CipherCard: A Token-Based Approach Against Camera-Based Shoulder Surfing Attacks on Common Touchscreen Devices

Teddy Seyed, Xing-Dong Yang, Anthony Tang, Saul Greenberg, Jiawei Gu, Bin Zhu, Xiang Cao

CipherCard: A Token-based Approach against Camera-based Shoulder Surfing Attacks on Common Touchscreen Devices

Teddy Seyed[1], Xing-Dong Yang[2], Anthony Tang[1], Saul Greenberg[1], Jiawei Gu[3], Bin Zhu[4], and Xiang Cao[5]

[1]Department of Computer Science, University of Calgary, Alberta, Canada
{teddy.seyed, tonyt, saul}@ucalgary.ca
[2]Department of Computer Science, Dartmouth College, Hanover, NH, USA
xing-dong.yang@dartmouth.edu
[3]Baidu Institute of Deep Learning, Beijing, China
gujiawei@baidu.com
[4]Microsoft Research Asia, Beijing, China
binzhu@microsoft.com
[5]Xiaoxiaoniu Creative Technologies, Beijing, China
xiangcao@acm.org

Abstract. We present CipherCard, a physical token that defends against shoulder-surfing attacks on user authentication on capacitive touchscreen devices. When CipherCard is placed over a touchscreen's pin-pad, it remaps a user's touch point on the physical token to a different location on the pin-pad. It hence translates a visible user password into a different system password received by a touchscreen, but is hidden from observers as well as the user. CipherCard enhances authentication security through Two-Factor Authentication (TFA), in that both the correct user password and a specific card are needed for successful 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.

Keywords: Shoulder-surfing attack, capacitive touchscreen, PIN entry, security

1 Introduction

Capacitive touchscreens have become the primary input mechanism for many security related applications such as access control systems (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 public spaces, the user is susceptible to shoulder-surfing attacks: unscrupulous individuals or cameras can see the password or PIN being entered into the system [1-6]. Further exacerbating the problem, user interfaces for touchscreens are often designed to be larger (because of the fat finger problem

[7]), making it difficult to shield input from observation. As well, the lack of haptic feedback on touchscreens makes eyes-free operation difficult, which means that users cannot easily shield the display from view.

To enhance user authentication security on touchscreen devices, we present CipherCard, a physical token that protects a users' PIN entry against camera-based shoulder-surfing attacks. CipherCard (Fig. 1.) is an opaque overlay that is placed atop a 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, this touch point is remapped to a different button location on the touchscreen via its internal wiring, hiding the actual input location from observers or hidden cameras. CipherCard translates the input sequence ("*user password*") into a distinct sequence ("*system password*") that is received by the touchscreen. For example, Fig. 1. illustrates a user entering their user password of '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 observers or hidden cameras. A shoulder-surfing attacker may acquire the user password, but without the user's CipherCard, the attacker cannot successfully authenticate.

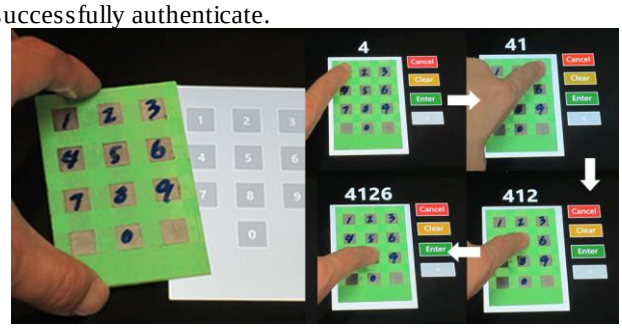

Fig. 1 CipherCard maps a touch into a different location.

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). Renewing a system password can be as easy as getting a new CipherCard or reconfiguring an existing one, where the user password can remain the same. CipherCard can be designed to automatically adapt to different password input UIs (e.g. size, orientation, and layout), and vice versa.

The CipherCard authentication scheme raises many engineering, security, and usability questions that warrant long-term research. At 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. To evaluate our progress, we conducted two studies: first, a feasibility evaluation with five usability professionals, where we identify usability issues of the three prototypes. Based on the findings, we proposed an improved de-

sign. Second, we conducted a separate workload evaluation to verify our improved design as well as the overall usability and security of CipherCard.

Our contributions in this paper are three-fold: (a) a novel CipherCard concept that allows remapping for key entries to protect PIN input against camera-based attacks. (b) An exploration of the design space of CipherCard concept through the implementation of three prototypes, and (c) a feasibility evaluation and a workload evaluation that validate the usability of CipherCard and its form factors.

2 Related Work

Researchers have developed several authentication schemes and interaction techniques with the goal of hindering shoulder-surfing attacks. However, many are still susceptible to camera-based attacks, where the input can be reviewed in detail at a later time.

Cognitive trap techniques increase the complexity of the authentication process, for example by showing distractive keystrokes or cursor movements to observers. This makes it more difficult for an observer to derive the password [8, 9] 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. [4, 10]). 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. CipherCard occludes the real password entered into the system, thus resisting camera-based attacks

Hiding PIN input techniques shield against the visibility of input actions. In its simplest form, people can position their bodies/hands to shield their actions from an observer's view [11]; however, most users do not do this [12]. While easy to perform, shielding is vulnerable to well-placed cameras [12] or thermal imaging after entry [13]. Many technical approaches also try to decrease the visibility of password entry. Examples include back-of-the-device PIN entry [3, 6], eye gaze input [14], pressure [11], and haptic/tactile feedback as secret input or output channels to assist password entry [1, 15, 16]. 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 [11]. Similarly, haptic/tactile feedback can be detected from a recorded sound track [16]. CipherCard allows PIN input to be completely hidden from observers, video and audio recording.

Biometric methods distinguish users based on biometric characteristics, e.g. fingerprint, hand geometry, retina, etc. [17] or their behavioral signatures [18]. 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 [18].

Physical token methods require that the user present a physical object to a reader (e.g. keyfob), 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 [20]. Of course, a lost or stolen token could still allow individuals to pass the authentication. This is not the case for CipherCard, which requires both the token and password to pass authentication.

Two-factor authentication (TFA) requires a user to present at least two 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) [21] to enhance authentication security. Combining a password with a physical token requires a dedicated hardware (e.g. RFID readers) that is usually unavailable on a typical touchscreen device. A popular solution involves asking a user to enter 2 codes (e.g. a PIN + a 1-time passcode generated by a token) through a 2-step authentication process [21]. A common problem in these solutions is that they are vulnerable to man-in-the-middle attacks. Whereas, CipherCard is resistant to such attacks. Additionally, CipherCard does not expose system passwords to attackers. This allows a number of novel applications to be carried out by CipherCard users. For example, a user can use an easy-to-memorize user password while the system can be protected by a strong system password. The user can also use a single user password to authenticate multiple systems each protected by a different system password. Finally, the user can renew a system password while continuing to use the existing user password (see details in Section 7).

3 CipherCard Concept and Design Space

CipherCard is an opaque overlay that is placed on top of a software authentication UI, acting as a physical proxy to the UI elements. Users can place it 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, however at a different input location on the underlying screen. This creates a randomly preset or user specified permutation between the two sets of locations on the two sides of the card. Thus, the CipherCard provides substitution cipher capabilities, translating the touch input sequence on the CipherCard (*user password*) to another unique sequence that is sent the touchscreen (*system password*). The system password is never exposed. So long as the touchscreen UI and card layout are compatible, which can be achieved through software modification on touchscreen devices, one CipherCard can be reused for an arbitrary number of different PINs for different applications.

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

3.1 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.

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.

3.2 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.

3.3 Form Factor

CipherCards should be easy to carry and deploy. For example, it can resemble a credit card carried in a wallet or purse, or ID tag worn on clothing. Alternatively, it can be integrated into flat daily personal belongings, e.g. a wallet or phone case (that flips open) or even an existing bank or credit card—this avoids the need to carry 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.

3.4 UI Alignment

A practical concern is the variety of sizes and layouts of PIN pads may not (by default) match that of a CipherCard. Having a mechanism that can automatically align the two interfaces may largely improve the practicality of the concept.

Fixed Alignment to PIN pad. A passive card must be made to match a particular touchscreen layout and is not scalable to UIs with different layouts or button sizes. An active card could utilize higher output resolution to be able to scale to different UI layouts and button sizes (within the card's physical dimensions).

PIN pad aligns to CipherCard. An alternative approach could be to have the software PIN pad align with a CipherCard (either passive or active) automatically. This could be achieved by, for example, adding spatial tags to CipherCard [22]. This would allow the touchscreen to identify the size, position, and orientation of a CipherCard, and align the PIN pad interface accordingly. We expect that the size and

layout of the buttons inside PIN pad boundaries can follow a common standard, thus once boundaries are detected by the touchscreen device, the buttons can be automatically aligned with the electrodes of the CipherCard. In cases when custom sizes and layouts are needed, this information can be pre-stored in the touchscreen devices, and can be loaded upon receiving an encoded 'request' from the CipherCard. A request can be encoded into a certain touch patterns (either static or dynamic) based on the pattern of electrodes interpreted by the touchscreen [22].

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.

4 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 are the subject of our feasibility and usability studies.

4.1 Card-shaped CipherCard

CipherCard works based on the fact that touch input can be simulated by any conductive object in contact with the 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 (Fig. 2a). For the sake of simplicity, this prototype does not employ utility electrodes for detecting the size and orientation of the card. Each electrode on the front side is uniquely connected to an electrode on the back side. 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 (Fig. 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 outside 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 it can be easily carried in a wallet.

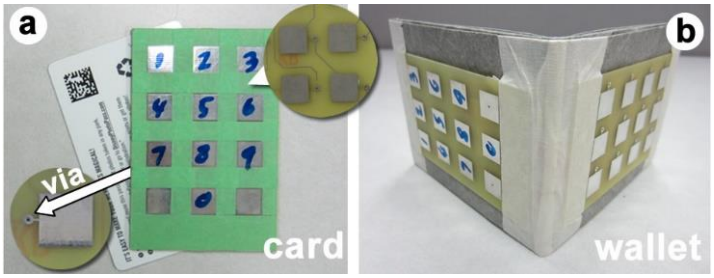

Fig. 2. Left: Card-shaped CipherCard with a 3×4 electrode grid. Top callout shows the internal wiring and bottom callout shows the via and electrodes on the back. Right: Wallet-based CipherCard.

4.2 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 (Fig. 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 connected to the opposite side through copper wires. Our prototype uses two hard PCB, but we expect a deployable version to use flexible materials, for example, tin paths printed on PET film. What is important is that the deployable version preserves the appearance, feel and functionality of a wallet.

4.3 Smartphone-based CipherCard

We explored the feasibility of an active CipherCard by creating a smartphone prototype (Fig. 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 boards. 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 thus 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.

Fig 3. Phone-based CipherCard.

5 Reconfigurable CipherCard Mappings

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

5.1 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 can be reconfigured on the fly. We designed (not implemented yet) one possibility, illustrated in Fig. 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 Fig 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 they are 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 as easily as switching positions of removable electrodes.

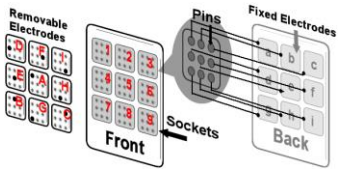

Fig 4. Design of a reconfigurable CipherCard.

5.2 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 (Fig. 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 (Fig. 5b). This way, when the key is tapped, the corresponding touch location is triggered (Fig. 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 (Fig. 6 right). Once done, tapping the number key triggers the associated locations in a sequence that the keys were created.

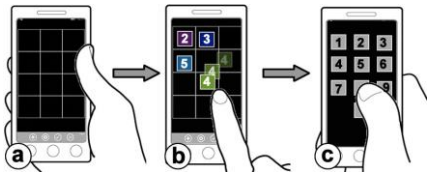

Fig. 5. (a) grid shows the position of the electrodes; (b) dragging a number key to inside a desired electrode to change a key mapping;(c) finished configuration.

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 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 (Fig. 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 algorithm triggers all the electrodes (if smaller than 2mm) that reside inside the key.

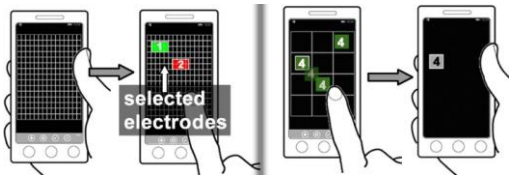

Fig. 6. Left: configuring a grid of 15 × 15 electrodes; Right: duplicating a number key to configure a *1-to-m* mapping.

5.3 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 on paper using a home printer and conductive ink [23]. This allows CipherCard to be widely adopted for home and office use.

6 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 can record (from multiple angles) all 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 without the possession of CipherCard.

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 difficult without physically possessing it. CipherCard can be deployed where the system password is not revealed to the user. This means the user would only know their user password, which in turn mandates CipherCard in user verification. This makes the user immune to the social engineering attacks [24], where the user may divulge the system password to attackers, who can then bypass the CipherCard to authenticate directly.

Losing both the user password and a CipherCard to an adversary will grant access to protected services or locations. Hacking into an active card, e.g. user's phone, or breaking into a computer which stores translation files may disclose a user's password mapping to an adversary. Individuals or organizations that design or have access to the design of the password mapping may also present risks 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) [25]. 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) [26]. 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 too easy to guess, the level of security of CipherCard 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 user password becomes extremely easy to obtain. Even so, the attacker would still need to somehow take possession of the CipherCard.

7 CipherCard Usage

CipherCard makes it possible for people to choose easy-to-remember PINs instead of using strong ones that are less memorable [29], which often imposes security concerns [3, 31]. 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 generating different system passwords), or a single active CipherCard (which would change the mapping based on the service). This allows the user to reuse passwords for multiple accounts without significantly impacting security [27].

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, 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 logged in. After selecting that option, a user can simply place the CipherCard on the authentication UI, and enter a new user password which, 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 to do with our reconfigurable designs. The actual mapping between the two can be done by any of the previously described methods.

Replacing a lost or damaged CipherCard. A new CipherCard can be authored and granted to the user if the original one is lost or damaged. The user can also make one at home (see 5.3). The design of the password mapping needs to be securely stored on a safe computer in order to preserve the security of CipherCard.

8 Feasibility Study

We conducted two studies to evaluate the concept of CipherCard, our designs, and identify potential usability issues. In our first study, we used heuristic evaluation to identify usability issues of the three prototypes. We were also interested in perceptions of the security of the scheme. We did not implement UI alignment for CipherCard. This allowed us to investigate the feasibility of actual deployment without modifying existing touchscreen devices. While we believe that CipherCard warrants a long-term field deployment study, in this early development stage, we deem this study a necessary step towards refining and improving the concept before they can be deployed and studied.

8.1 Participants

Our heuristic evaluation was conducted with usability experts. 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.

8.2 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 desk. To simulate a mobile scenario, the participant was asked to hold the tablet using their non-dominant 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. In order to

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 completing a questionnaire (7-point Likert scale) and an interview, in which the participants were asked about their perceived security and portability of CipherCard as well as their mental demand, physical demand, effort, frustration, and concentration when using the prototypes.

8.3 Results

Overall, the participants welcomed CipherCard as a method to resist camera-based shoulder-surfing. Their feedback confirmed the merits of the prototypes, e.g. security

and portability, but also identified issues that may cause cognitive overhead (Fig. 8. Left). The results reported below are using median.

Merits of CipherCard

Security. All of the participants perceived CipherCard as more secure against shoulder-surfing than current practices (6, 7 being the most secure; s.e. = 0.5), e.g. directly entering the PIN. For the participants, who had expressed interest of using CipherCard (e.g. 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, with 7 being strongly agree; all s.e. = 0) regarding the convenience in carrying them around. 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 do"-P3*.

Issues that cause cognitive overhead

UI alignment. Prior to entering the PIN, CipherCard needs to be physically aligned with authentication UI, e.g. PIN pad. 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. Misalignment resulted in touches being unregistered on the touchscreen, which had consequently caused frustration.

Slippery screens. 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 on tilted screens.

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 had sometimes occluded the number buttons that the 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, making 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 even more overhead, where they had to first unlock the phone, open the app, and then search for the desired card mapping.

9 Improved Design

After carefully reviewing the results from the first study, we came up with a number of solutions to resolve some of the most outstanding issues.

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 capacitive PIN pad does not match the one on the phone, the software can automatically load a key pad configuration that matches its specification.

For issues regarding UI alignment, slippery screens, and two-handed operation, we designed a card-holder that can be attached on top of the capacitive PIN pad. This allows the user to snap CipherCard into the right position on the screen without aligning or holding by hand. The card-holder guides the position of CipherCard, holds it onto the screen, and aligns it properly. We implemented a prototype on-screen card-holder to demonstrate the idea (Fig. 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. Alternatively, magnets can be attached to the card and screen to achieve the same goal, while preserving the flatness of the screen. This design is more suitable to phones and wallets as they do not have a uniform form factor. Thus, an on-screen holder may not work for them. 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 even adjust its button size and layout to fit those of CipherCard (see details in Section 4). We can envision many different approaches for developing the snap-in mechanism, exploring which are outside the scope of this paper. We thus leave them for future work. Instead, we focused our investigation on the effectiveness of our improved designs.

Fig. 7. Snap-in card holder allows one handed PIN entry.

10 Preliminary Workload Evaluation

The goal of this study was to verify the concept of our improved design 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 Fig. 7. For the wallet and phone, we used double-sided tape (on the back of different CipherCards) to simulate a magnetic snap-in effect.

10.1 Participants

Six participants (5 males and 1 female). All were adult office workers with prior experience using PIN pads and TFA.

10.2 Apparatus and Procedure

The procedure is similar to the feasibility study, 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.

10.3 Results

Reduced cognitive overhead

Overall, the participants rated cognitive workload being very low (1.5, with 1 being extremely low; s.e. = 0.1). The snap-in solution led to low mental/physical demand, effort, frustration, and concentration (Fig. 8. Right). The result of this study indicated the importance of having the snap-in feature before CipherCard can be deployed.

Participants found the wallet and phone had higher level of frustration than the card due to their cost value and 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.

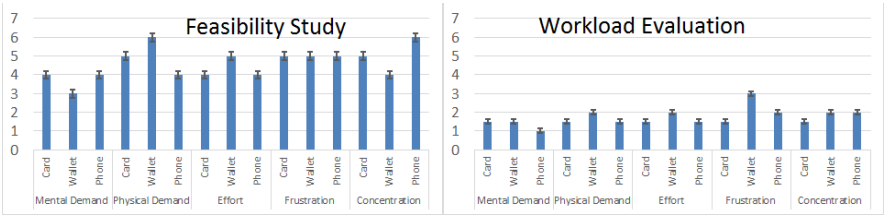

Fig. 8. Average responses on several measures of cognitive overhead from both studies (Likert-scales: 1: very low, 7: very high) (Error bar shows standard error).

11 Results from Both Studies

Security. The results from both studies confirmed that 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 against social engineering attacks), more participants preferred to have a 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 (4) expressed interest in using CipherCard on mobile devices for highly secured application, e.g. online banking. Others were less interested, feeling they had more control hiding their input on a mobile device.

Maintaining multiple passwords. Overall, the participants saw the merits of using CipherCard to help release the workload of memorizing multiple strong passwords, e.g. 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. More participants (7) 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 how much social pressure they may feel when using CipherCard in front of stranger, friend, and family, where the viewer may perceive it as an insult. They did not feel social pressure using CipherCard in front of strangers, friend, and family (all 7, with 7 being strongly disagree that they felt social pressure; s.e. = 0); They did not think they would feel uncomfortable if others (e.g. stranger, friend, or family) were to use CipherCard in front of them (1, with 7 being strongly mind; s.e. = 0).

12 Discussion and Limitations

In the 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 a 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 [24], their security behavior can be changed through education and proper design of security systems. We see that the features such as allowing easy-to-memorize user password, reuse of a user password for multiple authentication systems and for renewing a system password are handy trade-offs that may encourage users' authentication behavior by actively using CipherCard.

Convenience vs security. Our study 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 [28]. CipherCard tries to motivate user's security behavior (e.g. using TFA) by providing a set of handy 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 user password may reduce the security of CipherCard. In addition, if an adversary steals a wallet containing multiple CipherCards, all with the same user password, he will be able to access *all* associated accounts if he knows that password, even though their system passwords may differ. Future work needs to focus on convenient techniques without impacting security.

Modification of the existing authentication device. The UI alignment 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.

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

Study. CipherCard warrants a long-term field study, which will help 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 [12].

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.

13 Conclusion

In this paper, we introduced the concept of CipherCard to prevent PIN entry on capacitive touchscreens from camera-based 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. 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. Those studies (and our own experiences) also revealed various issues and tradeoffs that could affect its acceptance, its real-world use, and that should be considered in evolving designs. Of course, 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., Kwon, D.S.: The phone lock: audio and haptic shoulder-surfing resistant PIN entry methods for mobile devices. *TEI'11*, 197-20
2. Kumar, M., Garfinkel, T., Boneh, D., Winograd, T.: Reducing shoulder-surfing by using gaze-based password entry. *SOUPS'07*, 13-19.

3. Luca, A.D., Harbach, M., Zezschwitz, E.v., Maurer, M.-E., Slawik, B.E., Hussmann, H., Smith, M.: Now you see me, now you don't: protecting smartphone authentication from shoulder surfers. *CHI'14*, 2937-2946.
4. Shirazi, A.S., Moghadam, P., Ketabdar, H., Schmidt, A.: Assessing the vulnerability of magnetic gestural authentication to video-based shoulder surfing attacks. *CHI'12*, 2045-2048
5. Wiedenbeck, S., Waters, J., Sobrado, L., Birget, J.-C.: Design and evaluation of a shoulder-surfing resistant graphical password scheme. *AVI '06*, 177-184
6. Luca, A.D., Zezschwitz, E.v., Nguyen, N.D.H., Maurer, M.-E., Rubegni, E., Scipioni, M.P., Langheinrich, M.: Back-of-device authentication on smartphones. *CHI'13*, 2389-2398.
7. Vogel, D., Baudisch, P.: Shift: a technique for operating pen-based interfaces using touch. *CHI'07*, 657-666.
8. Kim, S.-H., Kim, J.-W., Kim, S.-Y., Cho, H.-G.: A new shoulder-surfing resistant password for mobile environments. *ICUIMC '11*, Article No. 27.
9. Tan, D.S., Keyani, P., Czerwinski, M.: Spy-resistant keyboard: more secure password entry on public touch screen displays. *OZCHI '05*, 1-10.
10. Kratz, S., Aumi, M.T.I.: AirAuth: a biometric authentication system using in-air hand gestures. *CHI '14 EA*. 499-502.
11. Kim, D., Dunphy, P., Briggs, P., Hook, J., Nicholson, J.W., Nicholson, J., Olivier, P.: Multi-touch authentication on tabletops. *CHI'10*, 1093-1102.
12. Luca, A.D., Langheinrich, M., Hussmann, H.: Towards understanding ATM security: a field study of real world ATM use. *SOUPS'10*, 1-10.
13. Mowery, K., Meiklejohn, S., Savage, S.: Heat of the moment: characterizing the efficacy of thermal camera-based attacks. *WOOT'11*, 6-6.
14. Luca, A.D., Denzel, M., Hussmann, H.: Look into my eyes!: can you guess my password? *SOUPS '09*, Article No 7.
15. Sasamoto, H., Christin, N., Hayashi, E.: Undercover: authentication usable in front of prying eyes. *CHI'08*, 183-192.
16. Luca, A.D., Zezschwitz, E.v., Hu, H., #223, mann: Vibrapass: secure authentication based on shared lies. *CHI'09*, 913-916.
17. Liu, S., Silverman, M.: A Practical Guide to Biometric Security Technology. IT Professional 3, 27-32, 2001.
18. Sae-Bae, N., Ahmed, K., Isbister, K., Memon, N.: Biometric-rich gestures: a novel approach to authentication on multi-touch devices. *CHI'12*, 977-986.
19. Roth, V., Schmidt, P., G, B., #252, ldenring: The IR ring: authenticating users' touches on a multi-touch display. *UIST'10*, 259-262.
20. Vu, T., Baid, A., Gao, S., Gruteser, M., Howard, R., Lindqvist, J., Spasojevic, P., Walling, J.: Distinguishing Users with Capacitive Touch Communication. *Mobicom '12*, 197-208.
21. Schneier, B.: Two-factor authentication: too little, too late. Commun. ACM 48, 136, 2005.
22. 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., Hung, Y.-P.: TUIC: Enabling Tangible Interaction on Capacitive Multi-touch Displays. *CHI'11*, 2995-3004,
23. Kawahara, Y., Hodges, S., Cook, B.S., Zhang, C., Abowd, G.D.: Instant inkjet circuits: lab-based inkjet printing to support rapid prototyping of UbiComp devices. *UbiComp'13*, 363-372.
24. Orgill, G.L., Romney, G.W., Bailey, M.G., Orgill, P.M.: The urgency for effective user privacy-education to counter social engineering attacks on secure computer systems. *CITC5'04*, 177-181.
25. Burr, W.E., Dodson, D.F., Polk, W.T.: Electronic Authentication Guideline. *NIST*, 2012.
26. Kim, D., Solomon, M.: Fundamentals of Information Systems Security. 2010.
27. Zezschwitz, E.v., Luca, A.D.: Survival of the Shortest: A Retrospective Analysis of Influencing Factors on Password Composition. *INTERACT'13*, 460-467.
28. Cranor, L., Garfinkel, S.: *Security and Usability*. 2005.