RFIDs and Secret Handshakes: Defending Against Ghost-and-Leech Attacks and Unauthorized Reads with Context-Aware Communications

Alexei Czeskis
University of Washington
aczeskis@cs.washington.edu

Karl Koscher
University of Washington
supersat@cs.washington.edu

Joshua R. Smith
Intel Research Seattle
joshua.r.smith@intel.com

Tadayoshi Kohno
University of Washington
yoshi@cs.washington.edu

ABSTRACT

We tackle the problem of defending against ghost-and-leech (a.k.a. proxying, relay, or man-in-the-middle) attacks against RFID tags and other contactless cards. The approach we take — which we dub *secret handshakes* — is to incorporate gesture recognition techniques directly on the RFID tags or contactless cards. These cards will only engage in wireless communications when they internally detect these secret handshakes. We demonstrate the effectiveness of this approach by implementing our secret handshake recognition system on a passive WISP RFID tag with a built-in accelerometer. Our secret handshakes approach is backward compatible with existing deployments of RFID tag and contactless card readers. Our approach was also designed to minimize the changes to the existing usage model of certain classes of RFID and contactless cards, like access cards kept in billfold and purse wallets, allowing the execution of secret handshakes without removing the card from one's wallet. Our techniques could extend to improving the security and privacy properties of other uses of RFID tags, like contactless payment cards.

Categories and Subject Descriptors

B.m [**Hardware**]: Miscellaneous

General Terms

Security

Keywords

Context-aware Communications, Gesture Recognition, Man-in-the-middle Attacks, Passive Gesture Recognition, Privacy, Proxy Attacks, Relay Attacks, RFID, RFID Device Selection, Skimming Attacks

CCS'08, October 27–31, 2008, Alexandria, Virginia, USA.

1. INTRODUCTION

Radio frequency identification tags (*RFIDs*) and other contactless cards (like *proximity cards* and *contactless smartcards*) are increasing in ubiquity. For example, large corporations often use RFIDs or proximity cards to regulate building access. American Express, VISA, and MasterCard all produce credit cards with embedded RFID tags. Many car keys also have embedded RFID tags to help protect against hot-wiring. While the security community has invested significant resources in understanding and addressing the security deficiencies of such cards — including documented attacks against and defensive recommendations for each of the above examples [5, 14, 16] — there exists one class of attacks that the community is still battling: the so-called *ghost-and-leech* attacks [8, 20]. It is this class of attacks that we tackle in this paper, and in doing so we introduce a new defensive approach that we refer to broadly as *secret handshakes* — or — on-card mechanisms for detecting and then communicating only when the card holder performs some action with the card indicative of legitimate use.

Ghost-and-Leech Attacks. A key challenge with RFIDs and other contactless cards is that they are indiscriminate with respect to what external devices they wirelessly communicate with. A ghost-and-leech attack exploits this indiscriminatory nature as follows. Consider, for example, the case where an RFID or proximity access card (or access badge) is used to grant entry into a building. Under a normal usage scenario, an employee — Bob — might keep his access card in his wallet, walk up to the door, take his wallet out of his pocket, and then place his wallet near the reader, thereby triggering the reader to unlock the door and grant Bob entry; see Figure 1(a). To mount a ghost-and-leech attack, two attackers — the *ghost* and the *leech* — coordinate their activities; see Figure 1(b). The ghost places his or her attack equipment near the door's reader, and the leech places his or her attack equipment near Bob's wallet, perhaps as Bob rides the bus or train to work or stands in line at a local coffee shop. By relaying all communications between the reader and Bob's access card, the ghost can surreptitiously gain access to the building [14].

One can apply similar ghost-and-leech attacks to other uses of RFIDs and contactless cards. Moreover, anti-cloning and strong cryptographic mechanisms *cannot* by themselves protect against the ghost-and-leech attack. This follows nat-

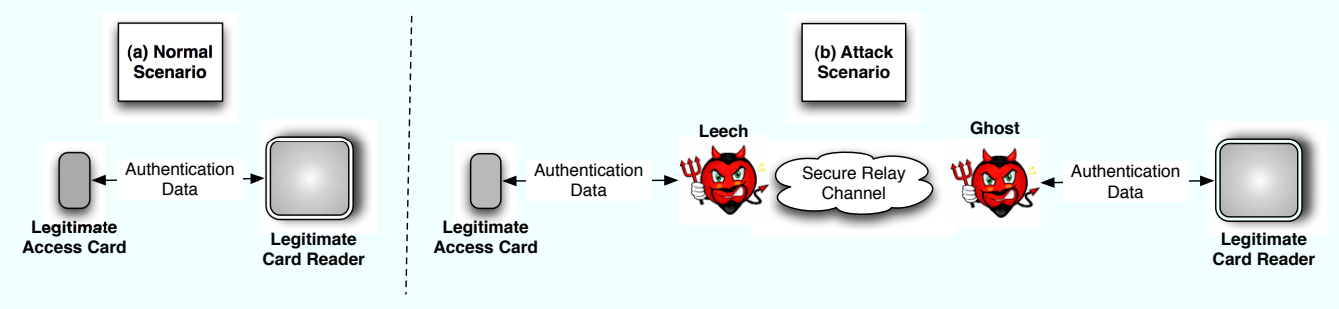

Figure 1: Typical authentication session with contactless card. Arrows represent data flow. Part (a) shows a normal scenario. Part (b) shows the *ghost-and-leech attack*. The ghost emulates a card and leech emulates a reader. The ghost-and-leech attack is also often referred to as a *man-in-the-middle attack* or a *relay attack*.

urally from the behavior of the ghost and leech: the ghost and leech do not need to modify, tamper with, or inspect the contents of the communications between the reader and Bob's access card — the ghost and leech simply need to relay the communications in a black-box manner. The inability for cryptography to defend against the ghost-and-leech attack creates a conundrum, and the only known solutions require either sophisticated processing on the reader or sometimes obtrusive changes to the usage model of the RFID or contactless card. As an example of the former, one solution — known as *distance bounding* [15] — is for the reader to gauge the physical proximity of the RFID or contactless card by measuring the amount of time it takes for the tag or card to respond to challenge messages; the assumption here is that the ghost's and leech's proxying step will introduce non-negligible time delays. As an example of the latter, some vendors are producing access cards with buttons that users must press in order to activate them [3], and third-party vendors are selling protective metallic sleeves that block contactless communications [1].

Our Approach: Goals. Motivated by the above discussion, we seek to improve the resistance of RFIDs and contactless cards to ghost-and-leech attacks subject to the following two central design constraints:

- *Backward Compatibility.* Our solution should not require modifications to existing, deployed RFID or access card readers.
- *Consistent Usage Model.* Our solution should require little or no change to the usage model of existing RFIDs and contactless cards.

By focusing on these two principle goals, we provide a unique balance in the design space for defending against the ghost-and-leech attack — a solution that is both incrementally deployable today without modifying existing readers, and that does not require users to, for example, remove an access card from their wallet and press a button before entering a building.

We acknowledge that our approach does not provide perfect resistance to the ghost-and-leech attack, and we experimentally evaluate this level of resistance via a prototype implementation. We find that for many applications our approach significantly improves the resistance of RFIDs and access cards to practical ghost-and-leech attacks. Nevertheless, and because other defensive mechanisms like reader-side distance bounding are also not 100%-effective, an additional goal of our system is that it be *composable* with other defensive mechanisms, like distance bounding on the reader. This composability allows a deployment to layer both mechanisms, thereby providing greater defense-in-depth.

Our Approach: Secret Handshakes. The intuition behind our technical approach is the following. Consider the standard usage scenario of an RFID or contactless card. When Bob wishes to enter an access-controlled building, he often subconsciously does a fixed set of motions to gain entry — he reaches for his wallet, draws his wallet near the door's reader, and pauses. Similarly, when Bob wishes to start his car, he takes the key out of his pocket, inserts it into the key shaft, and twists. The observation we make is that *if* it were possible for the RFID chip or contactless technology in the access card or car key to somehow internally detect exactly when these actions were being performed, then it would be possible to create logic on the tags and cards that would *only* allow the tags and cards to communicate *while these actions are being performed*. We call this approach *context-aware communications*, and we note here a related observation in Heydt-Benjamin *et al.* [16] that we expand on further in Section 2.

Context-aware communications must provide two properties to be useful in practice. Namely, (1) the *false negative* rate must be low, meaning that Bob should with high probability be able to enter his building or start his car on the first try. At the same time, (2) the *false positive* rate on the RFID tag or contactless card must also be low, meaning that there should be little chance for the access card or car key to accidentally conclude that Bob is trying to gain entry or start his car when in fact he is not. While some motions (like key insert and twist) are relatively unlikely to occur accidentally, other motions (like taking ones wallet out of one's pocket) are more likely to occur when Bob is not actually trying to gain entry to his building. The risk with false positives is clear: any situation leading to a false positive could potentially be exploited by a ghost-and-leech adversary. The argument here is, however, in fact more subtle than the above implies, and we return in more detail to the risks associated with false negatives later.

While we are the first to deeply investigate the notion of context-aware communications for RFID tags and contactless cards, we also introduce a new approach — called *secret handshakes* — for reducing the false positive rate while only moderately weakening the consistency of our usage model. In particular, we consider the creation of special movements

for the RFID tag or contactless cards — movements that are highly unlikely to occur naturally in one's day-to-day activities but that are easy to perform within a short period of time (less than one second) and that do not require direct physical manipulation of the card (e.g., no pressing a button). For example, one of our secret handshakes — 1.5-wave — would simply have Bob wave his wallet from left to right to left near his building's card reader.

Survey of Access Card Users. While obviously achieving context-aware communications with *no* modifications to the normal usage model is ideal, we argue that minor usage modifications like 1.5-wave offer many advantages over traditional solutions, like placing buttons on the cards or placing the cards in metal sleeves. Indeed, a survey of 554 people found that, of the 191 people who used a single contactless access card, 123 (64.4%) kept those cards in their billfold or purse wallets and placed those wallets in front of the access reader to gain entry. Our 1.5-wave secret handshake would introduce few modifications to how those participants currently use their access cards. Additionally, of the 232 participants who used a single contact-based access card (e.g., a magstripe card), 159 (68.5%) kept that card in their billfold or purse wallet when not being used, suggesting that these participants would benefit from our approach if they were required to switch to contactless access cards. We present more results from our survey in Section 5.4.

Prototype Implementation and Evaluation. To evaluate our approach, we implemented context-aware communications and secret handshake detection on the WISP UHF RFID tag [27], which is a passive (batteryless) RFID tag with a built-in accelerometer and modest computational capabilities. The existence of this accelerometer allowed us to use simple activity recognition techniques to infer whether or not the holder of that tag is performing a secret handshake, and to only respond to external RF communications after internally recognizing that handshake. A key challenge that we had to overcome was the WISP RFID tag's limited computational capabilities and memory. Our implementation therefore uses a variant of cross-correlation written in auto-generated, highly-optimized C code. We then experimentally verified, with three participants, that our approach was effective, allowing them to trigger the 1.5-wave secret handshake recognition system — with zero attempts failing out of 50 total attempts — while simultaneously reducing the risk of exposure to ghost-and-leech attacks.

Extensions. Our approach — while targeted at protecting against ghost-and-leech attacks — can also help improve the resistance of RFID tags and contactless cards to location tracking attacks and the leakage of other private information. For example, it is currently possible to sit next to someone in a bar and surreptitiously read their RFID credit card account number and other information [16], or to place RFID readers in many locations and track someone's location by monitoring consistent identifiers [17]. While there exist cryptographic approaches for improving resistance to these attacks (e.g., [24]), those solutions require modifications to both the tags and the readers or back-end systems. A separate approach is to use an external device, like a "blocker tag [18]." Secret handshakes could serve as a tag-local approach for limiting the exposure to these attacks.

As a related example, consider a scenario in which Bob has multiple cards in his wallet — a transit (bus) card, a credit card, and a gym card, all of which for backward-compatibility purposes reveal private information when read. Using secret handshakes could prevent the driver of the bus from surreptitiously reading Bob's credit card number while Bob waves his wallet near the reader on the bus.

Alternate Approaches. As an alternate to our secret handshakes approach, we observe another viable approach for protecting against ghost-and-leech attacks and other surreptitious communications *when* we can assume that the standard usage model involves a person physically touching his or her RFID tag or card. In this case, it would be possible to use a capacitive sensing [26] WISP RFID tag to detect when the tag is being held in someone's hand, and vary the RF communications and other activities accordingly. We do not investigate this approach in this paper because the usage model is more restrictive than the usage model we consider (albeit perhaps appropriate for certain classes of cards, like RFID credit cards), and because the technical challenges are less (capacitive sensing verses gesture recognition on a passive RFID tag). However, we do note that a capacitive sensing approach is likely more durable than a physical button because of the lack of moving parts.

2. BACKGROUND

2.1 The Ghost and Leech Attack

The ghost-and-leech attack was coined in 2005 by Z. Kfir *et al.* [20], but more general relay attacks have been known for over 30 years [8]. As illustrated in Figure 1, a pair of attackers — the *ghost* and the *leech* — mount a ghost-and-leech attack by relaying communications between a legitimate RFID or contactless card and a legitimate reader. The ghost-and-leech attack vector is practical and has been successfully demonstrated against proximity door-access cards [14], though one could also apply the ghost-and-leech attack to other tags and cards like SpeedPass™[5] and credit cards [16] if those devices didn't already use weak or non-existent cryptography. (In addition to these examples, there are demonstrable ghost-and-leech attacks to non-contactless systems, such as UK's EMV payment system [10].)

Threat Model: Goals and Non-Goals. Our goal is to reduce the vulnerability of RFID tags and contactless cards to ghost-and-leech attacks, while not significantly impacting the usage model for these tags and cards and not requiring changes to existing, deployed readers. We assume that the attackers have complete control over the radio environments surrounding both the legitimate RFID tag or contactless card and the reader. For our threat model, we explicitly do not consider attacks in which one of the attackers is in physical possession of the card. For example, we do not consider attacks in which the one of the attackers steals or temporarily borrows the card, or when one of the attackers is also the legitimate owner of the card wishing to grant remote building access to a conspirator while he or she is at a public event for alibi purposes.

Existing Defenses to the Ghost-and-Leech Attack. Known cryptographic techniques or anti-cloning mechanisms are unable to prevent ghost-and-leech attacks because the attackers only need to relay communications in order to mount such an attack (and not modify those communications or generate new communications). Several alternate suggestions have therefore been proposed for combating the

ghost-and-leech attack [19]. One standard solution is to use multi-factor authentication (e.g., requiring a PIN or password when swiping a card, requiring a biometric scan, or both). Active (battery-backed) tags incorporating some of these features are commercially available [2]. Other approaches include sleeves that require the tag to be removed [1] or a button to be pressed [3] in order to enable the tag to be read. However, these are rare in practice and change the usage model dramatically. A potential approach relies on tight timing constraints between the reader request and the card's reply, based on the fact that the ghost-and-leech attack will introduce some delay, e.g., see [14, 15]. However, this requires fine-grained timing mechanisms, modifications to existing readers, and is difficult to do with today's hardware.

2.2 Additional Related Work

We discuss some additional related work here.

RFIDs, Security, Privacy, and the WISPs. The WISP RFID tags [27] are a powerful tool for implementing stronger security mechanisms on passive RFID devices. Examples of this line of research include the papers by Chae *et al.* [6] and Halperin *et al.* [13]. Chae *et al.* demonstrate that it is possible to implement RC5 on the WISPs, and Halperin *et al.* demonstrate how WISP-like technologies could improve the security and privacy of implantable medical devices.

RFIDs and Context-Aware Communications. Related to our notion of context-aware communications for RFIDs is the work of Heydt-Benjamin *et al.* [16]. These authors discuss a large number of possible approaches to reducing the risk of fraud and privacy invasions associated with RFID credit cards, including the use of more sophisticated cryptographic techniques, protective sleeves, and buttons on the cards. Within their list of defensive approaches, the authors conjecture that motion sensors on an RFID card could detect the telltale tap-and-go motion typically associated with RFID credit card purchases. Our research dives deeply into this shared conjecture, in which we solidify our model for context-aware communications, introduce our new notion of secret handshakes, investigate the deployment and other considerations surrounding these models, and demonstrate — perhaps surprisingly — that it is in fact feasible to implement such techniques today on an existing passive RFID tag.

Additional RFID Security and Privacy Research. There exists a significant body of RFID security and privacy work that is further removed from the ghost-and-leech attacks and our research, e.g., research on RFID location tracking. We defer a detailed summary of such research to Juels [17].

Activity Recognition. There is a significant body of literature focused on activity recognition; see [9] for a survey. Work in this area has shown that accelerometers are sufficient for certain activity recognition tasks [4, 21], which motivates our use of the accelerometers on the passive WISP RFID tags. A critical difference between this existing research and our own is that we implement our activity recognition systems on the resource-constrained WISPs. We stress, however, that our WISP-based activity recognition system is only one aspect of our contributions, and that if one were to implement context-aware communications and secret handshakes on less resource-constrained tags or cards, then it would be possible to further leverage this existing body of activity recognition research. See [7] for an example of a richly provisioned platform for sensor based activity recognition.

Accelerometers and Security. There also exist other research utilizing accelerometers for security purposes and access control. For example, Patel *et al.* [25] present a method for a mobile device to authenticate with an untrusted, public terminal; for this research a person would shake the mobile device in a manner the public terminal specifies. Gafurov *et al.* [12] consider the problem of the illegal use of stolen mobile devices. To counter this problem, Gafurov *et al.* propose only unlocking mobile devices when they detect the gait (walking pattern) of the legitimate owner. At the highest level this research shares our goal of varying a device's behavior based on the activity that it detects, but our specific directions are different. Our goal is to disable communications and other activities while the device (tag or contactless card) is in the context of day-to-day activities, i.e., we wish to only enable communications during specific authentication activities associated with, for example, attempting to gain legitimate entry to a building. In contrast, Gafurov *et al.*'s approach is to enable actions in day-to-day activities like walking. Furthermore, Gafurov *et al.* were able to utilize more sophisticated activity recognition schemes on the less resource-constrained mobile phones. Mayrhofer and Gellersen [22] demonstrate secure device pairing through mutual observation of acceleration; here the model is that two mobile devices could be shaken together in such a way that they both can derive the same cryptographic key from the shared accelerometer readings but that the key would be hard for a third party observer to infer.

3. CONTEXT-AWARE COMMUNICATION

In Section 1 we motivated several of our key goals for a ghost-and-leech protection system, including the following utility, deployment, and security considerations:

i. Consistency of usage model;

ii. Backward compatibility with existing, deployed readers;

iii. Cost-effectiveness;

iv. Composability with other defensive mechanisms like reader-side distance bounding;

v. Utility, e.g., in the case of access badges Bob shouldn't be denied entry; and

vi. Attack resistance: the system should provide low exposure to ghost-and-leech attacks.

Here we discuss our overall approach for simultaneously meeting all of these goals. We refer to our approach as *context-aware communications* since an RFID tag or contactless card should only engage in communications when the user actually desires this action — i.e., when the *context* is correct.

3.1 Inferring Context

There are a broad range of potential techniques — some more desirable or feasible than others — for an RFID tag or card to determine whether the context is appropriate for

communications. These approaches span the spectrum from obtrusive — requiring the user to perform additional awkward or tedious tasks with the tag or card — to completely transparent to the user.

Obtrusive Inference Techniques. As our survey shows, 282 of the 423 (66.7%) people with access cards (contactless or non-contactless) keep those cards in their billfold or purse wallets; see Section 5.4. Our survey also indicates that many people with contactless access cards do not remove those cards from their wallets when they use their cards to gain access to a resource. Additionally, and anecdotally, users sometimes find it difficult to remove a (non-contactless) access card from a billfold or purse wallet, and this difficulty increases while wearing gloves. Consequently, while a button would clearly meet our reader-side backward compatibility, utility, and security goals, we believe that button-based and related context inference techniques would in many cases be undesirable in practice. Our survey also indicates that minimizing authentication time is extremely important to many users, implying that the use of PINs or biometrics (whether on the card or on the reader) would also be on the obtrusive end of the usability spectrum and would hence violate our goal of maintaining a consistent usage model.

Transparently Inferring Context. At the other end of the spectrum, we seek to transparently infer whether the user is in a legitimate authentication context. Here we can leverage the fact that users may move the RFID tag or contactless card in relatively unique ways during authentication; they may twist their hip, wave their wallet, tap the tag or card against the reader, or insert and twist a car key. This leads to one of our principle observations: *by placing an accelerometer on an RFID tag or contactless card, we can passively and transparently capture the physical information describing these movements.* Observe that *if* we can reliably and precisely detect these activities, then we can limit the tag or card to communicate only in valid authentication contexts. This would significantly reduce the access card's exposure to ghost-and-leech attacks, provided that the authentication context is not frequently detected during normal everyday activities.

One might consider broadening our context-aware approach to include, for example, pre-existing research on detecting whether two devices share a common physical environment. For example, LaMarca *et al.* propose a method whereby two devices verify their shared physical proximity based on the shared properties of received radio signals [28], and one could naturally create similar approaches based on audio or light signals. We avoid such approaches here, not only because they will be difficult to implement on resource constrained devices, and perhaps imperceivable by tags tucked inside wallets and purses, but also because these techniques violate our reader-side backward compatibility goal. In contrast, our approach for detecting context is local to the tags and contactless cards themselves.

Relaxations to Perfect Consistency. Precisely detecting the legitimate authentication context, while maintaining a perfectly consistent (unchanged) user experience and utilizing a low-resource device, may be very hard. However, in many cases it is sufficient to maintain an approximately consistent usage model. With small modifications to the usage model, the RFID or contactless card can begin to make good guesses about whether or not the tag or card is in the correct authentication context.

From a consistency perspective, one of our goals is that a person should not be forced to take a card out of their billfold wallet or purse wallet. Furthermore, it must be possible for users to perform all additional actions within at most one second. Note how techniques like the button press violate both criteria.

While intuitively motions like "key turn" might appear fairly unique, simply the act of taking one's wallet out of a pocket is not since a user may take their wallet out numerous times during the day for various reasons other than authentication. We consider asking the user to slightly modify their usage model with the addition of a *secret handshake*.

Secret Handshakes. Conceptually, a secret handshake is a series of time-constrained physical actions — or gestures — which an individual must perform with the RFID tag or contactless card (or wallet, if the tag or card is in the wallet) in order to "unlock" the card and permit it to communicate with a card reader. In this paper we also refer to secret handshakes as *activation schemes* and use these terms interchangeably. Some example secret handshakes include:

- Alpha – User moves card in an alpha (α) pattern in front of reader. (An example of this activation scheme is shown in Figure 2.)
- Key Twist – User makes key turning motion with the card.
- Hip Twist – User keeps card in his/her pocket and twists hip to bring pocket with reading range of the reader.
- Circle – User moves card in a circular manner, parallel with the surface of the reader.
- Double Circle – User makes two consecutive circles with the card.
- Triangle – User moves card in a triangular pattern parallel with the surface of the reader.
- 1-left-1-right – User waves card side to side (once left, once right) in front of the card reader.
- 1-right-1-left – User waves card side to side (once right, once left) in front of the card reader.
- 1.5-wave – User waves card side to side (once right, once left, once right) in front of the card reader. (An example of this activation scheme is shown in Figure 3.)

Ideally, the accelerometer fingerprint of these motions would significantly differ from those which are produced during everyday activities (such as sitting, eating, walking, running, and jumping) in order to prevent an attacker from capturing signals during those activities. In Section 5 we show that some schemes perform better than others and recommend the most efficacious activation scheme. We also stress the potential for more precise secret handshakes if one is able to equip the contactless card with a battery and additional computational resources.

3.2 Context Detection on the Card or Reader

Processing of the accelerometer readings can either be performed directly on the RFID tag or contactless card or can be encrypted and sent to the reader for processing. Specifically, signal processing on the card uses limited resources such as memory (for buffering samples) and CPU. However,

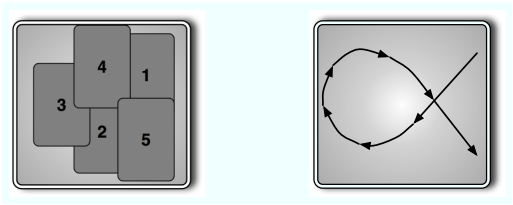

Figure 2: Example secret handshake/activation scheme. Both images show the alpha (α) motion performed with the card in front of the reader. In the left image, numbers indicate sequence of card positions across reader with time. In the right image, arrows show how the card moves across the reader with time.

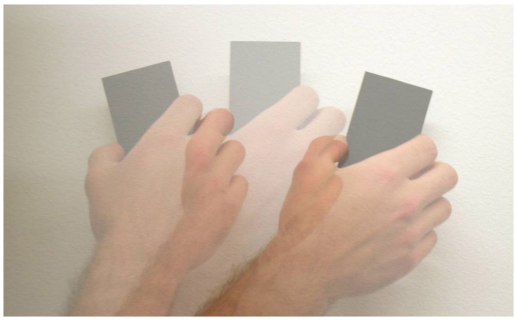

Figure 3: Example secret handshake/activation scheme. In this image, we demonstrate the 1.5-wave gesture.

signal processing on the card allows it to only transmit data when the user explicitly authorizes the transfer through the correct physical manipulation of the card and requires no modification of the reader, making it backwards compatible with current systems. Additionally, there may be instances when the tag is able to collect and process accelerometer data but unable to send it to the reader; for example, RF interference from other 900 MHz ISM-band devices could corrupt bits in the communication channel without significantly affecting the power received by the tag. Signal processing on the card is also independent of any reader behavior (except that the reader must be sending some RF energy to power the card), so selective forwarding attacks (where an attacker drops some samples in order to massage a signal into something a reader might accept) are not possible.

On the other hand, signal processing on the reader has greater resources. It requires the constant transmission of accelerometer data by the card. This also means that the card must encrypt and authenticate the data being sent in order to prevent easy replay attacks. Furthermore, this latter approach is not backwards compatible with current systems. After evaluating these tradeoffs, we decided on an implementation that is fully contained within the tag or contactless card.

4. IMPLEMENTATION

We implemented a prototype of context-aware communication and secret handshakes on a passive RFID tag. The usage model of the system is consistent with the regular usage scenario — the user is in possession of an RFID access card, which he must present to the RFID reader to gain access to a protected resource. In accordance with our design discussion in Section 3, we performed context detection on the RFID access card.

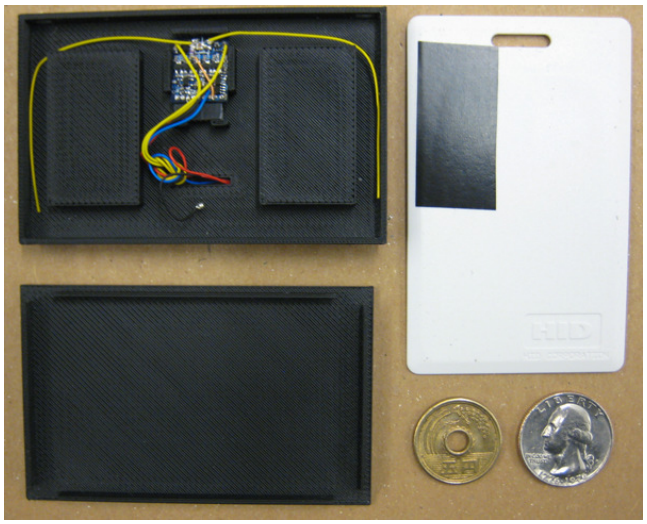

Figure 4: Our prototype WISP RFID system enclosed in a custom plastic box. Wires lead out the back for debugging and tethered experiments. Box cover shown at bottom left. Next to it is a standard HID proximity card.

4.1 WISP Overview

In order to meet our design goals of cost effectiveness and backwards compatibility with existing readers, we explore how to perform authentication context recognition on a completely passive RFID card. We chose to use the WISP platform from Intel Research Seattle [27] since it is currently the only passive RFID device that is completely programmable. The WISPs we use combine a Texas Instruments MSP430F1232 16-bit microcontroller (with 8 KB of Flash and 256 bytes of RAM) with an Analog Devices ADXL-330 3-axis, $\pm 3g$ accelerometer. Currently, the WISPs communicate using the EPC Class 1 Gen 1 standard [23], but future devices will use the EPC Class 1 Gen 2 standard [11]. While we acknowledge that EPC Gen 1 is not widely used in the card form factor, more common near-field protocols like ISO 14443 can provide even more power, so it is likely to be even more feasible to implement our technique with these protocols. For experimentation purposes and durability, we enclosed the WISP in a plastic box measuring 85.60mm x 53.98mm x 7.5mm. The width and length match standard ISO 7810 ID-1 size used by most access control badges. While the box is slightly thicker than the ISO 7810 ID-1 standard, an actual deployment of the WISP in access badges would come very close to or even match the standard width. An image of our prototype WISP system is shown in Figure 4.

4.2 Challenges to WISP-Based Context Detection

The WISPs have three main constraints: memory, power consumption and availability, and raw computational ability.

Memory Constraints. With only 256 bytes of RAM available, we have only enough memory to buffer 48 accelerometer samples. The samples are stored as 3-tuples of signed bytes, using 144 bytes of RAM. The remaining RAM is used for the RFID protocol and stack. Since we would like our activation schemes to take at most approximately one second (to keep the usage model reasonably consistent with the status quo), this naturally led to a sampling rate of approximately 48 Hz, which drove many of our other design decisions.

Power Constraints. Since the WISP is completely powered by the RF field generated by the reader, our power budget is tight. According to its datasheet, the microcontroller typically consumes 225μA at 3V when running at 0.75 MHz. Even more problematic is the fact that the accelerometer consumes 300μA when active, and requires being active 17 ms (for the sampling rate chosen) before the measurements are valid. In fact, most of the energy spent in our system is used powering up the accelerometer. Given that the WISP RFID tag receives power from the external reader, in Section 5.3 we discuss an adversary who might attempt to control the WISP's clock and therefore the sampling rate.

Computational Constraints. Finally, even if we had an infinite power budget, the WISP's core computational capabilities are rather limited. The microcontroller has no DSP features, such as a hardware multiplier or a square root lookup instruction. The microcontroller also has a maximum clock speed of 8 MHz. The restricted computational capabilities and the low clock rate on the WISP prevented us from using more sophisticated gesture recognition techniques, and instead we implemented our own variant of cross-correlation.

Cost. While more microelectronic components are being squeezed into smaller areas with every successive generation of process technology, Moore's law does not directly enable lower RFID tag costs. This is because the antenna cost and assembly cost represent a substantial fraction of the RFID tag cost. In large integrated circuits, these "per die" costs make a relatively small contribution to the cost of the finished chip. In small ICs like RFID tags, these "per die" costs become a cost floor for the tag. If we assume that these "per die" costs are fixed or vary only slowly over time, then what Moore's law should enable for RFID tags is not ever cheaper tags, but tags with ever more transistors for approximately the same cost as today's tags. Generalizing slightly, it appears that improvements in process technology should enable new capabilities (such as sensing and increased computing) in RFID tags for the same price as today's tags.

4.3 System Overview

Registration. Our overall system has three phases. In the first phase a person performs the target action one or more times, and the accelerometer readings for that action are recorded and serve as the *template* with which future actions will be compared. This template can be on a per-person basis or generic, meaning that multiple people performing the same action may have different templates. This template is then loaded on the WISP tag.

WISP States. Our WISP implementation itself has two general states: *activation scheme* (or *context*) *recognition*, and *RFID transmission*. In the *context recognition* state, the system continuously samples the accelerometer and runs a recognition algorithm against a window of these samples, until it runs out of power. If the recognition algorithm determines that an activation scheme has been performed, then our system transitions into the *RFID transmission* state, which merely implements the EPC Gen 1 protocol to transmit a static identifier for approximately two seconds. Note that, if desired, we can reuse the accelerometer sample buffer at this point and implement a cryptographic challenge / response protocol as done on the WISPs in [6] and [13]. In the following subsection we detail the implementation of the first *context recognition* state.

4.4 Context-Recognition on the WISP

To recognize a particular secret handshake, we compute the cross-correlation C of the accelerometer data window A against a template T of the scheme, both of size n data points:

$$C = \sum_{i=1}^{n} (A_{xi}T_{xi} + A_{y_i}T_{y_i} + A_{zi}T_{zi}) \quad (1)$$

We transition to the *RFID transmission* state when C exceeds a certain threshold. Ideally, we'd like to perform normalized cross-correlation to reduce the impact of high-magnitude samples, which is a standard technique in signal processing but more efficient than standard gesture recognition techniques. However, the lack of DSP features on the microcontroller make normalized cross-correlation extremely slow. Instead, we simply zero out samples where any of the axes exceed their magnitude thresholds, which are set such that they will rarely be exceeded when performing an activation scheme. Naturally, we except that an implementation of gesture recognition techniques on a more powerful tag or contactless cards with greater resources would yield recognition results at least as good as those we discuss in Section 5.

To minimize the number of cycles needed to perform the cross-correlation, we employ a number of tricks by taking advantage of the fact that each position in the accelerometer data window is multiplied by a fixed template value. The microcontroller can perform addition and subtraction in hardware, but cannot perform multiplication. Instead of calling the generic multiplication function for each sample and adding the results together, we can use the distributive property of integers to add the samples that will be multiplied by a common template value first, and then do the multiplication.

We can further optimize this by decomposing the multiplications into the summation of multiplications of powers of 2. We have eight registers (one per bit), and we add each accelerometer sample to each register that corresponds with a 1 bit in its template multiplier. For example, if accelerometer sample A_{x2} is multiplied by the template value 9, we add A_{x2} to registers R_1 and R_8.

Expressing the multipliers as canonical signed digit numbers allows us to minimize addition operations by substituting many of them for a single subtraction operation, which is equally cheap on the microcontroller. For example, if accelerometer sample A_{y7} is multiplied by the template value 31, instead of adding A_{y7} to R_1, R_2, R_4, R_8, and R_{16}, we add A_{y7} to R_{32} and subtract it from R_1. Each multiplication is then a simple bit shift, and adding the eight registers together gives us our cross-correlation value. We use a Perl

script to automatically produce optimized C code to perform these operations without any branching.

4.5 Building Context Templates

With cross-correlation implemented on the WISP, the final challenge is choosing values for the three major variables to our system: the context template, the magnitude filtering thresholds, and the cross-correlation detection threshold. To explore the space, we had three participants perform a variety of secret handshakes and carry the WISP around while performing day-to-day activities. To do this, we modified our prototype to allow it to be externally powered and provide accelerometer data over a serial line. We used Intel Research Seattle's Mobile Sensing Platform [7] to power the WISP and store its accelerometer data. We then imported the data into MATLAB for analysis.

Traces of template examples were automatically split into individual templates using a simple algorithm that found n 48-sample windows with the highest magnitude. Samples toward the middle of the window were weighted slightly higher to ensure that if the action was less than 48 samples long, it was performed in the middle of the sample window.

We then compared each instance of an action to the day-to-day activity traces and to other instances of the activity to determine the appropriate cross-correlation detection thresholds. In several instances, there were clear regions of the threshold value where no false positives or false negatives were detected. For the magnitude filtering thresholds, we chose to use the minimum and maximum value of each axis in the gesture instance. We tried multiplying these values by some constants, but in all cases our false positive rates went up while our false negative rates remained the same.

5. EVALUATION

In order to evaluate our approach, we studied our system in two phases: tethered, and untethered. Phase 1 is used to identify the most effective secret handshake templates by analyzing the false positive rate (when the system recognizes an authentication scheme when none occurred) and false negative rate (when the system does not recognize an authentication scheme when one did occur). For the former we consider an attacker that could constantly (or selectively) leech onto the WISP at all times during one's day-to-day activities. Phase 2 mainly concerns utility by analyzing the false negative rate with respect to missed data reads that may occur during real authentication sessions.

Our experimental evaluations in Sections 5.1 and 5.2 consider three participants, P_1, P_2, and P_3. In Section 5.3 we discuss security in further detail, and in Section 5.4 we study a survey on how people currently use access cards.

5.1 Phase 1: Tethered — Choosing a Secret Handshake Template

In phase one, we tethered the WISP to an external power source and data logger, as mentioned in section 4.5. This simulates an attacker with an RFID reader that is continuously close enough to the victim to constantly power the WISP and receive responses, and allows us to place an upper bound on the percentage of time an attacker will be able to elicit a response from the tag. The tethered WISP also allows us to collect more accurate secret handshake template candidates by streaming the accelerometer readings from the WISP to the connected device via a wire. The alternative would be to collect the templates either 1) on the WISP, which has little available memory or 2) transmit data over the RF channel, which is prone to being lossy, giving us an incorrect template. We then analyze these template candidates in MATLAB.

Action	P_1 Slack	P_2 Slack	P_3 Slack
Key Twist	**2500**	**3800**	-3600
Hip Twist	-700	-1100	-2000
Alpha	-400	**100**	**4800**
Circle	-800	**500**	-300
Double Circle	**11400**	**5200**	**1500**
Triangle	**1100**	**1600**	**300**
1-left-1-right	-100	**7400**	**1200**
1-right-1-left	**1600**	**3900**	**3800**
1.5-wave	**1700**	**19800**	**8600**

Table 1: Maximum slack.

Our MATLAB code automatically picks the best n template candidates (as described in section 4.5) for a secret handshake from traces collected of the secret handshake. We also collected accelerometer data from the WISP from a variety of day-to-day activities for the three participants, such as walking, sitting, standing, fidgeting, playing ping-pong, and riding a bicycle. Next, we ran the WISP's cross-correlation code, reimplemented in MATLAB, on these template candidates against the day-to-day activity traces in order to determine the false positive rate — how often each particular secret handshake gestures were recognized during these activities — for a spectrum of cross-correlation threshold (the threshold at which a match will be made between a template and the target data, see Section 4.5 for more details). This was done over the aggregate of all day-to-day activities for all three participants. For each type of secret handshake, we also compared each template candidate to all the other template candidates of the same type and for the same participant; this allowed us to gauge the expected false negative rate — how often secret handshake gestures would not be recognized when they did, in fact, occur.

For many of the secret handshakes, we found a region of thresholds where no false positives or false negatives occur; we define the slack of a secret handshake to be the range of this region. In more detail, consider a single template. Let Pos_r be the maximum threshold at which the false positive rate is non-zero, and let Neg_l be the minimum threshold at which the false negative rate is non-zero. Then the slack for this template is $\mathsf{slack} = \mathsf{Neg}_l - \mathsf{Pos}_r$. Negative values of slack indicate the width of the region where both the false positive and false negative rate is non-zero. We stress that these results are preliminary, and we do not expect perfect accuracy in actual use. However, the use of slack lets us find the most promising gestures. For each gesture, we selected templates, which resulted in the greatest positive slack, setting the threshold to the mean of the slack region. We report the maximum slack achieved by the optimal template for each secret handshake and each participant in Table 1. The double circle and 1.5 wave had the largest range of thresholds where the false positive and false negative rates were zero. The hip twist performed worst — it yielded a large section of thresholds where the false positive and false negative rate was non-zero.

Reader	P_1: 1.5-Wave	P_2: 1.5-Wave	P_3: 1.5-Wave
Speedway	0/50	0/50	0/50
Alien	0/50	0/50	0/50

Table 2: Untethered False Negative Rate. Each participant attempted the secret handshake 50 times in front of two different readers, every attempt resulting an a successful authentication.

5.2 Phase 2: Untethered — Utility and False Negatives

While in phase 1 the WISP had constant power, in actual use the WISP's power is constrained by the distance from the reader, angle with the reader, and loss of RF energy due to obstacles between the reader and the WISP. Consequently, if a user accidentally moves the WISP out of the range of the reader, the WISP won't receive enough power to gather accelerometer data, won't perform cross correlation to determine context inference, and ultimately won't let the user authenticate. In phase 2, we capture this effect by untethering the WISP and using a template selected from phase 1 to measure the actual false negative rate.

For each of the three participants in our study we used P_1's template for the 1.5-wave secret handshake, with the threshold set to the mean of the slack region for P_1's template. Each of the participants was allowed up to 5 minutes to refamiliarize themselves with the 1.5-wave motion. The participants then performed the secret handshake 50 times. If their movement permitted them to authenticate, we marked the attempt as successful, otherwise we recorded them as having a false negative. Each of the participants achieved 0 false negatives for 50 attempts of the secret handshake; these results are reported in Table 2. Each of the attempts lasted less than a second. Even though the participants' gestures varied as their hands fatigued through the 50 attempts, our approach was still robust enough to achieve a zero false negative rate.

We performed these experiments with two different RFID readers in order to eliminate any reader bias from our results. Our experimental setup consisted of Alien ALR-9780 RFID (EPC Gen 1 and 2) reader and an Impinj Speedway RFID Reader, which were both connected to a separate circularly-polarized, 6dBi gain antenna. While the Speedway reader is an EPC Gen 2-only reader, our WISP can be powered by Gen 2 readers (but cannot communicate with them); we therefore monitored a GPIO pin on the WISP with an oscilloscope to determine if a gesture has been recognized. Note that the WISP was entirely powered by RF in both instances, and upon recognizing the secret handshake, transmitted its ID to the Alien Gen 1 reader.

We did not evaluate false positives in the untethered mode since it does not model the worst case scenario, in which an attacker is able to constantly power the WISP with an RFID reader. This analysis was performed by the tethered analysis in phase 1.

5.3 Security Analysis

Since our model for context-aware communications and secret handshakes is new, it is prudent to reflect upon the security properties that our overall approach — and our WISP-based RFID prototype — provide. Recall that our goal is to protect against surreptitious reads and proxying of wireless RFID tag and contactless card communications, while at the same time allowing the tags and cards to achieve their primary functions (e.g., allowing a user to legitimately enter a building or start a car).

Given this goal, we believe that for many applications it would be sufficient to simply raise-the-bar against the ghost-and-leech attack by decreasing the windows of opportunities at which the tag or card would be vulnerable to attack — i.e., reduce the amount of times in which the tags or cards would incorrectly conclude that the secret handshake is being performed when in fact it is not. However, our preliminary experimental results here are very encouraging. For all three participants, five of our secret handshakes had threshold regions that resulted in both zero false positives and zero false negatives. But our results suggest that 1.5-wave and double circle might be a reasonable gestures to assign to all people in an organization or company. Alternately, recalling that templates can be chosen on a per-person basis, it would also be reasonable to allow participant P_1 to use double circle as his or her secret handshake, and P_2 and P_3 to use 1.5-wave as their secret handshakes.

While these results already serve as a foundation and indication for the utility and strength of our approach, we believe that further investigation is also merited. For example, while Table 1 seems to indicate that hip twist is less desirable than the other gestures and secret handshakes, it may be the case that the hip twist motion will still provide adequate security in some cases. First, the false positive rates recorded in our study emulate an *ideal attacker* monitoring a person during day-to-day activities, i.e., an attacker capable of placing an RFID reader near the victim card *constantly* throughout these activities such as bicycling. Real attackers might not have such capabilities. Additionally, if we were to weaken our utility goal and not demand near-zero false negatives, then we could set the threshold for hip twist to a higher value, thereby further reducing the false positives during day-to-day activities; we have not, however, experimented with this approach.

It is also important to evaluate the false positive rates in environments other than day-to-day activities. We hypothesize that some motions — like dancing with one's wallet in their pocket — may generate motions similar to some of our secret handshakes. Such correlations, if found to be true, may not significantly impact the usefulness of secret handshakes in practice since secret handshakes will still limit the exposure of one's tag or contactless card in the common case. Nevertheless, this discussion motivates a new class of attacks that we dub the *ghost-and-dancer* or *dancing-leech* attack, in which one attacker attempts to coerce a person into moving in a particular manner, perhaps by encouraging him or her to join in a dance in which the attacker is able to directly or indirectly manipulate the card's motion.

Lastly, we must consider more sophisticated adversaries that attempt to not only relay the the communications between the RFID tag or contactless card and the reader, but adversaries who attempt to directly modify the physical state on the tag or card. For example, since the WISPs are powered by the energy received from the RFID reader, we must ask whether a malicious RFID reader could influence the accelerometer sampling frequency, clock, or some property of the WISP. While this certainly appears infeasible given off-the-shelf readers, our preliminary analyses also suggest that such attacks would be difficult to mount even

with custom RFID readers. This is because the voltage from the WISP power harvester is regulated at approximately 1.8 volts, keeping the microcontroller's oscillator stable. If 1.8 volts is not available, the voltage supervisor resets the microcontroller before it is able to affect the operation of the WISP.

5.4 Survey of Access Card Usages

We conducted a survey of how students, faculty, and staff at the University of Washington use access control cards to gain entry to buildings, rooms, and other secured locations. A total of 554 people responded to this survey. 44 had never used an access card or access badge and were hence excluded from the rest of the survey. Of the 510 remaining participants, 354 were male, 150 were female, and 6 did not provide a gender. These participants were also predominantly in the 18–28 year range (403 participants), with 53 participants aged 29–38, 21 participants aged 39-48, 24 participants aged 49–58, and 5 participants aged 59–68; 4 participants did not provide an age.

We asked these 510 participants what kind or kinds of access cards they have or have had in the past: contactless, contact-based (e.g., magstripe or smartcard), or both. Of these participants, 208 are currently or have used one or more contactless access cards, 248 are currently or have used one or more contact-based access cards, and 6 have used one or more of both types of cards; 48 did not provide an answer.

Contactless Card Location. We then asked the participants a sequence of questions to ascertain where they keep or kept their access badges. Since one of our primary research goals is to protect against the ghost-and-leech attach while maintaining approximate consistency in the usage model for RFID tags and contactless cards, it is critical for us to ascertain how people are currently using their contactless access cards. Therefore, while we asked these questions of all participants, we focus the following discussions on *only* those participants that are currently using or have used contactless access badges in the past. In order to avoid double counting individual participants who had multiple cards (and perhaps kept them in the same or different locations), we focus the following on participants that only ever used one such contactless access card. This left 191 participants.

The questions were hierarchical in nature, structured in a way that would allow us to determine the precise location in which the participants normally carry their access cards. For example, if a participant indicated that he or she kept his or her access card in a wallet, we subsequently asked that participant where he or she usually kept that wallet (e.g., in a front pocket, in a back pocket, in a backpack, in a purse, around the neck, or in some other location).

Figure 5 presents a summary of these results. Of the 191 participants that qualified for this portion of the study, 123 (64.4%) kept them in their wallet, 32 (16.8%) loosely jammed the card in their pocket, 21 (11%) wore them on a lanyard around their neck and above their clothes, 10 (5.2%) loosely jammed them in their purse, and 5 (2.6%) wore their card on a lanyard below their clothes.

Contactless Card Usage. We next asked the 191 participants who have only used one contactless access badge how they use or used that access badge to gain entry to a physical resource. (For example, do participants with contactless access cards wave their wallets in front of the badge reader? Do they remove their cards from their wallets before pressing them to the badge reader? Etc.)

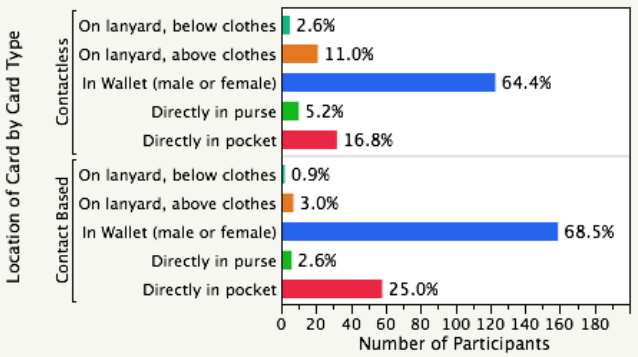

Figure 5: Locations where survey participants keep or kept their access cards by card type

Figure 6 presents a summary of these results. Over a third (31.9%) of the participants authenticated by taking their wallets out of their pockets and pressing their wallets against the card reader *without* removing their card from the wallet. Less than a fourth (24.6%) of the participants removed their card from their wallet, purse, or pocket and brought it near the reader. These results do indeed support our prior claim that a secret handshake, which would not require a user to take the access card out of their wallet, would be beneficial to a large portion of contactless card users.

Contact-Based Card Location. Our study included 232 participants who had never used a contactless card, but do have an electronic access card (e.g., magstripe). Again these results only include participants with one such card. Of these participants 159 (68.5%) kept the card in their wallet, 58 (25%) kept the card in their pocket, and 15 (6.5%) of the participants kept the card in a purse or lanyard. Figure 5 illustrates these results. If given a contactless card, these users may want to keep their card in the same location. Consequently, a secret handshake would allow these users to do just that, while even simplifying their usage model.

Security Versus Efficiency. We also asked the survey participants whether it was more important that their access card worked quickly or worked securely — where securely, in this context, was explained to be the inability of an attacker to clone their access card or gain entry as them. 287 (67.85%) people said that its important to be quick, while 92 (21.80%) said it's important to be secure; 44 people chose 'other' or did not respond. The distributions are similar for both the participants that had contactless cards and those that didn't. This highlights the fact that security administrators and users are often at odds with one another concerning authentication – users want speed and utility, administrators want security. We believe that secret handshakes can provide a nice compromise by making authentication more secure without incurring a major usage model modification.

Survey Summary. Our survey consisted of 554 people from a variety of backgrounds and ages. Although, for many

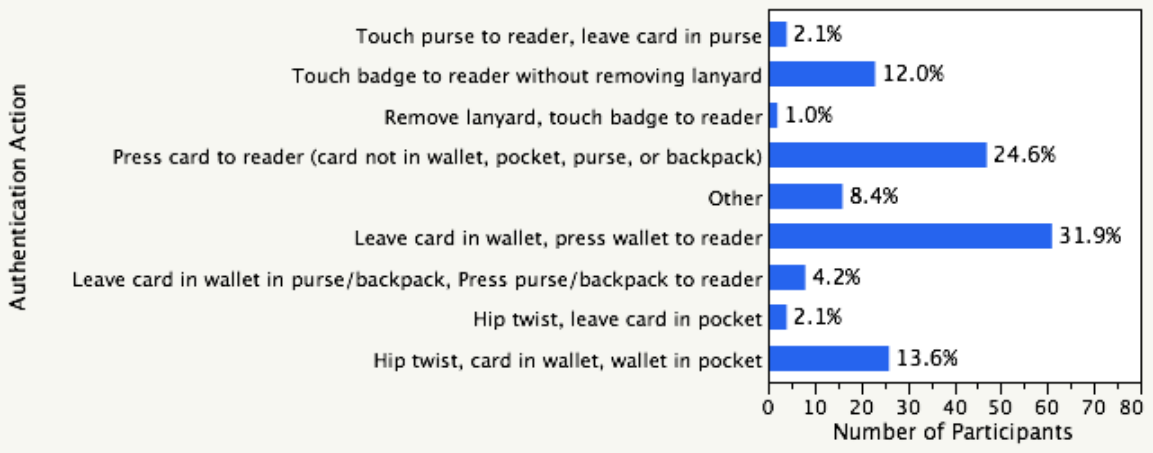

Figure 6: How survey participants use or used their contactless access cards.

of the question we did not have 554 total replies[1], the survey revealed several interesting facts:

1. Users mostly keep their access cards in their wallet.
2. A large portion of the users who have contactless access cards do not remove them from their wallet during authentication.
3. Users overwhelmingly value quickness over security.

These observations strengthen our proposal by showing that speed is important and that secret handshakes will be only negligibly impact the current authentication model.

6. CONCLUSIONS

Contactless technologies, including smartcards, proximity cards, and RFID-based devices, are constantly increasing in ubiquity in such applications as access control and payment systems. These contactless technologies are vulnerable to a class of relay attacks called *ghost-and-leech* attacks. In this paper we have presented a novel approach for adding a level of protection against the ghost-and-leech attack by limiting the context in which the contactless card can communicate with a reader. We show that by asking users to perform *secret handshakes* — short, small gestures with their cards during their authentication session — we are able to increase the resistance of the contactless cards to ghost-and-leech attacks without significantly altering the usage model.

We built a prototype that modeled an RFID access control system and used our approach for limiting the communication between card and reader to valid authentication contexts. We found that several secret handshakes exist that result in small false positive and false negative rates. Furthermore, we have shown that it is possible to fully implement our model on passive RFID tags, such as WISPs. This makes our solution backwards compatible with current RFID readers and minimizes cost of implementation and adoption.

To provide further context for our research, we surveyed more than 500 participants concerning their access card usage. We discovered that our approach would have negligible effect of the usage model for current access cards. We found that since all secret handshakes execute in less than 1 second, we are able to meet the survey participants' desire for speed during the authentication process. While our solution is by no means a panacea for all types of surreptitious access card reads or relay and ghost-and-leech attacks, it is also composable with other defensive mechanism, thus providing greater defense-in-depth in order to fortify and protect against these attacks.

[1]This was due to our institution's human subjects policies, which state that a research subject may selectively opt out of answering any survey question at any point in time.

7. ACKNOWLEDGMENTS

We wish to thank Daniel Yaeger for his assistance with the WISP, and Cynthia Matuszek for her helpful feedback on our early drafts. We thank Kevin Fu for suggesting the term "Secret Handshakes."

8. REFERENCES

[1] Identity Stronghold website. `http://idstronghold.com/`.
[2] Privaris plusID products. `http://www.privaris.com/products/index.html`.
[3] SMARTCODE solves the privacy issue relating to potential unauthorized reading of RFID enabled passports and ID cards. `http://tinyurl.com/ypodsz`.
[4] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. In A. Ferscha and F. Mattern, editors, *Proceedings of PERVASIVE*, 2004.
[5] S. C. Bono, M. Green, A. Stubblefield, A. Juels, A. D. Rubin, and M. Szydlo. Security analysis of a cryptographically-enabled rfid device. In *SSYM'05: Proceedings of the 14th conference on USENIX Security Symposium*, pages 1–1, Berkeley, CA, USA, 2005. USENIX Association.
[6] H.-J. Chae, D. J. Yeager, J. R. Smith, and K. Fu. Maximalist cryptography and computation on the WISP UHF RFID tag. In *Proceedings of the Conference on RFID Security*, July 2007.

[7] T. Choudhury, G. Borriello, S. Consolvo, D. Haehnel, B. Harrison, B. Hemingway, J. Hightower, P. P. Klasnja, K. Koscher, A. LaMarca, J. A. Landay, L. LeGrand, J. Lester, A. Rahimi, A. Rea, and D. Wyatt. The mobile sensing platform: An embedded activity recognition system. *IEEE Pervasive Computing*, 7(2):32–41, 2008.

[8] J. Conway. *On Numbers and Games*. Academic Press, 1976.

[9] N. Davies, D. P. Siewiorek, and R. Sukthankar. Activity based computing. *IEEE Pervasive Computing*, 7(2):20–21, 2008.

[10] S. Drimer and S. J. Murdoch. Keep your enemies close: Distance bounding against smartcard relay attacks. In *16th USENIX Security Symposium*, August 2007.

[11] EPCGlobal. Class 1 Generation 2 UHF Air Interface Protocol Standard. `http://www.epcglobalinc.org/standards/uhfc1g2`.

[12] D. Gafurov, K. Helkala, and T. S?drol. Biometric gait authentication using accelerometer sensor. *Journal of Computers*, 1(7):51–59, 2006.

[13] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno, and W. H. Maisel. Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses. In *IEEE Symposium on Security and Privacy*. IEEE Computer Society, May 2008.

[14] G. Hancke. A practical relay attack on ISO 14443 proximity cards, 2005. `http://www.cl.cam.ac.uk/?h275/relay.pdf`.

[15] G. P. Hancke and M. G. Kuhn. An RFID distance bounding protocol. In *Proceedings of IEEE/Create-Net SecureComm*, 2005.

[16] T. S. Heydt-Benjamin, D. V. Bailey, K. Fu, A. Juels, and T. O'Hare. Vulnerabilities in first-generation RFID-enabled credit cards. In *Proceedings of Financial Cryptography and Data Security*, 2007.

[17] A. Juels. RFID security and privacy: A research survey. In *IEEE Journal on Selected Areas in Communications*, 2006.

[18] A. Juels, R. Rivest, and M. Szydlo. The blocker tag: Selective blocking of RFID tags for consumer privacy. In *10th Annual ACM Conference on Computer and Communications Security*, 2003.

[19] D. Kaminsky. Soylent badges: An attack surface analysis of RFID, 2007. `http://www.law.washington.edu/LCT/Events/rfid/Dan_Kaminsky-RFID-Attack-%Surface.pdf`.

[20] Z. Kfir and A. Wool. Picking virtual pockets using relay attacks on contactless smartcard systems, 2005. `citeseer.ist.psu.edu/kfir05picking.html`.

[21] B. Logan, J. Healey, M. Philipose, E. Munguia-Tapia, and S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *Proceedings of Ubicomp*, 2007.

[22] R. Mayrhofer and H. Gellersen. Shake well before use: Authentication based on accelerometer data. In *Proc. Pervasive 2007: 5th International Conference on Pervasive Computing*. Springer-Verlag, May 2007. *to appear*.

[23] MIT Auto-ID Center. 860MHz - 930MHz Class I Radio Frequency Identification Tag Radio Frequency & Logical Communication Interface Specification Candidate Recommendation, 2002. `http://tinyurl.com/2ebjx7`.

[24] D. Molnar and D. Wagner. Privacy and security in library RFID issues, practices, and architectures. In *11th ACM Conference on Computer and Communications Security*, 2004.

[25] S. N. Patel, J. S. Pierce, and G. D. Abowd. A gesture-based authentication scheme for untrusted public terminals. In *UIST '04: Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 157–160, New York, NY, USA, 2004. ACM.

[26] A. P. Sample and J. R. Smith. A low-cost capacitive touch interface for passive RFID tags. Submitted for publication.

[27] J. R. Smith, A. P. Sample, P. S. Powledge, S. Roy, and A. Mamishev. A wirelessly-powered platform for sensing and computation. In P. Dourish and A. Friday, editors, *Ubicomp*, volume 4206 of *Lecture Notes in Computer Science*, pages 495–506. Springer, 2006.

[28] A. Varshavsky, A. Scannell, A. LaMarca, and E. de Lara. Amigo: Proximity-based authenticaiton of mobile devices. In *Proceedings of Ubicomp*, 2007.