GTmoPass: Two-factor Authentication on Public Displays Using Gaze-Touch passwords and Personal Mobile Devices

Mohamed Khamis¹, Regina Hasholzner¹, Andreas Bulling², Florian Alt¹
¹Ubiquitous Interactive Systems Group, LMU Munich, Germany
²Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
{mohamed.khamis, florian.alt}@ifi.lmu.de, bulling@mpi-inf.mpg.de

ABSTRACT
As public displays continue to deliver increasingly private and personalized content, there is a need to ensure that only the legitimate users can access private information in sensitive contexts. While public displays can adopt similar authentication concepts like those used on public terminals (e.g., ATMs), authentication in public is subject to a number of risks. Namely, adversaries can uncover a user’s password through (1) surfing users, (2) thermal attacks, or (3) smudge attacks. To address this problem we propose GTmoPass, an authentication architecture that enables Multi-factor user authentication on public displays. The first factor is a knowledge-factor: we employ a shoulder-surfing resilient multimodal scheme that combines gaze and touch input for password entry. The second factor is a possession-factor: users utilize their personal mobile devices, on which they enter the password. Credentials are securely transmitted to a server via Bluetooth beacons. We describe the implementation of GTmoPass and report on an evaluation of its usability and security, which shows that although authentication using GTmoPass is slightly slower than traditional methods, it protects against the three aforementioned threats.

Author Keywords
Multi-factor Authentication, Pervasive Displays, Eye Gestures

ACM Classification Keywords
K.6.5 Computing Milieux: Security and Protection: Authentication

INTRODUCTION
Public displays deliver various kinds of tangible benefits and are now deployed in train stations, airports, and streets. Meanwhile, there is an increasing demand for displays to offer personalized, context-specific content [7, 20, 22]. There are many cases in which users need to securely authenticate at public displays. For example, while a group of tourists examine places to visit on a large public display, the system could allow users to buy tickets for museums, buses, etc. One or more users could then authenticate in parallel by entering their passwords on their mobile devices. Using the mobile device’s MAC address and the provided password, the system validates the user’s credentials and charges the correct account for the ticket fee. While there are scenarios where it might be acceptable to continue the purchase and interaction on the mobile device, in many cases it is favorable to keep the user at the display to resume the primary task. In the aforementioned example, tourists could then be shown further suggestions for activities at the given place.

When exchanging sensitive data with a public display (e.g., login credentials), users are prone to several types of threats. Namely, adversaries can uncover a user’s password in public through: (1) Shoulder surfing attacks: exploiting the heat traces resulting from the user’s interaction with the interface [1, 23], and (3) Smudge attacks: exploiting the oily residues left after authenticating on touchscreens [3]. While the latter two risks were demonstrated to be feasible, shoulder surfing was shown to occur in daily contexts [14].

In this work we introduce GTmoPass, an authentication architecture that enables multi-factor user authentication on public
While this type of schemes does not require users to remember a password, a drawback of using this factor alone is that if an attacker wants to interact with the user’s mobile device, the user has to do anything other than launching the app and entering the password. The results of our evaluation show that users authenticate relatively fast (2.8 – 4.9 seconds), and that the authentication process is resilient to the aforementioned threats. Even if the password is leaked, the architecture of GTmoPass requires the adversary to additionally acquire the user’s mobile device.

BACKGROUND AND RELATED WORK

Authentication Factors

Researchers and practitioners have developed different ways for users to authenticate in order to be granted access to private or sensitive information. Three of the most popular authentication forms are (1) knowledge-based authentication, (2) possession-based authentication, and (3) inherence-based authentication (also known as biometric authentication).

Knowledge Factor

Knowledge-based authentication schemes rely on “something the user knows”. It is perhaps the most commonly used factor [2]. Examples are passwords, PINs, and graphical passwords. Researchers also developed ways to authenticate using eye movements [8, 12, 13, 16], mid-air gestures [24], and by recalling photographs [25]. Knowledge-based schemes allow changing passwords, and can be integrated into any system that accepts any kind of user input. On the other hand, knowledge-based passwords can be guessed by illegitimate users. Attackers can find the password by observing users during authentication [14]. Smudge attacks are possible when passwords are entered through touchscreens [3]. Graphical passwords such as Android lock patterns are particularly vulnerable to smudge attacks [28, 39]. Furthermore, many knowledge-based schemes are also vulnerable to thermal attacks, where heat traces resulting from the user’s interaction with the interface are exploited to find the password [1].

Possession Factor

The possession factor relies on “something the user possesses”. Physical keys and scanable personal IDs are examples of possession-based authentication. Researchers experimented with identifying users on public displays through the MAC address of their smartphone [27]. Davies et al. [7] exploit the user’s mobile device to identify the list of applications the user wants to interact with on a display. Others approaches include using Bluetooth devices [6] and wearable shutter glasses [31]. While this type of schemes does not require users to remember passwords, it requires keeping possession of the access token. A drawback of using this factor alone, is that if an attacker gets hold of the access token, the attacker can impersonate the user and gain unauthorized access.

Inherence Factor

The inherence factor relies on biometric data, such as fingerprints, user behavior (e.g., behavior on a touchscreen [10] or eye-gaze behavior [32]) and face detection [15]. While biometric authentication can be easy and fast to use, it is accompanied with a number of problems. Biometric passwords cannot be changed; once a user’s biometric data (e.g., fingerprint or iris scan) is leaked, there is no way the user can invalidate the leaked data. Face recognition can be bypassed using pictures of the legitimate user [21], and users leave fingerprints everywhere as they interact with surrounding objects. Furthermore, users are oftentimes concerned about disclosing their biometric data to third-party companies [26], especially after it was found that biometric data can be stolen remotely [33, 41].

Multifactor Authentication

Multifactor authentication refers to the use of two or more of the aforementioned factors for improved security. This approach is employed by ATM machines; users have to know a PIN, and have to possess an ATM card. The approach has also been adopted by multiple Internet services, such as Google, Facebook, Microsoft, and Dropbox; users have to know their username and password, and have to possess a previously identified mobile device on which they receive an additional one-time password, or confirm their log-in attempt.

Researchers developed systems where users authenticate at ATMs by entering PINs on their mobile phones [4, 29, 30]. De Luca and Fraudendienst introduced PocketPIN, where users can enter credit card data on their phones before being securely transmitted to public terminals [9].

The advantage of GTmoPass is that it employs multimodal authentication as a knowledge factor, and personal mobile devices as a possession factor. Multimodal authentication was shown to be highly resilient to shoulder surfing [19]. Furthermore, thermal and smudge attacks normally require the attacker to inspect the interface after the user had left [1, 3]. Our architecture complicates these attacks by relying on the user’s mobile device for input. This means that an attacker can only perform these attacks by stealing the mobile device fast enough before the heat or smudge traces can no longer be traced. And even by doing so, the attacker would not be able to identify the gaze-input.

Protecting Privacy on Public Displays

In addition to the aforementioned works by Davies et al. [6, 7] and Schaub et al. [27], other works exploited proxemics for showing content on public displays. For example, Vogel and Balakrishnan show private content on public displays only when the user is very close to the display [35]. Brudy et al. proposed some concepts to hide private data on public displays by partially hiding the private data from the observer’s view estimated by a Kinect [5].

GTMOPASS

GTmoPass is an authentication architecture that enables secure multifactor user authentication on public displays. In the
following we describe the concept and implementation of GTmoPass, and which threat models it is optimized against.

**Concept**

GTmoPass relies on two factors (1) a knowledge factor: we use a multimodal authentication scheme that combines gaze and touch input, and (2) a possession factor: users enter the multimodal passwords on their personal mobile devices.

The Knowledge Factor

For the knowledge factor in GTmoPass, we employ a modified version of GazeTouchPass [19], a state-of-the-art authentication scheme that is highly resilient to shoulder surfing attacks. GazeTouchPass is a multimodal scheme that employs combinations of touch-based PINs (0-9) and eye movements (gaze gestures to the left and to the right). This means that it uses a theoretical password space of $12^n$ (10 digits + 2 gaze gestures) where $n$ is the length of the password. A password could be: Gaze (left), Touch (2), Gaze (right), Touch (1).

The strength of GazeTouchPass lies in its use of two input modalities. This adds complexity to shoulder surfing attacks, as it requires the attacker to observe (1) the user’s input on the touchscreen, and (2) the eye movements of the user.

The Possession Factor

For the possession factor in GTmoPass, we rely on the user’s personal mobile device. The multimodal passwords are entered on the mobile device; touch input is detected through the touchscreen, and gaze input is detected through the front-facing camera. The mobile device then communicates securely with an authentication server, that validates the password and signals the display to show the private information.

**Implementation**

GTmoPass consists of two main components: (1) the authentication server, and (2) the user’s mobile device (client).

**Authentication Server**

Implemented in NodeJS, the authentication server is configured to receive HTTP requests. The server runs on a computer (e.g., in our setup we used a laptop) that is connected to a WiFi router. The IP address of the server is broadcasted using a BLE Beacon in Google’s Eddystone protocol. The IP is broadcasted in Eddystone-URL packets at a 10 Hz rate (i.e., it broadcasts once every 100 ms), with a transmission power of 0 dBm (≈ 50 meters).

**The Client**

GazeTouchPass [19] was implemented as an Android application. It uses the OpenCV library and the Viola-Jones classifier [34] to detect the user’s face and eyes. Afterwards, in a manner similar to previous work [40, 42], the gaze direction is estimated depending on the distance between each eye and the center of the user’s face.

We further extended GazeTouchPass to communicate with the authentication server. As soon as the modified app launches, it scans for beacons and connects to the nearest one that broadcasts Eddystone URLs. The URL is then saved in the local memory. Whenever the user provides four inputs, the system hashes the input using SHA 256 and sends an HTTP request to the server (see Figure 2). Similar to previous work [27], the mobile device is uniquely identified through its MAC address.

**EVALUATION**

We previously evaluated the usability and observation resistance of GazeTouchPass, to find that although it requires 1.6 more seconds that PINs [38], it is significantly more secure against shoulder surfing compared to traditional PINs [19]. We also found that the structure of a multimodal password has an influence on its security. Namely, passwords that contain several switches from one modality to another, are more difficult to observe compared to those that have less switches. For example, a password such as “left-2-right-1” has three modality switches, and is hence harder to observe compared to “left-right-2-1”, which has only one modality switch. The reason is that attackers would have to switch attention between the user’s fingers and the user’s eyes more often in the case of passwords with more modality switches.

Another interesting insight from our previous evaluation of GazeTouchPass, is that multimodal passwords that start or end with gaze input were perceived by participants to be more difficult to observe. While our previous study was designed to focus on the effect of the number of switches in input modalities on the observability of the password, in the currently presented work we focus on the influence of the position of the gaze input in the password on the usability and security of the GazeTouch password.

**Usability Study**

The goals of this study are to: (1) collect feedback about the use of GazeTouchPass in the proposed setup, and (2) understand the influence of the gaze-input’s position on usability.

---

2. [https://developers.google.com/beacons/eddystone-eid](https://developers.google.com/beacons/eddystone-eid)
3. [http://opencv.org/platforms/android.html](http://opencv.org/platforms/android.html)
We invited 16 participants to our lab (6 females), recruited to participate in a study about gaze input used in passwords. We connected a laptop computer running a NodeJS server that accepts HTTP requests. The server validates the received passwords and updates the display’s content accordingly. We recorded participants as they enter passwords from the side (see Figure 3). An HD camera was positioned such that it recorded participants as they enter passwords using an HD video camera. The recorded videos were used in the subsequent security study to simulate shoulder surfing attacks. Figure A shows a user entering a touch input Touch(7), while figure B shows a user performing a gaze gesture Gaze(Left).

Apparatus
We used a 48 inch Samsung TV (1920×1080 px) as a display. We connected a laptop computer running a NodeJS server that accepts HTTP requests. The server validates the received passwords and updates the display’s content accordingly. We recorded participants as they enter passwords from the side (see Figure 3). An HD camera was positioned such that it is close enough to show the touchscreen, and also the user’s eyes. These videos were recorded to be used in the subsequent security study, to simulate shoulder surfing attacks.

Design
Since we wanted to investigate the influence of the position of the gaze input in the password, we experimented with four conditions: (1) passwords that start with gaze input (GazeStart), (2) passwords that end with gaze input (GazeEnd), (3) passwords that start and end with gaze input (GazeStartEnd), and (4) passwords with gaze input in the middle (GazeMiddle).

The study was designed as a repeated measures experiment. Each participant performed 16 authentications (4 passwords × 4 conditions) using randomly generated passwords. In case of conditions GazeStart and GazeEnd, participants entered two passwords with one gaze input at the start/end of the password, while the other two passwords had two gaze inputs at the start/end of the password. Table 1 shows sample passwords.

Participants
We invited 16 participants to our lab (6 females), recruited through mailing lists and social networks. Participants were awarded with online shop vouchers or participation points. All participants had normal or corrected-to-normal vision.

Procedure
The experimenter first described the study and asked the participants to sign a consent form. She then handed the participant a mobile device with the modified version of GazeTouchPass installed, and explained how it works. Each participant was allowed to perform a training run per condition to get acquainted with the system. The trial attempts were not included in the analysis. At each authentication attempt, the experimenter read out the password to be entered according to a previously generated list that was randomized. The participant would then enter the password, and observe the feedback on the display, which indicated whether or not the correct password was detected. Afterwards participants were interviewed to learn about their feedback, ideas and concerns about GTMoPass.

Results
To measure the impact of the position of gaze input on the usability of the passwords, we measured the entry time and the error count while authenticating.

Entry time was measured starting from the first input until the last input was recognized. Figure 4 illustrates the time taken to enter a password at every condition. A repeated measures ANOVA (with Greenhouse-Geisser Correction due to violation of sphericity) showed a significant effect of the gaze input’s position on the time it takes to enter a password ($F_{1,8,27.5} = 9.1$, $p < 0.05$). Post-hoc analysis with Bonferroni correction ($\alpha = 0.05 / 6$ comparisons = 0.0083) showed significant differences in entry time between GazeStart ($M = 2863 \text{ ms}, SD = 1525 \text{ ms}$) and GazeMiddle ($M = 4959 \text{ ms}, SD = 3141 \text{ ms}$), between GazeEnd ($M = 3892 \text{ ms}, SD = 3045 \text{ ms}$) and GazeMiddle ($M = 4959 \text{ ms}, SD = 3141 \text{ ms}$), and between GazeStartEnd ($M = 3757 \text{ ms}, SD = 3852 \text{ ms}$) and GazeMiddle ($M = 4959 \text{ ms}, SD = 3141 \text{ ms}$). The other pairs were not significantly different ($p > 0.05$). This means that passwords with gaze in the middle are significantly slower than other cases.

Error count reflects the number of times the participant entered the password incorrectly. Errors could occur due to entering the wrong gaze or touch input, or due to the system detecting an incorrect gaze input due to poor lighting conditions. Figure 5 shows the number of errors at each condition.

Qualitative Feedback collected at the end of the study through semi-structured interviews revealed positive feedback towards the system. Many participants reported that they liked the idea of detecting eye movements through the smartphone’s camera, and would imagine using it to authenticate on ATMs instead of using cards and PINs. Some participants suggested using the system to open security doors. One participant suggested using it for authentication on other digital devices.

<table>
<thead>
<tr>
<th>Condition</th>
<th>Format</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>GazeStart</td>
<td>startOne</td>
<td>Gaze</td>
</tr>
<tr>
<td>startTwo</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>Gaze</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>Touch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gaze</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>Gaze</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>Gaze</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>Touch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GazeEnd</td>
<td>endOne</td>
<td>Touch</td>
</tr>
<tr>
<td>endTwo</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>GazeEnd</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>Gaze</td>
<td>Gaze</td>
<td></td>
</tr>
<tr>
<td>Gaze</td>
<td>Gaze</td>
<td></td>
</tr>
<tr>
<td>Touch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GazeEnd</td>
<td>Gaze</td>
<td></td>
</tr>
<tr>
<td>Gaze</td>
<td>Gaze</td>
<td></td>
</tr>
<tr>
<td>Touch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GazeStartEnd</td>
<td>Touch</td>
<td>Gaze</td>
</tr>
<tr>
<td>Gaze</td>
<td>Touch</td>
<td>Touch</td>
</tr>
<tr>
<td>Gaze</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7LL3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Previous work reported that the position of the gaze input was perceived by participants to have an influence on the password’s observability [19]. Hence we experimented with the above conditions to cover possible positions of the gaze input in the password.
Security Study

Unlike existing systems, GTmoPass is resilient to thermal and smudge attacks by design. Heat traces and oily residues can only uncover the touch part of the password, but not the gaze input. Therefore in this study we focus on GTmoPass’s resilience to observation attacks.

Although previous work evaluated the impact of the number of modality switches on the security of passwords [19]. Our aim in this study was to understand the influence of gaze input’s position on the observation resistance of the password.

Using the video material produced in the first study, we conducted a second observability study that simulated a shoulder surfing attack against a user authenticating using GTmoPass. To do this, we invited 16 different participants and asked them to simulate shoulder surfers by watching the recorded videos, and trying to find the entered passwords (see Figure 6).

Threat Model

In our threat model, the user and the attacker are in public space. The attacker is familiar with GTmoPass and how it works. The attacker observes the user from an angle that allows seeing both the touch input on the mobile device, and the eye movements (see Figure 3). The distance between the attacker and the user is close enough to see the touchscreen, but far enough to reduce the effort of switching focus back and forth between the user’s eyes and the device’s touchscreen. After unveiling the password, the attacker tries to get hold of the device and authenticate at the display.

Results

To understand the impact of the position of gaze input on the observability of the passwords, we measured the binary success rate and the Levenshtein distance.

Success rate reflects how many passwords of each condition were successfully observed. On the other hand, the Levenshtein distance indicates how close the guess is to the original password [1, 19, 37]. Since our participants provided 3 guesses against each password they observed, only the guess
While authentication using PINs is fast, it is known to be insecure and vulnerable to observation attacks [19, 36] and thermal attacks [1]. On the other hand, authentication using multimodal passwords is more secure, but takes longer time compared to PINs. While even a slight increase in authentication time on mobile devices has a big impact considering that users unlock their mobile devices more than 50 times a day [18], we argue that authentication on public displays does not happen as often and hence a slight increase (between 1.3 and 2.5 seconds in our case) is not very significant.

In addition to the overall trade-off, we found that having gaze-input in the middle of the password (GazeMiddle) is the least likely to be successfully attacked, but also requires the longest time to enter. In general, it was found that providing consecutive gaze inputs results in longer authentication times. This was the case in GazeStartTwo and GazeEndTwo and GazeMiddle (see examples in Table 1). This is due to the time it takes to perform a gaze gesture, look to the front again, then perform another gaze gesture. On the other hand, guesses against GazeEnd are the closest to the actual password. We expect that after observing two or three touch inputs, participants foresaw that the following inputs could be gaze-based.

**Perceived Difficulty of Shoulder Surfing**

It is interesting that the perceived difficulty of attacks reported by participants was more in-line with the Levenshtein distances rather than with the binary success rate. The Levenshtein distance metric evaluates how similar a guess is to the actual password, which means that it also reflects how many times digits or gaze gestures were observed correctly. This means that unlike the binary success rate, participant’s confidence in identifying particular inputs can be a valid indicator of low Levenshtein distances.

It is not surprising that the final success rate does not correlate with the perceived difficulty. In fact, previous work reported that attackers often underestimate the perceived difficulty of shoulder surfing. For example, in the work by George et al. [17], the perceived difficulty of performing shoulder surfing attacks changed drastically after trying to perform attacks.

**CONCLUSION**

In this work we showed that GTmoPass offers a secure authentication architecture for public displays. A usability and a security study showed that GTmoPass is usable and secure against shoulder surfing. We also discussed how thermal and smudge attacks are infeasible by design.

In the future, we want to evaluate more complicated threat models. For example, a combination of a thermal attack to uncover touch input and an observation attack to uncover gaze input, or multiple consecutive observations by insiders (e.g., family members or work colleagues). Another interesting threat model is the case of having two attackers: one observing the eyes, while the other observes the touchscreen. We also intend to conduct a field study to better understand how users perceive GTmoPass in the wild. A further direction for future work is to include a third inherence factor. This can be done by scanning the finger print or by face detection using the front-facing camera.

**ACKNOWLEDGEMENT**

This work was supported by a Google IoT Technology Research Award.

---

**Table 2.** We measured the binary success rate (i.e., whether or not the attacker’s guess is correct) and the Levenshtein distance (i.e., how similar the guesses are to the correct password). A low success rate and high Levenshtein distance indicate high observation resistance and hence higher security. The most secure passwords are the GazeMiddle, where only 13% of the passwords were observed successfully, and guesses were 55% similar to the correct ones. Participants reported the perceived difficulty of attacks before knowing whether their guesses were correct.

<table>
<thead>
<tr>
<th>Mean values</th>
<th>GazeStart</th>
<th>GazeEnd</th>
<th>GazeMiddle</th>
<th>GazeStartEnd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary success rate (0%=all incorrect;100%=all correct)</td>
<td>19%</td>
<td>16%</td>
<td>13%</td>
<td>16%</td>
</tr>
<tr>
<td>Levenshtein distance (0=identical;4=completely different)</td>
<td>1.36</td>
<td>1.18</td>
<td>1.21</td>
<td>1.27</td>
</tr>
<tr>
<td>Perceived Difficulty (1=very easy,5=very hard)</td>
<td>4</td>
<td>3</td>
<td>3.5</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 2 shows the average success rate and Levenshtein distance for each condition. While participants succeeded the most when attacking GazeStart, their guesses had the longest average distance from the correct password. Guesses against GazeEnd were closest to the correct password (i.e., low distance) possibly because the attackers knew that each password consists of 4 inputs, and after seeing two or three touch-inputs they foresaw that the following inputs are gaze-based.

We also collected the perceived difficulty which participants indicated through a Likert scale (5-points; 1=very easy; 5=very hard). Table 2 shows that participants found all types difficult to observe, but GazeStart and GazeStartEnd were perceived to be more difficult compared to GazeEnd and GazeMiddle. These values are in-line with the Levenshtein distances.

**DISCUSSION**

The proposed architecture enables usable and secure authentication on public displays. Users simply approach the display with their unmodified personal mobile device that has our app installed. The app retrieves the IP of the server that is broadcasted by the BLE beacon, allowing the user to directly authenticate without entering URLs or scanning QR codes.

A usability study shows that the use of multimodal passwords in that setup is feasible, well perceived, and is only slightly slower than the less secure PINs (von Zesselwitz et al. report 1.5 seconds for PIN-entry [38]). While users make few errors, previous work has shown that users are willing to correct errors on public displays [2]. A security study showed that authentication is robust against observation attacks (only 13%-19% successful attacks in optimal conditions), which is more secure than PINs and several recently proposed systems. For example, attacks against EyePassShapes [8], EyePIN [12], GazeTouchPass [19] and XSide [11] had success rates of 42%, 55%, 19% and 9% – 38% respectively. Furthermore, the fact that gaze-input does not leave traces on the device makes GTmoPass secure against thermal and smudge attacks even if the attacker gets hold of the mobile device.

**Trade-off between Usability and Security**

While authentication using PINs is fast, it is known to be insecure and highly vulnerable to observation attacks [19, 36] and thermal attacks [1]. On the other hand authentication using multimodal passwords is more secure, but takes longer time

---

The proposed architecture enables usable and secure authentication on public displays. Users simply approach the display with their unmodified personal mobile device that has our app installed. The app retrieves the IP of the server that is broadcasted by the BLE beacon, allowing the user to directly authenticate without entering URLs or scanning QR codes.

A usability study shows that the use of multimodal passwords in that setup is feasible, well perceived, and is only slightly slower than the less secure PINs (von Zesselwitz et al. report 1.5 seconds for PIN-entry [38]). While users make few errors, previous work has shown that users are willing to correct errors on public displays [2]. A security study showed that authentication is robust against observation attacks (only 13%-19% successful attacks in optimal conditions), which is more secure than PINs and several recently proposed systems. For example, attacks against EyePassShapes [8], EyePIN [12], GazeTouchPass [19] and XSide [11] had success rates of 42%, 55%, 19% and 9% – 38% respectively. Furthermore, the fact that gaze-input does not leave traces on the device makes GTmoPass secure against thermal and smudge attacks even if the attacker gets hold of the mobile device.

**Trade-off between Usability and Security**

While authentication using PINs is fast, it is known to be insecure and highly vulnerable to observation attacks [19, 36] and thermal attacks [1]. On the other hand authentication using multimodal passwords is more secure, but takes longer time compared to PINs. While even a slight increase in authentication time on mobile devices has a big impact considering that users unlock their mobile devices more than 50 times a day [18], we argue that authentication on public displays does not happen as often and hence a slight increase (between 1.3 and 2.5 seconds in our case) is not very significant.

In addition to the overall trade-off, we found that having gaze-input in the middle of the password (GazeMiddle) is the least likely to be successfully attacked, but also requires the longest time to enter. In general, it was found that providing consecutive gaze inputs results in longer authentication times. This was the case in GazeStartTwo and GazeEndTwo and GazeMiddle (see examples in Table 1). This is due to the time it takes to perform a gaze gesture, look to the front again, then perform another gaze gesture. On the other hand, guesses against GazeEnd are the closest to the actual password. We expect that after observing two or three touch inputs, participants foresaw that the following inputs could be gaze-based.

**Perceived Difficulty of Shoulder Surfing**

It is interesting that the perceived difficulty of attacks reported by participants was more in-line with the Levenshtein distances rather than with the binary success rate. The Levenshtein distance metric evaluates how similar a guess is to the actual password, which means that it also reflects how many times digits or gaze gestures were observed correctly. This means that unlike the binary success rate, participant’s confidence in identifying particular inputs can be a valid indicator of low Levenshtein distances.

It is not surprising that the final success rate does not correlate with the perceived difficulty. In fact, previous work reported that attackers often underestimate the perceived difficulty of shoulder surfing. For example, in the work by George et al. [17], the perceived difficulty of performing shoulder surfing attacks changed drastically after trying to perform attacks.

**CONCLUSION**

In this work we showed that GTmoPass offers a secure authentication architecture for public displays. A usability and a security study showed that GTmoPass is usable and secure against shoulder surfing. We also discussed how thermal and smudge attacks are infeasible by design.

In the future, we want to evaluate more complicated threat models. For example, a combination of a thermal attack to uncover touch input and an observation attack to uncover gaze input, or multiple consecutive observations by insiders (e.g., family members or work colleagues). Another interesting threat model is the case of having two attackers: one observing the eyes, while the other observes the touchscreen. We also intend to conduct a field study to better understand how users perceive GTmoPass in the wild. A further direction for future work is to include a third inherence factor. This can be done by scanning the finger print or by face detection using the front-facing camera.

**ACKNOWLEDGEMENT**

This work was supported by a Google IoT Technology Research Award.
REFERENCES


