9-1-2019

Using vibrations from a SmartRing as an out-of-band channel for sharing secret keys

Sougata Sen
Dartmouth College, Sougata.Sen@dartmouth.edu

Varun Mishra

David Kotz
Dartmouth College, David.F.Kotz@Dartmouth.EDU

Poster: Using Vibrations from a SmartRing as an Out-of-band Channel for Sharing Secret Keys

Sougata Sen
Dartmouth College
Sougata.Sen@Dartmouth.edu

Varun Mishra
Dartmouth College
Varun.Mishra.GR@Dartmouth.edu

David Kotz
Dartmouth College
David.F.Kotz@Dartmouth.edu

ABSTRACT

With the rapid growth in the number of Internet of Things (IoT) devices with wireless communication capabilities, and sensitive information collection capabilities, it is becoming increasingly necessary to ensure that these devices communicate securely with only authorized devices. A major requirement of this secure communication is to ensure that both the devices share a secret, which can be used for secure pairing and encrypted communication. Manually imparting this secret to these devices becomes an unnecessary overhead, especially when the device interaction is transient. In this work, we empirically investigate the possibility of using an out-of-band communication channel – vibration, generated by a custom *smartRing* – to share a secret with a compatible IoT device. Through a user study with 12 participants we show that in the best case we can exchange 85.9% messages successfully. Our technique demonstrates the possibility of sharing messages accurately, quickly and securely as compared to several existing techniques.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; *Mobile devices*; • **Security and privacy**;

KEYWORDS

smartRing; vibration; security; wearables; IoT

ACM Reference Format:
Sougata Sen, Varun Mishra, and David Kotz. 2019. Poster: Using Vibrations from a SmartRing as an Out-of-band Channel for Sharing Secret Keys. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2019 International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct), September 9–13, 2019, London, United Kingdom.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3341162.3343818

UbiComp/ISWC '19 Adjunct, September 9–13, 2019, London, United Kingdom

ACM ISBN 978-1-4503-6869-8/19/09.
https://doi.org/10.1145/3341162.3343818

1 INTRODUCTION

Over the last few years, there has been a tremendous growth in the number of deployed IoT devices with sensing and data transmitting capability – e.g., a wearable Continuous Glucose Monitors (CGM) can collect a user's blood-glucose level and wirelessly transfer the information to an individual's personal device such as a smartphone or a smartwatch or any other device that represents the user. Data acquired by these devices can sometimes reveal sensitive information. For example, blood glucose information can reveal an individual's dietary pattern. Furthermore, some IoT devices (e.g., a smart blood pressure monitor or a body-weight scale) can exist in a home and can be shared by members of a household. A user's interaction with such devices can be short-lived and does not justify permanent pairing between the IoT device and each user's personal device. Thus, it is important for these data-transmitting IoT devices to know who is *using* it, and for the personal device to know which IoT device is currently being used. To ensure that the IoT device knows who is using it, the personal device should be able to share a secret message with the IoT device. The devices can then use this secret message to bootstrap a secure channel that they can use to exchange encrypted information. This secret sharing process should be unobtrusive to the user.

A straightforward solution to the problem is to allow an individual to explicitly pair each IoT device with the legitimate personal device using a PIN code or similar technique. This manual key-exchange process, however, can be cumbersome for the individual, especially when the number of devices an individual interacts with is large and the interaction with these devices is short-lived. For example, when an healthcare practitioner needs to access real-time data from every visiting patient's CGM.

To make this temporary key exchange convenient, yet secure, we seek an answer for the question – *how can a smart IoT device quickly, securely and unobtrusively receive a secret from the individual who is interacting with it?* Although there are various styles in which an individual can interact with a

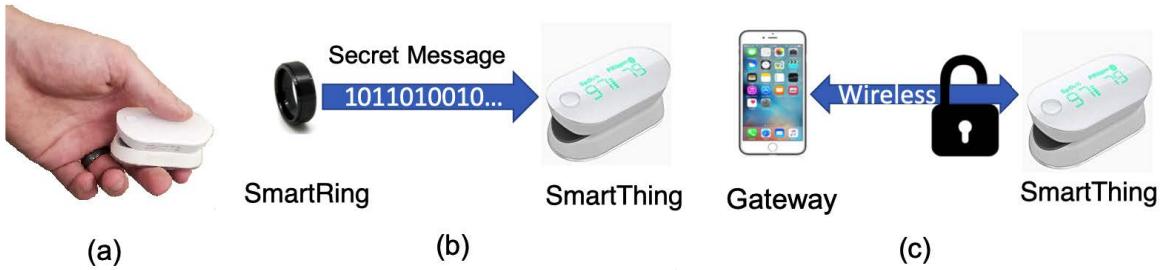


Figure 1: Operation of effortless out-of-band pairing using smartRing: (a) an individual picks a smartThing wearing a smartRing; (b) the ring automatically discovers the smartThing and shares a secret through the vibratory channel; (c) the secret is used to bootstrap a secure wireless connection between the smartThing and athe individual's personal device.

device, in this work we focus on the physical interaction of *picking up and holding* a device with the intention of using the device in its expected usage style.

To answer the question, in this paper we evaluate the possibility of using an out-of-band communication channel – vibration – generated by a smartRing to allow an individual's device to share a secret with a smart IoT object, a.k.a. *smartThing*. This secret obtained by the smartThing can eventually be used to bootstrap a secure session between the smart thing and the individual's personal device over an in-band wireless channel. We present the operation of such a system in Figure 1. The advantage of using vibration as out-of-band channel is the nature of attenuation in vibration signals: the signal decays as the distance from the vibration source increases. This attenuation (lossy property) makes it difficult for a distant adversary to gather information transferred through vibration. Since an accelerometer chip is rather inexpensive and many smartThings are equipped with an accelerometer, we use the smartThing's accelerometer to capture the secret transmitted through vibration.

An important usability requirement for such a system is to ensure that the key exchange is quick and it does not disturb the individual's natural interactions, i.e., the individual does not have to perform explicit additional actions such as bringing a vibrating smartphone in contact with the smartThing. It is intuitive that a finger-based vibration source, which is usually in close proximity to a handheld object, could be used for exchanging the secret. With the increasing interest in *smart jewelry*, especially smartRings (e.g., Motiv Ring [4]), and their constantly improving battery-life, they become an obvious candidate for the secret-sharing source. The shared secret can, in-turn, support a secure in-band wireless communication channel between the smartThing and a personal device. This entire process of discovering and securely connecting an individual's trusted device with the smartThing will be seamless to the individual as it will require no additional interaction from the individual.

Although researchers have previously explored the possibility of in-band information exchange through vibration [2, 6] or out-of-band exchange [3], as compared to prior work, our system is either faster or can work in naturalistic setting.

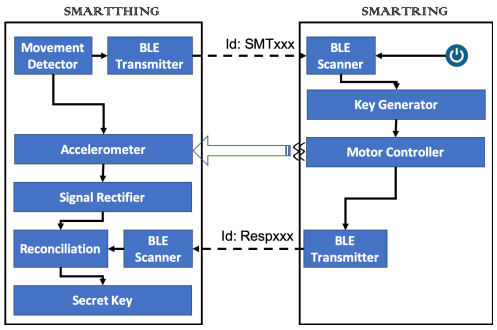
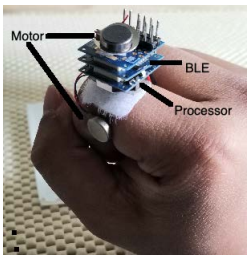


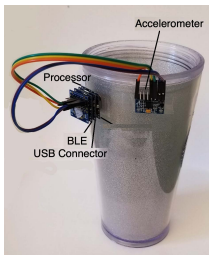
Figure 2: System overview of smartRing

2 SYSTEM OVERVIEW

Figure 2 pictorially depicts the working of the envisioned system. The system consists of a smartThing [T] and a smartRing [R]. [T] has a *pick* detection module. This module puts the components of [T] into a low-power sleep mode, until [T] is *picked up* by an individual. When [T] detects that it is being *picked up*, its BLE module starts advertising its presence. Once a user has picked up [T], the user presses a button on the [R] that the user is wearing. To make the process less obtrusive, in future, we could consider using a pick-up detection technique on [R] (e.g., as described in [5]) to initiate the key exchange. In this work, we assume that the aforementioned steps already exist. We work towards implementing the subsequent steps. At this point, [R] turns on the BLE scanner to listen for the presence of a smartThing. [R] identifies a smartThing by its name. On receiving an advertisement from [T], [R] generates a random n bit message, K. This n-bit message is transmitted by [R] in the form of vibrations. We currently use Manchester Encoding scheme for the transmission of the signal. When [T] detects the expected preamble for a vibration pattern, it extracts the subsequent n bits. Several time domain features are used



(a) The smartRing



(b) The smartThing

Figure 3: Prototype of the smartRing and smartThing board made using off-the-shelf boards for evaluation purpose

to extract the secret message. [T] also updates its BLE advertisement details to inform the smartRing that it is the smartThing which has received the vibration message. Finally to ensure that the secret has been exchanged properly, [R] and [T] perform a key reconciliation step and Hamming error correcting code for correcting up to 4 ambiguous bits. This step ensures that [T] has received the correct message and can use it to transmit its identity and other data.

Implementation

Figure 3 shows the prototype smartRing and smartThing. We used development boards available in the TinyCircuits store[1] to assemble these prototypes. The current prototypes are only for experimental and evaluation purpose; they would be much smaller and suitably sized if engineered as a product.

The smartRing: The *smartRing* consists of two coin type Eccentric Rotating Mass (ERM) vibration motors, a slightly modified Arduino Uno board with Atmega328P MCU, and a board with STMicroelectronics' BLE chipset.

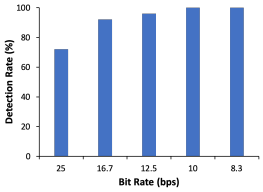
The smartThing: The *smartThing* consists of the same processor as the smartRing. The processor has a USB connector stacked onto it for transferring the accelerometer data to a computer. For RF communication, the smartThing uses the STMicroelectronics' BLE chipset. Lastly, the smartThing is connected to an external 3-axis accelerometer (MPU-6050, sampling at 500Hz) for collecting the acceleration data.

Data Collection

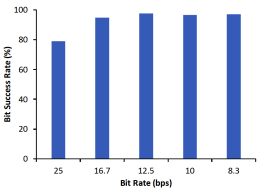
We attached the smartThing module to a hard plastic tumbler. The tumbler represents a hypothetical smartThing. We collected two sets of data for this study: a controlled study set, where we taped the smartRing to the smartThing and collected data, and a user study where we recruited 12 participants and instructed them to wear the ring prototype and pick up the smartThing.

For the controlled study, we attached the ring directly to the tumbler, 4 cm below the smartThing's accelerometer.

[1]https://tinycircuits.com/



(a) Preamble Success Rate



(b) Bit Success Rate

Figure 4: Performance of the system in the controlled study

The smartRing transmitted 10 randomly chosen unique messages, each of length 64 bits+8 bits preamble. The smartRing transmitted each message at bit rates of {8.3, 10, 12.5, 16.7, 25} bits per second (bps). For every bit rate, the smartRing transmitted each message 5 times. Thus in total we evaluate the performance of 250 messages of 72 bits and report the performance.

For the user study, we recruited 12 participants (5 males, 7 females), aged between 18 to 30 years. To understand the system's performance, each participant performed several distinct pick up gestures while holding the smartThing at various positions. We randomly selected 3 messages for this study. During every pick up gesture, the smartRing transmitted one among the three messages at {8.3, 10, 12.5, 16.7, 25} bps. Overall, 135 messages of 64 bits + 8 bits preamble were collected from each participant.

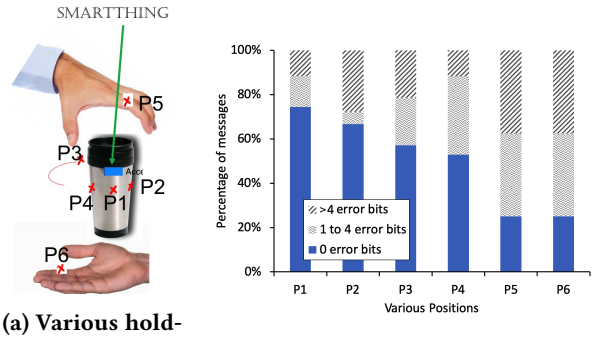
3 EVALUATION

We next evaluate the possibility of exchanging a message using vibration and effect of holding style on this exchange.

Possibility of Communicating through Vibration

To measure the reliability of sending a message from the smartRing to the smartThing through vibrations, we used the controlled study dataset.

The first step in obtaining the message is preamble detection. The preamble indicates the start of a message. Figure 4a shows the preamble detection rate at various bit rates. From the figure we can see that at 16.7 bps, we could successfully detect the preamble in 92% messages and it reaches 100% for 10 bps and 8.3 bps. It is intuitive that at higher bitrate, the bit-error rate (ratio of incorrectly received bits to total transmitted bits) will be higher. We found that at 25 bps, the number of preambles detected correctly was less than 80%. Although the bit-error rate in the preamble is much lower, we cannot detect the start of a message without detecting the preamble; the preamble-detection rate is the more relevant metric. At 12.5 bps, the preamble detection rate is 96% and the time required to transmit a 64-bit message is 5.12 seconds. Since the preamble-detection rate is reasonably high, while the time required for a 64-bit message transmission is reasonable, we use 12.5 bps as the default bit rate.



(a) Various holding positions

(b) Variation of *MER* at 12.5 bps

Figure 5: Effect of Holding Position on the *BER* and *MER*

Next, we evaluate the bit-success rate (BSR, the ratio of bits successfully received to total transmitted bits), for all the messages where the smartThing could detect the preamble. The BSR for messages where the preamble was correctly detected is presented in Figure 4b. We observe that at 12.5 bps, our system has a BSR of 97.5%, indicating that only 1.8 bits in a 64 bits message + 8 bits preamble were interpreted incorrectly. However, we observe that the bit error is not uniformly distributed across the messages. The effective message-success rate (MSR, the number of messages transferred successfully) at 12.5 bps after resolving disputed bits and applying error correcting code is 88.3%, indicating that the system could share a 64 bit secret in less than 6 seconds in over 88% instances. The time taken for smartRing to transfer the 64 bit secret is similar to the time taken to gesturally input a 7 digit PIN, as demonstrated by Ahmed et al. [1], but requires no effort or action by the user.

Effect of Holding Position on Message Transmission

We next use the user study dataset to analyze the effect of holding style on the Bit Error Rate (BER) and the Message Error Rates (MER). Figure 5a indicates the 6 approximate positions of the smartRing when the participants held the tumbler. During the study, we noticed that for positions $P1$ to $P4$, normally the user's palm and fingers wrapped around the tumbler. For $P5$, the fingers were in contact with the tumbler, while for $P6$, only the palm was in contact.

We evaluate the BER and MER for the 6 positions at various bitrates. However, due to space constraints, we present results for only a bitrate of 12.5 bps. To evaluate the BER and MER, we only consider the messages whose preamble was detected correctly. From the data we observe that for P1 to P4, the BER varies between 2.21% (at P1) and 4.93% (at P3), indicating that the system will perform well when the smartRing is near the smartThing's accelerometer.

Since the goal of the system is to share a secret from smartRing to a smartThing, we next compute the MER of the system for the 6 positions at 12.5 bps. Figure 5b presents the percentage of messages with no bit mismatch, up to 4 bit mismatch, or more than 4 bit mismatch (which includes all messages with error in preamble detection). From the figure we can see that for position P1, more than 85% messages had 4 or fewer than 4 erroneous bits and we could resolve them using the error-correcting code. From the figure we can also see that for positions where there is little contact between smartRing and the smartThing (P5 and P6), the percentage of messages with 5 or more erroneous bits is 36.1%. This indicates that for achieving low MER, the ring should be in contact with the smartThing. Although a user may need to adjust her grip to attain good contact, this effort is still lower than manually inputting a secret.

4 CONCLUSION AND FUTURE WORK

In this work we demonstrate the possibility of using vibration generated by a smartRing to share a secret with a smartThing. Through a controlled study we show that it is possible to share the secret reliably – with bit error rate of less than 2.5%. Additionally, through a user study we show that the system can successfully exchange over 85% messages in the best case. In future, we plan to evaluate the effect of bit rate and type of material on the message success rates. We also plan to evaluate various security threats (e.g., impersonation attacks or denial-of-service attacks) on such a system.

ACKNOWLEDGMENTS

This research results from a research program at the Institute for Security, Technology, and Society at Dartmouth College, supported by the National Science Foundation under award number CNS-1329686. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors.

REFERENCES

[1] Imtiaj Ahmed, Yina Ye, Sourav Bhattacharya, N. Asokan, Giulio Jacucci, Petteri Nurmi, and Sasu Tarkoma. 2015. Checksum Gestures: Continuous Gestures as an Out-of-Band Channel for Secure Pairing. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*.

[2] Inhwan Hwang, Jungchan Cho, and Songhwai Oh. 2012. Privacy-aware communication for smartphones using vibration. In *Embedded and Real-Time Computing Systems and Applications (RTCSA)*.

[3] Younghyun Kim, Woo Suk Lee, Vijay Raghunathan, Niraj K. Jha, and Anand Raghunathan. 2015. Vibration-based secure side channel for medical devices. In *Design Automation Conference (DAC)*.

[4] Motiv Inc. 2019. Motiv Ring. http://mymotiv.com

[5] Sougata Sen, Archan Misra, Vigneshwaran Subbaraju, Karan Grover, Meera Radhakrishnan, Rajesh K. Balan, and Youngki Lee. 2018. I4S: Capturing in-store shopping behavior. In *ACM International Symposium on Wearable Computers (ISWC)*.

[6] Takuro Yonezawa, Jin Nakazawa, and Hideyuki Tokuda. 2015. Vinteraction: Vibration-based information transfer for smart devices. In *International Conference on Mobile Computing and Ubiquitous Networking (ICMU)*.