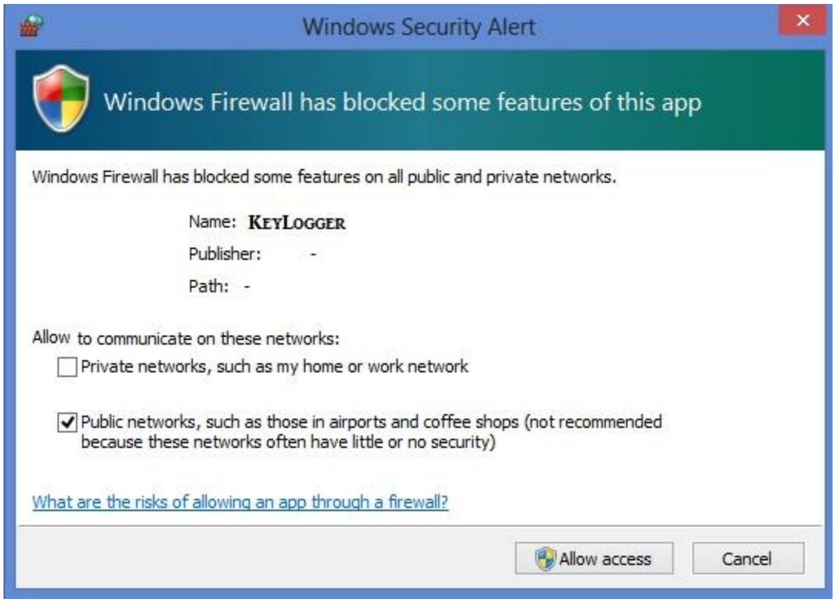
Other requirement is either an external firewall software or default windows firewall enabled.

C. Security Model (optional)

The security phase is not in the application layer. i.e. in the software but is user defined. As without this an attack will be performed and achieved. In the software developed the type of threat is email i.e. everything logged in the textbox is sent to the email if defined in the code of the software. This is self-triggered either at exit or when clear button is pressed and hence to avoid this from occurring, a rule in the windows firewall is set up denying the access. Consider the model below:



Setting the rule in the windows firewall avoids sending the email at termination or pressing the clear button. There are anti-viruses available which have their own firewall for use if not willing to use the windows firewall.



Windows Security Alert

Windows Firewall has blocked some features of this app

Windows Firewall has blocked some features on all public and private networks.

Name: KEYLOGGER
Publisher: -
Path: -

Allow to communicate on these networks:

□ Private networks, such as my home or work network

☑ Public networks, such as those in airports and coffee shops (not recommended because these networks often have little or no security)

What are the risks of allowing an app through a firewall?

Providing an external security measure is essential as the behavior of the key logger is only known to the developer at first and hence one can never be sure if there was an attack to occur. In this software there are two events defined.

1. Generate a file with the logs.

   This is stored on the local computer.

2. Send an email.

   Sends an email based on the text box apart from the log file. This is defined in the code.

III. Project Description

A key logger is a type of surveillance software (considered to be either software or spyware) that has the capability to record every keystroke you make to a log file. A key logger recorder can record instant messages, e-mail, and any information you type at any time using your keyboard. The log file created by the key logger can then be sent to a specified receiver. Some key logger programs will also record any e-mail addresses you use and Web site URLs you visit. Key loggers, as a surveillance tool, are often used by employers to ensure employees use work computers for business purposes only. Unfortunately, key loggers can also be embedded in spyware allowing your information to be transmitted to an unknown third party. Hence, the idea is to develop a key logger demonstrating safe environment use and not a spyware.

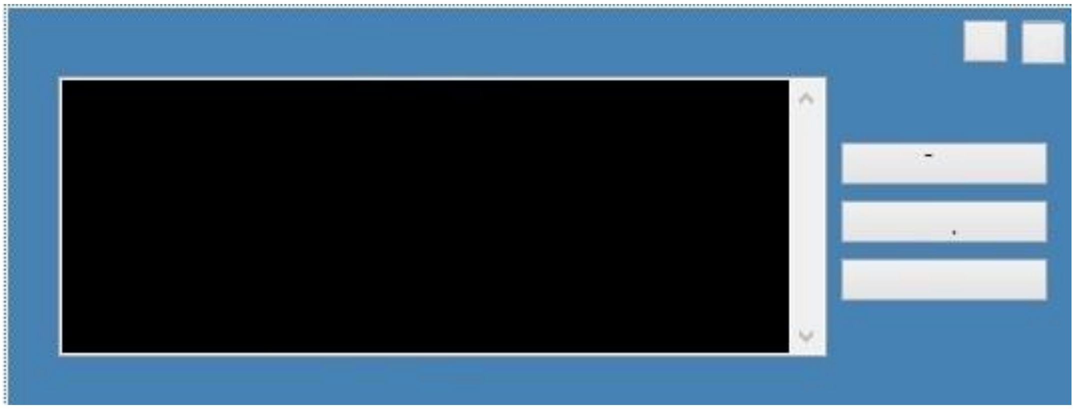
A. *Project Overview*

The project has five main tasks involved in its development.

B. *Task 1 : Designing.*

This was the initial step in this project was to design the form layout.

This form is visible to the user. Though what is visible is not always true, as in this case there are two text fields behind the black text box.

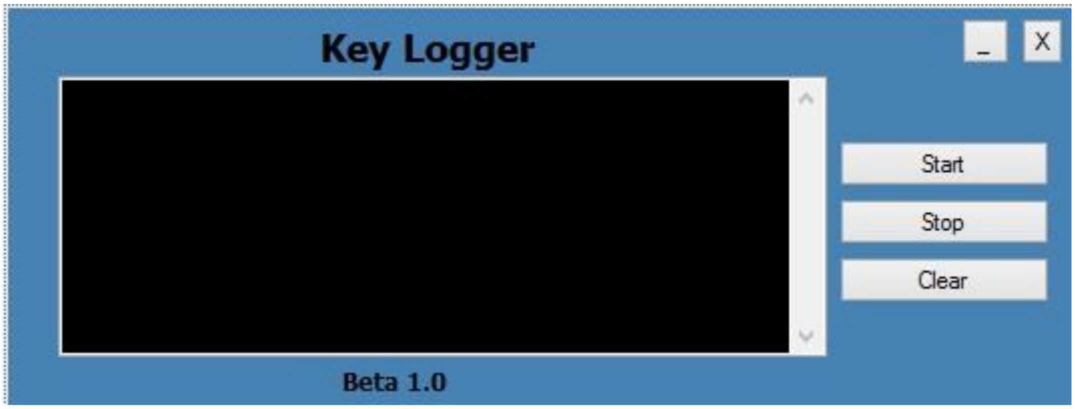


C. *Task 2: Planning*

This is the next task in queue, as the initial design of the form is ready hence planning the functions that will carried out on the form and how to carry them out. Hence as shown above naming the controls used in the form.

D. *Task 3: Implementation*

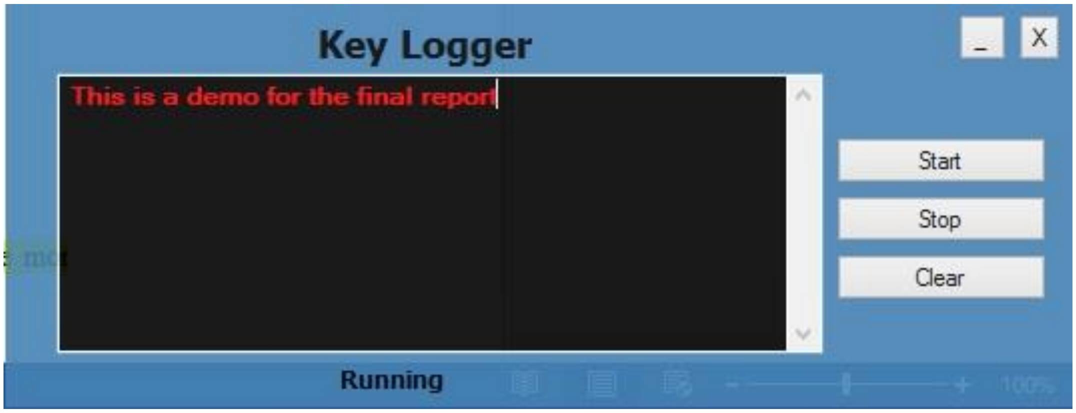
In this task the above two tasks mentioned need to be carried out i.e. as stated naming the controls as required and also start the coding once the form is ready.



The code is where the attack is set up, In this case consider the sample code as mentioned below: -

E. Task 4: Coding/Testing

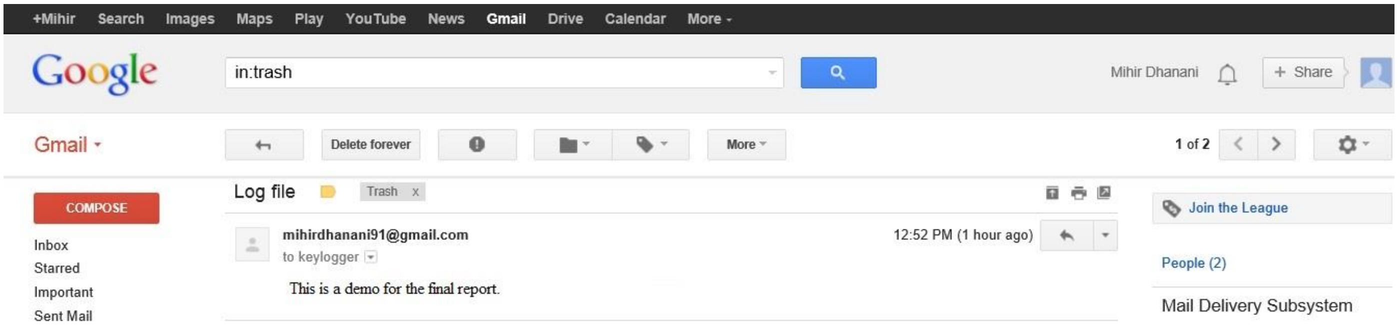
In this task testing the software for its functionality and sorting code errors was performed, after all the code had to be compiled for errors and also the software had to be checked for its logging capabilities.
The screenshots for the tests care displayed below:



This is a demo for the final report

Running

This image above displays a sample of the log recorded.



in:trash

Log file

mihirdhanani91@gmail.com
to keylogger
12:52 PM (1 hour ago)

This is a demo for the final report.

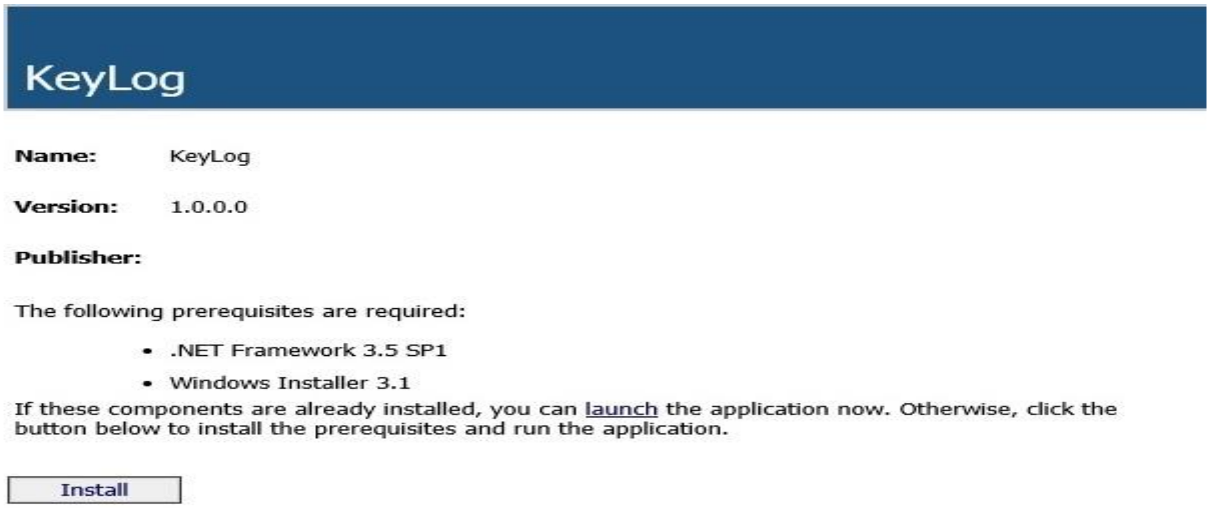
The image above displays the email received, triggered as mentioned on the click event stated in the coding task.

F. Task 5: Report

As per the initial task list mentioned above the fifth task is report. But what good is a software without packaging hence, this was performed as a sub type along with the reporting.

- Software can be published in 3 ways:

1. Through an website URL.(Pub\publish.htm)



KeyLog

Name: KeyLog

Version: 1.0.0.0

Publisher:

The following prerequisites are required:

- .NET Framework 3.5 SP1
- Windows Installer 3.1

If these components are already installed, you can launch the application now. Otherwise, click the button below to install the prerequisites and run the application.

2. Can be added to an ftp server.
3. Or General Setup Package.

G. Deliverables

The outcome of this project is to understand the countermeasures that need to be taken in general because key loggers use a variety of techniques to capture data and the countermeasure needs to be effective against the particular data capture technique. For example, an on-screen keyboard will be effective against hardware key loggers, transparency will defeat some but not all screen loggers and an anti-spyware application that can only disable hook-based key loggers will be ineffective against kernel-based key loggers. Also, key logger program authors may be able to update the code to adapt to countermeasures that may have proven to be effective against them.

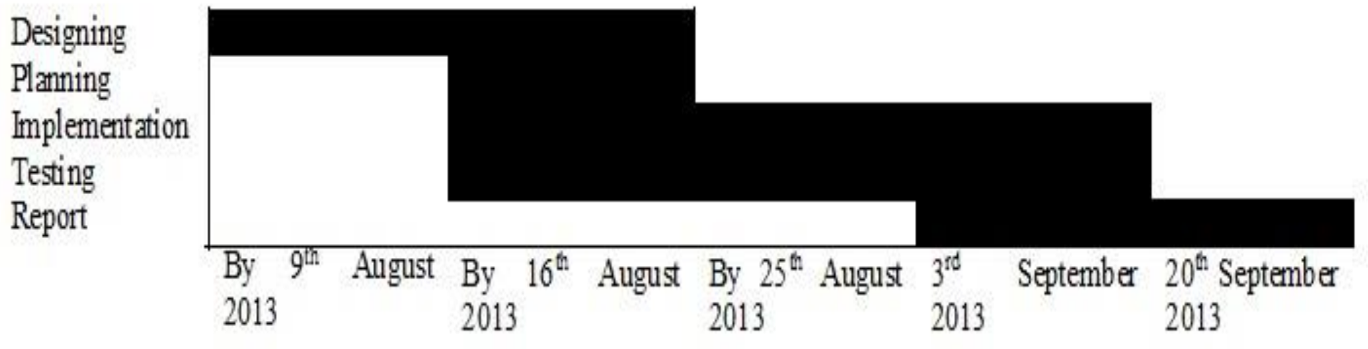
The project of making the key logger was established but the security measure used in this was external hence to make the software efficient, measures need to be taken from within the application itself which is a task that need to be established.

The main aspect of this project is to learn from the experience of others or from past events and evolve from it, consider a past example for key logger.

> [7][8] In 2000, the FBI used FlashCrest iSpy to obtain the PGP passphrase of Nicodemo Scarfo, Jr., son of mob boss Nicodemo Scarfo. Also in 2000, the FBI lured two suspected Russian cyber criminals to the US in an elaborate ruse, and captured their usernames and passwords with a key logger that was covertly installed on a machine that they used to access their computers in Russia. The FBI then used these credentials to hack into the suspect's computers in Russia in order to obtain evidence to prosecute them.

H. Project Timeline

Consider the chart below showing the roadmap for this project representing the tasks and timeline schedule for the same to be achieved.



IV. Discussion

The main problem faced was to actually understand the behavior of the software as a developer I knew about the email that's been sent but cannot be detected from within the software as the text field was hidden behind the text box logging the keys and due to transparency is not doubted. That actually makes one alert as a result one could automatically distrust public along with computers owned by friends is due to the possibility that it may contain malware. [2] *Mike Chapple, CISSP is an IT Security Professional with the University of Notre Dame* commented "Now that I think about it, I basically distrust any computer that I don't inherently own. Some may think that as a bit paranoid but that's mainly due to you forgetting one very basic computer security tenet: once a bad guy gets you to run his/her software on your computer, it's not your computer anymore. Once you hand over your personal information, even if it happened just one time, that information is not private anymore. All it takes is just one incident. So now you may be thinking, "Well Mr. Perfect, how do you know that the computer that you do own isn't infected with malware as well?" The correct answer is that I really can't 100% be sure as well. However, at least I know that I have taken all the right pre-cautions and practiced safe computing habits to greatly minimize the chances of it being infected with malware. Can you say the same for that computer you're sitting in front of at your public library? Can you say the same when you are using your friend's computer to log in to your financial website? I doubt it"

The true danger posed by key loggers is their ability to bypass encryption controls and gather sensitive data directly from the user. All the encryption in the world will not secure your data if a hacker watches you type your encryption key. He can then simply use that plaintext key to decrypt all of your protected communications from that point forward. Hence remedies for this is manually done as there is no direct approach such as monitor your intrusion-detection system (IDS) and keep the signatures current. If you're not able to block spyware from [9] phoning home, you might at least be able to detect it with your IDS and use the reports to identify infected systems.

Secondary, Prevent users from installing downloaded software. Most spyware installations are the result of users installing unauthorized software downloaded from the Internet. Spyware, and the associated crime of identity theft, is one of the most important battles currently facing information security professionals.

Future techniques to be considered as a part for awareness could include:

Automatic form filler programs[2]

Automatic form-filling programs may prevent key logging by removing the requirement for a user to type personal details and passwords using the keyboard. Form fillers are primarily designed for web browsers to fill in checkout pages and log users into their accounts. Once the user's account and credit card information has been entered into the program, it will be automatically entered into forms without ever using the keyboard or clipboard, thereby reducing the possibility that private data is being recorded. However someone with physical access to the machine may still be able to install software that is able to intercept this information elsewhere in the operating system or while in transit on the network. (Transport Layer Security (TLS) prevents the interception of data in transit by network sniffers and proxy tools.)

One-time passwords (OTP)

Using one-time passwords may be key logger-safe, as each password is invalidated as soon as it's used. This solution may be useful for someone using a public computer, however an attacker who has remote control over such a computer can simply wait for the victim to enter his/her credentials before performing unauthorized transactions on their behalf while their session is active.

Security tokens

Use of smart cards or other security tokens may improve security against replay attacks in the face of a successful key logging attack, as accessing protected information would require both the (hardware) security token as well as the appropriate password/passphrase. Knowing the keystrokes, mouse actions, display, clipboard etc. used on one computer will not subsequently help an attacker gain access to the protected resource. Some security tokens work as a type of hardware-assisted one-time password system, and others implement a cryptographic challenge-response authentication, which can improve security in a manner conceptually similar to one time passwords. Smartcard readers and their associated keypads for PIN entry may be vulnerable to keystroke logging through a so-called supply chain attack[26] where an attacker substitutes the card reader/PIN entry hardware for one which records the user's PIN.

V. Conclusion

No software-based anti-spyware application can be 100% effective against all key loggers. Many anti-spyware applications are able to detect some software based key loggers and quarantine, disable or cleanse them. However, because many key logging programs are legitimate pieces of software under some circumstances, anti-spyware often neglects to label key logging programs as spyware or a virus. These applications are able to detect software-based key loggers based on patterns in executable code, heuristics and key logger behaviors (such as the use of hooks and certain APIs).

Software-based anti-spyware cannot defeat non-software key loggers (for example, hardware key loggers attached to keyboards will always receive keystrokes before any software-based anti-spyware application).

However, the particular technique that the anti-spyware application uses will influence its potential effectiveness against software key loggers. As a general rule, anti-spyware applications with higher privileges will defeat key loggers with lower privileges. For example, a hook-based anti-spyware application cannot defeat a kernel-based key logger (as the key logger will receive the keystroke messages before the anti-spyware application), but it could potentially defeat hook- and API-based key loggers.

This was one example of tracking the attack and avoidance, there are other ways of improving this. [9] An example for such other could be network monitors (also known as reverse-firewalls) can be used to alert the user whenever an application attempts to make a network connection. This gives the user the chance to prevent the key logger from phoning home with his or her typed information.

ACKNOWLEDGMENT

I would like to thank our lecturer Dijiang Huang for his assistance towards making us understand the security issues that are involved nowadays and approving my proposal for the development of this software.

REFERENCES

[1] ^ "Anti Keylogging & Public Computers". *Anti Keylogging & Public Computers*. Archived from the original on 22 May 2011. Retrieved 10 May 2011.

[2] ^ "Privacy Watch: More Criminals Use Keystroke Loggers". *Privacy Watch: More Criminals Use Keystroke Loggers*. PC World About.

[3] ^ "Keylogger". Oxford dictionaries.

[4] ^ "What is a Keylogger?". PC Tools.

[5] ^ "SpyEye Targets Opera, Google Chrome Users". *Krebs on Security*. Retrieved 26 APR 11.

[6] ^ "Keylogger Removal". *Keylogger Removal*. SpyReveal Anti Keylogger. Retrieved 25 April 2011.

[7] ^ John Leyden (2000-12-06). "Mafia trial to test FBI spying tactics: Keystroke logging used to spy on mob suspect using PGP". The Register. Retrieved 2009-04-19.

[8] ^ John Leyden (2002-08-16). "Russians accuse FBI Agent of Hacking". The Register.

[9] ^ Christopher Ciabarra (2009-06-10). "Anti Keylogger". Networkintercept.com.

[10] ^ Cormac Herley and Dinei Florencio (2006-02-06). "How To Login From an Internet Cafe Without Worrying About Keyloggers" (PDF). Microsoft Research. Retrieved 2008-09-23