Bachelor thesis

ENHANCING PASSWORD-BASED USER AUTHENTICATION: IMPLICIT MOTOR SEQUENCE LEARNING IN THE SISL TASK

Sanne van Waveren
S1233920

HUMAN FACTORS AND ENGINEERING
COGNITIVE PSYCHOLOGY AND ERGONOMICS
UNIVERSITY OF TWENTE, THE NETHERLANDS

EXAMINATION COMMITTEE:
Prof. Dr. Frank van der Velde
Dr. Andreas Peter

UNIVERSITY OF TWENTE.

Abstract ─ The serial interception sequence learning (SISL) task is used in a user authentication system based on implicit motor sequence learning (Bojinov, Sanchez, Reber, Boneh & Lincoln, 2014). This study aims to investigate the influence of training length on performance of the SISL task. Participants were distributed across three conditions: a training phase of 480 trials, 960 trials and 1,440 trials. Experiment 1 comprised two phases: a training phase and a test phase. In the former, participants substantially trained on a fixed sequence, in the latter, sequence knowledge was tested, using both the trained sequence and new, random sequences. Results showed that participants performed significantly better on the trained sequence at both experiments and that training length did not significantly influence SISL task performance. From this, it can be concluded that a SISL-based authentication system could be successful in practice. This is supported by a performed security analysis. However, future research should explore the robustness of the SISL-based authentication system.

Key words: serial interception sequence learning task, implicit learning, user authentication system

Samenvatting ─ De serial interception sequence learning (SISL) taak is gebruikt voor gebruikersauthenticatie gebaseerd op het impliciet leren van motor sequenties (Bojinov, Sanchez, Reber, Boneh & Lincoln, 2014). In dit onderzoek is de invloed van trainingsduur op de prestatie op de SISL taak onderzocht. Deelnemers waren verdeeld over drie condities: een trainingsfase van 480 trials, 960 trials en 1440 trials. Experiment 1 bestond uit twee fases: een trainingsfase en een testfase. In de eerste trainden deelnemers op een vaste sequentie en in de laatste werd getest op sequentiekennis door de sequentie af te wisselen met nieuwe, willekeurige sequenties. In experiment 2 voltooiden deelnemers na vijf weken nogmaals een testfase om te kijken of sequentiekennis aanwezig blijft. Resultaten lieten zien dat proefpersonen in beide experimenten significant beter presteerden op de getrainde sequentie en dat de trainingsduur geen significante invloed op had SISL taak prestatie. Uit de resultaten kan worden geconcludeerd dat een beveiligingssysteem op basis van de SISL taak zou kunnen werken in de praktijk. Dit wordt ondersteund door een uitgevoerde beveiligingsanalyse. Echter, toekomstig onderzoek zal moeten worden uitgevoerd om de robuustheid van dit beveiligingssysteem in kaart te brengen.

Trefwoorden: serial interception sequence learning taak, impliciet leren, gebruikersauthenticatie systeem

TABLE OF CONTENTS

I. INTRODUCTION

A significant part of our daily lives is spent online. For instance, social networking sites such as Facebook and Twitter are incredibly popular, with more than 1.4 billion and 288 million users each month worldwide respectively (Statista, 2015). Besides social networking sites, there is an increase in the use of e-mail, online banking and other online services. With this, there has been a drastic growth in the amount of personal data that is shared digitally.

The information distributed via these media should be considered as sensitive and requires both confidentiality and integrity. This means, limited access to authorized users and preservation of the original information that is transmitted without it being changed by others. These accounts therefore often require usernames and passwords as authentication.

However, with the rise of the internet, weaknesses and deficits of password protection have often become apparent. For instance, an 18-year-old hacker recently hacked multiple Twitter accounts, including Barack Obama's, performing a *brute force attack* (exhaustive search of all password possibilities) using an automated password-guesser (Zetter, 2009). Furthermore, the social networking sites accounts of multiple institutions, such as Delta Airlines and U.S. Central Command, were recently hacked and in some cases, the hackers even modified the content into provoking messages (Mosendz, 2015; Rosenfeld, 2015; Stampler, 2015). It has been speculated that this could have been prevented using solid password protection (Rogers, 2015).

Besides the advantages provided by the internet, there is a major drawback as well. With the extensive use of social networking sites, a lot of personal information about its users can be found online. Including information that is often used to create a password, such as a date of birth of a pets name. The use of personal items as part of passwords may be a result of convenience. Given the fact that on average an internet user possesses 19 passwords (Cyber Streetwise, 2014), it seems no surprise that people tend to choose easy to remember passwords, and that the majority of people use just one password for multiple services (van Ammelooy, 2015). The need to remember seems to be incompatible with ensuring strong digital security (van Ammelooy, 2015).

The worst passwords of 2014 have been presented (SplashData, n.d.), leading to the following top three: 1. *123456*. 2. *password*. 3. *12345*. This was confirmed by the famous Adobe security breach, that revealed the most commonly used passwords of Adobe users: 1. *123456*, used by 1,911,938 users, 2. *123456789*, used by 446,162 users, and 3. *password*,

used by 345,834 users (Newman, 2013). From this, it becomes apparent that weak passwords often play a major role in making it feasible for adversaries to hack digital systems. It may only take three seconds to crack an eight letters long password using password guessing technologies (van Ammelooy, 2015).

In addition, several social engineering techniques may be used to obtain personal or sensitive data from users. These techniques make use of psychological tricks instead of computer programs and other technologies (Thompson, 2006). An event at the DEFCON computer security convention called Capture the Flag challenges social engineers to extract sensitive information from companies by phone. In the 2010 edition of Capture the Flag, people were persuaded during a phone call to go to a particular website in only 20 minutes from which the social engineer could extract details about the computer that the person was using (Gold, 2010). The same happened at DEFCON 2012, when a social engineer called a manager of Wal-Mart. With a simple lie, the social engineer extracted among others several details about the store, moments that managers were on a break, and about the computer, web browser and antivirus software the manager was using (Cowley, 2012). Other companies that at least gave a bit of information away while being called by a social engineer were for instance, Shell, Cisco, Hewlett-Packard and FedEx (Cowley, 2012). These examples clearly show the power of social engineering techniques.

One social engineering method is called *rubber hose attacks*. A rubber hose attack relies on the use of non-technical means such as persuasion and emotional manipulation (Bojinov et al., 2014; Thompson, 2006). For instance, an adversary coerces a user to reveal their password by aiming a weapon or making threats. This is a stressful situation, in which individuals tend to make a quick, disorganized evaluation of the situation (Keinan, 1987). Moreover, this influences the emotional state of the person that is being coerced, as they might experience pressure and fear, as well as their behavior, as they might reveal their secret under pressure, whereas they would not have revealed their secret if not being coerced. These attacks are a serious threat to security, mostly of high security facilities with authentication systems that require physical presence. Bojinov et al. (2014) claim that this kind of attacks is often the easiest way to outsmart user authentication.

Because of this, Bojinov et al. (2014) proposed an authentication system that should provide better security against rubber hose attacks. This authentication system enables users to unconsciously learn a password, so that it cannot be revealed under pressure. However, it may also be a way to prevent brute force attacks, because it is an alternative way of password-based user authentication eliminating the use of freely chosen passwords

The aim of the current study is to further investigate the use of implicit passwords to enhance security of password-based user authentication systems. In this chapter, the concepts and relevance of user authentication, human memory, explicit and implicit knowledge, language acquisition and motor skill are explained. Also, the purpose of the current study and the proposed authentication system are reported. In chapter 2 the method are described. Next, in chapter 3 the results of this study are reported. Finally, in chapter 4 the findings of the current study and suggestions for further research are discussed.

1.1. User authentication

In order to investigate the new password-based user authentication method described above, it is important to first consider what is meant by user authentication. User authentication is commonly used in systems that need to protect personal and sensitive information. One way to ensure this essential part of computer security is through password-based user authentication (Ives, Walsh & Schneider, 2004; Adams & Sasse, 1999). This comprises an identification and an authentication phase. The first identifies the user and the latter verifies that the user is not a fraud, but a legitimate owner of the identity presented in the first phase (Adams & Sasse, 1999).

With the emergence of the web, a need arose for numerous systems that require secure use to protect personal and sensitive data (Renaud & De Angeli, 2004). For instance, online banking typically requires a username and password in order to access accounts (Mannan & van Oorschot, 2007). Password-based user authentication helps digital systems to facilitate data confidentiality, which is the protection of sensitive data from passive attacks, such as wire tapping and eavesdropping. This keeps sensitive data from falling into unintended hands, because a person is only allowed to consult data they are supposed to see ("Data Classification, Access, Transmittal, and Storage", 2014). However, despite the widespread use, password-based user authentication carries a high potential risk for identity theft for the following reasons.

Firstly, users do not change their initial passwords (Adams & Sasse, 1999). This way, an adversary only needs to figure out a password once, instead of a set of passwords. However, regularly changing passwords compromises security as well. Adams and Sasse (1999) argue that due to limitations of human memory, having to remember multiple passwords is a complex process, and therefore users tend to write down their passwords.

Secondly, passwords generate the secret key in password-based user authentication. To ensure optimal security, secret keys should be truly random (Smart, 2003). However, this

requires passwords to be as long and random as possible and thus hard to remember. Most users tend to choose meaningful, language-based passwords, which in turn reduces randomness (Smart, 2003).

Thirdly, users tend to choose easy passwords, "(…) 'password' may be the most commonly selected password, where it is allowed…" (Burr, Dodson & Polk, 2004, p. 45). Typically, users minimize the characters used in their passwords (Adams & Sasse, 1999). Given the length and simplicity, passwords are vulnerable to several attacks including guessing and use of dictionaries of common passwords (Burr et al., 2013). A solution against dictionary attacks may be to select a random password from the total password-space *k*. However, this is again user unfriendly and may increase the likelihood of users writing down their passwords, which in turn compromises security (Davies & Ganesan, 1993). It becomes apparent that security involves both technology and people, but that the latter are often forgotten (Brostoff & Sasse, 2002).

Moreover, Davies and Ganesan (1993) argue that passwords are often chosen from a small password-space k_1 that is part of *k*. This small size of k_1 allows adversaries to efficiently do an exhaustive search of the password-space (Davies & Granesan, 1993). In an exhaustive search, the attacker simply checks all possible passwords until the correct one is found. Although the attacker still needs to consider several possible passwords, there are relatively few possibilities and therefore this search becomes feasible. For instance, the password "pass" contains only lowercase characters and no numeric values or symbols. Therefore, there are only:

$$26^4 = 456976\,,$$

possibilities to explore in an exhaustive search.

Also, a small password-space *k* causes the Shannon entropy of passwords to be low. Shannon entropy is hereinafter referred to as entropy. Burr et al. (2013) define entropy as "a measure of the amount of uncertainty that an attacker faces to determine the value of a secret" (p. 9). If a password is written in binary (0 or 1), the entropy *H* is the average number of binary digits required per character of the password. Burr et al. (2013) and Smart (2003) give the following mathematical definition of entropy with the probability that *X* has the value x_i (for *i* in x_i it holds that $1 \leq i \leq n$), with a probability distribution $p_i = p(X = x_i)$:

$$H(X) = -\sum_{i=1}^{n} p_i log_2 p_i \qquad (1.1)$$

For instance, a password *"PASS"* with event space *C*:

$$C = \{P, A, S, S\},$$

with the probabilities based on frequency of occurrence:

$$p(C = P) = p(C = A) = 0.25 \text{ and } p(C = S) = 0.5.$$

The entropy of *C* can be calculated as follows:

$$H(C) = -0.25\, log_2(0.25) - 0.25\, log_2(0.25) - 0.5\, log_2(0.5) = 1.5.$$

This means there are around one and a half bits of information about the password to be found (Smart, 2003). A small password-space causes entropy to be low, which in turn makes it feasible for an adversary to retrieve the password.

Last, as a user is consciously aware of a password, describing it to others is relatively easy. Several social engineering techniques are available, such as persuasion, in order to coerce someone to revealing their password (Weinshall & Kirkpatrick, 2004). If someone is aware of a password they may reveal it in some way when being coerced by an adversary.

1.2. Human memory

Usually, users tend to consciously know their passwords. In order to understand the difference between unconscious and conscious knowledge it is important to explain the mechanisms of the human memory first.

In cognitive psychology, knowledge is defined as information about the world that is stored in memory (Smith & Kosslyn, 2007). There is a distinction between long-term, short-term and working memory. The first has a vast capacity to store knowledge for over years, such as information about ten years ago. Short-term memory has a capacity of approximately four to seven items and depends on rehearsal (Kalat, 2013). For instance, information about this morning. Working memory involves actively processing information, such as holding a thought in your mind. If information is somehow lost from working memory, it has to be retrieved from short-term or even long-term memory (Smith & Kosslyn, 2007). This process also takes place in password-based user authentication. A password is often stored in long-term memory as it is used for a long time and possibly with long time intervals in between. Typing a password involves working memory, as knowledge is actively processed in the

individual's head right at that moment. When the password cannot be retrieved from working memory, short-term or even long-term memory should be consulted.

Within long-term memory, two forms of memory have been distinguished: declarative memory and non-declarative memory (Squire, Knowlton & Musen, 1993). Declarative memory can be divided into episodic memory and semantic memory. The first encompasses memory of events in the past and the latter general knowledge about things in the world and their meaning.

Language based passwords may be seen as part of the declarative memory, as they often can be described verbally. This type of memory is depending on the medial temporal lobe (MTL) memory system (Scoville & Milner, 1957; Reber, 2013; Bojinov et al., 2014). Studies have shown that damage to the MTL memory system, including stroke or amnesia, impairs the ability to acquire declarative knowledge. However, in case of such damage, the ability to acquire non-declarative knowledge remains intact (Bojinov et al., 2014; Squire & Zola-Morgan, 1991). Therefore, non-declarative knowledge seems to be independent of the MTL (Reber, 2013).

1.3. Implicit and explicit knowledge

Memory systems research has shown different types of long-term memory (Bojinov et al., 2014; Squire & Zola-Morgan, 1991; Reber, 2013). On the one hand, declarative, conscious, explicit knowledge, which often can be described verbally. On the other hand, non-declarative, implicit knowledge, which is expressed through performance and cannot be described verbally (Reber, 1976; Squire & Zola-Morgan, 1991). For instance, the knowledge that the big yellow circle in the sky is called the sun is learned explicitly and can be described consciously. However, riding a bicycle is a process that can be carried out efficiently, but cannot be described verbally. This is done implicitly: it is clear what is done, but not how it is done. DeShon and Alexander (1996) argue that explicit knowledge is acquired by first developing an internal representation. Explicit learning requires conscious thinking about a problem's initial state, goal state and the steps to get from the first to the latter (DeShon & Alexander, 1996). Implicit learning, however, depends on frequent repetition with no conscious awareness (DeShon & Alexander, 1996).

1.4. Language acquisition

As mentioned earlier, passwords commonly comprise, among others, existing words, dates and derivatives of words from natural languages (Davies & Granesan, 1993). Since passwords have a language-based content, the process of learning a password may be compared to the process of learning a language.

Language is based on fundamental links between sentence form and meaning (Gertner, Fisher & Eisengart, 2006). General knowledge on word meanings belongs to so called conceptual knowledge or semantic memory (Patterson, Nestor & Rogers, 2007). Languages are composed of complex rules and regularities. Knowledge of these rules and regularities is essential in order to master a language, but native speakers are often not aware of the underlying rules of the language (Batterink, Oudiette, Reber & Paller, 2014). The underlying principles about grammar can be acquired implicitly, which means that this knowledge is unconsciously abstracted from the environment (Reber, 1976; Ellis, 2008). However, an important difference between the type of knowledge that is language and the knowledge of passwords, is that a user learns a password explicitly, as it can often be verbally described by the user (Ellis, 2008).

1.5. Motor skill

Many daily activities consist of a variety of motor skills that can gradually be acquired by practice. This holds for basic as well as complex motor behaviors (Doyon, Penhune & Ungerleider, 2003; Savion-Lemieux & Penhune, 2005; Verwey, Lammens & Honk, 2001; Abrahamse, van der Lubbe & Verwey, 2009; Tzvi, Münte & Krämer, 2014). Learning motor skills, both explicitly and implicitly, has been studied extensively. Studies on motor sequence learning, like typing or playing the guitar, have shown that motor skills can be acquired implicitly (Schendan, Searl, Melrose & Stern, 2003; Bojinov et al., 2014; Abrahamse, van der Lubbe & Verwey, 2009; Tzvi et al., 2014; Rüsseler & Rösler, 2000).

1.6. The serial interception sequence learning (SISL) task authentication system

Because of the large amount of personal information we share digitally, password-based user authentication is very important nowadays. However, it have shown to be vulnerable to attacks.

Bojinov et al. (2014) proposed an authentication system based on implicit learning in the serial interception sequence learning (SISL) task as a method resistant to rubber hose

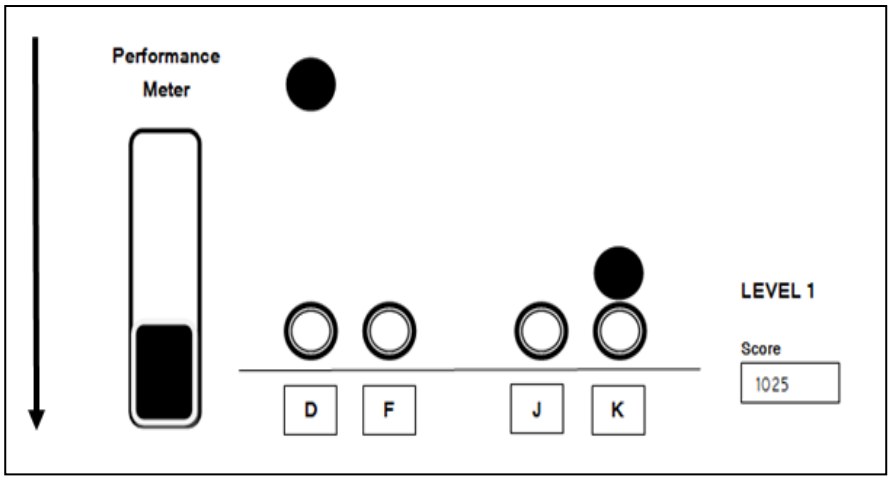

Figure 1. Schematic representation of the four-button serial interception sequence learning (SISL) task. The buttons 'd', 'f', 'j' and 'k' on a keyboard are used for data registration. The filled circles move downwards the screen. Participants need to press the corresponding key accurately when the filled circles are at the position of the open circles.

attacks. The SISL task, first introduced by Sanchez, Gobel and Reber (2010), lets participants train a particular sequence by repetition, but acquiring explicit knowledge is difficult. The task is similar to the popular game Guitar Hero, and moving cues should be intercepted by pressing corresponding keys (see Figure 1).

Bojinov et al. (2014) focused on high security facilities using local password-based user authentication which requires physical presence of users. In this scenario, if an attacker captures the user and coerces him or her to reveal their password, the user cannot do this as the trained sequence is learned implicitly.

Moreover, a SISL-based user authentication system may also provide a solution to the deficiencies of traditional password-based user authentication, because it has higher entropy than most commonly used passwords. The implicitly learned passwords comprise of motor skill sequences. A motor sequence that consists of 12 characters will have an entropy of 8 bits, as is calculated below. Whereas a traditional password of 12 characters, for instance, the word "degradations", has only 3.25 (for a detailed explanation on the calculation, see paragraph 1.1.).

As mentioned earlier, social networking accounts may also contain sensitive, personal information, but these accounts are often hacked. To prevent against brute force attacks a SISL-based authentication may be added to the security protocol.

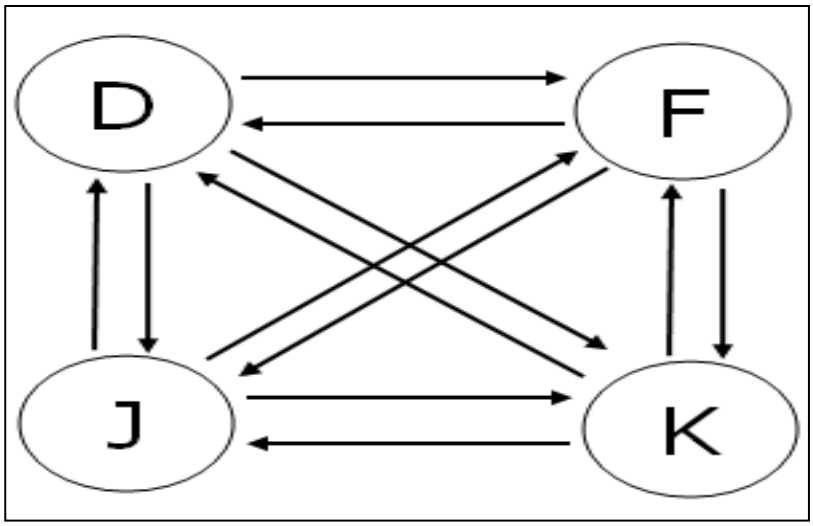

Figure 2. A set of Euler cycles in a directed graph. From this set 12-character sequences are generated.

The SISL task makes use of motor sequences. All these sequences are designed to minimize recognition and to avoid easy to remember patterns. This is done by using a set ∑ of all sequences that correspond to an Euler cycle, this means that all vertices in the graph have an even degree of non-zero degree and are connected (see Figure 2), and the characters used for the sequences are taken from set *S* {d, f, j, k}. An Euler cycle is a cycle that traverses each edge of a directed graph exactly once (Dickman, 1996). So, all sequences are made of six bi-grams (such as 'df' and 'fk') that are never repeated within a sequence (Bojinov et al., 2014). An example of a sequence with the property that every non-repeating bigram over 'S' appears exactly once is *"djfkdkjkfdfj"*.

Bojinov et al. (2014) describe the computation of the number of keys with the BEST theorem and Cayley's formula. Cayley's formula states how many different trees (an undirected graph in which any two vertices are connected by exactly one path) can be constructed on *n* vertices. On every tree there can be multiple Eulerian circuits. For instance, when vertices switch places, the possible routes change and with this, the possible Eulerian circuits. In this formula (see Formula 1.2), for every positive integer $m \in \mathbb{Z}$, where $\mathbb{Z}$ denotes the set of all integers, the number of trees on *n* labelled vertices is n^{n-2} (Casarotte, 2006):

$$|T_n| = n^{n-2} \qquad (1.2)$$

The BEST theorem can be used to count the number of Eulerian circuits in a directed graph (Brightwell & Winkler, 2004):

$$\mathrm{ec}(G) = t_w(G) \prod_{v \epsilon V} (\deg(v) - 1)! \qquad (1.3)$$

In this formula (see Formula 1.3), $t_w(G) = |T_n| = n^{n-2} = 4^2$. Every vertex's in-degree is denoted by deg (v) and has the value 3 in the graph in Figure 2. For the Euler graph in figure 2 corresponding to a four-button SISL task the BEST theorem gives:

$$\#\text{ Keys} = |\Sigma| = 4^2 \cdot 2^4 = 256 = 2^8. \qquad (1.4)$$

Thus, the trained sequence in the SISL task has about 8 bits of entropy (see Formula 1.4). Although this entropy is significantly smaller than with the six-button SISL task (which has 38 bits of entropy), it is still higher than the entropy of traditional language-based password. For instance, the commonly used password *"password"* has only 2.75 bits of entropy. This holds for passwords including numbers and symbols as well. For instance, the password "*Pas$w0rD2O!0*" has only 3.42 bits of entropy, which is far less than the 8 bits of the sequence in the SISL task.

The SISL-based authentication protocol used in the current study is almost similar to the one used by Bojinov et al. (2014). However, it used three 12-character sequences: the trained sequence and two random sequences. Then, the protocol used in this study looks as follows:

- Let k_0 be the trained 12-character sequence and let k_1 and k_2 be two random sequences chosen from $\sum$.
- The system presents the user with a SISL task with the following sequence of 540 items = 45 x 12 characters. Every sequence is repeated five times in a row and followed by the next sequence until every sequence is repeated fifteen times: $k_0, k_0, k_0, k_0, k_0, k_1, k_1, k_1, k_1, k_1, k_2, k_2, k_2, k_2, k_2, \ldots, k_2, k_2, k_2, k_2, k_2$. Where k_0 is the trained sequence.
- For $i = 0, 1, 2$, let p_i be the percentage correct on the sequence k_i during the authentication. Authentication is successful if

$$p_0 > \text{mean}\,(p_1, p_2) + \sigma. \qquad (1.5)$$

In this formula $\sigma > 0$ should cover the possibility that the performance gap occurred by chance. However, the exact size of σ still needs to be determined as this has not been done yet (Bojinov et al., 2014).

1.7. Current study

The current study aims to investigate use of SISL-based authentication in which users implicitly learn their secret key: a motor sequence. This authentication may provide several advantages to original password-based authentication: 1) higher entropy compared to original password-based user authentication, 2) inability of users to reveal the secret key.

Up until now, the length of the training phase for learning motor sequences implicitly, did not receive much attention. The six-button SISL task used by Bojinov et al. (2014) requires at least 30-45 minutes of training before a user can be authenticated. As has been described earlier, users tend to be complacent when it comes to passwords. Once chosen, they do not change their passwords and they tend to use easy and predictable passwords. This way, creating and managing passwords takes as little time as needed. Therefore, it is important to reduce training time as much as possible, to increase the practical use of sequence-based passwords.

In this study a four button SISL task was used with 12-character sequences to reduce training time. As a consequence, entropy was lower compared to a six-button SISL task using 30-character sequences. However, the SISL-based authentication system presented here aimed to improve original password-based user authentication, and therefore did not require very high entropy. To investigate the influence of training length on SISL task performance, three different training phase lengths were compared in an experiment with the four-button SISL task. The longest training length consisted of 1,440 trials, the middle training length consisted of 960 trials and the shortest training length consisted of 480 trials. Possibly, there is a minimum length the training phase needs to have in order to facilitate implicit learning of a motor sequence. This study intended to answer the following main question:

1. *What is the influence of training time on the extent of implicit motor sequence learning in a four-button SISL task?*

Motor sequence learning was expressed by the percentage correct on the trained sequence compared to the percentage correct on untrained sequences. It was expected that participants perform better on the trained sequences compared to random sequences. To ensure motor sequence learning was implicit, the nature of sequence knowledge of the participants was evaluated. Two tasks checked for any explicit knowledge about the sequence trained in the experiment. The recognition task required participants to rate sequences on familiarity and the recall task required participants to reproduce their trained sequence. This leads to the following hypotheses:

Hypothesis 1. *Performance, expressed in percentage correct, on a trained sequence on a four-button SISL task differs across the three conditions;*
Hypothesis 2. *Average recall score differs across the three conditions;*
Hypothesis 3. *Average recognition score differs across the three conditions.*

Implicit learning tasks have shown an increase in performance with practice (e.g., Bojinov et al., 2014; Doyon et al., 2003; Savion-Lemieux & Penhune, 2005; Verwey et al., 2001; Abrahamse et al., 2009; Tzvi, Münte & Krämer, 2014), and therefore, it was expected that performance increases with training length. Moreover, it was expected that the ability to recognize and to recall the trained sequences increased with training time, because it is reasonable that, although knowledge remains implicit, participants do create a stronger association with practice.

It is important for an authentication system not to use one-time only passwords (OTP), and therefore it was investigated whether users can authenticate over time. To check if sequence knowledge remains implicitly in memory, a follow-up experiment was done five weeks after the first experiment, in which participants complete authentication and a recognition and recall task.

II. METHOD

1. Experiment 1

1.1. Participants

37 adults living in the Netherlands participated in this experiment. One participant was excluded from the study, because of software malfunctioning, which resulted in a group of 36 participants (19 male and 17 female). Participants were selected through convenience sampling. The participants were between 18 and 60 years of age (mean age = 24.1, SD = 8.7). All participants were tested individually. Prior to the study, all participants gave their written informed consent. 50% were psychology students from the university of Twente and 17 participants received course credits for their participation. The study was approved by the ethics committee of the Faculty of Behavioral Sciences at the University of Twente.

1.2. Apparatus

Matlab R2014b was used to present the stimuli and register data. Psychtoolbox was installed in order to visualize the stimuli and to record responses. The experiment ran on a HP ProBook 4530s laptop. The ‘d’, ‘j’, ‘f’ and ‘k’ keys of a regular QWERTY-keyboard were used for data registration.

1.3. Task and procedure

The task involved a serial interception sequence learning task in which a 12-character sequence was pressed in response to sequence-specific visual cues.

Prior to the experiment, a 24-trial demo was run to make the participants familiar with the four-button SISL task. Then, the actual experiment was run. Subsequently, a recognition and a recall task were executed.

In the experiment, participants placed their left middle, left index, right index and right middle finger on the ‘d’, ‘f’, ‘j’ and ‘k’ keys of a QWERTY-keyboard respectively. The experiment started with instructions presented in the middle of the screen. The participant had to press any key to continue to the first training level. Each trial started with a blue circle (cue circle) at the vertical position of one of the four dashed yellow rings (target circles), with an interval of 500 ms in between consecutive trials. The cues circles scrolled from the top to the bottom of the screen. They initially scrolled with a speed of 10 cm/s, however, speed was made adaptive to keep participants challenged during the task. A good performance caused the speed to increase and a poor performance decreased the speed. The cues reached the target

in 2.0 s, travelling a total distance of 25 cm from the top to the bottom of the screen in 2.5 s. Participants received immediate feedback on their responses, however it might have been difficult to pay attention to upcoming cues and the given feedback simultaneously. Whenever the participants pressed a key, the initially yellow target circles temporally changed color. The target circles turned red or green for incorrect and correct responses, respectively. A response was counted as correct if the participant pressed the corresponding key while a cue circle overlapped with the target circle as much as possible. Incorrect responses, multiple simultaneous responses and non responses were counted as errors.

Participants were randomly distributed over three conditions. During the training phase, condition S had to perform a training phase of one level, condition M had to perform a training phase of two levels, and condition L had to perform a training phase of three levels. One level consisted of 8 blocks in which the trained sequence repeated five times, giving a total of 480 trials per training level. After completion of the training phase, all participants completed a test phase of one level. In this test level, the repeated sequence and two foil sequences were all repeated 15 times, with a total of 540 trials. Between levels participants were offered a self-paced break.

2. Experiment 2

2.1. Participants

15 of the participants of the initial experiment (7 male and 8 female) participated in the follow-up study five weeks after the initial experiment. The participants were between 18 and 60 years of age (mean age = 26.7, SD = 12.9). Three participants were initially in condition S, four in condition M and eight were in condition L. All participants were tested individually. Prior to the study, all participants gave their written informed consent, which was approved by the local ethics committee. Two participants received course credits for their participation.

2.2. Apparatus

Matlab R2014b was used to present the stimuli and register data. Psychtoolbox was installed in order to visualize the stimuli and to record responses. The experiment ran on a HP ProBook 4530s laptop. The ‘d’, ‘j’,. ‘f’ and ‘k’ keys of a regular QWERTY-keyboard were used for data registration.

2.3. Task and procedure

Five weeks after initially participating in the first experiment, participants completed the follow-up experiment. The task involved a serial interception sequence learning task similar to the one in the initial experiment. A 12-character sequence was pressed in response to sequence-specific visual cues. The task was the same as in the initial experiment, except for the initial speed, which was set to the final speed of the participant in the initial experiment. The follow-up experiment consisted of 180 random trials as a warm up, followed by a retention test of 540 trials of 15 times the trained sequence, 15 times a random sequence used in the initial experiment and 15 times a random sequence.

After the test level was finished, the participants again completed the recognition task and recall task. These were exactly the same as in the initial experiment.

III. RESULTS

1. Experiment 1

1.1. SISL performance

1.1.1. training phase.

During the training phase the average performance (percent correct) was 65.76%, with an average performance of 68.45% in condition S, 64.69% in condition M and 64.15% in condition L. There was a significant better average performance in condition S compared to the conditions M and L, $F(2, 35) = 5.47$, $p < 0.010$, 95% CI [0.002, 0.074] and [0.007, 0.079] respectively.

Learning effects were analyzed with a repeated measures ANOVA with Time (2; first block versus last block) as within-subjects factor and Group (3; S, M and L) as between-subjects factor. This produced significant main effect of Time (See Figure 3), $F(1, 33) = 9.10$, $p < 0.010$, indicating that participants' average performance on the first block in all three conditions significantly differed from average performance on the last block they completed. The difference is expressed in decreased performance in the last block compared to the first block.

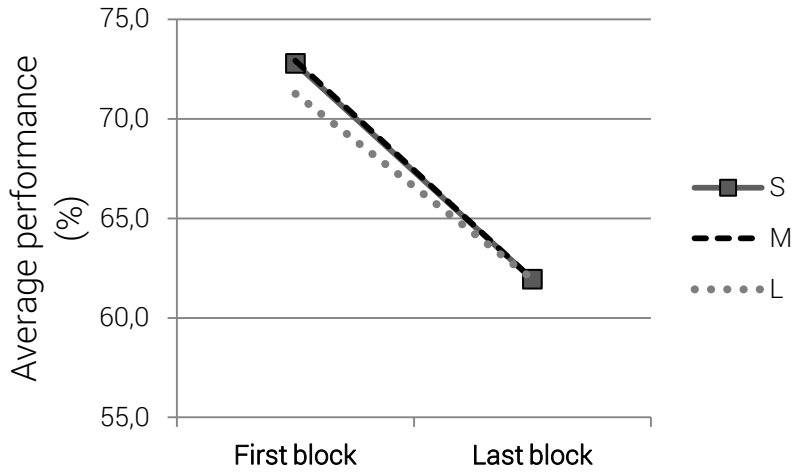

Figure 3. Average performance (%) in the first block and the last block for all three conditions S, M and L on the four-button SISL task.

1.1.2. Test phase

The test phase (authentication) comprised of the last level of the SISL task, in which both the trained sequence and two random sequences are presented to participants 15 times each, resulting in a level of 540 trials.

A one-way between groups ANOVA showed a non-significant effect for training length on average performance on the trained sequence, $F(2, 35) = 0.30$, $p = 0.743$; for random sequences, $F(2, 25) = 2.57$, $p = 0.092$; and for trained and random performance together represented as SISL score, $F(2, 35) = 0.34$, $p = 0.712$.

Analysis of the average percentage correct on the trained sequences and the average percentage correct on the two random sequences in the test level showed that all participants performed at an average rate of 63.63% (SD 14.37%) correct for the trained sequence, and on average 57.65% (SD = 9.84%) correct on random sequences. The overall performance was on average 59.64%. A paired-samples t-test showed a significant difference between performance on the trained sequence compared to average performance on the random sequences, $t(35) = 3.02$, $p < 0.01$, indicating that participants performed significantly different on the trained sequence compared to the random sequences (See Figure 4).

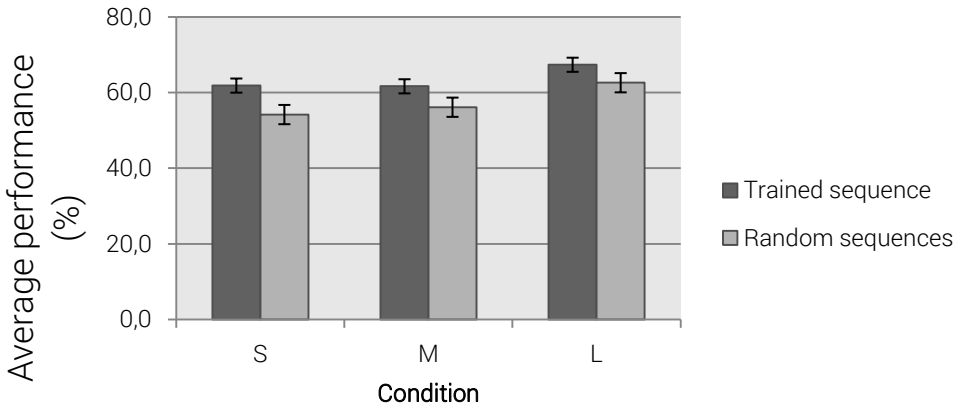

Figure 4. Average performance (%) on the trained sequence and random sequences of all participants in the three conditions S, M and L on the four-button SISL task in the test level. Error bars represent standard errors.

A one-way ANOVA was conducted to compare the SISL score for all three conditions. The SISL score is calculated by subtracting the mean performance on the two random sequences from the mean performance on the trained sequence. There was not a significant difference between the conditions, $F(2, 35) = 0.19$, $p = 0.832$, indicating that training length did not significantly influence performance on the trained sequence compared to random sequences (SISL score) reliably. The average SISL score was 5.98. Analysis of the three conditions showed that participant's difference in average performance on the trained sequence and random sequences was greatest in condition S, followed by condition M and with smallest difference in condition L (See Table 1). However, this difference between the conditions was not significant, $F(2, 35) = 0.19$, $p = 0.832$.

Table 1

Mean performance (%) on trained sequence and random sequences and difference between them for all three conditions (S, M and L)

Condition	Trained sequence (%)	Random sequences (%)	Difference (%)
S	61.85	54.19	7.66
M	62.78	56.20	6.58
L	66.25	62.55	3.70

1.2. Recognition score

Five participants rated their trained sequence correctly with certainty (a score of 10). Participants rated the trained sequence moderately positive familiar with an average score of 3.58 (SD = 5.60). The random sequences were rated moderately negative familiar with an average score of -2.05 (SD = 2.98). The average recognition score was 5.63 (SD = 6.88). A one-sample t-test showed it was significantly greater than zero, $t(35) = 4.91$, $p < 0.001$, indicating some explicit sequence knowledge of the participants. However, expression of sequence knowledge (high recognition score) did not correlate with SISL task performance, $r(34) = 0.23$, $p = 0.170$. Also, expression of sequence knowledge did not reliably predict performance on the SISL task, $R^2 = 0.05$, $F(1, 35) = 1.97$, $p = 0.170$. There was no significant difference in sequence knowledge between the three conditions.

1.3. Recall score

The recall score consists of the largest chunk of trials participants could recall (for instance, ‘djk’ is a chunk of three). The average recall score for the trained sequence was 4.47 (SD = 1.65). However, a high recall score did not correlate with performance, $r(34) = 0.11$, $p = 0.509$. Also, expression of sequence knowledge did not significantly predict performance on the SISL task ($R^2 = 0.01$, $F(1, 35) = 0.45$, $p = 0.509$).

2. Experiment 2

2.1. SISL performance

After a warming up of 180 random trials to let participants adapt to the speed, all three 12-character sequences are presented 15 times, resulting in a level of 720 trials. One

sequence was the trained sequence, one a new random sequence and one a random sequence that was used in the initial experiment.

A one-way between groups ANOVA was conducted in order to compare the effect of training length on performance on the trained sequence in the test level of the four-button SISL task in the three conditions S, M and L. There was not a significant effect of training length on performance, $F(2, 14) = 1.35$, $p = 0.297$. Participants did not perform significantly different on the trained sequence in the three conditions.

Table 2

Mean performance (%) on trained sequence and random sequences and difference between them for all three conditions (S, M and L)

Condition	Trained sequence (%)	Random sequences (%)	Difference (%)
S	63.0	64.0	1.0
M	71.7	58.7	13
L	71.0	62.0	9

Analysis of the average percentage correct on the trained sequences and the average percentage correct on the two random sequences in the test level showed that all participants performed at an average rate of 69.63% (SD 8.09%) correct for the trained sequence, and on average 61.09% (SD = 4.40%) correct on random sequences. The overall performance was on average 63.94% (SD = 2.79%). A paired-samples t-test showed there was not a significant difference in performance on the two random sequences in the authentication, $t(14) = -0.10$, $p = 0.926$. However, a paired-samples t-test showed a significant difference between performance on the trained sequence compared to average performance on the random sequences, $t(14) = 3.00$, $p = 0.009$, indicating that participants performed significantly different on the trained sequence compared to the random sequences (See Table 2). In an analysis of mean performances, condition M showed with 13 percent the biggest difference between performance on the trained sequence and the random sequences.

A one-way ANOVA was conducted to compare the SISL score for all three conditions. The SISL score is calculated by subtracting the mean performance on the two random sequences from the mean performance on the trained sequence. There was not a significant difference between the conditions, $F(2, 14) = 1.73$, $p = 0.219$, indicating that training length did not significantly influence performance on the trained sequence compared to random sequences (SISL score). The average SISL score was 8.54.

2.1.1. Retention

A paired-samples t-test was used to compare average performance on the trained sequence, random sequences and the calculated SISL score from the first experiment and the follow-up experiment. This showed that only performance on the random sequences in the follow-up was significantly higher compared to the first experiment, $t(14) = 2.65$, $p = 0.019$, 95% CI [0.85, 8.00]. The average SISL score in the first experiment was 5.98, and in the follow-up it was 8.54.

2.2. Recognition score

Three participants rated their trained sequence correctly with absolute certainty (a score of 10). Participants rated the trained sequence moderately positive familiar with an average score of 5.33 (SD = 6.86). The random sequences were rated moderately negative familiar with an average score of -3.10 (SD = 5.25). The average recognition score was 8.43 (SD = 8.45). A one-sample t-test showed the recognition score was significantly greater than zero, $t(14) = 3.87$, $p = 0.002$, indicating some explicit sequence knowledge. However, expressed sequence knowledge (high recognition score) did not significantly predict performance on the SISL task, $R^2 = 0.05$, $F(1, 14) = 0.03$, $p = 0.873$. Expression of sequence knowledge also did not correlate with SISL task performance, $r(13) = 0.05$, $p = 0.873$. There was no significant difference in average recognition score between the first experiment and the follow-up, $t(14) = 0.98$, $p = 0.343$.

2.3. Recall score

The recall score consists of the largest chunk of trials participants could recall (for instance, ‘djk’ is a chunk of three). The average recall score for the trained sequence was 5.07 (SD = 2.31), which was in the first experiment 4.47 (SD = 1.65). However, a high recall score did not correlate with performance, $r(13) = 0.03$, $p = 0.896$. Also, expression of sequence knowledge did not reliably predict performance on the SISL task ($R^2 = 0.03$, $F(1, 14) = 0.02$, $p = 0.896$). There was not a significant difference between average recall score of the first experiment and follow-up, $t(14) = 1.35$, $p = 0.196$.

IV. DISCUSSION

The goal of the current study was to determine the influence of training length on SISL task performance in a four-button SISL task. The study shows that training length does not influence performance of the SISL task. Below, the implications of these findings are discussed.

Performance in training phase. Firstly, both experiments show that participants performed less than the 70% correct that was aimed for in the task by adapting the speed of the moving cues, which is not in line with the performance found by Bojinov et al. (2014). This implies participants do not learn the trained sequence perfectly, nor do they become so familiar with task that they can perform it with a 100% correct rate. The 70% correct that was aimed for by adapting speed in this study was taken from the study by Bojinov et al. (2014), however, there are two more important requirements that performance needs to meet for the SISL-based authentication system to work: a) performance is not perfect, because perfect performance may imply that participants have obtained explicit sequence knowledge and would negate the SISL-based authentication system, and b) a difference exists between trained and random sequences.

Secondly, experiment 1 shows there is no learning effect during the training phase in the sense that average performance on the trained sequence does not improve. Average performance was expected to enhance with practice, given the motor skill required to perform the SISL task (Doyon et al., 2003; Savion-Lemieux & Penhune, 2005; Verwey et al., 2001; Abrahamse et al., 2009; Tzvi, Münte & Krämer, 2014), but in fact the opposite is the case as performance reduced over time. This contradicts earlier findings in a study by Gobel, Sanchez & Reber (2011) which showed an enhanced performance on a four-button SISL task in 1,440 trials (similar to the condition L in this study). There is an important difference that should be considered in the design of the experiments. Learning in the study by Gobel et al. (2011) was shown by significant drops in performance when switched from trained sequence to random trials and not by an overall improvement of performance on the trained sequence. In this study, this was shown in the test phase, which included the trained sequence and random sequences. It seems plausible to assume that participants perform less well on random sequences compared to trained sequences after practice. Also, it is plausible that this would have been the case in this study if training phase would contain not only a trained sequence (constantly repeating), but also made a comparison with random sequences. This is not tested in this study, because to facilitate as much learning as possible, a short training phase was

prioritized. Therefore, the training phase consists of the trained sequence only. Considering this fact, there is a learning effect in this study as well. It is just defined differently: learning effect is shown by trained sequence advantage.

Performance in authentication phase. Firstly, results from both experiments converged with previous findings (Bojinov et al., 2014) that showed the presence of a learning effect, which is expressed in the difference in performance on trained and random sequences. Average performance on the trained sequence is significantly higher when compared to average performance on random sequences. Training length did not influence performance significantly, disproving hypothesis 1. Experiment 2 shows that a random sequence from earlier authentication can be reused in next authentication, because there was no performance difference is found between the reused and a new random sequence. This indicates that the authentication system could use the same sequences over and over again, and there is no need to change them.

Secondly, despite the significant difference between the trained sequence and random sequences, the average performance of participants was only 59.64% in experiment 1 and 63.93% in experiment 2. This finding contradicts the notion that adapting the speed could keep the aimed average performance at a correctness rate of 70% correct (Bojinov et al., 2014). In condition L the training phase consisted of 1,140 trials, which is similar to the training phase in a study by Gobel et al. (2011), but performance of participants in condition L was often poorer than performance of participants in the study by Gobel et al. (2011). This relatively poor performance may not be an issue, it implies that users of the SISL-based user authentication system do not learn the task nor the trained sequence perfectly. With this, there is little likelihood of gaining explicit knowledge by performing the task. It was expected that authentication success rate is based on training, but this study showed that there is no significant difference between the three different training lengths. Thus, a training length of 480 trials, approximately four minutes, can be sufficient for this authentication system.

Interestingly, results show that participants improve over a period of 5 weeks (an increase in average overall performance of 4.30%). It can be speculated that participants were more familiar with the SISL task the second time they performed the task. However, it is very important to consider that only 15 participants from experiment 1 also participated in experiment 2. The data shows that participants in study 2 did not perform better on average than in study 1, so the difference was not caused by a selection bias. It may be that performance learned in the SISL task needs time to sink in, a process known as

consolidation (Brashers-Krug, Shadmehr & Bizzi, 1996; Walker, Brakefield, Morgan, Hobson & Stickgold, 2002; Karni et al., 1997). Further research can investigate SISL task performance over time and on more than two separate occasions.

Furthermore, study data indicates that recognition and recall scores do not seem to significantly predict SISL performance. This is consistent with findings from previous studies by Gobel et al. (2011), Sanchez et al. (2010) and Bojinov et al. (2014). Moreover, these scores do not depend on training length. Explicit sequence knowledge does not seem to increase with training length, disproving hypotheses 2 and 3. These results support the notion that users do not gain explicit knowledge of the trained sequence during authentication. As mentioned earlier, performance does not seem to reach to point of perfect hit rate. Therefore, it is unlikely that explicit knowledge of the trained sequence develops. In addition, if the SISL-based authentication system is used for social networking sites and users authenticate themselves on a daily basis, based on previous research, it can be argued that knowledge on the sequences remains implicit (e.g., Karni et al., 1997; Walker et al., 2002). However, it is important to note that the motor sequences used in this study are build from characters referring to particular keys on a regular keyboard. As the sequence is only 12 characters long and also related to keys (‘d’, ‘j’, ‘f’ and ‘k’), it may be feasible for users to memorize the sequence explicitly if they repeatedly authenticate themselves. On the one hand, findings from a previous study by Bojinov et al. (2014) showed that users do not obtain explicit sequence knowledge as they show no ability to recall the sequence after a training phase of 3780 trials. On the other hand, items in the beginning and end of a sequence are remembered more accurately than items in other positions (Nevins, 2010). If the sequence in the SISL-based authentication system is deliberately linked to meaningful characters, it could become explicit knowledge. It is also known that memory is limited to seven items plus minus two items (Miller, 1956) and a 12-character sequence can easily be memorized, or divided into either six chunks of two items or four chunks of three items. Moreover, healthy participants are able to report (parts of) the sequence verbally, if the sequence is relatively short (four or twelve items). This is shown in both previous research and the current study (Reber & Squire, 1998). Thus, memorization of the sequence may be a potential limitation of SISL-based user authentication and future research can investigate this.

Moreover, the SISL scores (trained sequence advantage) on both experiments do not significantly differ (6 in experiment 1 and 8.5 in experiment 2). This indicates that users are still able to authenticate after a 5-weeks delay. There is no decrease in SISL score over time, which is opposed to the findings of Bojinov et al. (2014).

To conclude, a SISL-based user authentication system seems to have several implications and limitations that should be considered before its implementation into practice.

Firstly, training length does not seem to influence performance and therefore, it must be investigated what is the actual antecedents of the learning effect expressed in trained sequence advantage. The current study shows that, if a significant difference between performance on the trained sequence and random sequences is sufficient for authentication, the SISL task may be suitable for authentication. However, it is important to further investigate performance on the SISL task with an even shorter training, because an even shorter training may already be sufficient. This increases usability, as is known that people do not want to put too much effort in managing their passwords (Adams & Sasse, 1999). However, as motor skill is acquired by practice, it seems unlikely that very short training would be sufficient. Nevertheless, this study shows there is no influence of training length on SISL scores and therefore, it is important not to prematurely dismiss this possibility.

In addition, the process of consolidation is important in the sense that future research should determine what amount of training is followed by what amount of memory consolidation and what level of implicit sequence knowledge is sufficient to accurately authenticate users.

Thirdly, memorization is important to take into account, because the implicitly trained sequence cannot become explicit knowledge for the sake of security. This also includes the aspect of performance, because it seems highly likely that knowledge becomes explicit if performance is perfect.

Lastly, it is important to consider that experiments were conducted in an experimental setting using a select group of individuals. Password-based user authentication is used in different, real life circumstances and by many different people. To create better external and ecological validity, it is highly recommendable, before implementing this new authentication system, to test it with a large group of individuals in a natural environment.

4.1. Security analysis

First, it is important to consider potential implications and mitigations of the authentication system to ensure security ─ this is done in the current paragraph. Then, in paragraph 4.2. the newly proposed authentication system protocol is described in detail. In paragraph 4.3. a possible password guessing attack is considered. Last, in paragraph 4.4. a general conclusion is given.

Although it is difficult for an adversary to remember the sequences at the speed of the task and it seems hard to construct the sequence, Bojinov et al. (2014) propose the use of fragments of sequences to create a chance of $\frac{1}{500}$ that an attacker who slows down on random sequences will pass the test. However, this is inadvisable when using sequences of only 12 characters long, as fragments with a length of less than five do not accurately assess sequence knowledge (Bojinov et al., 2014). Then, the sequence could be split in two. This creates two fragments of the trained sequence and four fragments of random sequences. Moreover, Bojinov et al. (2014) propose a system that switches to a random-mode if it observes that the user is not able to demonstrate a significant trained sequence advantage. Still, if an adversary authenticates multiple times, it is possible to learn the fragments of the sequence. Therefore, Bojinov et al. (2014) propose a system that shuts down when knowledge on only a subset of fragments of the trained sequence is shown. In addition, the authentication protocol may start with a random sequence of random length, which makes it harder to distinguish the start and end of the actual sequence. Despite these design measures to make it more difficult for an adversary to hack the authentication system, there are some potential implications and mitigations to take into consideration.

Firstly, if the system blocks a user, how could this be undone? With this, an important question is: Is performance consistent enough in order for this to be a good authentication system that rules out denying a legitimate user having a bad day or being under a lot of stress (for instance, someone who is in a hurry)? These circumstances should be investigated, because the authentication system is based on performance and performance is known for its sensitivity for manipulations (e.g., Lazarus, Deese & Osler, 1952). Also, would performance be consistent over and over again? Should previous performances be considered? In the protocol the success of the authentication is based on $p_0 >$ mean $(p_1, p_2) + \sigma$, in which σ needs to be determined. Future research needs to concentrate on the value of σ for this formula.

Secondly, the system shuts down after it detects a lack of trained sequence advantage. Because of this, the attacker could obtain little bits of knowledge every time he or she attempts to unlock the device. A possibility to prevent this is to use honey passwords, which presents a dishonest user with a fake SISL task when an incorrect key is used. Using honey passwords, the attacker does not get to see the real SISL task at all. If a user is coerced to enter their credentials and play the game, so that the adversary can watch a trained user to learn from him or her, a honey password misleads the adversary. For a further elaboration on

honey passwords see the next section of the newly proposed SISL-based authentication protocol.

Thirdly, at some point in the procedure, the sequence needs to be visualized to the user. This should be done either with encrypted sequences, when they need to be stored or visualization should be possible without storage of the sequences. It is important to consider this and to design authentication software that is capable of handling the visualization in this way.

4.2. SISL-based authentication system (SAS) protocol

This section extends upon the short description on the SISL-based authentication system (SAS) given in the introduction and the authentication system described by Bojinov et al. (2014). Below, the protocol is described and a schematic representation is given (see Figure 5).

When the user first uses the SAS, a trained sequence and two random sequences need to be created. This happens at the client-side, because the protocol is designed in such a way that the sequences are never revealed in the clear to outsiders or the server. This is to prevent from so called man-in-the-middle attacks (an attack in which data is intercepted during transfer), such as eavesdropping (Yoon, Shin, Jeon & Yoo, 2010). To identify him- or herself, the user needs to enter a user ID and a password, the latter is hashed by a hash function. A hash function is a cryptographic function which outputs encrypted data, but is practically impossible to revert (Dimitriou, 2005). Because of this, the password is never known by anyone besides the user him- or herself. The client sends the user ID and hashed password to the server, which then checks if this matches the hashed password that belongs to that particular user ID stored on the server. If this matches, the server sends back an acceptance code and the client gets permission to create the sequences. The sequences are all immediately encrypted under key *k*, which is derived from the password that is entered by the user. The creation of sequences happens only the first time, when the user becomes a trained user. The system presents the user with the secret sequence and lets him or her train on this sequence. After this, users enter their user ID and passwords in order for authentication to start.

Then, if a registered user wants to authenticate, the user enters a user ID and password. The client computes a hashed password = h(pw), in which h() is a one-way hash function. If the hashed password matches the one stored on the server for that user ID, then the server sends the encrypted sequences that were stored during registration of that particular user. Again, these sequences are the trained sequence and the two random sequences

encrypted under key *k*. However, they are split in two, to create two fragments of each sequence, denoted in Figure 5 as ([Seqfrag$_{train0}$]$_k$, [Seqfrag$_{train1}$]$_k$, [Seqfrag$_{rand0}$]$_k$,…, [Seqfrag$_{rand3}$]$_k$). Based on the security enhancement proposed by Bojinov et al. (2014) this creates a chance of $1/\binom{6}{2} = \frac{1}{15}$ of successful authentication if an adversary randomly slows down on two out of six sequences, instead of the original $\frac{1}{3}$ chance. The server permutes the order of the fragments sequences and stores this permutation. In this way, it is more difficult to figure out which (fragment of) sequence is the trained sequence. For a six-tuple as described above, consisting of six fragments of sequences, this means that there are 720 possible permutations:

$$\frac{6!}{(6-6)!} = 720. \qquad (4.1)$$

When the client sends back the performance scores on the sequences, the server is able to figure out by using the known permutation, which sequence was the trained sequence and check if difference with the random sequences is sufficient for authentication. This way, the server does not need to know the exact sequence, but it only needs to distinguish the trained sequence from the random sequences. To continue, the encrypted sequences in a permutated order are sent to the client. In addition, a bogus sequence is added to make it harder to distinguish the actual beginning of the sequences. This bogus sequence has a random length of maximum 120 random cues (this way, not more than one minute is added to authentication) and is not included in performance analysis. The client decrypts the sequences with the key *k*, and lets the user play the SISL game. Afterwards, the performance scores on the fragments are encrypted and send back to the server, denoted in Figure 5 as ([Pscorefrag$_{train0}$]$_d$, [Pscorefrag$_{train1}$]$_d$, [Pscorefrag$_{rand0}$]$_d$,…, [Pscorefrag$_{rand3}$]$_d$). The fragments performance scores are computed back into scores for complete sequences by the server. The server can then verify these scores by their order with the stored permutation and decide whether authentication is successful. In others words whether the following is true, where p_0 is the performance on the trained sequence, p_1 and p_2 are performance on the two random sequences:

$$p_0 > \text{mean}\,(p_1, p_2) + \sigma. \qquad (4.2)$$

In addition, users can type in a honey password instead of their real password, whenever they are being coerced by an adversary (see Appendix A for a schematic

representation of the honey passwords protocol). Imagine the situation in which an adversary coerces the user to enter their password, play the game and let the adversary record it. The goal of the adversary in this case, is to figure out the trained sequence and practice on it him- or herself at home. In addition to the design measures described in the security analysis that make is more difficult for an adversary to gain knowledge on the sequences, the use of honey passwords adds even more difficulty. When the user enters his or her user ID, but with a honey password, the server checks the hashed honey password that is entered with the hashed honey password stored on the server. If they match, a fake SISL task is started from which the adversary cannot extract any correct sequence information. However, the user never successfully authenticates when performing the honey SISL task. A reason that can be given by the user is that this task is too difficult when under a lot of stress (caused by the coercion). With this, it is important to be careful with malicious adversaries who may get angry and put the user in danger, but the SAS fulfils its role in protecting the system.

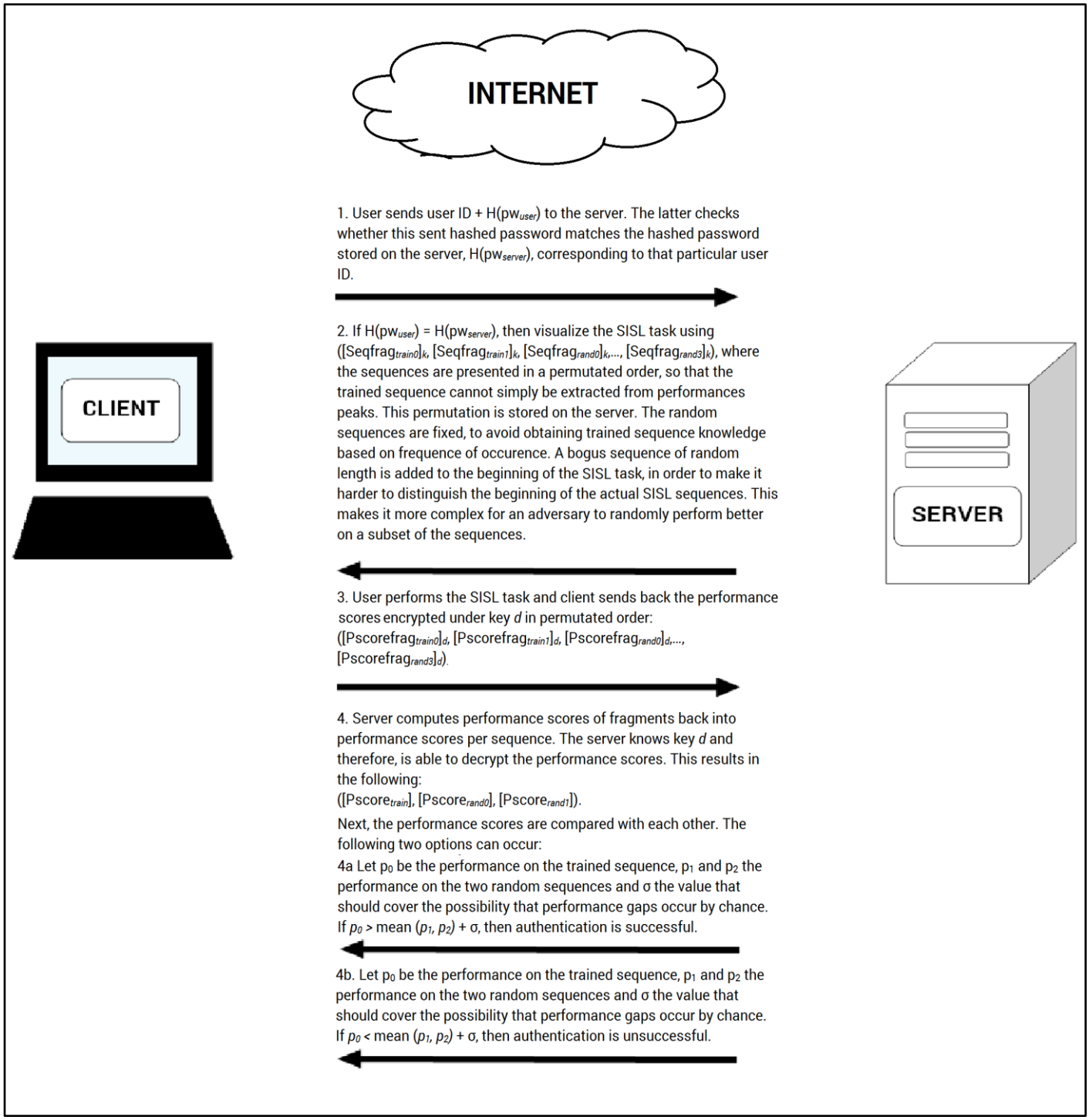

Figure 5. Schematic representation of the SAS protocol. A two-factor authentication using both a user id + password combination, and a secret key embedded in motor skill in the SISL task. Here, $[\text{Seqfrag}_i]_k$ refers to an encryption of the fragments of sequences under key *k*. Key *k* is only known by the client. At step 4, the server computes the performance scores of the sequences with key *d*, from the performance scores on the fragments of sequences and their stored permutation. Then, it checks the performance scores of the sequences and is authentication either successful or unsuccessful.

4.3. Password guessing attacks

Let Eve be an active attacker who wants to hack the proposed SAS by performing a password guessing attack. It has been discussed that traditional password authentication is vulnerable to these attacks, because of low entropy and predictable passwords. With this authentication system this kind of attack is not very likely to succeed.

The trained sequence serves as secret key in the SAS. Eve does not have any knowledge of the sequence, but needs to perform exactly as expected during the authentication SISL task. This means Eve needs to randomly create a drop in performance on four of the six sequence fragments. However, this is made highly complex, because a) a bogus sequence is added to the beginning of the SISL task, which creates extra difficulty in identifying the beginning of the sequences, b) the task is performed at a significant speed, which makes it hard to deliberately distinguish sequence fragments, and c) the chance of successfully authenticating by dropping performance on random fragments of sequences, is only $1/\binom{6}{2} = \frac{1}{15}$. This chance seems reasonably high, but it should be considered that every time the user authenticates, sequence appear in a different order (recall, there are 720 possible orders) and there is a different bogus sequence to confuse the adversary. For instance, if the bogus sequence consist of the maximum of 120 cues, the chance to successfully authenticate described above, becomes $1/\binom{16}{2} = \frac{1}{120}$. Recall, the password "Pas$w0rD2O!0". The chance of guessing this one correct is the first time: $\frac{1}{94}$, because the character set consists of lower- and uppercase letters, numbers and symbols (26 possibilities + 26 possibilities + 10 possibilities + 10 possibilities, respectively). However, to guess a traditional password, password guessing technologies can be used, in contrast to the SISL task in which an attacker needs to perform the task manually, which demands a lot more time and effort. The first time Eve tries the SISL task in the SAS, it is reasonable to expect that performance is not exactly as required to successfully authenticate (i.e., creating a performance gap on the four random sequence fragments compared to the trained sequence fragments). Consequently, the system knows, by noticing this failure to show a trained sequence advantage, it is under attack. It switches to random-mode and Eve cannot extract anymore sequence knowledge from performing the task. So, besides the difficulty to extract sequence knowledge by performing a password guessing attack, it is made even more difficult, because the system misleads the attacker in a sense. If this happens multiple times, the system may even shut down and force the user to train for a different password. Which brings Eve back to the starting point.

4.4. Conclusion

The aim of this study was to further investigate the SAS, initially proposed by Bojinov et al. (2014). The study showed that authentication based on implicit sequence learning may have future potential. Results showed that a short training is sufficient to create trained sequence advantage and that sequence knowledge remains over time. Evidence was found for the presence of memory consolidation. To enhance security, random bogus sequences and honey passwords were added to the SAS. It was showed that with the SAS it is difficult to dishonestly authenticate using password guessing attacks, at least more difficult than in most cases of traditional password authentication.

Future work includes research on the robustness of these two additions, the external and ecological validity of this SAS, the learning curve including the process of consolidation, finding the exact value of σ for the authentication formula, and conducting experiments to investigate the resistance of the SAS against multiple different attacks.

Acknowledgements

I would like to thank my supervisors prof. dr. Frank van der Velde and dr. Andreas Peter for their enthusiasm, valuable feedback and guidance. I am also thankful for the help that Daniel Sanchez offered me with rewriting the Matlab scripts that were used in this study. Thank you for all your efforts. Moreover, I would like to say thanks to Pim Hendriks, Renske van Waveren and Niels van Berkel for providing me with valuable feedback during my writing process. Last but not least, I would like to thank all participants that offered their help in this research.

V. REFERENCES

Abrahamse, E. L., van der Lubbe, R. H. J., & Verwey, W. B. (2009). Sensory information in perceptual-motor sequence learning: visual and/or tactile stimuli. *Experimental Brain Research, 197,* 175-183.

Adams, A., & Sasse, M. A. (1999). Users are not the enemy: why users compromise computer security mechanisms and how to take remedial measures. *Communications of the ACM, 42*, 40-46.

Ammelrooy, P., van. (2015, April 11). Wachtwoord wAcHTwoOrd wctAHoRDwo WTRoaOChdw. De Volkskrant, pp. A13, A17.

Batterink, L. J., Oudiette, D., Reber, P. J., & Paller, K. A. (2014). Sleep facilitates learning a new linguistic rule. *Neuropsychologia, 65*, 169-179.

Bojinov, H., Sanchez, D., Reber, P., Boneh, D., & Lincoln, P. (2014). Neuroscience meets cryptography: Crypto primitives secure against rubber hose attacks. *Communications of the ACM, 57,* 110-118.

Brashers-Krug, T., Shadmehr, R., & Bizzi, E. (1996). Consolidation in human motor memory. *Nature, 382*, 252-255.

Brostoff, S., & Sasse., M. A. (2002). Save and sound: A safety-critical approach to security. *Association for Computing Machinery*, 41-50.

Burr, W. E., Newton, E. M., Perlner, R. A., Polk, W. T., Sabari, G., & Nabbus, E. A. (2013). Electronic Authentication Guideline. *NITS Special Publication, 800*, 1-111.

Burr, W. E., Polk, W. T., & Dodson, D. F. (2004). Draft recommendation for electronic authentication. *NIST special publication, 800,* 1-51.

Cowley, S. How a lying 'social engineer' hacked Wal-Mart. Retrieved June 2, 2015, from http://money.cnn.com/2012/08/07/technology/walmart-hack-defcon/

Cyber Streetwise (2014). Three quarters of Britons risking online safety. Retrieved April 28, 2015, from https://www.cyberstreetwise.com/blog/three-quarters-britons-risking-online-safety

Data classification, access, transmittal, and storage. (2014, May, 29). Retrieved from http://web.stanford.edu/group/security/securecomputing/dataclass_chart.html

Davies, C., & Ganesan, R. (1993). BApasswd: A new proactive password checker. Retrieved February 12, 2015, from http://www.findravi.com/downloads/BApasswd-ANewProactivePasswordChecke.pdf

DeShon, R., & Alexander, R. A. (1996). Goal setting effects on implicit and explicit learning of complex tasks. *Organizational Behavior and Human Decision Processes, 65*, 18-36.

Dickman, P. (1996). Incremental, distributed orphan detection and actor garbage collection using graph partitioning and Euler cycles. Retrieved February 20, 2015, from http://download.springer.com/static/pdf/289/chp%253A10.1007%252F3-540-61769-8_10.pdf?auth66=1424437683_c940bf1be4764a56b1d8b04dc10d4934&ext=.pdf

Dimitriou, T. (2005). A lightweight RFID protocol to protect against traceability and cloning attacks. In *Security and Privacy for Emerging Areas in Communication Networks.* 59-66.

Doyon, J., Penhune, V., & Ungerleider, L. G. (2003). Distinct contribution of the cortico-striatal and cortico-cerebellar systems to motor skill learning. *Neuropsychologia, 41,* 252-262.

Ellis, N. (2008). Implicit and explicit knowledge about language. *Encyclopedia of Language and Education, 2,* 1-13.

Gertner, Y., Fisher, C., & Eisengart, J. (2006). Learning words and rules: Abstract knowledge of word order in early sentence comprehension. *Psychological Science, 17,* 684-691.

Gobel, E. W., Sanchez, D. J., & Reber, P. J. (2011). Integration of temporal and ordinal information during serial interception sequence learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 37*, 994-1000.

Gold, S. (2010). Social engineering today: psychology, strategies and tricks. *Network Security, 11*, 11-14.

Ives, B., Walsh, K. R., & Schneider, H. (2004). The domino effect of password reuse. *Communications of the ACM, 47,* 75-78.

Kalat, J. W. (2013). Biological psychology. Wadsworth: Cengage Learning.

Karni, A., Meyer, G., Rey-Hipolito, C., Jezzard, P., Adam, M. M., Turner, R., & Ungerleider, L. G. (1997). The acquisition of skilled motor performance: Fast and slow experience-driven changes in primary motor cortex. *Proceedings of the National Academy of Sciences, 95,* 861-868.

Keinan, G. (1987). Decision making under stress: Scanning of alternatives under controllable and uncontrollable threats. *Journal of Personality and Social Psychology, 52*, 639-644.

Lazarus, R. S., Deese, J., & Osler, S. F. (1952). The effects of psychological stress upon performance. *Psychological Bulletin, 49,* 293-317.

Mannan, M., & van Oorschot, P. C. (2007). Using a personal device to strengthen password authentication from an untrusted computer. *Financial Cryptography and Data Security,* 88-103.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limitations on our capacity for processing information. *The Psychological Review, 63,* 81-97.

Mosendz, P. (2015). Newsweek Twitter account hacked by group claiming ISIS affiliation. Retrieved February 13, 2015, from http://www.newsweek.com/newsweek-twitter-account-hacked-isis-affiliated-group-305897

Nevins, A. (2010). Two case studies in phonological universals: A view from artificial grammars. *Biolinguistics,* 218-233.

Newman, J. (2013). 123456: Millions of Adobe hack victims used horrible passwords. Retrieved March 5, 2015, from http://www.pcworld.com/article/2060825/123456-millions-of-adobe-hack-victims-used-horrible-passwords.html

Patterson, K., Nestor, P. J., & Rogers, T. T. (2007). Where do you know what you know? The representation of semantic knowledge in the human brain. *Nature Publishing Group, 8,* 976-988.

Reber, A. S. (1976). Implicit learning of synthetic languages: The role of instructional set. *Journal of Experimental Psychology, 2,* 88-94.

Reber, P. J. (2013). The neural basis of implicit learning and memory: A review of neuropsychological and neuroimaging research. *Neuropsychologica, 51,* 2026-2042.

Reber, P. J., & Squire, L. R. (1998). Encapsulation of implicit and explicit memory in sequence learning. *Journal of Cognitive Neuroscience, 10,* 248-263.

Renaud, K., & De Angeli, A. (2004). My password is here! An investigation into visuo-spatial authentication mechanisms. *Interacting with computers, 16,* 1017-1041.

Rogers, J. (2015). Centcom hack: Military tightens password security. Retrieved March 5, 2015, from http://www.foxnews.com/tech/2015/01/13/centcom-hack-military-tightens-password-security/

Rosenfeld, E. (2015). FBI investigating central command Twitter hack. Retrieved March 5, 2015, from http://www.cnbc.com/id/102330338

Rüsseler, J., & Rösler, F. (2000). Implicit and explicit learning of event sequences: evidence for distinct coding of perceptual and motor representations. *Acta Psychologica, 104,* 45-67.

Sanchez, D. J., Gobel, E. W., & Reber, P. J. (2010). Performing the unexplainable: Implicit task performance reveals individually reliable sequence learning without explicit knowledge. *Psychonomic Bulletin & Review, 17,* 790-796.

Savion-Lemieux, T., & Penhune, V. B. (2005). The effects of practice and delay on motor skill learning and retention. *Experimental Brain Research, 161,* 423–431.

Schendan, H. E., Searl, M. M., Melrose, R. J., & Stern, C. E. (2003). An fMRI study on the role of the medial temporal lobe in implicit and explicit sequence learning. *Neuro, 37,* 1013-1025.

Scoville, W. B., & Milner, B. (1957). Loss of recent memory after bilateral hippocampal lesions. *Journal of Neurology, Neurosurgery, and Psychiatry, 20,* 11.

Smart, N. (2003). Cryptography: An introduction. New York: McGraw-Hill.

Smith, E. E., & Kosslyn, S. M. (2007). *Cognitive psychology: Mind and brain.* Upper Saddle River, N.J: Pearson/Prentice Hall.

SplashData (n.d.). "123456" maintains the top spot on our annual “worst passwords" list. Retrieved June 15, 2015, from http://splashdata.com/splashid/worst-passwords/

Stampler, L. (2015). Delta Airlines’ Facebook page got hacked. Retrieved February 13, 2015, from http://time.com/3703640/delta-airlines-facebook-page-got-hacked/

Statista (2015). Retrieved May 26, 2015 from http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/

Squire, L. R., Knowlton, B., & Musen, G. (1993). The structure and organization of memory. *Annual Review of Psychology, 44,* 453-495.

Squire, L. R., & Zola-Morgan, S. (1991). The medial temporal lobe memory system. *Science, 253,* 1380-1386.

Thompson, S. T. C. (2006). Helping the hacker? Library information, security, and social engineering. *Information Technology and Libraries*, *25,* 222-225.

Tzvi, E., Münte, T. F., & Krämer, U. M. (2014). Delineating the cortico-striatal-cerebellar network in implicit motor sequence learning. *NeuroImage, 94,* 222-230.

Verwey, W. B., Lammens, R., & van Honk, J. (2002). On the role of the SMA in the discrete sequence production task: a TMS study. *Neuropsychologia, 40,* 1268–1276.

Walker, M. P., Brakefield, T., Morgan, A., Hobson, J. A., & Stickgold, R. (2002). Practice with sleep makes perfect: sleep-dependent motor skill learning. *Neuron, 35,* 205-211.

Weinshall, D. & Kirkpatrick, S. (2004). Passwords you'll never forget, but can't recall. *Proceedings of the CHI'04 Extended Abstracts on Human Factors in Computing Systems, New York,* 1399-1402.

Yoon, E. J., Shin, Y. N., Jeon, I. S., & Yoo, K. Y. (2010). Robust mutual authentication with a key agreement scheme for the session initiation protocol. *IETE Technical Review, 27,* 203-213.

Zetter, K. (2009). Weak password brings 'happiness' to Twitter hacker. Retrieved March 5, 2015, from http://www.wired.com/2009/01/professed-twitt/

Appendix A

Systematic representation of the SISL-based authentication protocol using a real password (A) and using a honey password denoted as hpw (B).