Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 3, March 2015, pg.180 – 184

RESEARCH ARTICLE

INTERACTIVE KEYLOGGING SYSTEM TO DISCOVER ABUSER ACTIVITY

Ms. J.RAJESHWARI
B.E-CSE
IFET COLLEGE OF ENGG
VILLUPURAM

Mr. R.RAJESH, M.TECH
Asst.Professor (CSE)
IFET COLLEGE OF ENGG
VILLUPURAM

rajijagan94 @gmail.com

ABSTRACT:

The windows based application which helps the admin to track the activities of the user. It works as a spy in a computer. It tracks the user keyboard entries such the key strokes entered by the user and it forms a word then it saves into the database. It has a login in order to authenticate the admin whether the user has rights turn on or off the application to track the user keystrokes. It forms a word using the separator as a space and wildcards such as question mark, full stop, comma, etc. The key logging system is

done using GlobalHook technique. It receives the user keystrokes before it reach the user interface. The list of keystrokes are unruffled in this method and form a meaningful word by identifying the separator such as space, comma, semi colon, etc.

Key Terms- keystroke logging, access control, globalhook technique.

1. Introduction**:**

Keyboard is perhaps the most common human input device. We use keyboard to input a variety of information, some of which are highly expensive, such as passwords, social sanctuary numbers, and glory card numbers. It came as no surprise that keystroke logging is a favorite tool of trade by attackers. The attacker can install a Trojan program on the victim computer to log keystrokes, or use out of posse channel to infer keystrokes. Acoustic key logger, can infer keystroks from acoustic frequency signatures, timings between two keystroks, or language models. Electromaganetic emanations of keyboards are also studied for keylogging Touch screen smartphones have changed the paradigm of user interaction. Most touch screen smart phones have no corporeal keyboard. As an substitute, the user types on the software upright on the panel. Since there is neither sound nor electromagnetic emanation from a virtual keyboard, the attacker can no longer infer keystrokes based on these signals. Moreover, many smart phone operating systems, such as Android and iOS, restricts privileges granted to applications. In most cases, an application cannot read keystrokes unless it is active and receives the focus on the panel. It seems that key loggers, at least the traditional ones described above, are facing severe obstacles on touch screen smart phones. Accelerometers and gyroscopes, may be used to infer keystrokes. When the user types on the soft keyboard on her smart phone (especially when she holds her phone by hand rather than placing it on a fixed facade), the phone vibrates. Most emission, including acoustic keyboard secretion, are not uniform across different occurrences, even Different users on a single keyboard or different keyboards (even of the same model) emit different sounds, making reliable recognition hard. They achieved relatively high recognition rate (approximately 80%) when they trained neural networks with textACM .labeled sound samples of the same user typing on the identical upright. Their attack is correlated to a known-plaintext attack on a cipher – the cryptanalyst has a sample of plaintext (the keys typed) and the corresponding ciphertext (the recording of aural secretion). This labeled training sample requirement suggests a inadequate attack, because the attacker needs to obtain training samples of wide-ranging length. Apparently these could be obtain from video scrutiny or network sniffing. However, video scrutiny in most cases should render the acoustic attack extraneous, because even if passwords are masked on the screen, a video shot of the upright could directly reveal the keys being typed.

2. Related Works:

2.1 Inferring keystrokes via device orientation:

We designed and implemented key Logger, an Android tool to infer keystrokes on the soft keyboard of smartphones from the device orientation. More precisely, key Logger infers the landing locations of the typing finger based on the device orientation and then looks up the corresponding keys based on the current soft keyboard configuration.

2.2 Inputlog 4.0

Inputlog allows researchers to record writing route data, generate different data files, integrate various types of data from other programs and playback the recorded session.

2.3 Record a writing session

Inputlog enables researchers to record data of a writing session in Microsoft Word and other Windows based programs (e.g. Internet explorer, Mozilla, Powerpoint, eInputlog logs every keystroke, every mouse movement and click, and - if available - speech input from Dragon Naturally talking. Furthermore, all the windows that the writers opens in different programs, They are branded and logged. So, if a critic uses Google when lettering a report, Inputlog logs the URL of the web page access collectively with a time. This enable researchers to take writers' search activities into account.

Inputlog 4.0 offers five different data analyses:

2.3.1 General file:

An XML file containing basic logging information of the writing session in which every line represents an input action (keyboard, mouse click or movement and – if present – speech, window information);

2.3.2 Linear text:

A plain linear text in XML-format containing the complete linear production of the text (keyboard and speech) including mouse actions and pauses. The linear scrutiny is divided into two options: on the one hand researchers can generate a linear output in which the writing activities are divided into periods (fixed time durations of x seconds, free to choose) or intervals (fixed number of equal timeslots in which the writing process is to be divided) of their choice. In both options the verge for the pause length can be adapted to meet the requirements of a particular study.

2.3.3 Summary data:

An XML file containing basic statistical information of the writing session on a more aggregate level. Several route uniqueness are shown, such as the number, mean and standard deviation of words, sentences, and paragraphs produced, pause times (based on the threshold entered in the interface) and the use of the different writing modes.

2.3.4 Pause analysis:

An XML file containing analyses of every non-scribal episode. The threshold for the pause can generally be set to 1, 2 or 5 seconds or to any user distinct level superior than 1 millisecond. Recess data are spawn on a more general level: number of pause, imply and standard deviation of pause length, and on a more explicit interval level in which the writing session is divided into 10 equal timeslots. Finally, pauses are appraise per word, sentence and paragraph locality.

2.3.5 Revision analysis:

An XML file containing a basic analysis of the number, level and the kind of revision that has taken place during the script conference. To define revision we have developed an algorithm and a set of policy. The alteration scrutiny first of all defines critical events in the writing process that can be linked to a revision and then evaluates these instances by comparing the operations in the isolated writing episode to the revision rules in the algorithm. Inputlog sequentially analysis the beginning of the revision, the range of the text to revise or the positioning of the cursor, the (possible) deletion of the text and the end of the revision.

3. Global Hook Technique:

A hook is a mechanism by which an application can interrupt events, such as post, mouse measures, and keystrokes. A function that intercepts a particular type of event is known as a hook procedure. A hook route can act on each occurrence it receives, and then modify or discard the event. The system supports many different types of hooks; each type provides access to a different phase of its message-handling method. For example, an relevance can use the WH_MOUSE hook to scrutinize the message traffic for mouse messages.

4. Conclusion:

A stealthy keylogger that runs directly on a PC, allowing it to evade current protection mechanisms that run on the host CPU. We have implemented and evaluated the keylogger on PCs. Besides recording keystrokes, the architecture of modern graphics processors enables our prototype to benefit from their excess computational capacity for analyzing the captured data. As part of our future work, we plan to port our prototype implementation for Windows, and explore similar techniques for performing other malicious activities, including the acquisition of sensitive data, such as cryptographic keys, credentials for web banking accounts, web-camera snapshots, screenshots, and open documents located in the file cache.

REFERENCES

[1] Android developer's documentation: Sensor event. http:/developer.android.com/reference/android/ hardware/SensorEvent.html
[2] ASONOV, D., AND AGRAWAL, R. Keyboard acoustic emanations. In IEEE Symposium on Security and Privacy (2004).
[3] CAI, L., MACHIRAJU, S., AND CHEN, H. Defending against sensor-sniffing attacks on mobile phones. In Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds (New York, NY, USA, 2009), MobiHeld '09, ACM, pp. 31–36.
[4] FOO KUNE, D., AND KIM, Y. Timing attacks on pin input devices. In Proceedings of the 17th ACM conference on Computer and communications security (New York, NY, USA, 2010), CCS '10, ACM, pp. 678–680.
[5] PACHAL, P. Google removes 21 malware apps from android market. Accessed March 18, 2011.
[6] POPESCU, A., AND BLOCK, S. DeviceOrientation event specification, editor's draft 9. http://dev.w3.org/geo/api/ spec-source-orientation.html, February 2011.
[7] SONG, D. X., WAGNER, D., AND TIAN, X. Timing analysis of keystrokes and timing attacks on ssh. In Proceedings of the 10th conference on USENIX Security Symposium - Volume 10 (Berkeley, CA, USA, 2001), USENIX Association, pp. 25–25.
[8] VUAGNOUX, M., AND PASINI, S. Compromising electromagnetic emanations of wired and wireless keyboards. In Proceedings of the 18th conference on USENIX security symposium (Berkeley, CA, USA, 2009), SSYM'09, USENIX Association, pp. 1–16.
[9] XU, N., ZHANG, F., LUO, Y., JIA, W., XUAN, D., AND TENG, J. Stealthy video capturer: a new video-based spyware in 3G smartphones. In Proceedings of the second ACM conference on Wireless network security (New York, NY, USA, 2009), WiSec '09, ACM, pp. 69–78.
[10] ZHANG, K., ZHOU, X., INTWALA, M., KAPADIA, A., AND WANG, X. Soundminer: A stealthy and context-aware sound trojan for smartphones. In Proceedings of the 18th Annual Networkand Distributed System Security Symposium (2011), NDSS '11.
[11] ZHUANG, L., ZHOU, F., AND TYGAR, J. D. Keyboard acoustic emanations revisited. ACM Trans. Inf. Syst. Secur. 13 (November 2009), 3:1–3:26
[12] Bergh, H.; Barbier, M.-L.; Spinelli-Jullien, N. (2009) On-line tools for investigating writing strategies in L2. German as a foreign language (in this issue).
[13]Braaksma, M.A.H.; Rijlaarsdam, G.; Van den Van Hout Wolters, B.H.A.M. (2004) Observational learning and its effects on the orchestration of writing processes. Cognition and Instruction, 22, 1-36
[14] Flower, L.; Hayes, J.R. (1980) The Dynamics of composing: making plans and juggling constraints. In: L.W. Gregg; E.R. Steinberg (eds.) Cognitive processes in writing. Hillsdale, NJ: Erlbaum, 31-50.

