Side-Channel Attacks on Mobile and Wearable Systems

Ani Nahapetian
Computer Science Department
California State University, Northridge (CSUN)
Los Angeles, USA
ani@csun.edu

***Abstract*— This paper describes a variety of side-channel attacks on mobile and wearable computing systems, exposing vulnerabilities in their system and software architectures. Specifically addressed are malware approaches that passively leverage sensors on-board the systems to monitor user information for sensitive information retrieval. Some potential countermeasures at the system and user interface level are provided.**

Keywords—Mobile security, mobile computing, wearable computing, side-channel attacks, keystroke inference.

I. INTRODUCTION

With the proliferation of mobile devices, including smart phones, in the marketplace, the opportunities and the benefits of software attacks targeting these devices is growing. Wearable devices, with examples including smart watches, Fitbits, and Google Glass, form the most recent wave in the mobile explosion and open a new set of security vulnerabilities.

Information on mobile devices, specifically smart phones, makes them uniquely attractive targets. Mobile devices commonly have access to personal information (including saved passwords, login information, and email access), location information (including GPS data, daily schedules), and now with wearable computers physiological information (such as heart rate, ECG).

Mobile devices are equipped with a large number of high-end sensors. The fusion and processing of data from these sensors has been demonstrated to allow a great deal of user information to be inferred, both purposefully and maliciously. Moreover, many of these sensors, notably the motion sensors, lack any explicit access control mechanism, with requirements for apps to have permissions to access their data.

With broadband internet access, transferring data to a malicious agent in the cloud is feasible. The data capture and transfer is not likely to have a demonstrated drain on system resources such as memory and power, making it less likely for a user to notice the intrusion. As computation can be easily passed on to a remote server, device processing power or memory limitations due not deter or prevent the attack.

Post manufacturer software downloads from software marketplaces, such as iTunes, Google Play, and 3rd party marketplaces provide opportunities for malware to gain access to the devices. These apps, typically impersonating legitimate apps, can gain access to device sensors. Determining the app is using the data for malicious purposes is not readily possible, as the data can be captured for legitimate reasons and then used also for attacks. For example, motion data can be captured, relayed, and used for a fitness app, but then the same data can also be used for password inference. These third party apps, even if reviewed by the app marketplaces, do not provide information about how the captured data can be further leveraged for security breaches.

This paper examines side-channel attacks on mobile and wearable systems, which leverage data captured passively and non-intrusively from sensors on-board the mobile devices.

Unlike the well-known smudge attack [2] which requires possession of the smart phone to maneuver under the appropriate lighting conditions, all of the approaches considered here assume the smart device is not removed from the user's possession.

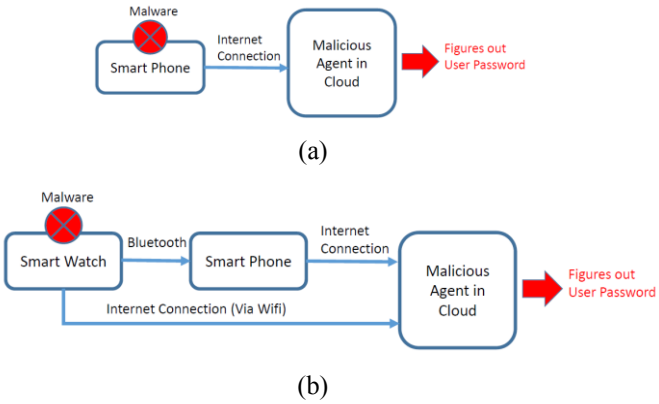

Fig. 1. Attack overview (a) in smart phones (b) in smart watches (or other paired wearable devices. Malware installed on the smart device accesses the on-board sensors and transmits the data to a malicious agent in the cloud which determines the user information.

Authorized licensed use limited to: University of Florida. Downloaded on November 26,2024 at 21:48:26 UTC from IEEE Xplore. Restrictions apply.

978-1-4673-9292-1/16/$31.00 ©2016 IEEE

The attacks considered leverage physical phenomena associated with private data entry (such as smart device movement) to determine what data is entered or produced by the user. The attacks considered included capturing data from the attacked device directly or using it to capture information from an adjacent computer, a paired smart phone, or the user.

Once side-channel data is retrieved from the sensors, it is transmitted to a malicious agent in the cloud via the internet connection of the device itself or a paired device (as is common with wearable devices such as smart watches). The computation is carried out in the cloud to uncover the private user information and complete the attack. Figure 1 provides an overview of the attack components, for both smart phone and wearable systems.

II. Attack Model

Side channel attacks on mobile and wearable systems involve malware accessing sensor data, such as the gyroscope and accelerometer, potentially filtering the data, then transmitting it to a remote server. Computation on the remote server learns features of the data with the goal of determining private information about the device's users. For example, by using the accelerometer and gyroscope data, keystrokes can be inferred, enabling the recovery of password information.

To apply many established learning algorithms to the data, training data is required. This training data can be obtained for a general user audience, and applied to individual user being attacked. However, more potently, the user can be tricked into installing malware on the device, which collects personalized training data. This malware can impersonate a game, as with TapLogger [7], where the user types or swipes, with the location of the types or swipes being recorded for the purposes of labelling the motion data.

A. Data Extracted

The side-channel attacks aim to capture private sensitive information, such as passwords, via surreptitious access to device sensors. The attacker's wish list includes:

- Personal Identification Numbers (PINs);
- Passwords;
- Patterns for unlocking smart phones;
- Entered text;
- Surrounding private information, such as conversations or neighboring computing activity.

Attackers are eager to capture password or numeric passwords (known as PINs). For unlocking phones, capturing the pattern that a user enters in the locked screen is valuable. Additionally, information entered, such as Google search terms, is desirable. Note that text, including words and sentences, is easier to infer, as the inherent redundancies of language and its constructs help to deal with the errors in measurement or classification.

In addition to capturing information entered onto a smart phone, information such as neighboring computing activity or conversations near a smart device have been demonstrated.

B. Leveraged Sensors

Smart devices are equipped with a plethora of sensors. Examples include:

- Location sensors, such as GPS, proximity;
- Motion sensors, such as accelerometers, gyroscopes, magnetometers;
- Environmental sensors, for such information as ambient light, temperature, barometer;
- Biometric sensors for wearable systems, providing heart rate, ECG;
- Audio and video sensors, namely cameras and microphones.

Attacks leveraging mobile device cameras or microphones has been demonstrated. However, these sensors are protected with permissions that users are informed of and are required to enable for an app to gain access. The motion and environmental sensors, however, do not have an access control mechanism [1], making them more likely targets.

C. Attach Components

Most of the machine learning approaches taken for keystroke classification have been supervised. As a result, these approaches require the collection of labelled training data. To collect the labelled training data for the supervised learning approaches, the attacks rely on the user downloading a malicious app that captures screen tap locations in conjunction with paired motion data. Thus providing labels to the captured motion data.

These apps, possibly impersonating as a game, require the user to tap at all points of the screen that match up to keyboard character or PIN pad digit locations. Then with both the actual keystroke locations captured directly from the game and the motion data captured surreptitiously by the malware, the paired data can be used to train the classifier used by the malicious agent. This is summarized in Figure 2.

This approach is a one-to-one classification. A one-to-many classification, although by its nature less accurate, does not require the training data to come from the user. Instead non-user data is used to train the classifier. In this case, the malware only need to passively observe the sensor data and transmit it to the cloud. The app can be impersonating as a fitness app, accessing the motion data and transmitting it onwards.

Authorized licensed use limited to: University of Florida. Downloaded on November 26,2024 at 21:48:26 UTC from IEEE Xplore. Restrictions apply.

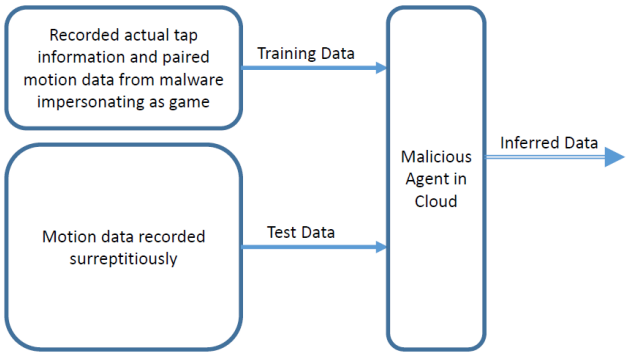

Fig. 2. Using a decoy app, the attacker can capture individual labelled data to train the classifier for each user.

D. Event versus Keystrock Classification

When an attack is classifying keystrokes on a smart phone, there are two classification challenges. The first challenge is to classify the keystrokes. In the case of PIN inference, differentiating a '1' keystroke from a '2' keystroke for example. However, there is also a second challenge, differentiating a non-tap event and a tap event to determine when a tap event is taking place. A considered approach groups non-tap events and taps events together and then carries the classification as the same time.

III. Mobile Device Side-Channel Attacks

In this section, we consider an array of passive side-channel attacks that leverage smart phone sensors. Previous work has examined the leakage of sensitive data via access to the device camera looking at the movements of the phone during taps [3], via access to location information [4], and via access to the microphone, camera, and GPS [5].

The movement of the smart phone as the user taps the smart phone screen has become an especially rich area of attack. TouchLogger used motion and orientation of the gyroscope to infer PINs [6]. TapLogger leveraged accelerometer and gyroscope data for PIN inference [7]. Cai et al examined how these results translate to a variety of devices, with different dimensions [8]. TapPrints, using the accelerometer and the gyroscope, was able to determine letters typed on the smart phone QWETRY keyboard [9]. Using only the accelerometer, ACCessory demonstrated that six-character passwords entered on the phone could be inferred [10]. Aviv et al examine PIN and pattern entry for unlocking using smart phone accelerometers [11].

IV. Mobile Devices Comprimised to Eavesdrop

In addition to capturing sensitive information entered onto a smart phone, attacks using smart phones sensors to eavesdrop on users have been considered.

Smart phones are pervasive in user lives, and so even when using a computer, a user is likely to place the smart phone next to the computer they are using, potentially compromising the data entered onto the computer.

Researchers have examined the success of recovering adjacent keyboard activity using the smart phone's microphone [12] and accelerometer [13].

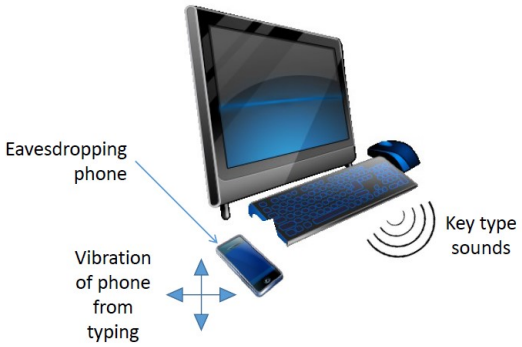

Fig. 3. Smart phone used as an eavsedropping device for nearby computing activity.

Figure 3 provide an overview of this attack set-up. The user is typing at a computer keyboard, and has placed his/her smart phone on the table next to the keyboard. With vibrations of the smart phone caused by the typing movements, search keywords could be extracted. Similarly, the sounds of the typing was used to infer the text being typed.

Stanford researchers have demonstrated that conversations carried out near smart phones can be captured and identified using only the smart phones gyroscope [14].

These attacks capture motion sensor data, use well-known classifiers, and benefit from language constructs whose inherent redundancy allows erroneous classifications to be identified and disregarded.

V. Side-Channel Attacks on Smart Watches

The majority of wearable devices sales are in wrist-worn technology, including fitness monitors and smart watches, with smart watches becoming a common utility for users.

Using custom hardware, the data from wrist-worn sensors has been shown to successfully classify smoking gestures [15] and eating gestures [16]. Motion sensor data from smart phones held in hand has been shown to provide enough information to impersonate a stylus [17] and a mouse [18]. Using smart watches, gesture classification using accelerometer and gyroscope data has also been examined [19].

Very recent research has demonstrated that malware accessing smart watches motion sensors can be used to capture sensitive information, such as PINs entered on smart phones [20][21]. Additionally, text entered on a computer keyboard can be inferred using the smart watch motion sensors [22].

Smart watches are especially susceptible to side-channel attacks, as they house powerful sensors and they are worn continuously. The only consolation is that only the non-

Authorized licensed use limited to: University of Florida. Downloaded on November 26,2024 at 21:48:26 UTC from IEEE Xplore. Restrictions apply.

dominant hand information can be directly captured. However, as is the case with [21] and [22], dominant hand information can be extrapolated.

VI. Countermearsures

To combat the array of side channel attacks presented in this paper, the following countermeasures can be considered.

At the operating system level, the flow of sensitive data can be tracked with approaches such as TaintDroid [4]. A complementary approach would involve limiting access to sensors. Just as access control mechanisms are in place for the cameras and the microphone, they can be added for the motion sensors. Limiting access to the sensors during private data entry is another related approach. Relaying data access events with an LED is yet another approach to alert the user.

At the user interface level, modifying the keyboard or the number pad during private data entry (such as passwords) can counteract most of the presented attacks. The user is burdened with a lengthier password entry process, as they have to search for the characters to enter. However, in this way nearly all the side-channel attacks presented in this paper are thwarted.

Finally, passwords can be securely saved in memory and relayed only after some alternative form of authentication is completed. Smart phone sensor data has also been used for authentication via differences in tap timing [23] and bioimpedence [24]. Examples of mobile biometric authentication include the use of gait [25] [26], walking patterns [27], and gaze [28].

VII. Conclusion

With the ubiquity of mobile and now wearable devices; and the increased value of the data entered on these devices, the potential for attack is apparent. This paper presented an array of side-channel attacks that leverage mobile devices sensor data to determine private user information.

References

[1] Android Developer: Security Permissions, http://developer.android.com/guide/topics/security/permissions.html.

[2] Adam J. Aviv, Katherine Gibson, Evan Mossop, Matt Blaze, and Jonathan M. Smith. 2010. Smudge attacks on smartphone touch screens. In *Proceedings of the 4th USENIX conference on Offensive technologies* (WOOT'10). USENIX Association, Berkeley, CA, USA, 1-7.

[3] Michael Backes , Markus Dürmuth , Dominique Unruh, Compromising Reflections-or-How to Read LCD Monitors around the Corner, Proceedings of the 2008 IEEE Symposium on Security and Privacy, p.158-169, May 18-21, 2008.

[4] William Enck , Peter Gilbert , Byung-Gon Chun , Landon P. Cox , Jaeyeon Jung , Patrick McDaniel , Anmol N. Sheth, TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones, Proceedings of the 9th USENIX conference on Operating systems design and implementation, p.1-6, October 04-06, 2010, Vancouver, BC, Canada.

[5] Liang Cai, Sridhar Machiraju, and Hao Chen. 2009. Defending against sensor-sniffing attacks on mobile phones. In *Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds* (MobiHeld '09). ACM, New York, NY, USA, 31-36.

[6] Liang Cai and Hao Chen. 2011. TouchLogger: inferring keystrokes on touch screen from smartphone motion. In Proceedings of the 6th USENIX conference on Hot topics in security (HotSec'11). USENIX Association, Berkeley, CA, USA, 9-9.Forman, G. 2003.

[7] Zhi Xu, Kun Bai, and Sencun Zhu. 2012. TapLogger: inferring user inputs on smartphone touchscreens using on-board motion sensors. In *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks* (WISEC '12). ACM, New York, NY, USA, 113-124.

[8] Liang Cai and Hao Chen. 2012. On the practicality of motion based keystroke inference attack. In *Proceedings of the 5th international conference on Trust and Trustworthy Computing* (TRUST'12), Stefan Katzenbeisser, Edgar Weippl, L. Jean Camp, Melanie Volkamer, and Mike Reiter (Eds.). Springer-Verlag, Berlin, Heidelberg, 273-290.

[9] Emiliano Miluzzo, Alexander Varshavsky, Suhrid Balakrishnan, and Romit Roy Choudhury. 2012. Tapprints: your finger taps have fingerprints. In *Proceedings of the 10th international conference on Mobile systems, applications, and services* (MobiSys '12). ACM, New York, NY, USA, 323-336.

[10] Emmanuel Owusu, Jun Han, Sauvik Das, Adrian Perrig, and Joy Zhang. 2012. ACCessory: password inference using accelerometers on smartphones. In *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications* (HotMobile '12). ACM, New York, NY, USA, , Article 9 , 6 pages.

[11] Adam J. Aviv, Benjamin Sapp, Matt Blaze, and Jonathan M. Smith. 2012. Practicality of accelerometer side channels on smartphones. In *Proceedings of the 28th Annual Computer Security Applications Conference* (ACSAC '12). ACM, New York, NY, USA, 41-50.

[12] Michael Backes, Markus Dürmuth, Sebastian Gerling, Manfred Pinkal, Caroline Sporleder, Acoustic side-channel attacks on printers, *Proceedings of the 19th USENIX conference on Security*, August 11-13, 2010, Washington, DC.

[13] Philip Marquardt, Arunabh Verma, Henry Carter, and Patrick Traynor. 2011. (sp)iPhone: decoding vibrations from nearby keyboards using mobile phone accelerometers. In *Proceedings of the 18th ACM conference on Computer and communications security (CCS '11)*.

[14] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. 2014. Gyrophone: recognizing speech from gyroscope signals. In *Proceedings of the 23rd USENIX conference on Security Symposium* (SEC'14). USENIX Association, Berkeley, CA, USA, 1053-1067.

[15] Abhinav Parate, Meng-Chieh Chiu, Chaniel Chadowitz, Deepak Ganesan, and Evangelos Kalogerakis. 2014. RisQ: recognizing smoking gestures with inertial sensors on a wristband. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services* (MobiSys '14). ACM, New York, NY, USA, 149-161.

[16] Y. Dong, A. Hoover, J. Scisco, and E. Muth. A new method for measuring meal intake in humans via automated wrist motion tracking. Applied psychophysiology and biofeedback, 37(3):205--215, 2012.

[17] Sandip Agrawal, Ionut Constandache, Shravan Gaonkar, Romit Roy Choudhury, Kevin Caves, and Frank DeRuyter. 2011. Using mobile phones to write in air. In *Proceedings of the 9th international conference on Mobile systems, applications, and services* (MobiSys '11). ACM, New Yo, NY, USA, 15-28.

[18] Sangki Yun, Yi-Chao Chen, and Lili Qiu. 2015. Turning a Mobile Device into a Mouse in the Air. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services* (MobiSys '15). ACM, New York, NY, USA, 15-29.

[19] Danial Moazen, Seyed Sajjadi Ani Nahapetian. AirDraw: Leveraging Smart Watch Motion Sensors for Mobile Human Computer Interactions. *IEEE Conference Consumer Communications and Networking (CCNC)*, January 2016.

Authorized licensed use limited to: University of Florida. Downloaded on November 26,2024 at 21:48:26 UTC from IEEE Xplore. Restrictions apply.

[20] Allen Sarkisyan, Ryan Debbiny, Ani Nahapetian. WristSnoop: Smartphone PINs Prediction using Smartwatch Motion Sensors. *IEEE Workshop on Information Forensics and Security (WIFS)*, November 2015.

[21] Anindya Maiti, Murtuza Jadliwala, Jibo He, Igor Bilogrevic. (Smart)watch your taps: side-channel keystroke inference attacks using smartwatches. In *Proceedings of the ACM Internationl Symposium on Wearable Computing (ISWC)*, Septmeber 2015, 27-30.

[22] He Wang, Ted Tsung-Te Lai, and Romit Roy Choudhury. 2015. MoLe: Motion Leaks through Smartwatch Sensors. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking* (MobiCom '15). ACM, New York, NY, USA, 155-166.

[23] K. Majdanik, C. Giuffrida, M. Conti, H. Bos,"I Sensed It Was You: Authenticating Mobile Users with Sensor-enhanced Keystroke Dynamics.", *In Proceedings of the 11th Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA 2014)*, July 2014.

[24] Cory Cornelius, Jacob Sorber, Ronald Peterson, Joe Skinner, Ryan Halter, and David Kotz. 2012. Who wears me? bioimpedance as a passive biometric. In *Proceedings of the 3rd USENIX conference on Health Security and Privacy* (HealthSec'12). USENIX Association, Berkeley, CA, USA, 4-4.

[25] Hong Lu, Jonathan Huang, Tanwistha Saha, and Lama Nachman. 2014. Unobtrusive gait verification for mobile phones. In *Proceedings of the 2014 ACM International Symposium on Wearable Computers* (ISWC '14). ACM, New York, NY, USA, 91-98.

[26] Muhammad Muaaz and René Mayrhofer. 2014. Orientation Independent Cell Phone Based Gait Authentication. In *Proceedings of the 12th International Conference on Advances in Mobile Computing and Multimedia* (MoMM '14). ACM, New York, NY, USA, 161-164.

[27] Pierluigi Casale, Oriol Pujol, and Petia Radeva. 2012. Personalization and user verification in wearable systems using biometric walking patterns. *Personal Ubiquitous Comput.* 16, 5 (June 2012), 563-580.

[28] Manu Kumar, Tal Garfinkel, Dan Boneh, and Terry Winograd. 2007. Reducing shoulder-surfing by using gaze-based password entry. In *Proceedings of the 3rd symposium on Usable privacy and security* (SOUPS '07). ACM, New York, NY, USA, 13-19.

Authorized licensed use limited to: University of Florida. Downloaded on November 26,2024 at 21:48:26 UTC from IEEE Xplore. Restrictions apply.