Information & Computer Security

User-centred authentication feature framework

Alain Forget Sonia Chiasson Robert Biddle

User-centred authentication feature framework

Alain Forget
CyLab Usable Privacy and Security, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, and
Sonia Chiasson and Robert Biddle
School of Computer Science, Carleton University, Ottawa, Canada

Received 31 August 2014
Revised 19 November 2014
4 January 2015
Accepted 5 January 2015

Abstract

Purpose – This paper aims to propose that more useful novel schemes could develop from a more principled examination and application of promising authentication features. Text passwords persist despite several decades of evidence of their security and usability challenges. It seems extremely unlikely that a single scheme will globally replace text passwords, suggesting that a diverse ecosystem of multiple authentication schemes designed for specific environments is needed. Authentication scheme research has thus far proceeded in an unstructured manner.

Design/methodology/approach – This paper presents the User-Centred Authentication Feature Framework, a conceptual framework that classifies the various features that knowledge-based authentication schemes may support. This framework can used by researchers when designing, comparing and innovating authentication schemes, as well as administrators and users, who can use the framework to identify desirable features in schemes available for selection.

Findings – This paper illustrates how the framework can be used by demonstrating its applicability to several authentication schemes, and by briefly discussing the development and user testing of two framework-inspired schemes: Persuasive Text Passwords and Cued Gaze-Points.

Originality/value – This framework is intended to support the increasingly diverse ecosystem of authentication schemes by providing authentication researchers, professionals and users with the increased ability to design, develop and select authentication schemes better suited for particular applications, environments and contexts.

Keywords Framework, Computer security, Memory, Authentication, Persuasion, Computer users

Paper type Conceptual paper

1. Introduction

For over 20 years, researchers have proposed numerous varieties of novel authentication systems, yet no proposals have globally replaced text passwords. Researchers are advocating the use of different schemes suitable for different contexts and applications, rather than completely replacing text passwords (Herley and van Oorschot, 2012). Many novel authentication schemes' designs primarily address security challenges, placing usability challenges as secondary. However, poor usability is the primary cause of insecure authentication practices (Stobert and Biddle, 2014).

This work was supported by the Natural Science and Engineering Research Council of Canada (NSERC), as well as partial funding from the NSERC Internetworked Systems Security Network (ISSNet). The second author acknowledges NSERC funding for her Canada Research Chair in Human-Oriented Computer Security.

Information & Computer Security
Vol. 23 No. 5, 2015
pp. 497-515

2056-4961
DOI 10.1108/ICS-08-2014-0058

Without usability, users cannot authenticate securely. Thus, we advocate for a more user-centric design and analysis of novel authentication schemes.

This paper presents the User-Centred Authentication Feature Framework, which is a taxonomy of some of the key user-centred features that schemes can support. This conceptual framework is intended to assist with the understanding, innovation and usage of knowledge-based authentication schemes. It enables authentication researchers and scheme developers to make clear and deliberate design choices supported by research literature. Systematically examining this taxonomy during the scheme design process can highlight features warranting further exploration that may otherwise have been overlooked. Although the features we present certainly impact security, the framework focuses on users' experiences and the underlying psychological concepts. We foreground these user-centred issues, since design solutions to authentication problems have often been decided somewhat haphazardly without carefully considering the solutions' full impact on the user.

In this paper, we first cover relevant background, including how our framework assists in scheme *design* while Bonneau *et al.* (2012) focus on *assessing* authentication technologies. We then describe each of our framework's classes and their member features, and we provide examples of existing schemes supporting them. We discuss the possible applications of our framework by identifying the features supported by existing schemes. We briefly illustrate how a systematic application of our framework's features has propagated advances in authentication research through two authentication schemes. Finally, we offer some concluding remarks.

2. Background

Our framework focuses primarily on features found in *knowledge-based authentication* (*KBA*) schemes, defined as a human entity authenticating itself by proving knowledge of some shared secret to the entity requesting the authentication (Renaud, 2005). This definition excludes biometric, token-based and key-based authentication systems (Renaud, 2005).

KBA is popular because it is relatively inexpensive to implement and typically requires no additional hardware. KBA schemes can theoretically be very secure, but their practical security is often limited by the lack of uniqueness and complexity of the shared secrets that humans can remember (Herley and van Oorschot, 2012).

Bonneau *et al.* (2012) have published the most recent survey of novel KBA schemes. They propose a framework for measuring and comparing authentication systems' security, usability and deployability. This framework is excellent for evaluating the practical trade-offs between existing authentication systems. By comparing multiple authentication systems in their framework, they made the important observation that choosing one scheme over another necessarily results in choosing one set of trade-offs over another and that no scheme is perfect. They conclude that text passwords are unlikely to be the best choice for all requirements, contexts and threat models.

Bonneau *et al.*'s framework and our own take complementary views of authentication systems. Their framework *assesses* existing authentication technologies', while our framework informs and supports the creative *design* process of authentication schemes. The User-Centred Authentication Feature Framework explicitly defines high-level design features largely grounded in one of two sources:

(1) human capabilities identified in psychology; and
(2) cognitive science research and existing authentication schemes which demonstrated particularly unique and promising design techniques.

Our framework simplifies the task of designing KBA systems by serving as a single reference for the known human characteristics and distinctive scheme features that can aid in the design of successful novel authentication schemes.

3. User-centred authentication feature framework

KBA schemes typically have two phases: registration and login (Figure 1). The registration phase involves the KBA scheme assigning a password to the user or prompting the user to generate a password, which they must memorise. During the login phase, the user must retrieve the password from memory and provide it to the scheme. Schemes may facilitate these processes by supporting particular types of features.

As listed in Table I, the User-Centred Authentication Feature Framework consists of a set of features that KBA schemes may support. Features are classified into one of the following four feature classes:

(1) *Persuasion*: It features attempt to influence the user to select more secure or memorable passwords than they would otherwise. These features may be particularly useful for authentication schemes where it may not be obvious to the user how to generate a secure and memorable password. However, these features should be used carefully, as some users may object to persuasion that is too forceful.
(2) *Memory*: Its features identify the various cognitive methods used to remember passwords. Some memory features focus on users' working memory to encode passwords into long-term memory. Other memory features support storing passwords in different formats in long-term memory. Users may have different abilities for particular memory types, and would prefer schemes that support particular memory features.
(3) *Input and output*: Its features categorise different methods of inputting passwords and receiving feedback. These features are an important consideration when designing authentication schemes for devices with novel interfaces or for particular groups of users. For example, users with disabilities often need alternative input and output modalities, so designers should carefully consider which method of input and output will best support users.

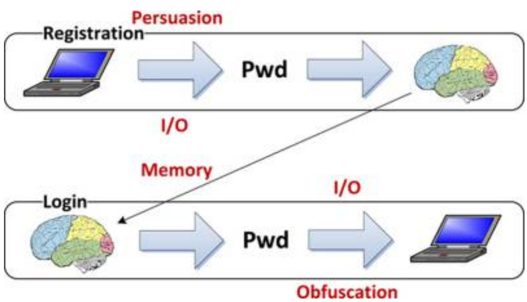

Figure 1. Illustration of the elements common to all knowledge-based authentication (KBA) schemes

Table I.
User-centred authentication feature framework

Section	Feature class	Feature name
3.1	Persuasion	Simplification Personalisation Monitoring Conditioning Social interaction
3.2	Memory	Lexical memory Visual memory Semantic memory Episodic memory Procedural memory Perceptual memory Cueing
3.3	Input and output	Keyboard Mouse Touch Eye gaze Haptic Vocal Visual output Tactile feedback Auditory
3.4	Obfuscation	Obscured input Obscured feedback Challenge-response

(4) *Obfuscation*: It features attempt to make it more difficult for illicit observers to intercept users' passwords. These features can significantly improve security, but often at the cost of decreased usability. Thus, the scheme's threat model should be carefully considered before using obfuscation features. The implementation and testing of those features should also be carefully executed to ensure that the scheme meets the expected usability benchmarks.

Each of these classes encompasses multiple features with inherent advantages and disadvantages. A given authentication scheme may support features from any class. A scheme that supports more features is not necessarily superior to another that supports fewer features. This conceptual framework is intended as a taxonomy of features that schemes may support. We believe that it covers the most relevant or popular features characterising authentication schemes, but the framework can be expanded to include features we may have overlooked, novel features from advances in authentication technology or additional dimensions to feature support, such as when a feature is only supported conditionally or to varying degrees across schemes. We will now describe in detail each of the framework's current classes and features.

3.1 Persuasion

Authentication systems may leverage *persuasion* to guide users to generate more secure and memorable credentials and promote secure behaviour in general. Persuasive Technology (PT) (Fogg, 2002) is popular in many domains seeking "interactive computing systems designed to change people's attitudes and behaviours". PT can influence users to behave in some desired manner by using well-established theories from behavioural, personality and social psychology. This section groups and describes the PT principles that can be reasonably integrated into KBA systems, as not all of PT theory is practically applicable to authentication.

3.1.1 Simplification. Authentication tasks should be as simple as possible. This includes reducing the process to the fewest actions and lowest complexity. Users can more easily form an accurate mental model of simpler authentication processes. Users are less likely to circumvent security tasks which are easier to perform. Ideally, the desired actions should form the path of least resistance (Chiasson *et al.*, 2008b), whereby users can more easily perform the authentication properly than to evade it.

For example, password managers reduce the burden on users by having the computer generate and/or remember complex passwords for them. Users typically need only enter one master password to activate the program, yet each of their accounts is protected by a distinct complex password generated by the password manager.

3.1.2 Personalisation. Customised information for individual users typically offers a more personal and engaging experience, which could be more persuasive than generic information. Users are concerned with security and privacy if they understand the implications and consequences of their actions (Stobert and Biddle, 2014). By offering appropriately timed personalised advice relating to the individual's needs, preferences or context, the system can provide details about why users' current behaviour is insecure and how it can be modified to be more secure. Because the information is personalised and offered at the moment it is most relevant, it is more likely to help improve users' mental models of security and help them understand the relevance of behaving securely.

For example, users could list some general interests to a system that would customise a mnemonic phrase (Yan *et al.*, 2004) to help users remember a system-assigned random password. The given mnemonic phrase could further include system-related content or pronounceable tokens (White *et al.*, 2014), helping users to rehearse and mentally link their mnemonic phrase and password to the system. This teaches users coping strategies for remembering passwords that can be applied to other randomly-generated passwords as well, thereby encouraging the use of both secure and memorable passwords.

3.1.3 Monitoring. When aware that they are being observed, users are more likely to perform the desired behaviour. A system tracking user performance or status can report it directly to the users, who may then adjust their behaviour in accordance with security policies. The system should provide the opportunity for users to learn how to behave more securely. This monitoring can be automated by the system or report to administrators who then take action. Furthermore, events that threaten security often happen in the background, over a long period of time, or as a result of a series of user actions. It may not be obvious to users that these events are occurring and may result in a security breach. In these cases, monitoring can help the system recognise these circumstances and bring them to the users' attention. It is especially important for the system to notify the user of dangerous behavioural errors that threaten their security, without revealing information to attackers.

The most widely used form of monitoring in authentication is password strength meters (Egelman *et al.*, 2013). When the user is creating a text password, a password strength meter illustrates the estimated strength of the user-entered password. As users change the characters in their password, many strength meters update in real-time, providing users with an effective motivation for creating a password that is deemed secure by the meter.

3.1.4 Conditioning. Computer security is concerned about potential threats and risks to the system. However, most users have little direct experience with the consequences of an attack. When users perform a mental risk analysis, they often believe the additional burden of correctly performing the security tasks outweighs the probability of being attacked. In these cases, we need to artificially induce the correct behaviour, as it is not supported by the users' natural environment. With user authentication, we want to convince people to use secure passwords even though it is a secondary task. For users to learn from any conditioning strategy, there should be other techniques at work to help users understand *how* to create effective passwords to receive the rewards for behaving securely. Examples of conditioning inducements in authentication systems (that may warrant study in future work) include:

- Longer sessions before a time-out occurs, requiring users to re-enter their password less frequently.
- Access to extra features, benefits and customisations.
- Faster system response.
- A golden lock icon with encouraging messages like, "Your password is very secure! Good job!"

3.1.5 Social interaction. Authentication is an activity that typically occurs in isolation. In contrast, physical security leverages social norms to influence behaviour and encourage secure behaviour. For example, the presence of security personnel may cause someone to reconsider entering a building without proper credentials. The social interaction principle leverages social norms by repositioning user authentication as a social activity.

A system that shares users' attitudes, traits, personality and membership is more persuasive. Such traits can be conveyed through language similar to the user's, conveying a sense of "team" and encouraging cooperation. Positive and supportive language, such as personally greeting, befriending and praising users, may further compel users to behave securely. Additionally, the system can represent an authority figure, adding more persuasive power for users who respond well to authority.

For example, users can be taught that their own insecure behaviour puts others at risk. Through careful wording and presentation by the security system, users may develop a sense of belonging and duty towards their colleagues and organisation. For example, users can be told:

- insecure accounts compromise not only their own account but the entire system;
- everyone is counting on them to do their part;

- their efforts at keeping the organisation secure are crucial and appreciated; and
- "other employees have passwords this strong. You don't want to be the weakest link".

3.2 Memory

Memory is critical to knowledge-based authentication, as its very nature depends on humans' ability to recall credentials. There are two types of memory: short-term and long-term.

3.2.1 Short-term memory. Short-term memory (STM) or *working memory* is defined as the human capacity to mentally retain information to perform the current task (Baddeley *et al.*, 2009). The more a piece of information is processed and used in STM, the more likely it is to be encoded and easily accessible in *long-term memory (LTM)*. The most widely accepted model for STM (Baddeley *et al.*, 2009) includes a *central executive* responsible for managing attention, decision-making and correlating information encoded by two slave systems: the phonological loop and the visuospatial sketchpad. The *phonological loop* retains verbal, auditory and textual information through rehearsal. The *visuospatial sketchpad* holds visual, spatial and possibly kinesthetic information.

This suggests that KBA secrets can be represented either *lexically* and/or *visually*. Users currently strongly rely on text passwords and personal identification numbers (PINs), which are both lexical KBA systems. However, research has suggested that people's memory for visual stimuli is better than text (Standing *et al.*, 1970). Thus, usable authentication researchers are exploring graphical passwords (Biddle *et al.*, 2012) to improve memorability.

The primary role of STM in authentication is to encode credentials into and later retrieve them from, LTM. Information is typically encoded into LTM through *rehearsal* in STM. The more often and deeply information is rehearsed in STM, the more easily it can be retrieved from LTM (Baddeley *et al.*, 2009). Elaborative rehearsal (e.g., associating a number with a meaningful date) facilitates information recall more than maintenance rehearsal (e.g., repeating a number).

LTM includes cognitive functions that store information for later retrieval and use. Without persistent information retrieval, KBA would be impossible. KBA system developers should be knowledgeable about LTM. Understanding memory may result in a user experience that facilitates the encoding of credentials into LTM. Figure 2 classifies some general types of LTM.

3.2.2 Explicit memory. Explicit (or *declarative*) *memory* is the deliberate and conscious retrieval of information from LTM. There are two types of explicit memory (Tulving and Donaldson, 1972). *Episodic memory* represents the recollection of events or situations from the past, such as a previous birthday or where someone recalls seeing

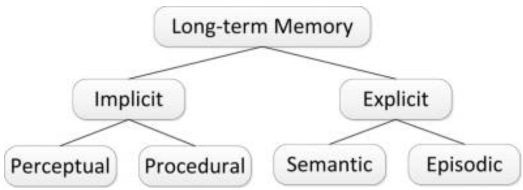

Figure 2. Classification of long-term memory types

their keys. *Semantic memory* represents the storage of factual knowledge without remembering where or how such knowledge was learnt. This includes information about people and objects (e.g., names and descriptions). Semantic memory is often supported by episodic memory. For example, when trying to remember someone's name (using semantic memory), people sometimes try recalling when and where they previously encountered said person (using episodic memory).

Text passwords are the most obvious example of an authentication scheme that uses explicit memory. Users may choose to leverage semantic memory by basing their password on information of relevance to them. They may also derive their password from a particular event through episodic memory.

3.2.3 Implicit memory. Implicit memory refers to the ability to automatically recall some piece of information without consciously retrieving it. Implicit memory can be subdivided into two types. *Procedural memory* involves remembering how to perform actions. People do not need to explicitly recall how to perform actions stored in procedural memory; they "just know" how to drive a car or access a website. *Perceptual memory* (or *priming*) is the cognitive function where a recent experience subconsciously influences a person's behaviour. This effect is illustrated in experiments asking participants to complete a set of words beginning with some letters. Participants who read a word list before the task are significantly more likely to unknowingly use words from the list than participants who had not read the list.

Explicit memory is slower and more effortful to access than implicit memory (Rovee-Collier *et al.*, 2001), which suggests a clear advantage for authentication schemes leveraging implicit memory. Bojinov *et al.* (2012) proposed the first scheme to use only procedural memory. Users perform a Serial Interception Sequence Learning (SISL) task (Figure 3), whereby on-screen columns each correspond to a keyboard button. Objects descend the columns at constant speed. The user's goal is to press the correct column's key when an object reaches the bottom of the column (as in "Guitar Hero" or "Dance Dance Revolution"). During a 30- to 60-minute registration, users perform hundreds of SISL tasks, unaware that 80 per cent contain covertly embedded repeating sequences that users are implicitly trained to perform more accurately (through repetition) than random sequences. At login, the user is granted access if they perform the trained sequences better than random sequences. A user study demonstrated that SISL worked as designed two weeks after registration. Although SISL's registration and login time is too long for most practical uses, this work demonstrates the potential of procedural memory for authentication.

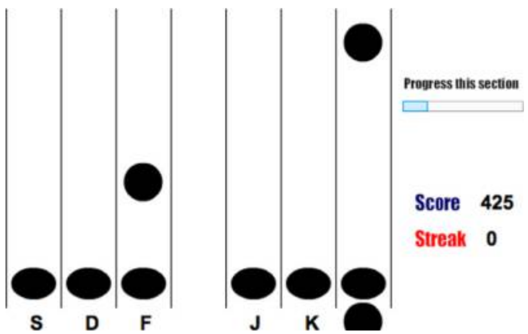

Figure 3.
Serial Interception Sequence Learning (SISL) task (Bojinov *et al.*, 2012)

3.2.4 Cueing. It is difficult to spontaneously recall information without context-specific assistance leveraging prospective memory or *memory cueing* (Baddeley *et al.*, 2009). This suggests that authentication schemes should provide users with cues to facilitate memory retrieval. However, care must be taken, as ill-considered cues may reveal password information to adversaries.

Cues support both explicit and implicit memory, depending on context and the cues presentation (Rovee-Collier *et al.*, 2001). The saliency of a presented cue does not seem to improve memory any more than a cue that is present but not particularly salient (Ellis and Kvavilashvili, 2000). Thus, the mere presence of a cue within the authentication context should assist users in recalling their credentials.

Recognition and cued-recall graphical passwords (Biddle *et al.*, 2012) are the KBA systems that best utilise cueing thus far, to varying degrees. For example, in Cued Click-Points (Chiasson *et al.*, 2007) (and derivatives), the user chooses a click-point on each image in a sequence. Each image serves as a memory cue for each click-point location. A widely deployed example is Windows 8's picture password scheme.

Users may also need some guidance on how to use the cue to create a password that is both memorable and secure. Otherwise, they may simply fall back on less secure password creation strategies (Stobert and Biddle, 2014). For example, two-thirds of Chiasson *et al.*'s (2009) study participants who were required to create several passwords for different mock systems (e.g., bank, e-mail, blog) referenced the system in their password. It appears that users take advantage of memory cues whenever possible, whether or not cues are deliberately provided. However, adversaries attacking all system accounts could incorporate cue-based information in a horizontal password guessing attack. This vulnerability can be mitigated if different users are provided different cues.

3.3 Input and output

It may sometimes be advantageous to use an entry method other than the standard mouse, keyboard or number pad. One ubiquitous example is *touch* devices (e.g., smartphones, tablets), where users interact with devices by directly touching the display. Unfortunately, it is difficult to perform some tasks with touch devices, such as entering text passwords, as typing is more difficult on touch devices than keyboards (MacKenzie and Soukore, 2002). The growing popularity of touch devices only increases the need for authentication methods designed for touch input.

Most work in varying the input modality has been to defend against shoulder-surfing attacks. Examples include a text password entry system where users would *gaze* at on-screen keyboard keys to "type" their password (Kumar *et al.*, 2007), a *haptic* keypad that generates vibrations providing user feedback regarding the keys' values (Bianchi *et al.*, 2010) and a brain–computer interface to measure brains' binary responses to presented stimuli (Thorpe *et al.*, 2005).

Additional input and output (I/O) modalities could enable people with difficulties using standard I/O methods to also use the given scheme. For example, most KBA systems use a *visual* output modality, which poses clear challenges to users with visual impairments, both for inputting their credentials and receiving feedback. Visually impaired users may benefit from authentication schemes supporting a *vocal* input or *auditory* output modality (Chiasson *et al.*, 2008a). Furthermore, standard I/O modalities are sometimes unavailable, inconvenient or dangerous. When driving, if users need directions to their destination, but first must authenticate to their mobile device, any authentication scheme requiring use of their hands or visual attention poses a risk to their lives. This risk could be avoided if the authentication scheme supported other input and output than touch and visual display, such as *vocal* input and *auditory* output. By methodically examining this framework's I/O modalities and carefully considering their advantages for a novel authentication scheme design, novel schemes could accommodate more users and use cases, including users with accessibility needs or different preferences and usage contexts beyond the standard computer user.

3.4 Obfuscation

A KBA concern is *observation attacks* when an adversary observes the authentication process and discovers part of or the whole authentication secret (Schaub *et al.*, 2012). Anyone can *shoulder-surf* by watching or recording the authentication within a user's physical proximity. *Obscured feedback* offers the simplest defence to shoulder-surfing, whereby sensitive feedback provided to the user is hidden. For example, text password systems hide users' passwords with dots. Feedback may be obscured to varying degrees, such as how Apple iPhones only mask password characters after one second or after another character is typed. Another shoulder-surfing defence is *obscured input*, where the method of credential input is hidden. Examples of this include covered keypads or various PIN alternatives for touch displays (Kim *et al.*, 2010).

Some KBA systems resist both observation and *social engineering attacks*, where users are deceived into revealing their authentication secret. In the so-called *challenge-response* schemes, users prove their knowledge of the secret without revealing the entire secret itself, thereby hiding it from observers. For example, GrIDsure (Weber, 2006) (Figures 4 and 5) users choose a pattern on a grid during registration. At login, users are shown a grid, each square containing a randomly-chosen digit (0 to 9). Users prove knowledge of their pattern by typing the digits on their pattern's squares. Because each digit appears least twice, users' patterns are resistant to an observation attack (although multiple observations may compromise the pattern).

4. Application of the framework

The User-Centred Authentication Feature Framework can be used either to identify the features supported by existing authentication schemes or to produce novel schemes by examining the framework for features that could improve existing schemes or novel scheme designs. To demonstrate the former usage, we describe the features identified in the authentication schemes listed in Table II. These schemes are a mixture of well-known schemes (e.g., text passwords, GrIDsure [Weber, 2006], PassPoints [Wiedenbeck *et al.*, 2005], Passfaces [Real User Corporation, 2004]) and schemes that uniquely emphasise features in unique ways (e.g., SISL Bojinov *et al.*, 2012), PTP (Forget *et al.*, 2008), CGP (Forget *et al.*, 2010).

4.1 Text passwords

Standard text passwords assume a *keyboard* input modality. The process and result of the password entry attempt is *visually output* to a monitor. People use *lexical memory* to process text passwords. Text passwords may be based on either a past event in *episodic memory*, some factual information in *semantic memory*, or both. With practice, passwords can be typed quickly and automatically from *procedural memory*, without consciously recalling semantic or episodic password contents. Most text password systems implement *obscured feedback* by hiding characters in password fields.

4.2 GrIDsure

As described in Section 3.4, GrIDsure requires a *visual output* device to show the grid and a *keyboard* to type digits. GrIDsure passwords are a *visually memorised* pattern of squares. GrIDsure does not support lexical memory, as digits' locations on the grid change for every login. Users' chosen grid patterns are likely stored in *semantic memory* in relation to some meaningful object. The primary benefit of GrIDsure is its *challenge–response* feature. An adversary observing the grid of digits and the user's input cannot determine the secret pattern with certainty because each digit is present in at least two grid squares (because there are 10 digits for 25 squares).

4.3 PassPoints

PassPoints (Wiedenbeck *et al.*, 2005) is a cued–recall graphical password scheme, whereby users enter a sequence of click-points (with their mouse) on an image as their password. A *visual output* device is required to display the image and allow the user to select their click-points with a *mouse*. PassPoints passwords leverage users' *visual memory*. When selecting click-points, users may choose locations that contain *memorable semantic* details or symbolise *episodic memories*. The presented image acts

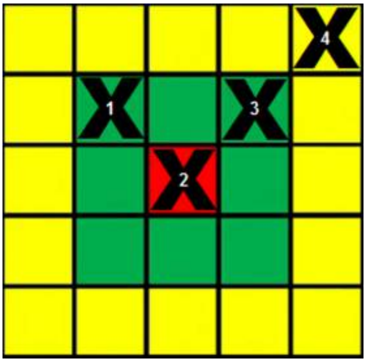

Figure 4.
Illustration of GrIDsure (Weber, 2006) pattern during registration. Numbers represent the order of selected squares

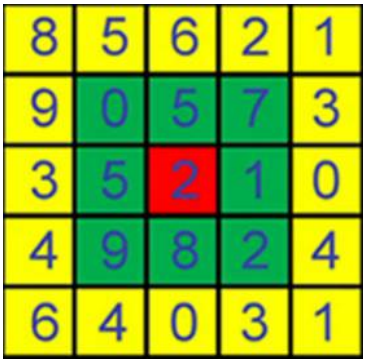

Figure 5.
Illustration of GrIDsure (Weber, 2006) login grid

	Text	GrIDsure	PassPoints	Passfaces	SHK	SISL	PTP	CGP
Scheme								
Simplification							✓	
Personalisation							✓	
Monitoring								
Conditioning							✓	
Social interaction								
Lexical memory	✓						✓	
Visual memory		✓	✓	✓				✓
Semantic memory	✓	✓	✓	✓	✓		✓	✓
Episodic memory	✓		✓		✓		✓	✓
Procedural memory	✓		✓		✓	✓	✓	✓
Perceptual memory				✓				
Cueing			✓	✓				
Features								
Keyboard	✓	✓		✓		✓	✓	
Mouse			✓	✓				
Touch				✓				
Eye gaze								✓
Haptic					✓			
Vocal								
Visual	✓	✓	✓	✓		✓	✓	✓
Tactile feedback					✓			
Auditory								
Obscured input				✓	✓			✓
Obscured feedback	✓				✓			✓
Challenge-response		✓						

Note: Table II is an example of how the User-Centred Authentication Feature Framework can be used to compare schemes' supported features

Table II.
Comparison of schemes' supported features

as a memory *cue*. After repeated logins, users will likely implicitly recall their click-points through *procedural memory*.

4.4 Passfaces

Passfaces (Real User Corporation, 2004) is a recognition-based graphical password scheme where users are assigned random faces at registration. Users logging in must sequentially select their assigned faces amongst distractor faces. Passfaces supports *visual memory*. Users may leverage *semantic memory* to remember specific aspects of their assigned faces. Passfaces utilises *perceptual memory*, as humans can rapidly recognise familiar faces. Passfaces also uses *cueing*, where the visual display of the user's assigned face implicitly cues the user's memory. Passfaces relies on a *visual output* device to display the faces. Finally, Passfaces *obscures input* by randomising the location of the correct face amongst the distractors on the grid. Thus, when using a number pad or *keyboard* to choose the correct face, it may not be immediately clear to observers which face the user selected. However, when using a *mouse* or *touch* device, users must directly click or touch the correct face, which clearly shows the selected faces to an observer.

4.5 Secure Haptic Keypad

The Secure Haptic Keypad (SHK) (Bianchi *et al.*, 2010) is a special keypad with three haptic keys, which vibrate at different frequencies. To login, the user must press the haptic keys vibrating at the frequency matching their password's sequence of frequencies. Before each key press, the keys vibration frequencies are randomised, preventing observers from visually discerning users' passwords.

SHK users may choose to use *semantic* and/or *episodic memory* to link their vibration sequence to similar sensations, such a heartbeat, engine or hummingbird. *Procedural memory* may enable users to reflexively enter their password based on the tactile sensations. SHK supports *obscured input*, since observers cannot easily determine the pressed key's vibration frequency. SHK also *obscures feedback* by providing no feedback whatsoever during login. A limit to SHK's adoption may be the required special *haptic* device providing *tactile feedback* at specific frequencies.

4.6 Serial Interception Sequence Learning

The Serial Interception Sequence Learning authentication scheme (SISL, Section 3.B.iii) (Bojinov *et al.*, 2012) uses *keyboard* input and *visual* output. Unique amongst authentication schemes, SISL relies entirely on *procedural memory*. This can be viewed as both a strength and a weakness of the scheme. Users must perform the SISL task for at least 30 minutes during registration, which may affect user adoption. Furthermore, SISL users must strongly trust the authentication system and their own procedural memory, since the shared secret cannot be deliberately retrieved or recorded. However, this is also a great strength: SISL users' credentials do not need to be explicitly remembered, cannot be deliberately divulged and are difficult for attackers to guess. For this reason, the SISL authentication scheme is still a valuable addition to the field, as it illustrates the potential of procedural memory, which is rarely leveraged. A novel SISL-based scheme may be more practical by supporting additional memory features from this framework for users to leverage when learning their credentials, thereby potentially reducing registration time. This illustrates how our framework can identify particularly valuable features in existing schemes and suggest additional improvements towards more secure or usable authentication schemes.

In addition to analysing existing schemes, the User-Centred Authentication Feature Framework can support the innovation of creative and novel authentication schemes. This can be accomplished by either isolating beneficial features in one scheme and implementing them into another scheme or substituting one feature for another, thereby producing a novel scheme with different properties, while retaining many of the original scheme's benefits. Sections 4.7 and 4.8 illustrate both approaches by describing the framework-inspired design and analysis of *Persuasive Text Passwords* (*PTP*) and Cued Gaze-Points (CGP), respectively.

4.7 Persuasive text passwords

The cued-recall graphical password system Persuasive Cued Click-Points (PCCP) (Chiasson *et al.*, 2008b) demonstrates how persuasion can influence users to choose more secure passwords. Despite their advantages, cued-recall graphical passwords could pose accessibility problems for users with eyesight or fine-motor control challenges. They are also susceptible to observation attacks.

Forget *et al.* (2008) developed a persuasive approach to influencing users to create more secure text passwords named *persuasive text passwords (PTP)*. At registration, once PTP users choose a password, PTP suggests improvements to passwords' security by placing random characters at random positions in users' passwords (Figures 6 and 7). Users may *shuffle* for an alternative improvement they may find more memorable.

PTP supports the following features identified in our framework. Because PTP is largely based on text passwords, the two schemes share many features. Their textual nature clearly relies on users' *lexical memory*. Users may remember their password with *semantic memory* or *episodic memory*, depending on the chosen initial password. As with text passwords, we believe that with sufficient practice, users' *procedural memory* would facilitate password entry without requiring the conscious retrieval of the password's contents from memory. Both schemes also *obscure feedback* by masking the echoed password on a *visual output* device during login, which users input with a *keyboard*. Text passwords and PTP differ in their use of persuasive features. While text passwords do not leverage any form of persuasion, PTP leverages the following persuasive features:

- *Simplification (Section 3.1.1)*: Because the PTP system takes on the responsibility of ensuring the password is secure, users can focus on making their password memorable, thereby simplifying the password creation task. Furthermore, users' *path-of-least-resistance* is to comply with the system's initial suggestion, which is more secure than shuffling until a weaker set of characters (such as all lowercase characters) is found. Thus, when creating a new password, PTP makes the most secure (i.e., random) choice the least burdensome.
- *Personalisation (Section 3.1.2)*: Because users choose their pre-improvement password, users are likely to feel a kinship towards their password and thus are

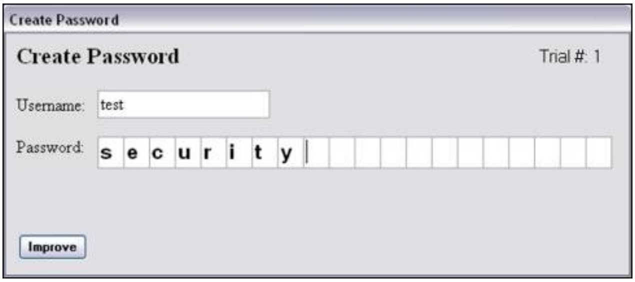

Figure 6.
PTP (Forget *et al.*, 2008) password creation before applying the persuasive improvement

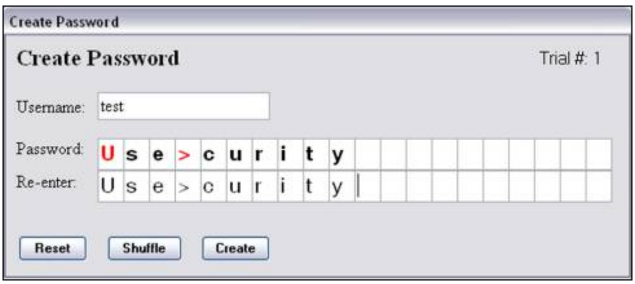

Figure 7.
PTP (Forget *et al.*, 2008) password creation after applying the persuasive improvement

more likely to comply with the system's suggestions. Furthermore, we expect that users are most likely to be open to password suggestions when creating a password. Thus, PTP applies its persuasion at the most opportune moment. The persuasion may also develop users' mental model of secure passwords, potentially leading them to apply the PTP random-character placement method to their other passwords.

- *Conditioning (Section 3.1.4)*: Shuffling repeatedly to find a specific set of (less random) system-assigned characters can be tedious. The PTP system makes less secure choices less attractive, hence guiding users away from poor security decisions.

User study results suggested that PTP helps users create more memorable and secure passwords than standard passwords (Forget *et al.*, 2008). However, users may have difficulty remembering multiple PTP passwords. To address this problem, we can systematically examine the User-Centred Authentication Feature Framework for features that may support multiple password recall. Cueing (Section 3.2.4) may serve this purpose, and has been shown to increase password memorability in cued-recall graphical password schemes. A similar effect may be possible by providing users with some kind of cue to remind them of the correct textual response without compromising security. Ideally, this memory-enhancement may also enable users to remember more secure credentials.

4.8 Cued gaze-points

While graphical passwords propose to leverage the human ability to more easily recognise and recall images over text (Standing *et al.*, 1970), click-based graphical passwords are particularly susceptible to *shoulder-surfing*, where attackers may observe or record users as they enter passwords. Text passwords may also be vulnerable to similar attacks (Tari *et al.*, 2006). To address this issue, a systematic examination using the User-Centred Authentication Feature Framework suggests some possible solutions. Because the mouse cursor reveals users' click locations, we could directly apply the *obscured feedback* feature to make the mouse cursor more difficult for observers to identify. For example, the mouse cursor could be obscured or hidden amongst multiple mouse cursors (De Luca *et al.*, 2013). However, such a system may be difficult to use. Thus, we can explore other features that may be used in concert with *obscured feedback*. For instance, the *input modality* could be changed from the *mouse*, which requires an on-screen cursor, to *eye-gaze*, where the user implicitly knows where they are gazing without any feedback obvious to an observer, thereby also applying the *obscured feedback* feature.

Forget *et al.* (2010) implemented these features as *Cued Gaze-Points (CGP)*, an eye-gaze version of Cued Click-Points (CCP) (Chiasson *et al.*, 2007), where users select points on a sequence of images with their eye-gaze instead of the mouse cursor. CGP leverages *visual memory* for locations or objects on images. Users may choose their gaze-points based on *semantic* details or *episodic* recollections. With some practice, users could implicitly remember their gaze-point locations from *procedural memory* and the presentation of the *cue*. Users' gaze-points are *obscured input*, as an attacker observing the login process would have difficulty determining their locations. Furthermore, *obscured feedback* is supported, as observing attackers cannot easily determine users' gaze locations. Obviously, users' *eye gaze* is the intended input modality, and a *visual output* modality (e.g., a monitor) is required.

Forget *et al.*'s (2010) user study results showed a clear trade-off between usability and security. They found the smaller tolerance (target) size too difficult to use. However, the larger tolerance size is more vulnerable to password guessing attacks, due to fewer possible distinct gaze-point locations. This would be an acceptable trade-off in certain environments, such as ATMs, where CGP is has greater password space and shoulder-surfing resistance than PINs. The authors found that 93 per cent of login attempts in the larger tolerance condition were eventually successful, indicating that users are capable of using the system with additional practice. Participants were also confident they could improve with practice.

5. Conclusion

Innovations in usable authentication have thus far been designed in an unstructured manner. This framework's principled set of features highlights that there are more opportunities for novel and useful schemes by more systematically classifying and applying beneficial authentication features. For example, the work on graphical passwords (Biddle *et al.*, 2012) largely addresses memorability, but we highlight the rich memory knowledge and other aspects of authentication that remain largely unexplored. This framework aims to make explicit the features that authentication schemes can support, to inspire the conception of novel and useful authentication schemes. By methodically using our framework to identify beneficial features in one scheme and transfer them to alternative schemes of varying designs, an ecosystem of multiple secure and usable authentication schemes can be developed.

Choosing amongst a wide array of proposed authentication schemes can be time-consuming and confusing. This framework can also assist people in selecting a scheme best suited to their needs and usage context. IT professionals can use our framework to identify features they and their users require, thereby shortlisting schemes that support these features. For example, an organisation with accessibility concerns could focus on authentication schemes with particular input or output modalities. An organisation building touchscreen software may favour schemes designed for touch interfaces. Administrators may wish to avoid using certain spectra of a feature type. For example, applications for blind people should avoid schemes relying heavily on visual output modalities. While this framework is designed to support authentication mechanisms, it may also benefit the design and classification of *support mechanisms* (e.g., password managers, writing down, single sign-on).

Authentication research and technology advances will invariably lead to novel features and classes, bringing them to the attention of researchers and designers for further study and implementation into novel and useful schemes. Adding novel features to the framework can also assist security professionals and users identify and choose schemes that best suit their needs. Ultimately, we hope this will lead to a diverse ecosystem of authentication schemes where users can choose schemes best suited their abilities and preferences (Forget *et al.*, 2015).

References

Baddeley, A., Eysenck, M. and Anderson, M. (2009), *Memory*, Psychology Press, New York, NY.

Bianchi, A., Oakley, I. and Kwon, D. (2010), "The secure haptic keypad: a tactile password system", *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, Atlanta, GA, 10-15 April, ACM, New York, NY, pp. 1089-1092.

Biddle, R., Chiasson, S. and van Oorschot, P.C. (2012), "Graphical passwords: learning from the first twelve years", *ACM Computing Surveys*, Vol. 44 No. 4, p. 19.

Bojinov, H., Sanchez, D., Reber, P., Boneh, D. and Lincoln, P. (2012), "Neuroscience meets cryptography: designing crypto primitives secure against rubber hose attacks", *Security Symposium, USENIX*, Bellevue, WA, 8-10 August, pp. 129-141.

Bonneau, J., Herley, C., van Oorschot, P.C. and Stajano, F. (2012), "The quest to replace passwords: a framework for comparative evaluation of web authentication schemes", *Symposium on Security and Privacy*, San Francisco, CA, 20-23 May, IEEE, pp. 553-567.

Chiasson, S., Forget, A. and Biddle, R. (2008a), "Accessibility and graphical passwords", paper presented at the Symposium on Accessible Privacy and Security (SOAPS), Pittsburgh, PA, *23* July, available at: http://cups.cs.cmu.edu/soups/2008/SOAPS/chiasson.pdf (accessed 26 July 2014).

Chiasson, S., Forget, A., Biddle, R. and van Oorschot, P.C. (2008b), "Influencing users towards better passwords: persuasive cued click-points", *British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction*, British Computer Society, Swinton, pp. 121-130.

Chiasson, S., Forget, A., Stobert, E., Biddle, R. and van Oorschot, P.C. (2009), "Multiple password interference in text and click-based graphical passwords", *Conference on Computer and Communications Security (CCS)*, Chicago, IL, 9-13 November, ACM, New York, NY, pp. 500-511.

Chiasson, S., van Oorschot, P.C. and Biddle, R. (2007), "Graphical password authentication using Cued Click Points", *European Symposium On Research In Computer Security (ESORICS)*, Dresden, 24-26 September, Springer, Berlin, pp. 359-374.

De Luca, A., von Zezschwitz, E., Pichler, L. and Hussmann, H. (2013), "Using fake cursors to secure on-screen password entry", *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, Paris, 27 April- 2 May, ACM, New York, NY, pp. 2399-2402.

Egelman, S., Sotirakopoulos, A., Muslukhov, I., Beznosov, K. and Herley, C. (2013), "Does my password go up to eleven? The impact of password meters on password selection", *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, Paris, ACM, New York, NY, pp. 2379-2388.

Ellis, J. and Kvavilashvili, L. (2000), "Prospective memory in 2000: past, present, and future directions", *Applied Cognitive Psychology*, Vol. 14 No. 7, S1-S9.

Fogg, B.J. (2002), *Persuasive Technology: Using Computers to Change What We Think and Do*, Morgan Kaufmann Publishers, San Francisco, CA.

Forget, A. (2012), "A World with many authentication schemes", PhD Thesis, *School of Computer Science*, Carleton University, Ottawa, Ontario, October.

Forget, A., Chiasson, S., van Oorschot, P.C. and Biddle, R. (2008), "Improving text passwords through persuasion", *Symposium on Usable Privacy and Security (SOUPS)*, Pittsburgh, PA, 23-25 July, ACM, New York, NY, pp. 1-12.

Forget, A., Chiasson, S. and Biddle, R. (2010), "Shoulder-surfing resistance with eye-gaze entry in click-based graphical passwords", *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, Atlanta, GA, 10-15 April, ACM, New York, NY, pp. 1107-1110.

Forget, A., Chiasson, S. and Biddle, R. (2015), "Choose your own authentication", *New Security Paradigms Workshop (NSPW)*, ACM, New York, NY, The workshop was held in Twente, The Netherlands.

Herley, C. and van Oorschot, P.C. (2012), "A research agenda acknowledging the persistence of passwords", *IEEE Security & Privacy*, Vol. 10 No. 1, pp. 28-36.

Kim, D., Dunphy, P., Briggs, P., Hook, J., Nicholson, J., Nicholson, J., Olivier, P. (2010), "Multi-touch authentication on tabletops", *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, Atlanta, GA, 10-15 April, ACM, New York, NY, pp. 1093-1102.

Kumar, M., Garfinkel, T., Boneh, D. and Winograd, T. (2007), "Reducing shoulder-surfing by using gaze-based password entry", *Symposium on Usable Privacy and Security (SOUPS)*, Pittsburgh, PA, 18-20 July, ACM, New York, NY, pp. 13-19.

MacKenzie, I. and Soukore, R. (2002), "Text entry for mobile computing: models and methods, theory and practice", *Human-Computer Interaction*, Vol. 17 Nos 2/3, pp. 147-198.

Real User Corporation (2004), "The science behind Passfaces", available at: www.realuser.com/published/ScienceBehindPassfaces.pdf (accessed 26 July 2014).

Renaud, K. (2005), "Evaluating authentication mechanism", in Cranor, L.F. and Garfinkel, S. (Eds), *Security and Usability: Designing Systems that People Can Use*, O'Reilly, Sebastopol, CA, pp. 103-128.

Rovee-Collier, C., Hayne, H. and Colombo, M. (2001), *The Development of Implicit and Explicit Memory*, John Benjamins Publishing, Amsterdam.

Schaub, F., Deyhle, R. and Weber, M. (2012), "Password entry usability and shoulder surfing susceptibility on different smartphone platforms", *International Conference on Mobile and Ubiquitous Multimedia (MUM)*, Ulm, 3-6 December, ACM, New York, NY, pp. 13-22.

Standing, L., Conezio, J. and Haber, R. (1970), "Perception and memory for pictures: single-trial learning of 2500 visual stimuli", *Psychonomic Science*, Vol. 19 No. 2, pp. 73-74.

Stobert, E. and Biddle, R. (2014), "The password life cycle: user behaviour in managing passwords", *Symposium on Usable Privacy and Security,*USENIX, Menlo Park, CA, 9-11 July, pp. 243-255.

Tari, F., Ozok, A. and Holden, S. (2006), "A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords", *Symposium on Usable Privacy and Security (SOUPS)*, Pittsburgh, PA, 12-14 July, ACM, New York, NY, pp. 56-66.

Thorpe, J., van Oorschot, P.C. and Somayaji, A. (2005), "Pass-thoughts: authenticating with our minds", New Security Paradigms Workshop (NSPW), ACM, Lake Arrowhead, California, US, pp. 45-56.

Tulving, E. and Donaldson, W. (1972), *Organization of Memory*, Academic Press, New York.

Weber, R. (2006), "The statistical security of GrIDsure", Technical Report, University of Cambridge, Cambridge.

White, A.M., Shaw, K., Monrose, F. and Moreton, E. (2014), "Isn't that fantabulous: security, linguistic and usability challenges of pronounceable tokens", *New Security Paradigms Workshop (NSPW)*, ACM, Victoria, British Columbia.

Wiedenbeck, S., Waters, J., Birget, J., Brodskiy, A. and Memon, N. (2005), "PassPoints: design and longitudinal evaluation of a graphical password system", *International Journal of Human-Computer Studies*, Vol. 63 No. 1, pp. 102-127.

Yan, J., Blackwell, A., Anderson, R. and Grant, A. (2004), "Password memorability and security: empirical results", *IEEE Security & Privacy*, Vol. 2 No. 5, pp. 25-31.

Further reading

Chiasson, S., Stobert, E., Forget, A., Biddle, R. and van Oorschot, P.C. (2012), "Persuasive Cued click-points: design, implementation, and evaluation of a knowledge-based authentication mechanism", *IEEE Transactions on Dependable and Secure Computing (TDSC)*, Vol. 9 No. 2, pp. 222-235.

About the authors

Alain Forget is currently a Post-Doctoral Research Scientist at Carnegie Mellon University. His PhD thesis research (Forget, 2012) at Carleton University in Ottawa, Canada, focused on various aspects of usable authentication, including users' mental models of passwords, using Persuasive Technology to improve users' mental models of authentication and computer security and exploring various solutions to the challenges users have with contemporary authentication. Alain Forget is the corresponding author and can be contacted at: aforget@cmu.edu

Sonia Chiasson is an Assistant Professor in the School of Computer Science at Carleton University in Ottawa, Canada, and the NSERC Canada Research Chair in Human Oriented Computer Security. Her main research interests are in usable security: the intersection between human–computer interaction (HCI) and computer security. Her current projects are on user authentication, usable security for mobile devices, and computer games for teaching about computer security.

Robert Biddle is a Professor in the School of Computer Science and Institute of Cognitive Science at Carleton University in Ottawa, Canada. His research is in Human–Computer Interaction and Software Design. His current research projects are on usable security, especially authentication, security decision-making and on large-scale multi-touch devices, especially environments for collaborative design and visualization.