Body Keylogging

Paz Hameiri

E-mail: bodykeylogger@gmail.com

Numeric keypads are now used to protect money, valuable goods, information and physical access, which makes them lucrative targets for criminals. The custom architecture of these devices makes it hard for an attacker to design or to deploy either a hardware-based or a software-based key-logging device. Cameras can be used for keypad tracking, but are limited by light conditions, shooting angles, line of sight, etc. In this paper, I suggest a new type of key-logging device that detects keystrokes by analyzing the interaction between the user's body and the device. Time-of-Flight sensors can be used to track body movements and by crossing this information with the rigid layout of the keypad, it is possible to reveal which key was pressed at any time. If the device generates audio feedback, the sound can be tracked by a microphone. This can improve successful code detection. To explore the body key-logging approach, I've designed and built a body key-logger and have used it on a commercially available safe. A field test of the device yielded a success rate of 92% for key press detection. In this paper, I introduce the device, its tracking techniques and the algorithms used for keystroke detection. I review the device's performance, discuss countermeasures for blocking this kind of attack and suggest future research.

Introduction

Cyber criminals and security researchers have employed different approaches to capture keystrokes on keyboards and keypads. Devices used to capture keystrokes are known as key-loggers. While the common numeric keypads used in safes and electronic door locks may offer an attacker immediate entry, that person needs intimate knowledge of the architecture of the device's hardware and software in order to build a customized key-

logger. Deployment of a key-logger is difficult since manufacturers build the devices so that only trained personnel know how to access the circuits without damaging the device or tripping the tamper alarm.

In this paper, I propose a new approach to key-logging. Since common keyboards and keypads have rigid user interfaces, it is possible to detect keystrokes by tracking the user's body movements and crossing that information with the layout of the keypad. Body tracking technology is commercially available and already in use for gesture recognition and computer vision.

The aim of this paper is to alert users to the risks of body tracking technology for the purpose of key-logging. To explore these risks, I designed and built a body key-logging "proof-of-concept" device from commercially available components and demonstrated its functionality on the keypad of a commercially available safe.

Malicious key loggers

Malicious key loggers' most fundamental requirement is to track keystrokes of an unsuspecting user in order to reveal the data to the person who planted the key-logger. Researchers, including Olzak [1] and Creutzburg [2], divide key-loggers into two main categories – software-based and hardware based. Software-based key-loggers are installed on the victim's device or on a device that is connected to the victim's device. Hardware-based key-loggers are based on dedicated hardware, whose main purpose is to act like a key-logger. Hardware-based devices are either connected to the victim's device or installed close to the victim's device to monitor various physical emissions. Simple hardware key-loggers are physically connected to keyboards and are able to extract keystrokes using the keyboard interface. More sophisticated key-loggers track measurable physical properties of the keyboard, like electrical properties, acoustics, electromagnetic emissions, and more. Another approach to hardware-based key logging is to use a well-placed surveillance camera to recover keystrokes from captured images, as demonstrated by Snopes.com [3] and Maggi et al [4].

When deploying a hardware-based key-logger, the attacker is required to connect the hardware to the victim's device or place it near the device. This is done by either physically accessing the device or by installing it close enough for the key-logger to track the data. When deploying a camera-based key-logger, installation locations are limited by the conditions needed for successful data extraction. The attacker needs to take into account the location of the keys, the location of the fingers, the camera angle, the light conditions and any other factor that might limit the image processing algorithms to recover the data from the captured images.

Numeric keypads under attack

A numeric keypad is a set of buttons arranged in a block that mostly bear digits. Numeric keypads are found on devices such as ATMs, safes, combination locks, and digital door locks. When using these devices, the user is required to enter an access code to access locked products, money or information. Since the access code is the key to an immediate profit, the keypad is a natural candidate for a key-logging attack. But planting a key-logger on such a device is hardly easy for the following reasons:

- In many cases, the hardware and software are embedded (e.g. Oke Alice et al. [5] and Lawan et al. [6]). In order to design a dedicated hardware key- logger or a dedicated software key-logger, the attacker needs to be familiar with the device's circuitry and code.
- Device designers are aware that the circuits and the keypad are the key to locked goods and make an effort to stop unauthorized personnel from accessing the device's control unit (e.g. Sargent & Greenleaf Inc [7] and Nortek Security & Control [8]).

Plore [9] demonstrated an electronic safe lock attack by analyzing the current consumption of the device. This attack did not use a key-logger by definition, but it resembles a key- logger attack in the sense that it measured and analyzed the electrical properties of the device. This attack is done by tampering with the device. Such an operation on a public device will draw much attention to the attacker and most likely will leave evidence that the safe has been tampered with.

Camera-based key-loggers exploit the interaction between the victim's fingers and the device keypad. This approach is harder to detect since the compromised device is not tampered with. The greater the distance between a disguised key-logger and a compromised device, the harder it is to link the two and expose the attack. The attacker does not need to be familiar with the device's circuitry or software, making it easier to focus on the development of the key-logger. Since a camera-based key-logger relies on image processing, it entails requirements for sensors, algorithms, processing power and battery usage. It is also limited by the limitations of photography such as the need for a clear line of sight and sufficient lighting – a keypad would be hard to photograph if the victim stands close to the keypad and blocks either the view of it or the light.

Body keylogging

When a user presses the keys on a keypad, an interaction is taking place between the user and the device. On one side of the interaction there's the device – the hardware, the software and the mechanics. On the other side of the interaction is the user – mind, senses, limbs and fingers. In the middle there's the interaction – the keys of the keypad are pressed one at a time and in some cases there's physical feedback to the user, indicating a successful key press (either visible or audible). Most key-loggers target the device side of the interaction. A camera-based key-logger targets the interaction between the user and the device from a viewpoint. Martinovic et al. [10] conducted experiments whose goal was to extract PIN numbers from the victim's brain. I propose a method to target the interaction between the user and the device from the user's side of the interaction.

Each keypad has a defined layout and dimensions. Therefore, the user is forced to press keys that have a well-defined position in space. This can be a vulnerability since eventually the user will press these positions in space in order to enter a code. A well-positioned key-logger based on a 3D camera (a camera with an ability to record spatial information) will be able to record the user's movements. Since the keypad's layout and dimensions are rigid and known to the attacker (either in advance or upon key-logger

deployment), an algorithm may be found to link the finger positions and the keypad layout in order to detect the code. This link can be based on the absolute position of the keys (coordinates of each key in space) or on a relative position of the keys (by following the distance between each key press and using one of the keys for spatial registration). If the device has a user feedback mechanism that the key-logger can track, the 3D problem can be reduced to a 2D problem since the pressing event can be detected by other means.

Time-of-Flight (ToF) sensors

An optical time-of-flight sensor measures the distance between the sensor and an object. It is based on the time difference between the emission of light and its return to the sensor after being reflected by an object. Some sensors emit a short pulse towards the object and measure the time it takes for the light to return. Others emit modulated light toward the object and measure the phase delay of the returning light. Simple time-of-flight sensors are comprised of a laser source and a single receiver. More sophisticated sensors are comprised of an array of receivers and are considered as 3D time-of-flight cameras. Arrays of 320 × 240 pixels are commercially available while products having bigger arrays (e.g. Teledyne e2v [11]) and higher depth resolution (e.g. Li et al. [12]) are being developed.

Body key-logger "proof-of-concept"

To explore the body key-logging approach I built a body key-logger. The target device I chose was a safe with a keypad (Yale YSV/200/DB1 Electronic Safe, EAN: 5010609182200). The safe's keypad is shown in Figure 1. To open the safe, using the keypad, a user is required to perform the following tasks:

- Enter the numeric code, digit by digit, by pressing the numeric keys of the keypad. Upon each successful keystroke, the device makes a noticeable sound and lights an indicator to indicate a numeric keypress.
- Press one of two "code entered" keys – either the "Enter" key or the "Key" key. Upon a successful keystroke, the device makes a noticeable sound and lights an indicator to indicate a successful or unsuccessful code entry.

- Rotate and pull a handle to open the safe door (assuming the code entry was successful).

Figure 1: Safe's Keypad

The vulnerabilities I decided to exploit in the user–device interface were:

- Each key has a fixed position
- Each key has a fixed function
- Audio feedback indicates a successful key press
- After entering a personal code, the user is forced to press either the “Enter” key or the “Key” key.

The circuit I designed is shown in Figure 2. It is comprised of a line of optical time-of-flight sensors. When scanned periodically, the line of sensors creates a detection plane that is used to track the horizontal movement of the key pressing finger in front of the keypad. The design assumes that the user is pressing each key with a single finger and that the remainder of the fingers are held in a fist, which does not change from one key press to another. Two properties are read from each sensor: the measured distance to the user’s finger and return signal rate.

Figure 2: Body key-logger circuit

The circuit is also comprised of a microphone which is sampled periodically to detect successful key press events. Other major components are an STM32F303K8T6 microcontroller, an ambient light sensor and an IR LED. The microcontroller executes the body key-logger software. To save on battery power, it is assumed that the safe is not exposed to light when it's not in use (e.g. the safe is installed in a drawer or a closet). The ambient light sensor is used to detect the decrease in ambient light (keypad not in use) or its increase (keypad in use) and to set the power consumption mode of the key-logger accordingly. The IR LED is used to transmit the logged key presses to an external terminal, upon request, using IR light.

The key-logger device was designed to be disguised as a magnet or a sticker, as shown in Figure 3. It could have been designed to be deployed in other forms (e.g. placed on a wall next to the safe).

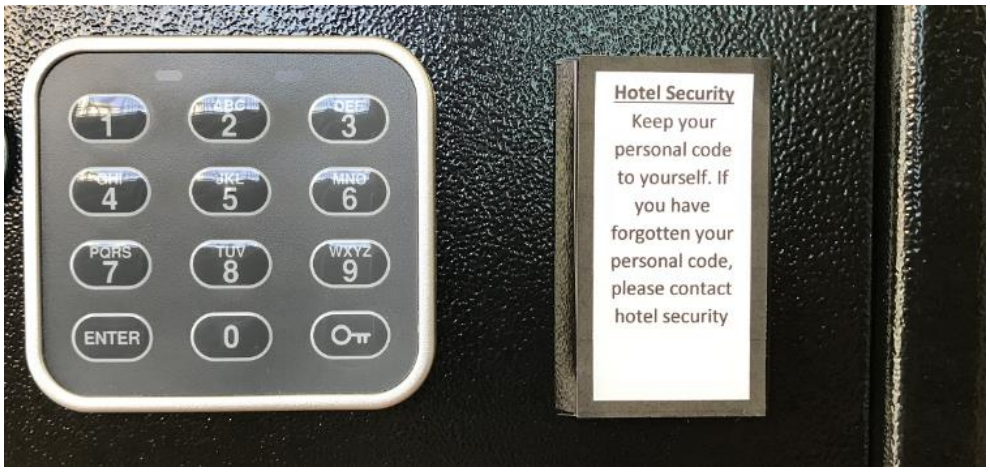

Figure 3: Body key-logger deployment

When not in sleep mode, the software scans the time-of-flight sensors waiting for object detection. When the victim's finger enters the detection plane, the software stores detection data records in a buffer until a "successful key press" audio event is detected. When the audio event is detected, the software stores the data records in the key press buffer. These records comprise the information derived from the user's finger position at the time of the "successful key press" audio event. When the attacker requests code extraction, the software performs the following steps for each key press event:

1. Finds the last data record before the audio event
2. Selects the readings with the highest return signal rate
3. Estimates the object's position on the sensors' axis using the following average:

$$\bar{X} = \sum_{i=1}^{m} \widehat{p_i} x_i$$

   $\bar{X}$ is the average position
   $\widehat{p_i}$ is the return signal rate of sensor i
   x_i is the position of sensor i
4. Calculates the range to the object by doing a linear interpolation on the range data of the two sensors closest to the estimated object position

The software then determines if the last key pressed was the "Enter" key or the "Key" key:

- If a key pressed was to the right of the last pressed key and the range from the last pressed key was larger than 2/3 of the keypad key column margin then the last pressed key most likely was the lower left key, or the “Enter” key.
- Otherwise, if a key pressed was to the left of the last pressed key, and the range from the last pressed key was larger than 2/3 of the keypad key column margin, then the last pressed key most likely was the lower right key, or the “Key” key.
- Otherwise, if the last pressed key range was beyond the distance between the detectors and the middle column of the keypad then the last pressed key most likely was the lower left key (the “Enter” key).
- Otherwise, most likely the lower right key was pressed (the “Key” key).

The last two steps solve the ambiguity problem in the case where the code is limited to a single keypad column. The two steps assume that the distance between the key-logger and the middle column of the keypad is known. A different approach can be taken by recovering keys pressed twice – once for the left column and once for the right column. In this case the attacker’s interrogation will yield two recovered codes instead of one. One of the recovered codes will be correct.

After choosing the role of the last key pressed, the software performs the following steps:

1. Finds the closest key grid to the detection grid (closeness defined as the sum of the minimum distances).
2. Determines the numeric value of each pressed key by finding the closest distance to a key at the closest key grid.
3. Transmits an encoded message via the IR LED (that is attached to the attacker’s reading device).

Proof-of-concept tests results

The "proof-of-concept" tests were mostly conducted with the key-logger placed one inch to the right of the keypad. The pointer finger was used to press the keys while the rest of

the fingers were clenched. The tests were performed using both left and right hands and similar results were obtained. An example of key position recovery is shown in Figure 4.

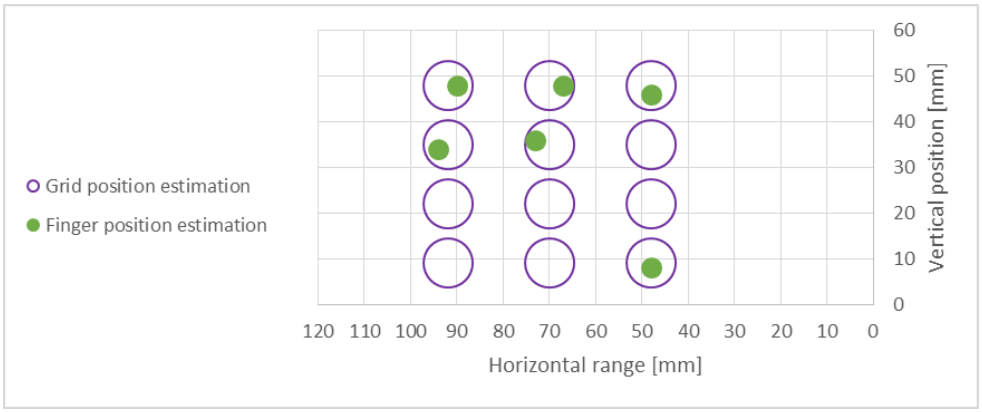

Figure 4: Finger position estimation example for "1-2-3-4-5-Key" code

An example of key position recovery and matching return signal rate is shown in Figure 5 and Figure 6.

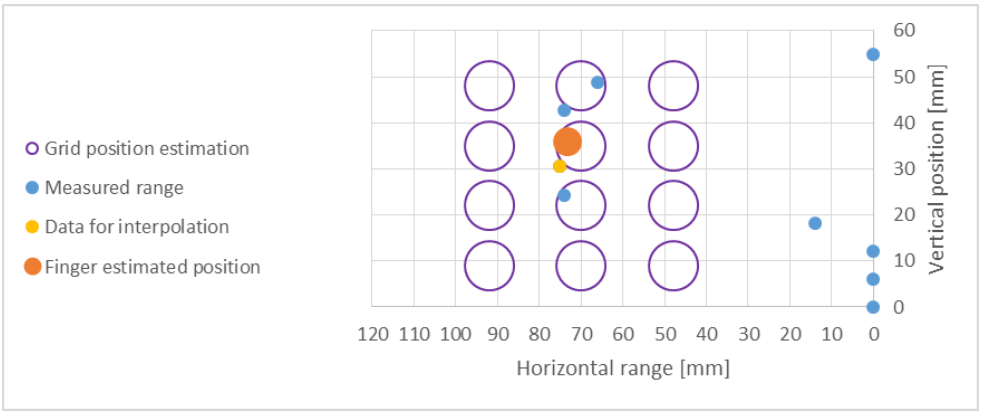

Figure 5: An example of Finger position estimates for the "5" key

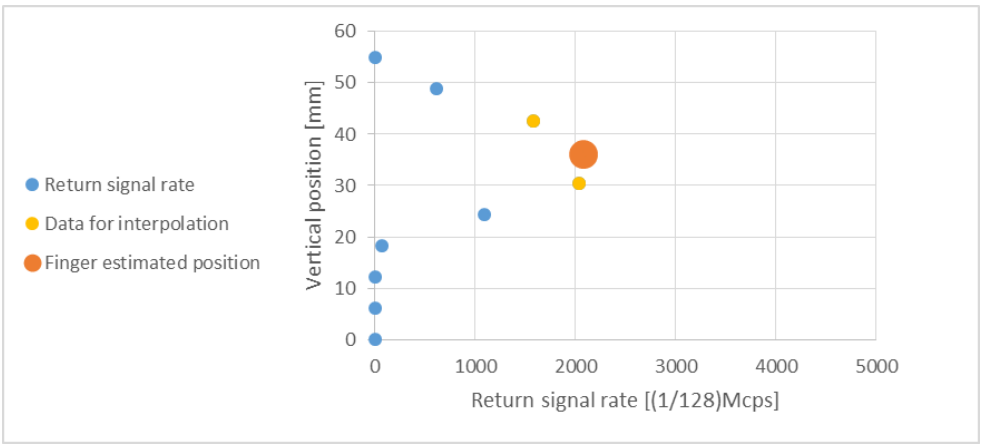

Figure 6: An example of Return signal rate for the "5" key

Each sensor used in the device was comprised of a light source with a 25 degree illumination cone. To avoid the keypad's frame detection, the sensors were tilted, as show in Figure 7. The wide illumination cone causes side detections in the horizontal plane, shown in Figure 5 and Figure 6. Since only a single key is pressed at a time, it is relatively easy to recover the physical location of the finger. On the vertical plane, the wide illumination cone influences the ability to detect the pointing finger. When the finger is short or when the finger is not perpendicular to the keypad the side detections reflects the side view of the fist. By blocking the upper and lower parts of the lens of the light source, the angle of the illumination cone was reduced and the probability of successful detection was improved.

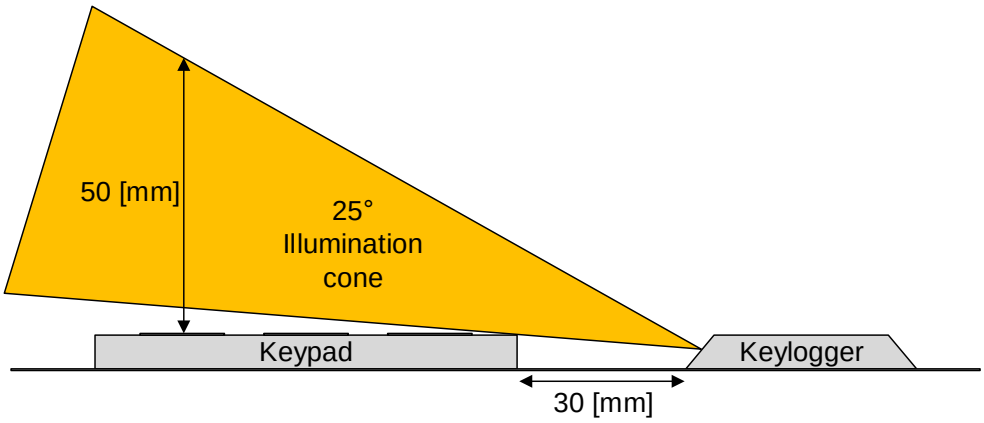

Figure 7: Time-of-flight illumination cone

I conducted tests to evaluate the probability of successful keystroke detection. The tests were performed by 7 people, each entering the following codes: 1-2-3-4-5-Key and 1-2-3-4-5-Enter, in an alternating manner. In every test, the codes were entered 25 times (a total of 150 key presses). The average probability of successful detection was 92%. Tests results per subject can be seen in Table 1.

Subject Number	Hand	Successful detection probability [%]
1	Right	100
1	Left	100
2	Right	96
3	Right	93
4	Right	92
5	Right	86
6	Right	85
7	Left	83

Table 1: Successful probabilities of recovery test results

Battery consumption

Based on the current consumption of the circuit, battery capacity and circuit activity period per day, the battery time was calculated. Calculated battery time vs activity period per day is shown in Figure 8.

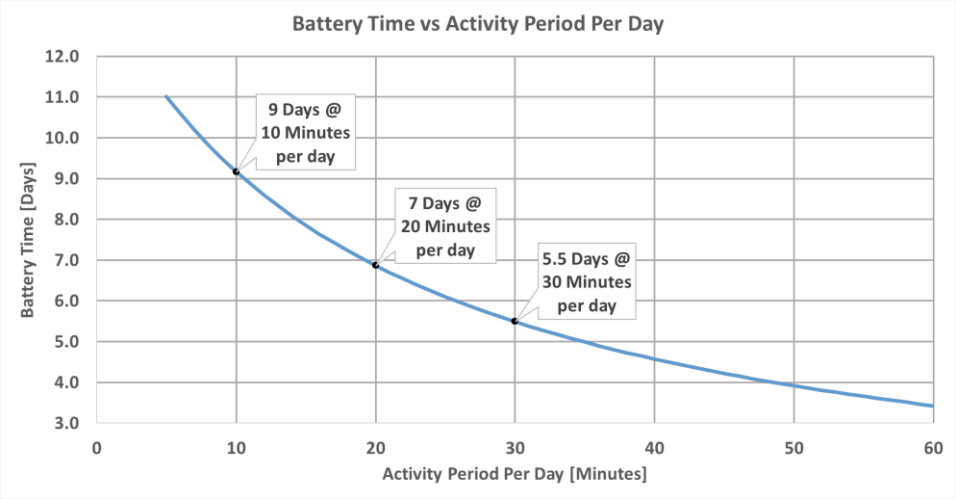

Figure 8: Battery time vs activity period per day

Discussion

Common keyboards and keypads have rigid user interfaces, making it easy to extract keystrokes by following the body movements of the user and correlating the data to the key layout. This would have been harder to do if the user interface was not rigid. Touch screens as well can be used to achieve this goal if, at each iteration, the layout changes. An example of an arbitrary keypad layout is shown in Figure 9.

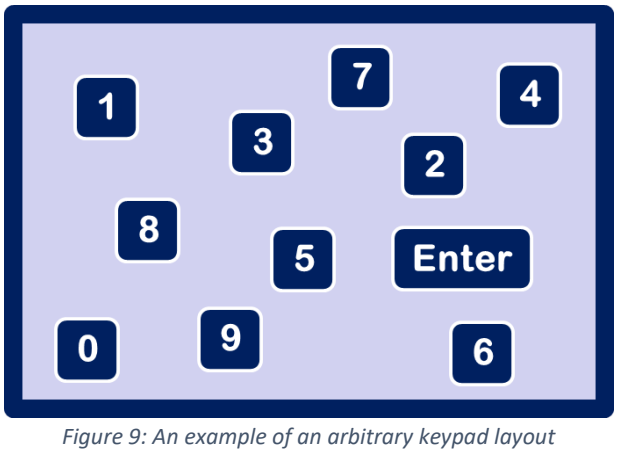

Figure 9: An example of an arbitrary keypad layout

Snyder et al. [13] show that skilled typists' explicit knowledge of the key locations is incomplete and inaccurate. This emphasizes the importance of the key layout. To improve the user's ability to remember the code, I suggest that graphic signs other than a numeric keypad keys be used. Intelligent Environments [14] suggests replacing numeric PIN codes with emoji codes. Other graphic signs that could be used are colors, letters, icons, emoticons, etc. Audio feedback is relatively easy to detect and exploit to improve the probability of key detection. It may be replaced with a narrow field of view visual sign that is visible only to the user.

Future directions

The device used for the "proof-of-concept" can be improved in several ways. The sensor positioning and the data processing algorithm can be improved to reduce the device's physical dimensions. The tracking approach can also be changed. One approach could track the side view of the hand, instead of tracking the finger. A different approach can track the wrist or the forearm.

3D time-of-flight cameras should be explored as they offer a wider range of tracking options. They may also increase the physical range at which the key-logger is deployed.

Conclusions

In this paper, I propose a new key-logging approach, targeting the interaction between a victim and a device on the victim's side of the interaction. To explore the concept, I introduced a body key-logging device based on an array of time-of-flight sensors, exploiting the vulnerabilities found in the interaction between a user and a keypad-protected safe. Both the key-logger hardware and the algorithms were discussed and the tests results were presented. Finally, suggestions were made on how user interfaces can be improved in order to foil similar attacks. Future directions were suggested.

On the web

Body Keylogging device demonstration:

https://www.youtube.com/watch?v=qbZf4IQH6WQ

Body Keylogging device late code extraction demonstration:

https://www.youtube.com/watch?v=VXioBVI60LE

Data analysis demonstration of the body keylogging device:

https://www.youtube.com/watch?v=sp5XtLMGNyU

General keypad finger tracking demonstration using the boy keylogging device:

https://www.youtube.com/watch?v=KZPcYwh50-E

Body Keylogging website:

https://bodykeylogger.wixsite.com/bodykeylogging

Acknowledgements

I would like to thank my wife and our children for supporting my endeavor. I would also like to thank you, the reader, for your interest in my work.

References

[1] T. Olzak, "Keystroke Logging (Keylogging)", 2008.
[2] R. Creutzburg, "The strange world of keyloggers - an overview, Part I" Electron. Imaging, vol. 2017, no. 6, pp. 139–148, 2017.
[3] Snopes.com, "ATM camera", https://www.snopes.com/fact-check/atm-camera/.
[4] F. Maggi, A. Volpatto, S. Gasparini, B. Simone, G. Boracchi, S. Zanero, "A fast eavesdropping attack against touchscreens", 7th International Conference on Information Assurance and Security, IEEE, 2011.
[5] O. Oke Alice, A. Adigun Adebisi, S. Falohun Adeleye, F.O. Alamu, "Development of a Programmable Electronic Digital Code lock system", International Journal of Computer and Information Technology, Volume 02– Issue 01, 2013.
[6] M. B. Lawan, Y. A. Samaila, I. Tijjani, "Microcontroller Based Electronic Digital Lock with Security Notification", Journal of Engineering Research and Reports, Vol.: 2, Issue: 3, 2018.
[7] Sargent & Greenleaf Inc, "Easy View/Tamper Resistant Keypad for Comptronic® Locks Installation Instructions", Document part number: 630-614, Revised 04/13/2006.
[8] Nortek Security & Control, "212iLW & 242iLW Standalone Keypad Installation & Programming Manual", Document number: 6-050700 X2, 2015.
[9] Plore, "Side-channel Attacks on High-security Electronic Safe Locks", DEF CON 24, 2016.
[10] I. Martinovic, D. Davies, M.Frank, D. Perito, T. Ros, D. Song, "On the Feasibility of Side-Channel Attacks with Brain-Computer Interfaces," USENIX Security Symposium Bellevue, WA, 2012.
[11] Teledyne e2v, "1.3MP BORA CMOS SENSOR", https://www.e2v.com/products/imaging/cmos-image-sensors/bora-1-3-time-of-flight-sensor/
[12] F. Li, F. Willomitzer, P. Rangarajan, M. Gupta, A. Velten, O. Cossairt, "SH-ToF: Micro resolution time-of-flight imaging with superheterodyne interferometry". ICCP 2018, 2018.
[13] K. M. Snyder, Y. Ashitaka, H. Shimada, J. E. Ulrich, & G. D. Logan. "What skilled typists don't know about the QWERTY keyboard". Attention, Perception, & Psychophysics, 76, 162–171. 2014.
[14] Intelligent Environments, "Now You Can Log Into Your Bank Using Emoji," Jun. 2015, http://www.intelligentenvironments.com/info-centre/press-releases/now-you-can-log-into-your-bank-using-emoji-1.