CipherCard: Enhancing Security on Common Touchscreen Devices using Two-factor Authentication

Teddy Seyed[1], Xing-Dong Yang[1], Anthony Tang[1], Saul Greenberg[1], Jiawei Gu[2], Bin Zhu[3], Xiang Cao[4]

[1]University of Calgary
Calgary, AB, Canada
{teddy.seyed, xdyang, tont, saul}@ucalgary.ca

[2]Baidu Institute of Deep Learning
Beijing, China
gujiawei@baidu.com

[3]Miscrosoft Research Asia
Beijing, China
binzhu@microsoft.com

[4]Lenovo Research & Technology
xiangcao@acm.org

ABSTRACT

We present CipherCard, a physical token that defends against shoulder-surfing attacks on user authentication on touchscreen devices. Placed over a touchscreen pin-pad, CipherCard remaps a user's touch points on the physical token to different locations on the pin-pad (i.e. as a substitution cipher). It translates a visible *user password* into a different *system password* received by a touchscreen, hiding the system password from observers. CipherCard enhances authentication security through Two-Factor Authentication (TFA), in that both the correct *user password* and a specific card are needed for authentication. We explore the design space of CipherCard, and describe three implemented variations each with unique capabilities. Based on user feedback, we discuss the security and usability implications of CipherCard, and describe several avenues for continued exploration.

Author Keywords

User authentication, two-factor authentication, capacitive touchscreen.

ACM Classification Keywords

H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

INTRODUCTION

Capacitive touchscreens have become the primary input mechanism for many security applications such as access control (e.g. door locks), public kiosks (e.g. ATMs, cash registers, point of sales via large screens), or mobile authentication (e.g. payment through personal mobile devices). Because user authentication through touchscreens is often carried out in a public space, users are susceptible to shoulder-surfing attacks: unscrupulous individuals or cameras can see the password or PIN being entered into the system [1, 11, 14, 17, 23, 28]. Further exacerbating the problem, user interfaces for touchscreens are often designed to be larger (due to the fat finger problem [26]), making it difficult to shield input from observation. As well, the lack of haptic feedback on touchscreens makes eyes-free operation difficult, meaning users cannot easily shield the display from view.

Cite as: Seyed, T., Yang, X-D., Tang, A., Greenberg, S., Gu, J., Zhu, B., and Cao, X. (2014) CipherCard: Enhancing Security on Common Touchscreen Devices using Two-factor Authentication. Research Report 2014-1063-14, Department of Computer Science, University of Calgary, Calgary, AB, Canada T2N 1N4.

To enhance user authentication security on touchscreen devices, we present CipherCard, a physical token that enables two-factor authentication (TFA) on capacitive touchscreen devices. As Figure 1 illustrates, CipherCard is an opaque overlay that is placed atop the touchscreen's password input area (e.g., a touchscreen PIN pad), where it serves as a physical proxy for the touchscreen's original password input UI. When a user touches a button on the CipherCard, the input is remapped to a different button location on the touchscreen. Internal wiring hides the actual input location and mapping from both observers and hidden cameras. CipherCard translates the input sequence ("*user password*") into a distinct sequence ("*system password*") that is received by the touchscreen. For example, Figure 1 illustrates a user entering his user password '1 3 5 8' into CipherCard, which is translated to the system password of '4 1 2 6'. Thus, the system password is hidden from the observer. A shoulder-surfing attacker may acquire the user password, but without the user's CipherCard, the attacker cannot pass authentication.

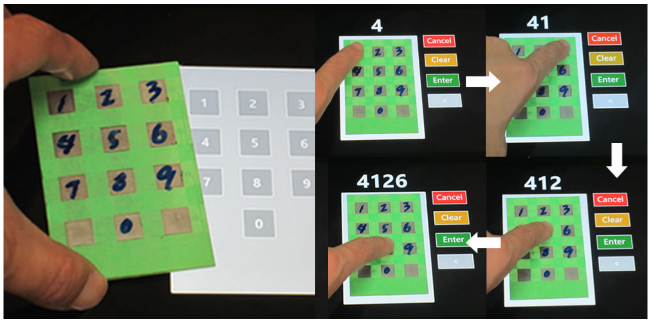

Figure 1. CipherCard maps a touch into a different location.

CipherCard mappings can be permanent (where they are manufactured with a single translation for use on particular systems) or reconfigurable (where a user can specify the translation between user and system passwords). CipherCard allows a user to choose a set of easy-to-memorize user passwords, and use them as proxies for "strong" system passwords (e.g. PINs with random combinations). In either case, if multiple CipherCards are used for different security systems, the user can specify a single user password as a proxy, where that user password is remapped to the particular system password. This minimizes the number of passwords a user needs to memorize for different services or locations. Renewing a system password can be as easy as getting a new CipherCard or reconfiguring an existing CipherCard, where the user password can remain the same.

The CipherCard authentication scheme raises many engineering, security, and usability questions. In this early stage, we focus on a thorough exploration of the design space. We implemented three prototypes: card-shaped, wallet-based, and phone-based CipherCards, each with a unique form factor and usability features. We ran a design study with five usability professionals, following which, we evolved the physical design of our prototypes. Initial user feedback from a usability study suggests that CipherCard concept is easily understood and can be used easily.

Our contributions in this paper are three-fold: (a) a shoulder-surfing resistant two-factor authentication scheme; (b) an exploration of the design space of CipherCard where we implement three prototypes, and (c) two user studies that validate the value of CipherCard and its form factors.

RELATED WORK

CipherCard's design is informed by previous work in three related areas: authentication schemes and techniques that impede shoulder-surfing, two-factor authentication, and password memorability.

Techniques to impede shoulder-surfing

Researchers have developed several authentication schemes and interaction techniques with the goal of hampering shoulder-surfing attacks. These frequently impose extra work on the user, but mainly only guard against casual observation. Most are still susceptible to camera-based attacks, where the input can later be reviewed in detail.

Cognitive trap techniques increase the complexity of the authentication process (e.g. by showing extraneous information). This makes it more difficult for an observer to derive the password [9, 25] from observing a single authentication instance. However, these schemes are still susceptible to camera-based attacks that afford analyzing multiple authentication instances in detail. In contrast, some schemes allow the user to create a customized 3D gesture (e.g. [10, 23]). These have been demonstrated to be challenging to forge from simple video review; however they are still susceptible to automated video capture and analysis from depth cameras.

Hiding PIN input shields against the visibility of input actions. At its simplest form, people can position their bodies/hands to shield their actions from an observer's view [7]; however, most users do not do this [15]. Many technical approaches try to decrease the visibility of password entry. Examples include back-of-the-device PIN entry [14, 17], eye gaze input [13], pressure [7], and haptic/tactile feedback as secret input or output channels to assist password entry [1, 16, 21]. Although resistant to direct human observation, all remain susceptible to video- or audio-based observation attacks. For example, a pressed fingertip can be detected from the change of its color [7]. Similarly, haptic/tactile feedback can be detected from a recorded sound track [16].

Biometric methods distinguish users based on biometric characteristics (e.g. fingerprint, hand geometry, retina, etc. [12] or their behavioral signatures [20]). Biometric methods are effective against video-based attacks but suffer major drawbacks, preventing them from wide deployment in real-world applications. For example, physiological biometric characteristics are not renewable after the attacker has successfully forged them. Furthermore, behavioral biometric signatures are prone to high recognition error, making them impractical [20].

Physical token methods require that the user present a physical object to a reader, and these are immune to shoulder-surfing [19]. However, these require entry systems to have special hardware to detect the physical object, such as an RFID card readers. Recent advances have begun to explore general capacitive touchscreens as sensors; however this is still in its infancy [27]. Of course, a lost or stolen token could still allow individuals to pass authentication.

Two-factor authentication

Broadly, user authentication methods rely on at least one of three factors: knowledge ("something you know", e.g. password or PIN), inherence ("something you are", e.g. biometric characteristics), and possession ("something you have", e.g. physical token). Two-factor authentication (TFA) requires a user to present at least two of these factors [22] to enhance authentication security. For example, most banking machines require both a bank card (possession factor) and a PIN (knowledge factor). This boosts security since attackers need access to both factors to pass authentication. However, TFAs are still vulnerable to man-in-the-middle or Trojan attacks [22]. While early research suggested that TFA introduced a substantial usability burden [5], TFAs are becoming increasingly common in daily life [4] (e.g. users now embrace TFA in their daily use of bank and credit cards). Furthermore, while TFA was initially used in government agencies and enterprise, it is now widely used by many consumer services (e.g. Google, PayPal, Facebook now offer TFA for login) in response to attacks on user accounts.

Memorizing multiple passwords

While there are many authentication methods that arguably provide more security, passwords and PINs are still the most prevalent form of individual authentication. To increase the security of the scheme, many authentication systems require the user to use strong passwords that: 1) cannot be found in dictionaries; 2) have a minimum length; and 3) include a random combination of letters, numbers or special characters. Strong passwords are less susceptible to guessing and cracking. However, they are less memorable [29], and represent a trade-off between usability and security [31]. Consequently, people often choose to write down their passwords, or reuse them for multiple service accounts [3], and rarely renew them [31]. Graphical passwords [24] were proposed to tackle this problem, but memorizing multiple graphical passwords still remains a challenge for users.

We will show that CipherCard as a TFA method addresses security by hiding the actual system password, while still allowing users to create easy-to-remember personal passwords.

CIPHERCARD CONCEPT

CipherCard provides two-factor authentication capabilities to existing password-based authentication on capacitive touch-sensing devices. Users place the CipherCard on a touchscreen, and use it as a pin-pad to enter PINs. When touching the front side of the card (e.g., a button), the card generates a touch point on its back, at a different input location on the screen underneath. It is thus a substitution cipher, and this cipher is either a randomly preset or user specified permutation between the two sets of locations on each side of the card. The touch input sequence on the CipherCard (*user password*) is translated to another unique sequence that is sent to the touchscreen (*system password*). The system password is never exposed. So long as the touchscreen UI and the card geometry are compatible, one CipherCard can be reused for an arbitrary number of different PINs for different applications, without requiring special purpose card readers.

CIPHERCARD DESIGN SPACE

The CipherCard concept can be realized in a number of different ways. We articulate the factors that describe this design space, and the trade-offs these present.

Passive *vs.* active

CipherCards can be made either passive or active. A *passive* CipherCard translates touch via electrical wiring, and requires no battery or external power source. The passive CipherCard can be cheap to design, produce, and customize for various touchscreen devices and authentication UIs. It can be disposed and replaced when a user needs a different pattern (e.g. to renew a system password) at a minimal cost. It can also be made reconfigurable (e.g., via jumpers) with more engineering effort and monetary cost. A passive card must be made to match a particular touchscreen layout and is not scalable to UIs with different layouts or button sizes.

In contrast, an active CipherCard receives user touches, and uses a control circuit and electrodes to remap those to the touches matching those required by the capacitive sensor on the authentication device. This mapping can be reconfigured through software, giving the user control over the substitution cipher. Furthermore, it is possible to generate complex mappings, where a single touch on the front side generates multiple fake “touches” on the back side (i.e. a *1-to-m* mapping). An active CipherCard has chips, circuits, and software, and thus is more costly.

Input/output resolution

The input and output resolution of CipherCard is determined by the number of electrodes on either side of the card. The output resolution of CipherCard is also determined by authentication UIs and the input required. We restricted our early explorations to simple PIN pads of 10 electrodes (to enable 0-9 number entry); however, it is possible to scale CipherCard keypads with more keys, and even to gestural entry, given higher resolution and electrode density. An active card can utilize higher output resolution to be able to scale to different UI layouts and button sizes (within the card’s physical dimensions).

Form factor

CipherCards should be easy to carry and deploy. It can resemble a credit card that a user can carry in their wallet, or an ID tag worn on clothing. Alternatively, it can be integrated into flat daily personal belongings, e.g. a wallet or phone case (one that flips open) or even an existing bank or credit card. This avoids the need of carrying an extra card. Integrating a passive card into personal belongings can be relatively easy due to its simplicity, but can be challenging for an active card without significantly impacting the normal usage of the personal item. Finally, an active CipherCard can be integrated into existing personal electronic devices, e.g. smartphones or tablets.

CIPHERCARD PROTOTYPES

We developed three proof-of-concept prototypes based on this design space: a passive credit card sized prototype; a passive wallet-based prototype, and an active smartphone-based prototype. These prototypes are described below and were used in both our feasibility and usability studies.

Card-shaped CipherCard

CipherCard works based on the notion that touch input can be simulated by any conductive object in contact with a capacitive sensor and electrically connected to the user’s hand (or body). We implemented a card-shaped passive CipherCard using a printed circuit board (PCB). Each side of the card contains an identical 3×4 grid layout of electrodes (Figure 2a). Each electrode on the front side is discretely connected to an electrode on the back. Connections can be either randomized or pre-specified by the user (so they can use a particular user/system password mapping) at the time of manufacture. The electrodes (1×1cm) and connecting paths (0.025cm width) were printed using a thin layer of tin. To prevent attackers from deciphering the mapping by visual inspection, CipherCard must be constructed in a manner that hides the connecting paths (e.g. by a surface material or by using a multi-layer PCB design). In our prototype, the connecting paths were covered by paper tape.

To connect the electrodes on both sides, we used tin-coated holes ("vias"). For each electrode on the bottom, we connected it to a via (diameter: 0.2cm and hole size: 0.071cm) placed 0.1 cm away from its edge (Figure 2a). Connecting an electrode to a via from the top connects it to the corresponding electrode on the bottom. Finally, the connecting paths were covered by tape to shield the connection pattern from visibility on the outside of the card. The finished prototype measures 8.6×5.4×0.15 cm (L×W×H), only slightly thicker than a standard credit card, and can be easily carried in a wallet.

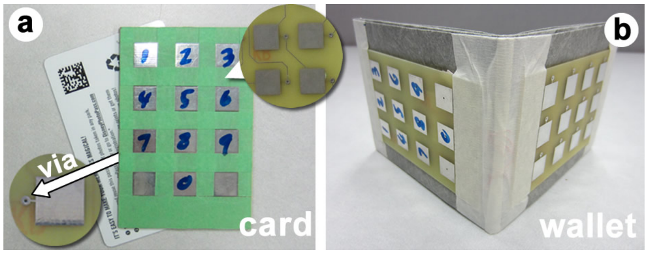

Figure 2. Left: Card-shaped CipherCard with a 3×4 electrode grid. Top callout shows the internal wiring and bottom callout shows the via and electrodes on the back. Right: Wallet-based CipherCard.

Wallet-based CipherCard

To demonstrate that CipherCard can be integrated into a daily personal belonging, we built a second passive prototype based on a conventional wallet (Figure 2b) of size 10×8.2cm when folded. Similar to the card, each side of the wallet has a 3×4 grid of electrodes, where one side's electrodes are connect to the other side through copper wires. Our prototype uses two hard plastic boards, but we expect a deployable version to use flexible materials, for example, tin paths printed on PET film. It is important that the deployable version preserves the appearance, feel and functionality of a wallet.

Smartphone-based CipherCard

We explored the feasibility of an active CipherCard by creating a smartphone prototype (Figure 3). We used an HTC 8X Windows Phone as our platform. Input is handled by the phone's native touch input API, and passed via WiFi to a Spark Core development board. The Spark Core drives a 3×4 grid of electrodes printed on a plastic board. A touch is simulated by programmatically connecting one of the pins of the Spark Core to the ground (e.g. configuring the pin as output and set its voltage to 0V). We found that the ground of Spark Core could not reliably trigger a touch, and therefore solved this by connecting the battery jack to the phone body: when the phone is held by user's hand, the Spark Core is grounded through the user's body, and generates simulated touch points reliably. While our prototype is unwieldy, we expect that the deployable version would integrate the logic into the phone hardware, and integrate the simulated touch circuitry into the phone's body.

To simplify our explorations, we constrained the output resolution of our CipherCard prototypes to a fixed PIN layout. However, further engineering efforts would allow resolution of the electrodes to be significantly increased (e.g. 20×20 2×3mm electrodes), thus taking advantage of the higher input resolution available from the smartphone.

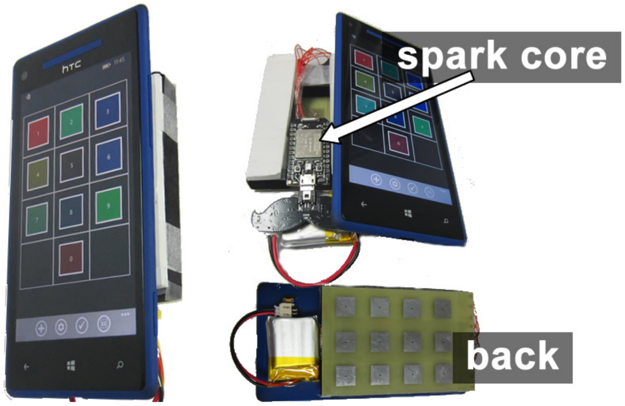

Figure 3. Phone-based CipherCard.

RECONFIGURING CIPHERCARD MAPPINGS

In this section, we present our reconfigurable designs for both passive and active CipherCard.

Passive CipherCard reconfigurable design

Completely passive cards are cheap to produce. However, they must be replaced when a user needs a different pattern (e.g., to change the desired user or system password). It would be more convenient to design CipherCard so they could be reconfigured on the fly. We designed (not implemented yet) one possibility, illustrated in Figure 4, which allows a user to reconfigure the connection pattern by rearranging the positions of the electrodes on one side (here the front side). Using a 3×3 grid layout as an example, each electrode (numbered *A-I* in Figure 4) on the front side can be freely removed and re-plugged into any of the 9 sockets (numbered 1-9), and the permutation order that is plugged in, intuitively defines how each socket location maps to one of 9 fixed-position electrodes on the back side (numbered *a-i*). To make this possible, we must provide a mechanism to ensure the same removable front-side electrode (e.g., *A*) always connects to the same back-side electrode with the matching letter (*a*), regardless of which socket it (*A*) is plugged into. This is enabled by having 9 small conductive pins (3×3) inside every socket. Each pin is hardwired to one of the back-side electrodes with the corresponding relative position (e.g., top-left for *a*). Each front-side electrode has only one pin on its bottom, the relative position of which corresponds to that of the back-side electrode with the matching letter (again top-left for *a*). Thus, whichever socket that electrode *A* is plugged into, its pin always contacts the socket pin that connects to electrode *a*, and so on. Therefore, changing the electrode for a certain socket position will change the position of its associated touch point seen by a touchscreen. With this simple design, a CipherCard pattern can be reconfigured easily by switching positions of removable electrodes.

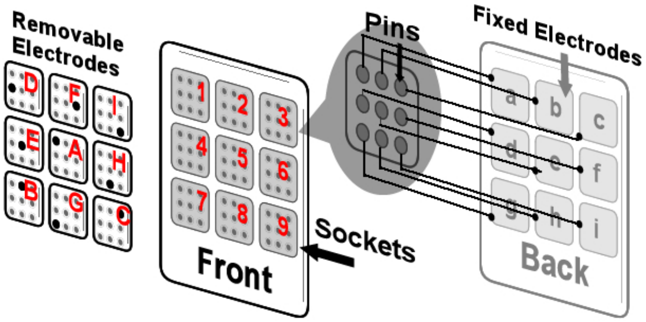

Figure 4. Design of a reconfigurable CipherCard.

Mobile app for reconfiguring active CipherCard

We also implemented a Windows Phone app, which illustrates one interface for reconfiguring the previously described active CipherCard. By default, our app shows a 10 key PIN pad when it starts (Figure 5a). The touch locations (e.g. electrodes) are shown to guide the reconfiguration of the key mappings. To change a key mapping, the user drags a number key to inside a desired touch location (Figure 5b). This way, when the key is tapped, the corresponding touch location is triggered (Figure 5c). Once the configuration is confirmed, the activated touch location is highlighted. When needed, the user can add or remove a number key. To configure a *1-to-m* mapping, the user can duplicate a number key, and then drags each of the duplicated keys to a desired location (Figure 6 right). Once done, tapping the number key triggers the associated locations in the sequence that the keys were created.

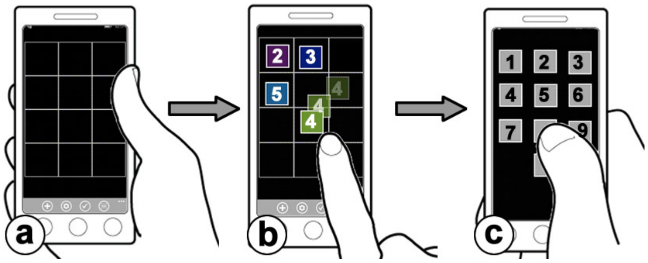

Figure 5. (a) grid shows the position of the electrodes; (b) dragging a number key to inside a desired electrode to change a key mapping;(c) a finished configuration.

Although not implemented in the hardware, our software also supports configurable key sizes and layouts. First, a variety of standard keyboard layouts can be selected, where each matches the keys and layout of a particular touch-screen security system. Second, layouts can be designed from scratch, although the interactions to do so are more complex. For example, the user can specify the dimensions of the key, and drag it to a desired location. The software also allows the user to scale key sizes using pinch gestures. Our software also automatically identifies the candidate location(s) that need be triggered for simulating a touch at the position of the key. To do so, the user first specifies the resolution of the touch locations (or electrodes). The software then walks through the locations and associates one with the key, which has the largest overlap with that key (Figure 6 left). Notice that most of the capacitive sensors ignore touches that are smaller than a threshold size (e.g. 3 mm for Microsoft Surface). To accommodate this, our software triggers all the electrodes (if smaller than 3 mm) that reside inside the key.

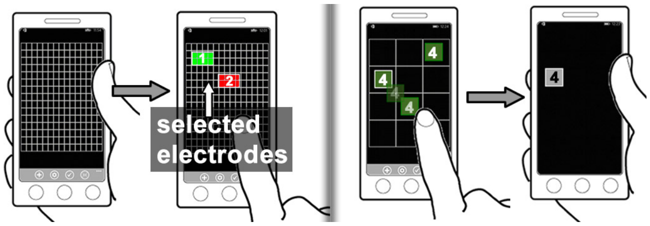

Figure 6. Left: configuring a grid of 15 × 15 electrodes; Right: duplicating a number key to configure a *1-to-m* mapping.

Authoring a new card

At the current stage, passive CipherCards can be designed using popular circuit software (e.g. Altium), and built using a standard PCB. A much easier way is to print them on paper using a home printer and conductive ink [6]. This would allow CipherCard to be widely adopted for home and office use.

SECURITY ANALYSIS

Our assumption of a threat model is based on real world threats, under which a shoulder-surfing attack may take place. We assume the user is in a public environment fully controlled by the attacker, who has hidden a number of high-resolution cameras in that environment. The cameras videotape, from different angles, the user's actions on a software PIN pad on a capacitive touch-sensing device. Multiple authentication sessions of the same user can be recorded, which can then be reviewed in order to extract the PIN. We also assume that the adversary has direct access to the authentication system but doesn't possess a CipherCard.

By combining two authentication factors into a single action, CipherCard boosts security of user authentication on capacitive touch-sensing devices. An attacker who has observed the user password is unable to pass authentication without possessing the user's CipherCard. Copying the card configuration is also extremely difficult without physically possessing it. CipherCard can even be deployed in a manner where *the system password is not revealed to the user*. This means the user would only know their user password, which in turn mandates having a CipherCard (and thus TFA) for authentication. In turn, this makes the user immune to a majority of the social engineering attacks [18], where the user is susceptible to revealing their user password that is frequently a system password, to an attacker. With CipherCard, a user does not know the system password that is mapped to the card. This means that an attacker is unable to perform harmful actions without a CipherCard.

Similar to all TFAs, losing both factors to an adversary will grant access to protected service or location. Hacking into an active card, e.g. user's phone, or breaking into a computer which stores translation files may disclose user's password mapping to adversary. Additionally, individuals or organizations that design or have access to the design of the password mapping may also impose risk to the security of CipherCard.

CipherCard does not prevent attacks directly on the authentication device. With the *1-to-1* translation of a password, CipherCard does not increase the overall entropy (i.e. the total number of possible authentication inputs seen by the system) [2]. We therefore assume the original password mechanism on the device is sufficiently strong on its own (e.g., with an appropriate password length and a limited number of trials) against direct attacks, e.g. brute-force attacks (i.e. enumerating all possible passwords) [8]. CipherCard however protects against dictionary attacks – a user may choose a common word as the user password, yet the translated system password is highly unlikely to be in the dictionary of guessed passwords. The entropy of the *1-to-1* mapping is equal to the total number of permutations of the n electrodes, i.e. $n!$. For example, a 3×3 grid layout offers $9! = 362880$ unique CipherCard patterns. In contrast, a *1-to-m* translation of a password increases the overall entropy by allowing a longer system password, thus providing a higher level of security. Note that m can vary for each character in the user password. Brute-force attacks can be extremely difficult if the length of the system password is unknown to the attacker, and can be thwarted simply by limiting the number of incorrect entries and/or by introducing time delays between attempts. Notice that if a chosen user password is easy to guess, the level of security of TFA can be reduced. For example, in an extreme case of *1-to-m* mapping, a user password can contain only one character, which serves as a shortcut to a longer system password. This way, the knowledge factor (i.e., the user password) becomes extremely easy to obtain. Even so, the attacker would still need to somehow take possession of the CipherCard.

CIPHERCARD USAGE

CipherCard makes it possible for people to choose easy-to-remember PINs. Furthermore, CipherCard makes it possible to have a single user password associated with different system passwords, either through the use of multiple passive CipherCards set to that user password (but each with a different mapping, generating different system passwords), or a single active CipherCard (which would change the mapping based on the service). This allows a user to reuse passwords for multiple accounts [31] without significantly impacting security. We describe three password management scenarios that arise because of this.

Changing system passwords. Renewal of passwords is commonly enforced by organizations to enhance security, but places undue burden on users to generate or remember new passwords. CipherCard allows for refreshed system passwords while allowing users to continue using their existing user passwords in two ways: they can use a new (passive) CipherCard, or the internal mapping in the CipherCard can be changed. Either way, a given user password now maps to a new system password without loss of security.

Changing a user password. Many systems allow the user to change their system password after they have correctly authenticated. After selecting such an option, a user can simply place the CipherCard on the authentication UI, and enter a new user password. This, in turn, generates the new system password.

Setting a user password based on an existing system password. In many scenarios, a system password is shared by many people (e.g. door entry); in this case, it is desirable to keep the system password, but also allow for a mapping between this and a user-chosen password. Because different CipherCards can generate the same system password from different user passwords, this becomes easy with our reconfigurable designs. The actual mapping between the two can be done by any of the previously described methods.

STUDY 1: CONCEPT FEASIBILITY STUDY

We conducted two studies to evaluate the concept of CipherCard, our designs, and identify potential usability issues. At this early development stage, we deem our studies a necessary step towards refining and improving the CipherCard concept before they can be deployed and studied in the field. The main goal of our first study was to identify usability issues of the three prototypes. We were also interested in perceptions of the security of the concept.

Participants

We recruited five professional usability engineers (25-40 years old) from industry. Two participants had one year of industry UX experience, one had >3 years, and two had >5 years of experience.

Apparatus and Procedure

At the beginning of the study, we showed the participants the three CipherCard variations, e.g. card-shaped, wallet-based, and phone-based CipherCard. We then walked them through three CipherCard usage scenarios: entering a PIN into 1) a touchscreen door lock, 2) a public kiosk, e.g. ATM and POS terminal, and 3) a personal mobile device (e.g. a tablet). To simulate the ATMs or door locks, we used Microsoft Surface tablets positioned in different ways. For example, to simulate a door lock, the tablet was hung on a vertical surface. To simulate a public kiosk, the tablet was tilted 35° on a desk. To simulate a mobile scenario, the participant was asked to hold the tablet using their nondominant hand and authenticate using CipherCard with the other hand. For each usage scenario, the participants were asked to enter a 4-digit PIN into a PIN pad application running on the tablet. After a PIN was entered, the application indicated whether the authentication succeeded or failed. Participants were encouraged to put themselves in the mindset of someone using these systems in real-life usage situations (e.g. taking the card from their pocket before use). They were allowed to try and use the prototypes for as long as they wanted prior to a questionnaire (7-point Likert scale) and an interview.

Results

Overall, the participants welcomed CipherCard as a method to resist shoulder-surfing schemes. Their feedback confirmed the merits of the prototypes, e.g. security and portability, but also identified issues that may cause cognitive overhead.

Merits of CipherCard

Security. All of the participants perceived CipherCard as more secure against shoulder-surfing schemes than current practices (median response: 6, with 7 being the most secure), e.g. directly entering the PIN. For participants who had expressed interest in using CipherCard (P1, P5), they found it highly attractive to have an extra layer of security. Some of the positive comments included "*I shield my PIN entry, it is my habit but I don't feel I have to (with CipherCard)" -P1* and "*I see myself using CipherCard to unlock my door because now I have the security of a bankcard, if someone wants to break into my house, they need to get my card as well."-P5*.

Portability. All prototypes were rated highly portable, (e.g. card: 7, wallet: 7; phone: 7—all median responses, with 7 being strongly agree) regarding the convenience of carrying the devices around. For example, P1 commented that it would be convenient to carry the phone-based CipherCard because, *"It is something I carry around anyways."* The wallet received similar comments, e.g. *"I don't have to carry something else as I already carry one"-P5.* While the card-shaped CipherCard is considered an extra burden (i.e. a new thing to carry), our participants found it easy to carry as well: "*I have a lot of cards anyways, so I don't think if carrying a lot of them (cards) will be an issue*" *-P1*, and *"I will be ok to carry it around if the credit card company decides everybody needs to"-P3.*

Issues that cause cognitive overhead

UI alignment. Prior to entering the PIN, CipherCard needs to be physically aligned with authentication UI. This was seen as an unwanted extra step. Among the three prototypes, the card-shaped design was the easiest to align, while the rest were initially challenging for the first time users. Mis-alignment had resulted in touches being unregistered on the touchscreen, which caused substantial frustration for the participants.

*Slippery screen*s. Touchscreens are slippery. This had made alignment even more difficult. The participants had to spend extra effort when holding the prototypes steadily, especially when the screen was tilted. The participants also worried about dropping their phone when the screen was tilted.

Two-handed operation. Entering a PIN on the prototypes while making sure the device did not slide required using two hands. Two-handed operation introduced unnecessary effort for the participants to prepare for using the card. For example a participant commented that, by requiring two hands, "*I will have to put my bag down and use both hands to operate"-P1*. Additionally, the holding hand sometimes occluded buttons that a participant wanted to tap.

Orientation. The translated output locations are dependent on the orientation of the card and the side that is used for input. The phone- and card-based prototypes have clear visual affordance, which had made it easier for the participants to identify the desired side and orientation to use. However, the wallet is symmetrical in its appearance, thus requiring extra effort from the participant to figure out the right direction.

Preparation effort. All of the aforementioned issues had introduced unnecessary preparation efforts from the users prior to entering the CipherCard's user password. Overhead also includes the effort to take out the device from where it is carried. The card-shaped device is less convenient than the other two prototypes in the sense that the users will have to take out the wallet first (assuming the card is not worn as an id tag). With the phone, the participants noted more overhead, as they first had to unlock the phone, open the app, and then search for a desired card mapping.

IMPROVED DESIGN

After reviewing the results from the first study, we devised a solution to resolve some of the most outstanding issues. Our solution allows the user to snap CipherCard into the right position on the screen without aligning or holding by hand. This can be achieved by attaching a card holder on top of the touchscreen PIN pad. The card holder guides the position of CipherCard, holds it onto the screen, and aligns it properly. Alternatively, magnets can be attached to the card and screen to achieve the same goal, while preserving the flatness of the screen. We implemented a prototype on-screen card holder to demonstrate the idea (Figure 7). To use it, the user simply slides the card-shaped prototype into the holder from the top and enters a PIN. This allows single-handed operation without the need for user alignment.

Phones and wallets are more problematic, as they do not have a uniform form factor. Thus, an on-screen holder may not work for them. An alternative approach could be letting the software PIN pad to adjust its position and orientation to align with CipherCard. This can be achieved for example, by adding spatial tags to CipherCard. These allow the touchscreen to identify its position and orientation [30]. The user can snap CipherCard onto the screen (e.g. using magnets) in an arbitrary orientation. The software pin-pad aligns with CipherCard automatically. It can also adjust its button size and layout to fit those of CipherCard. We can envision many different approaches for developing the snap-in mechanism, but its exploration is outside the scope of this paper. We thus leave it for future work.

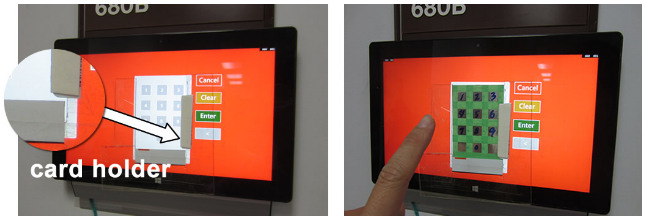

Figure 7 snap-in card holder allows one handed PIN entry

To reduce the preparation time for phone-based CipherCard, we implemented a new function, which allows the phone to automatically load a desired mapping by tapping it on a Near Field Communication (NFC) tag. In circumstances that the size and layout of the touch-screen PIN pad does not match the one on the phone, the software can automatically load a key pad configuration that matches its specification.

Finally, the system software can be used to consider only the relative locations (rather than the absolute locations) of the generated touches, where it can match it against a given pattern. That is, the numeric password is treated as a gestural password. While this means that the CipherCard can be placed anywhere on the system screen, security is somewhat reduced as some key combinations may create the same gestural pattern.

STUDY 2: WORKLOAD EVALUATION

The goal of this study was to verify the concept of our improved design based on feedback from Study 1, as well as the overall usability and security of CipherCard. To focus on the concept rather than the implementation, we mocked up the snap-in mechanism using the on-screen card holder shown in Figure 7. For the wallet and phone, we used double-sided tape (on the back of different CipherCards) to simulate a magnetic snap-in effect.

Participants

We recruited six participants (5 male). All were adult office workers with prior experience using PIN pads and TFA.

Apparatus and Procedure

The apparatus used and procedure followed is similar to Study 1, except with the phone-based prototype, the participants were asked to tap the phone on a NFC tag to load the app before entering a PIN. Participants were trained on the use of the snap-in guide for the card prototype. For the wallet- and phone-based prototypes, we asked participants to snap them onto the software PIN pad and imagine that alignment would be automatically adjusted.

Results

Reduced cognitive overhead

Overall, the participants rated cognitive workload being very low (median response: 1.5, with 1 being extremely low). Figure 8 shows the ratings of mental/physical demand, effort, frustration, and concentration from the two studies. The result of Study 2 indicated the importance of having the snap-in feature before CipherCard can be deployed. It should be noted that the population groups between Study 1 and Study 2 were different, thus potentially explaining the difference in responses.

Participants found the wallet and phone had higher levels of frustration than the card due to the potential danger of exposing them in the public. For example, *"In places that is not safe, I don't want to pull out my wallet because muggings are really common, and there is way too much personal information in the wallet"-p7.* Although tapping a NFC tag still requires extra effort from the users, participants found it much easier to do than searching through an application list on a mobile phone.

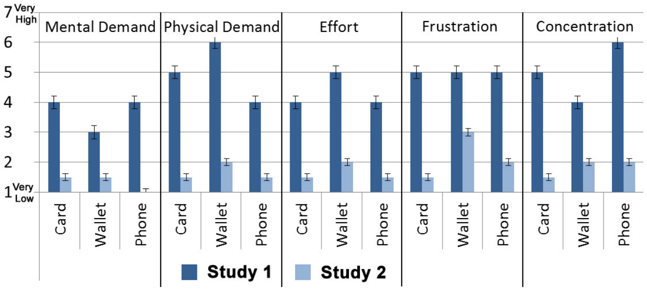

Figure 8 Average responses across participants on several measures of cognitive overhead for Study 1 and 2 (Likert-scales: 1: very low, 7: very high) (Error bar shows standard error)

RESULTS FROM BOTH STUDIES

Security. The results from both studies confirmed that the CipherCard was perceived more secure against shoulder-surfing than conventional PIN entry. When asked if they felt comfortable not knowing their system password, a slight majority of participants (6/11) said they preferred knowing it. Although all understood the security benefit of not knowing the system password (e.g. enforce TFA or protection against social engineering attacks), more participants preferred to have some sort of backup in case a CipherCard was lost, stolen or otherwise not available. Seven of the 11 participants expressed interest in using CipherCard in a public kiosk or door lock. A minority of participants (4/11) expressed interest in using CipherCard on mobile devices for highly secured application (e.g. online banking). Others were less interested, as they felt they had more control hiding their input on a mobile device.

Maintaining multiple passwords. Overall, the participants saw the merits of using CipherCard to minimize the workload of memorizing multiple strong passwords, e.g. median response: 6, with 7 being strongly agree. When choosing between using one CipherCard with multiple user passwords and multiple CipherCards with a single user password, all participants leaned towards using one card. They explained it would be easier than carrying multiple cards. Most participants (7 out of 11 participants) also leaned towards getting a new (passive) card when renewing a system password, as they could benefit by keeping the current user password (assuming the card is associated with only one PIN).

Social pressure. Participants were asked to rate the social pressure they may feel when using CipherCard in front of strangers, friends, and family, who may perceive its usage as an insult. They did not feel social pressure using CipherCard in front of strangers, friends, and family (all median response: 7, with 7 being a strongly disagree that they felt social pressure); They did not think they would feel uncomfortable if others (e.g. stranger, friend, or family member) were to use CipherCard in front of them, e.g. median response: 1, with 7 being strongly mind.

DISCUSSION AND LIMITATIONS

In this section, we discuss the insights and limitations we discovered from our own experiences designing CipherCard.

Change of authentication behavior. While CipherCard does not change the way a user enters a PIN, it changes a user's authentication behavior (i.e., it requires the user to carry the card and put it on the touchscreen prior to entering a PIN). Users may be resistant to this extra work. Like any other security system, users are always the key to ensure the success of CipherCard. While people have been found to be the 'weakest link' in the computer system [18], their security behavior can be changed through education and proper design of security systems.

Convenience vs security. Our studies showed that people do understand the importance of security. However, it is often the case that people sacrifice security for the sake of control or convenience [3]. CipherCard tries to motivate user's security behavior (e.g. using TFA) by providing a set of convenience features. While welcomed by our participants, users need to be aware that some of the features may introduce potential security risks. For example, using a single user password for multiple accounts or updating CipherCards but never changing the user password may reduce the security of TFA. In addition, if an adversary steals a wallet containing multiple CipherCards, all with the same user password, they will be able to access *all* associated accounts if they know the password, even though their system passwords may differ. Future work needs to focus on convenient techniques without impacting security.

Size and layout constraints. Our vision requires a CipherCard that is easy to carry, which – even considering our three form factors – suggests it should be modest in size. Yet authentication UIs can present quite large keypads for input, where its area is much larger than expected CipherCard sizes. Similarly, for the electrodes to work, they must match the touch key locations on the screen, which may not be the case without extensive CipherCard configuration. This introduces a mismatch between the two. This can be solved by creating a standard that specifies particular layouts and a maximum size of the input area on the touchscreen, where the CipherCard would also conform to that standard. While a low cost solution, pin-pad vendors and manufacturers would need to be willing to accept the standard.

Modification of the existing authentication device. The snap-in technique needs to be developed before CipherCard can be deployed in the field. This, however, requires minor augmentation of the existing hardware and/or software, which would increase the cost of deployment.

Configurability. While we described how passive and active CipherCards can be reconfigured by end-users, we do not address this extra work as part of our user studies. Reconfiguration will require extra learning and work, and it is unclear how many users will want to assume this overhead, even though they may benefit from the flexibility it provides.

Applications. We demonstrate CipherCard on capacitive touchscreen PIN pads, but we envision that the same concept can be applied to popular gestural pattern and QWERTY-based keyboards.

Study. CipherCard warrants a long-term study in the field, which will be helpful in understanding its practical usability in real-world use. The results from a field study might be more nuanced from the results from a laboratory environment due to artificial setups [15].

Prototypes. Our prototypes were designed as proof of concepts. Deployable systems will need more attention to how CipherCards appear, the cost of manufacture, and the reliability of the electronics.

CONCLUSION

In this paper, we introduced the concept of CipherCard to prevent PIN entry on capacitive touchscreens from being susceptible to shoulder-surfing attacks. CipherCard remaps a user's touch point to a different location on the touchscreen, thus translating the visible user password into a hidden system password received by the touchscreen. CipherCard enhances the authentication security through Two-Factor Authentication (TFA), where both the correct user password and the card are needed for a successful authentication. We explore the design space of CipherCard, and implemented three proof-of-concept prototypes. We evaluated the CipherCard concept with two user studies. The first study identified several usability issues, where we then proposed solutions that were the subject of the second study. User feedback from both studies confirmed the promise of CipherCard. The studies (and our own experiences) also revealed various issues and tradeoffs that could affect its acceptance, its real-world use, and need to be considered in evolving designs. As we are still in the early stages, future work will evolve CipherCard's design, ideally resulting in a field deployment form which real-world usage data and its practicality can be better understood.

REFERENCES

1. Bianchi, A., Oakley, I., Kostakos, V. and Kwon, D. S. The phone lock: audio and haptic shoulder-surfing resistant PIN entry methods for mobile devices. *TEI'11*. 197-200.
2. Burr, W. E., Dodson, D. F. and Polk, W. T. Electronic Authentication Guideline. 2012.
3. Cranor, L. and Garfinkel, S. Security and Usability. O'Reilly Media, Inc., 2005.
4. Gunson, N., Marshall, D., Morton, H. and Jack, M. User Perceptions of Security and Usability of Single-factor and Two-factor Authentication in Automated Telephone Banking. *Computer Security*, 30 (4), 208-220, 2011.
5. Johnston, J., Eloff, J. H. P. and Labuschagne, L. Features: Security and human computer interfaces. *Computer Security*, 22 (8), 675-684, 2003.
6. Kawahara, Y., Hodges, S., Cook, B. S., Zhang, C. and Abowd, G. D. Instant inkjet circuits: lab-based inkjet printing to support rapid prototyping of UbiComp devices. *UbiComp'13*. 363-372.
7. Kim, D., Dunphy, P., Briggs, P., Hook, J., Nicholson, J. W., Nicholson, J. and Olivier, P. Multi-touch authentication on tabletops. *CHI'10*, 1093-1102.
8. Kim, D. and Solomon, M. Fundamentals of Information Systems Security. *Jones & Bartlett Learning*, 2010.
9. Kim, S.-H., Kim, J.-W., Kim, S.-Y. and Cho, H.-G. A New Shoulder-surfing Resistant Password for Mobile Environments. *ICUIMC '11,* Article No. 27.
10. Kratz, S. and Aumi, M. T. I. AirAuth: a biometric authentication system using in-air hand gestures. *CHIEA'14*, 499-502.
11. Kumar, M., Garfinkel, T., Boneh, D. and Winograd, T.. Reducing shoulder-surfing by using gaze-based password entry. *SOUP'07*. 13-19.
12. Liu, S. and Silverman, M. (2001). A Practical Guide to Biometric Security Technology. *IT Professional*, 3 (1), 27-32, 2001.
13. Luca, A. D., Denzel, M. and Hussmann, H. Look into My Eyes!: Can You Guess My Password?? *SOUPS'09*. Article No. 27.
14. Luca, A. D., Harbach, M., Zezschwitz, E. v., Maurer, M.-E., Slawik, B. E., Hussmann, H. and Smith, M. Now you see me, now you don't: protecting smartphone authentication from shoulder surfers. *CHI'14*. 2937-2946.
15. Luca, A. D., Langheinrich, M. and Hussmann, H. Towards understanding ATM security: a field study of real world ATM use. *SOUP'10*. 1-10, 2010.
16. Luca, A. D., Zezschwitz, E. v., Hußmann, H. Vibrapass: secure authentication based on shared lies. CHI'09. 913-916.
17. Luca, A. D., Zezschwitz, E. v., Nguyen, N. D. H., Maurer, M.-E., Rubegni, E., Scipioni, M. P. and Langheinrich, M. Back-of-device authentication on smartphones. CHI'13, 2389-2398.
18. Orgill, G. L., Romney, G. W., Bailey, M. G. and Orgill, P. M.. The urgency for effective user privacy-education to counter social engineering attacks on secure computer systems. *CITC'04*. 177-181.
19. Roth, V., Schmidt, P., G, Güldenring, B. The IR ring: authenticating users' touches on a multi-touch display. In Proceedings of the *UIST'10*. 259-262.
20. Sae-Bae, N., Ahmed, K., Isbister, K. and Memon, N. Biometric-rich gestures: a novel approach to authentication on multi-touch devices. *CHI'12*. 977-986.
21. Sasamoto, H., Christin, N. and Hayashi, E. (2008). Undercover: authentication usable in front of prying eyes. *CHI'08*. 183-192.
22. Schneier, B. Two-factor authentication: too little, too late. *ACM Communication*, 48 (4), 136, 2005.
23. Shirazi, A. S., Moghadam, P., Ketabdar, H. and Schmidt, A. Assessing the vulnerability of magnetic gestural authentication to video-based shoulder surfing attacks. *CHI'12*, 2045-2048.
24. Suo, X., Zhu, Y. and Owen, G. S. Graphical Passwords: A Survey. *ACSAC'05*, 463-472.
25. Tan, D. S., Keyani, P. and Czerwinski, M. Spy-resistant keyboard: more secure password entry on public touch screen displays. *OZCHI '05*, 1-10.
26. Vogel, D. and Baudisch, P. Shift: a technique for operating pen-based interfaces using touch. *CHI'07*. 657-666.
27. Vu, T., Baid, A., Gao, S., Gruteser, M., Howard, R., Lindqvist, J., Spasojevic, P. and Walling, J. Distinguishing Users with Capacitive Touch Communication. *Mobicom'12*. 197-208.
28. Wiedenbeck, S., Waters, J., Sobrado, L. and Birget, J.-C. Design and evaluation of a shoulder-surfing resistant graphical password scheme. *AVI '06*, 177-184.
29. Yan, J., Blackwell, A., Anderson, R. and Grant, A. (2004). Password Memorability and Security: Empirical Results. *IEEE Security and Privacy*, 2 (5), 25-31, 2004.
30. Yu, N.-H., Chan, L.-W., Lau, S.-Y., Tsai1, S.-S., Hsiao, I.-C., Tsai, D.-J., Cheng1, L.-P., Hsiao, F.-I., Chen, M. Y., Huang, P. and Hung, Y.-P. (2011). TUIC: Enabling Tangible Interaction on Capacitive Multi-touch Displays. *CHI'11*. 2995-3004.
31. Zezschwitz, E. v. and Luca, A. D. (2013). Survival of the Shortest: A Retrospective Analysis of Influencing Factors on Password Composition. *INTERACT'13*. 460-467.