CONTEMPLATING SKILL-BASED AUTHENTICATION

Karen Renaud, Joe Maguire* and Johan van Niekerk† and Demetris Kennes‡

* *School of Computing Science, University of Glasgow, United Kingdom*
E-mail: karen.renaud,joseph.maguire@glasgow.ac.uk
† *School of ICT, Nelson Mandela Metropolitan University, South Africa*
E-mail: Johan.VanNiekerk@nmmu.ac.za
‡ *Enterprise Risk Services, Deloitte Limited, Limassol, Cyprus.*
E-mail: dkennes@deloitte.com

Abstract: Humans develop skills as they go through their lives: some are fairly common, such as reading, but others are developed to maximise employment opportunities. These skills develop over a long period of time and are much rarer. Here we consider whether we can exploit this reality in the security arena, specifically to achieve a stronger form of authentication. Authentication has traditionally been performed based on what users *know*, *hold* or *are*. The first is the most popular, in the form of the password. This is often referred to as "knowledge-based" authentication. Yet, rigorously following guidelines for password creation produces forgettable gibberish and nonsense strings, not knowledge. Nonsense is hard to remember and users engage in a number of coping strategies to ameliorate this, and these tend to weaken the authenticator. It would be beneficial to find a way of reducing this memorial load, to identify a more usable mechanism. This is hard: usually reducing the memorial load also makes the secret easier to guess. The challenge is in finding a way to reduce memory load while holding the line as far as strength is concerned. Here we contemplate exploiting recognition of artefacts resulting from experts practicing their craft: "skill-based" authentication. This should reduce the memorial load and effort, but also, crucially, make it harder for a random intruder to replicate. We report on how we trialled SNIPPET, a prototype of an authentication mechanism that relied on an expert programmer identifying his/her own code snippets from successive challenge sets. We found that our participants were all able to identify their own code snippets and that other participants were unable to guess these, even when they observed the legitimate person authenticating beforehand. These findings are not conclusive given the small number of participants but they do show promise and suggest that this is an area worth pursuing. We conclude by returning to the three NIST-identified forms of authentication and consider how SNIPPET can be positioned within the general authentication arena.

Key words: Authentication, Knowledge, Skills

1. INTRODUCTION

The PIN challenge issued by the ubiquitous ATM (Automatic Teller Machine) is a good example of an authentication mechanism encountered by the man and woman in the street in the course of their everyday lives. There is no report of complaints about having to remember the secret PIN when ATM machines were deployed in the 1960s. This is probably because in those days people only had to remember one or two PINs, not the multiple PINs and passwords they have to remember today.

As computers permeated all aspects of business life, the password was the obvious choice for restricing access, given the fact that the end-user had probably had experience of an ATM machine and could thus rely on a prior understanding of the concept. Fernado J. Corbató, the project leader behind one of the first systems to use passwords, Compatible Time Sharing System (CTSS) [1], explained that although passwords seemed theoretically strong, in practice many problems emerged. People routinely compromised security by choosing weak passwords [2], and by writing them down and sharing them [3]. A lot of this behaviour was driven by the fact that they had too many passwords [4, 5], and because they had previously forgotten passwords and had no desire to repeat the experience. Blaming the users is the natural response, and the obvious next step is to try to persuade or coerce them into abandoning these behaviours. This, while intuitively the right course of action, is bound to fail, since it does not eliminate the cause of the behavioural effect: users don't want the inconvenience of a forgotten password (Figure 1). If we can reduce the prevalence of the cause, the resulting uncceptable behaviours might be less likely to occur.

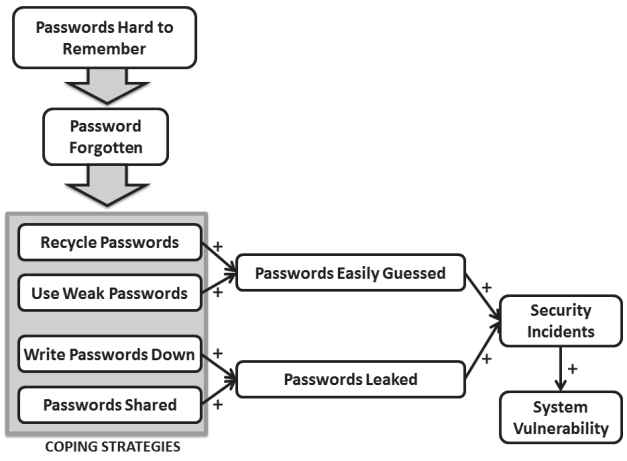

Figure 1: Coping Behaviours and Antecedents

Based on "SNIPPET: Genuine knowledge-based authentication", by K. Renaud, J. Maguire, J. van Niekerk and D. Kennes which appeared in the Proceedings of Information Security South African (ISSA) 2013, Johannesburg, 14 to 16 August 2013.

Perhaps in response to the wide-spread issues related to traditional secret-based mechanisms, Apple recently released an iPhone with a fingerprint sensor that essentially introduced biometrics into the mainstream consumer market. The device sold an estimated 9 million in its first few days after release [6]. Pankati [7] predicted at the turn of the century that biometric-based authentication was the future. He argued that since tokens were easily misplaced and it was easy to forget passwords, the only future direction for authentication was the dependable and indisputable biometric [7]. It is interesting that Apple appears to have come to the same conclusion, albeit 13 years later.

There are naturally concerns about the use of such an authentication mechanism. The approach is easily fooled by fake fingers [8]. Moreover, it appears to dissuade device sharing which is something many phone owners want to be able to do [9]. It is interesting that Apple has decided to include such a relatively novel authentication mechanism in their mainstream products. The convenience of access control for the device owner is probably considered a selling point although the recent revelations by Edward Snowden [10] might well give iPhone owners pause with respect to the potential destination of their fingerprint template [11, 12]. There is, however, a certain clarity in the choice of this authentication solution. In theory, the mechanism relies on both the owner and device being co-present and one can readily see the attraction and simplicity of such a guarantee as far as security is concerned.

Despite Apple's recent innovation, however, the reality is that biometric-based authentication remains relatively novel and passwords not only persist, they reign supreme, as the *de facto* authentication approach across the globe. In effect, passwords have become the default authentication solution for almost every context and user. This brings us back to the apparently intractable problem related to passwords: the tension between strength and memorability. Here we offer a way of ameliorating this problem.

The rest of the paper is structured as follows. Section 2. explores the concept of "What you Know" authentication. Section 3. explores the idea of a genuine knowledge-based authentication mechanism, leading to the concept of "skill-based" authentication. Section 4. reports on a survey of programmers to determine whether they thought they would be able to identify their own and others' programming code. The survey results suggested that empirical verification would be beneficial. Section 5. reports on a pilot study we carried out to test a "skill-based" authentication mechanism. Section 6. reconsiders authentication in general and positions our mechanism, SNIPPET, within the authentication arena. Section 7. concludes.

2. "WHAT YOU KNOW" AUTHENTICATION

"What you know" authentication is the process of confirming a claimed identity through knowledge of a secret, one known only to you and the authenticating party. Since it is a secret, individuals are advised to memorise it and not to record or share it. The secret itself could be a public event or record, but the use thereof must not be revealed.

The alphanumeric password is the best known implementation of "what you know" authentication. There are two reasons for this:

1. the concept of passwords is one which is centuries old and is easily understood by both users and developers; and
2. the interaction mechanism, i.e. the keyboard, is over a century old and one can easily enter passwords without additional training or expense.

This made passwords the authentication mechanism of choice for early systems, such as CTSS [1], and operating system designers such as Ken Thompson and Dennis Ritchie.

The problems with passwords emerged soon after their initial deployment. They immediately proved difficult to use and remember [13]. The situation has barely improved as technology has advanced. If anything, as the world becomes increasingly connected, the ubiquitous use of passwords becomes even more problematical. News stories detailing the problems caused by the improper use of passwords are not a rare occurrence. The Federal Trade Commission, for example, has recently taken legal action against Wyndham Hotels after the organisation failed to properly protect the financial information of 500,000 customers, resulting in damages of $10.6 million [14]. The organisation generated weak and simple passwords that were compromised by attackers and allowed them to install software to capture information.

The use of simple passwords is not particularly surprising as users will create simple passwords to avoid the inconvenience of not be able to complete a task, since they have probably forgotten a password previously and do not want to repeat the experience [15]. The following excerpt, extracted from a complaint submitted by the Federal Trade Commission, offers evidence of the use of simple passwords in the aforementioned case, as follows:

> "For example, to allow remote access to a hotel's property management system, which was developed by software developer Micros Systems, Inc,. Defendants used the phrase "micros" as both the user ID and password"
> **Federal Trade Commission Compliant [14, p. 11]**

The use of such simple strings for the convenience of a few individuals led to dramatic inconvenience for 500,000 paying guests. A great deal of expense, in terms of time and money, was spent rectifying the problems caused by this irresponsible authorisation mechanism.

However, passwords that are difficult to remember also incur costs for organisations. The estimated cost of password bureaucracy, such as replacement and recovery, is an estimated $17 per call [16]. Moreover, an estimated 30% of call volumes are associated with passwords [16]. Consequently, not only is there a cost associated with each call, there are also a considerable number of calls to cope with.

Despite these problems, the vast majority of authentication in 2013 falls into the "what you know" category. This is often termed *knowledge-based authentication*, which seems intuitively correct. This seems to be based on the assumption that there is a natural mapping, allowing one to substitute "what you know" with the word "knowledge". Actually we are going to argue that this is misguided, that the terms are not as interchangeable as they seem. In fact "what you know" may, over time, progress into knowledge, depending on its nature, but such a progression is by no means guaranteed. To support this argument we need to examine the distinction between data, information and knowledge (Figure 2).

- *Data:* Data is simply data: no use to anyone until someone provides the context. So, for example, consider the number: 2.5, a simple piece of data. There is no way of knowing what that number refers to.
- *Information:* If we add context and explain that this is the number used to convert a measurement from inches to centimetres, the data has become information, because it now has meaning. It is not yet knowledge, however.
- *Knowledge:* Knowledge is defined by the Oxford dictionary as: "*the theoretical or practical understanding of a subject*". In other words, knowledge implies an understanding of how to use the information to solve some problem. If one is given the dimensions of a room in inches and asked to calculate the area of the room in cm^2, the information just provided would be applied in order to solve the problem. The person would also have to know how to work out area using the width and breadth and know how to multiply the dimensions by the conversion value to arrive at the correct result. This implies an understanding of how to use the information, and success suggests that you do indeed possess that knowledge.

Knowledge and skills take time to develop, and this process cannot be short-circuited [17]. The benefit is that knowledge and skills are not easily disrupted. The nature

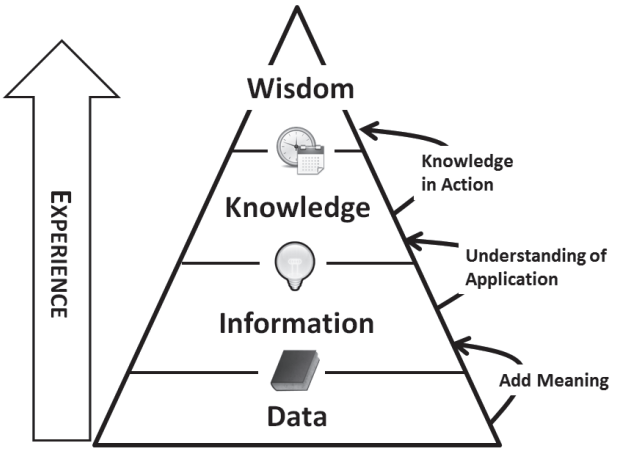

Figure 2: Data, Information, Knowledge, Wisdom (DIKW) Pyramid

of the knowledge and skill acquisition process seems to make a durable footprint on the user's mind that does not easily decay, even with age, especially when learnt before retirement [18]. Moreover, retrieving the knowledge requires less effort than recalling a nonsense data string effortfully memorised and possibly forgotten. Nonsense is forgotten because the brain is economical and performs neural pruning on networks that are not deemed essential [19]. The more interesting and stimulating something is, the more easily it will be remembered. Nonsense is neither stimulating nor interesting, and is deliberately pruned.

It is also of interest to note that the above mentioned "levels" as one progresses from data to knowledge also, to a certain extent, map to the first three levels of Bloom's well-known taxonomy of the cognitive domain [20]. The following lists the first three levels as presented by [20], and briefly shows how these levels relate to the distinction between data, information, and knowledge.

- *Remember:* This is the lowest level of cognition. Remember is the ability to *retrieve* relevant facts from memory but does not include the ability to relate the retrieved facts to a specific context.
- *Understand:* If we add context to remembered data a person has the ability to understand the data, "construct the meaning of instructional messages" [20, pp. 30], but does not necessarily have the ability to apply it correctly.
- *Apply:* The third level of the cognitive domain is being able use the information correctly in a given situation or context. This level of cognition thus clearly requires the person to have *knowledge*, as defined above.

Now consider authentication. Here is some advice given by CERT [21] for choosing a password:

- Don't use passwords that are based on personal information that can be easily accessed or guessed.
- Don't use words that can be found in any dictionary of any language.
- Develop a mnemonic for remembering complex passwords.
- Use both lowercase and capital letters.
- Use a combination of letters, numbers, and special characters.
- Use passphrases when you can.
- Use different passwords on different systems.

A password chosen according to these guidelines is more akin to data than it is to knowledge. If a password has meaning, it has become information. If it is information then attacks become easier to carry out. Users use information instead of data as passwords so that the password will not be forgotten, Such an information-based password has meaning, usually something related to the user him or herself. This action potentially weakens the password since an attacker who knows the user will be more likely to be able to guess it. Figure 3 shows how the drive for strong passwords conflicts with users' motivation to choose memorable and meaningful passwords.

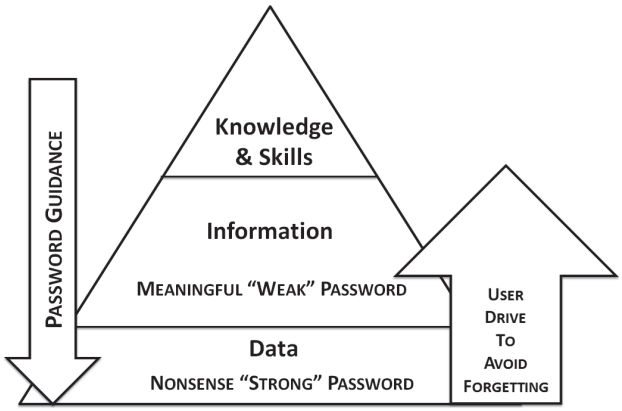

Figure 3: Passwords Positioned within the DIKW Pyramid

Thus a more realistic moniker for current recommended usage of "what you know" authentication would be "nonsense-based" authentication. This begs the question: what would actual knowledge-based authentication actually look like? Some pertinent aspects immediately become evident and will be referred to here as the *constraints* of genuine knowledge-based authentication, what we will call *skill-based authentication* (Figure 4).

C_1 *Appropriate Elicitation:* We have to test someone's skills or understanding of an area, which is much harder than asking them to produce an alphanumeric string. Moreover, testing this kind of knowledge requires provision of context, since knowledge is always applied within a particular context. Such context should not constitute a cue to any would-be intruder.

C_2 *Soundness:* It should not be possible for another expert in the area to authenticate: we have to ensure that the mechanism authenticates only the legitimate expert user [22].

C_3 *Cost-Benefit Balance:* It must be possible for a user to demonstrate this knowledge quickly and easily, so that authentication does not become too time-consuming or inconvenient [23].

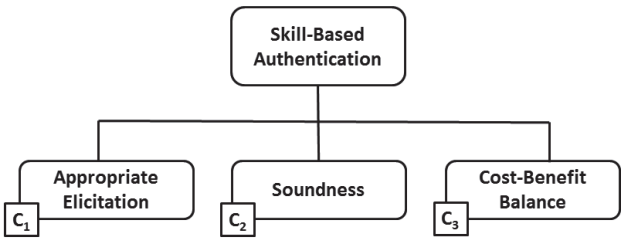

Figure 4: Constraints of S-Based Authentication

There are clear challenges inherent in testing genuine knowledge that meets these constraints. This kind of authentication is a relatively unexplored category, which is understandable given these constraints. The following section explores the issue of testing knowledge and skills in an authentication setting.

3. MOVING UP THE PYRAMID

Generally, one can test "what you know" in one of three ways: *recall, cued-recall* or *recognition* [24]. All of these require some memorial effort with effort decreasing from recall, to cued-recall to recognition. Testing recall-based memory offers the recaller no assistance: they are required to remember the item unainded. This becomes increasingly difficult as users age [25]. Moreover, since knowledge is applied in context, pure recall-based testing is unsuitable since it does not satisfy the first constraint.

Cued-recall mechanisms provide cues to help the user to recall the authentication secret. The provision of cues in this setting, while essential in testing knowledge, is problematical since cues have to assist the legitimate user but not any random intruder who happens to be skilled in the same area.

An example of the use of cues in authentication is the Cueblot mechanism [26] which displays an inkblot-like image to trigger the user's memory when they have to authenticate. Since the cueblot is sufficiently abstract it does not act as a cue for other users, but only for the legitimate user. What this paper confirms is the difficulty of providing a legitimate user with a cue that will not make sense to another user. The cueblot cue does not really test expert knowledge, however, so this particular

technique will not be useful in implementing genuine knowledge-based authentication.

Another example of a cued-recall mechanism is Zviran's associative passwords which probe a user's personal experience [27]. This quiz-based approach extracts several pieces of knowledge from the user at enrolment. The individual is presented a series of *fact-based* and *opinion-based* questions. A fact-based question would be 'What was the first school you attended?', while an opinion-based question would be 'What is your favourite film?'. The problem with this mechanism is that it is too time-consuming at authentication, thus not satisfying the third constraint: adequate balance of cost and benefit. However, this example exploits an aspect of skilled practice that will be very useful to this research: the *experience* of the user. We might be able to exploit this to meet constraint number two: distinguishing different experts from each other, since every expert has different life experiences.

Cued-recall authentication provides the essential context the skilled user needs to demonstrate possession of skills but it does so in a way that makes authentication time-consuming, and, as such, is probably infeasible. We will thus explore the last remaining possibility: relying on recognition.

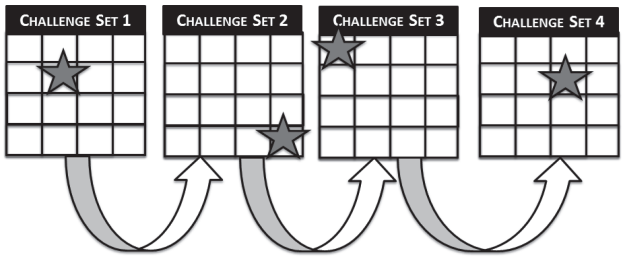

Figure 5: Authentication - Four Challenge Sets

Recognition-based mechanisms most often display grids of images and require the user to click on their own image from the challenge set (Figure 5). A number of these have been proposed [28–31] asking users to identify faces, representational or abstract images from challenge sets. Recognising is easiest for users, since all they have to do is click on their own secret image in order to authenticate: it is cognitively the least demanding mechanism. It meets the first constraint since it provides context. It also comes closer to meeting the third constraint since it takes less time than a cued-recall mechanism.

The second constraint is harder to meet. Most recognition-based authentication mechanisms do not personalise the images used by the mechanism, using the same images for the entire user population. Unfortunately, when users are allowed to choose from a common dictionary their choices are predictable [28, 32]. Perhaps they are still trying to find meaning in their secrets so as to prevent the secret from being forgotten.

How can we ensure that only the legitimate user can recognise and identify the correct image in the challenge set? Here we deploy the concept that Zviran [27] highlighted: the experience of the user. Experts often produce artefacts as they practice their skills. If we test recognition of these artifacts, rather than mere expert knowledge, we ensure that the user possesses both the skills and the actual experience. They should be able to remember that they engaged in a practice that produced the artifact. It must be admitted that not all skills leave artefacts: medical doctors, for example, do not necessarily produce artefacts. Other professions, though, do: examples include programmers and artisans such as carpenters and builders.

The second constraint, soundness, can be split into further sub-categories. Renaud and De Angeli [33] argue that the security (soundness) of an authentication mechanism means that it will be *unpredictable, abundant* and *undisclosed*. The first two seem to be focused primarily on the strength that comes from the size of the dictionary a secret authenticator is drawn from, which refers to the unpredictability of the mechanism. The third appears to be more related to the obscurity of the mechanism than the dictionary size: the need to keep knowledge of the secret from others. Moreover, this particular list of requirements does not include the need for the knowledge to be easily memorable, which undeniably contributes towards its soundness as an authenticator. Hence soundness must incorporate the following (Figure 6):

C_2(a) *Undisclosed:* The *non-availability* of the authenticator can be assured in two ways. The first is secrecy, ensuring the imposter does not gain knowledge of the authenticator. The second is security, keeping the autheticator out of the reach of would-be imposters even though it may not be secret. If the secrecy technique is used the authenticator does not need to be unique but if the authenticator is secured by keeping it out of reach then it has to be unique or at least arguably unique.

C_2(b) *Unpredictability* has two aspects:

C_2(b) i *Dictionary Size:* It should not be easily possible to attribute the artefact to the creator or at least to narrow down the possible identity of the artefact based on knowledge of the user. The size of the dictionary is only relevant when a potential intruder cannot predict which element someone will choose. Hence the selection process must be unpredictable, but, having made that choice, it should be impossible for someone easily to guess it. The artifact should not be in the public domain if a recognition-based mechanism is going to be used. So, for example, one could not make use of a famous artist's paintings to allow the artist herself to authenticate. Other examples of easy attribution exist. For example Argamon [34] shows that it is possible to determine the gender of a writer

from their written text. Estival et al. [35] show how analysis of an email can tell you even more about the author. This means that a paragraph written by a skilled writer would be unsuitable for use in authentication.

C_2(b) ii *Abundance:* It must be possible to find viable distractor images. For example, if we make use of handwritten mathematical proofs to authenticate mathematicians, we would have to display the user's proof, and then as distractors in the challenge set a number of proofs written by other mathematicians. We would expect the expert to identify their own proof, in their own handwriting.

C_2(c) *Recognisability (Memorability):* It must be possible for users easily to recognise their own artefact. Since they have created the artefact themselves, this should help them to recognise it [36].

C_2(d) *Matching Process Capability:* Does the matching process deliver a definitive answer, or does it deliver a confidence level? The former constitutes more soundness than the latter.

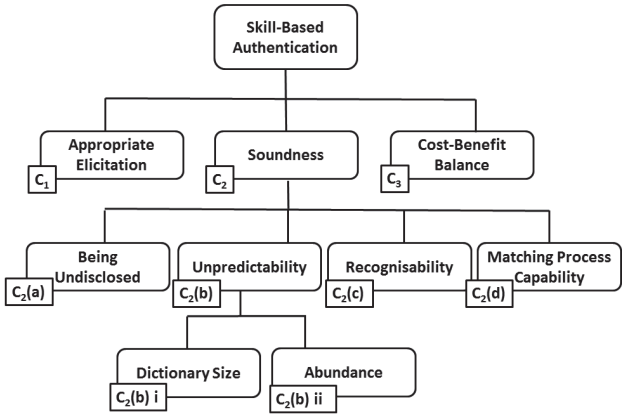

Figure 6: Extending Constraints of Skill-Based Authentication from Figure 4

We will attempt to meet these constraints by *personalising* the authentication secret. Users already do this intuitively when they choose passwords that are related to themselves i.e. information rather than data. Here we propose to advance another level up the pyramid (Fig. 3).

3.1 Personalising Authentication Secrets

Humans can recognise a lot of things about themselves. For example their own voices [37], their own handwriting [38, 39], their own performance (pianists) [40] odour [41] and gait [42]. Hence images that are related to the user should make them easy to recognise but it might well also make them easier to guess. There are other ways of maximising recognition success. For example, a graphical mechanism using facial images could be tailored to maximise recognition by tailoring the entire challenge set to the age [43], race [44] and gender [45] of the user. This would help the user but not make things easier for an attacker. All these variations would personalise the images to maximise the legitimate user's chances of being able to remember and identify their images.

Some authentication schemes have attempted to make use of personalised images. Dynahand [46] relies on the user being able to recognise his or her own handwriting (Fig. 7). It collects 10 examples of participants' handwritten numerals at enrolment. It then generates random PINs using the user's own handwritten numerals, and generates distractors from other users' handwritten numerals. Four challenge sets are displayed, and each time the user picks out the displayed PIN written in his or her own handwriting. A casual observer has less chance of gaining access to the user's account later because what is being tested, i.e. handwriting recognition, is relatively obscure and less easily cracked than a straightforward set of pictures. Moreover, it is completely effortless for the user.

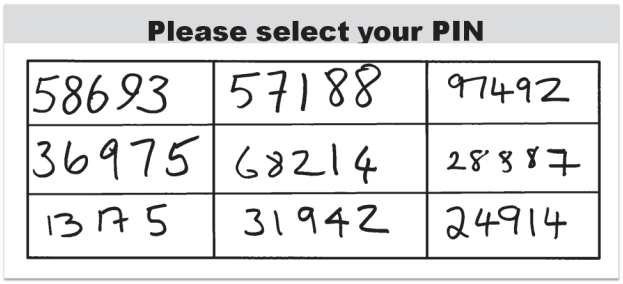

Figure 7: A Dynahand Challenge Set

Renaud [30] deployed this technique as one stage of the Handwing authentication mechanism to control access to a website used by a community group where the community members very successfully identify their own handwriting to authenticate. The mechanism also exploits the user's ability to recognise their own hand-drawn doodle and has been very successful — and is still being used 10 years later. Renaud [47] also tested the same concepts with a graphical authentication mechanism that used Mikon (my icons) images (Fig. 8). Users drew these using a browser-based engine. The majority of the participants in the study were able to remember all their Mikons successfully after a three month period of non-use. These examples serve to show that personalised images are recognisable but we don't yet know how predictable they will be.

3.2 Personalising Secrets for Experts

The schemes mentioned thus far did not exploit a particularly stringent or rare skill: almost everyone can write and draw. They do, however, demonstrate that people have the potential to remember, and to recognise artefacts resulting from their skilled practice. In the case of the drawn images, the images are more memorable than passwords because they rely on visual, lexical and kinaesthetic memory [48] rather than mere textual memory.

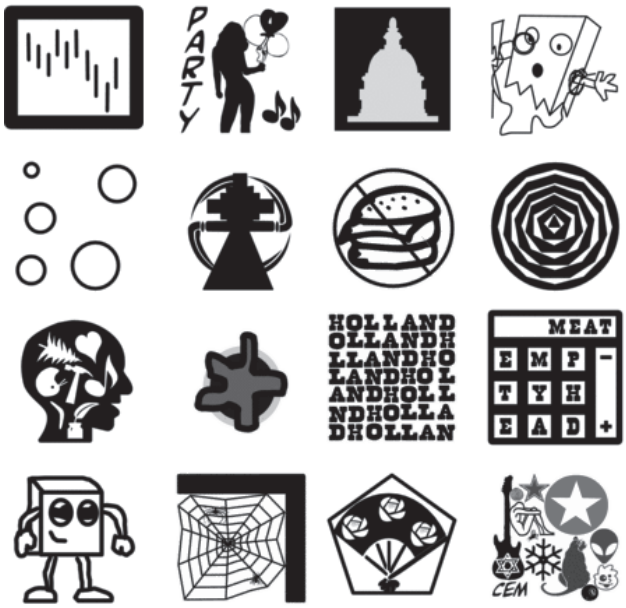

Figure 8: A Mikon Challenge Set

We propose to extend this concept to test whether experts can recognise the outputs from their own skilled actions, in this case programming language code. It takes thousands of hours to become a competent programmer [49]. Although there are millions of programmers in the world, the number is significantly smaller than those who can write and draw.

Let us consider programming code snippets in terms of the constraints introduced earlier in this section.

1. *Undisclosed:* Programming code is often not in the public domain — it is essentially hidden from view. Open source code is the obvious exception but it is not clear that a programmer's particular style would be recognised by anyone else.
2. *Unpredictability:* Requires empirical testing.
3. *Abundance:* Finding viable distractor images is trivial if the snippet is of a widely used programming language. It is also entirely possible to automate the generation of such distractors, which would make abundance a non-issue.
4. *Recognisability:* It should be possible for programmers to recognise their own code. Craik and Tulving [50] argue that the development of memory traces should be considered in terms of *depth of processing*. Programming is a cognitively demanding task and so the production of an artifact should lay down strong memory traces. The advantage is that using snippets of code would not require the user deliberately to memorise anything. This addresses the primary root cause of insecure password behaviours.
5. *Matching Process Capability:* We can perform an exact match at authentication.

From the above list, we see that *unpredictability* and *recognisability* need to be verified. Before we proceeded to testing these aspects though, we wanted to find out from skilled programmers whether they thought this scheme had any merit.

4. FACT FINDING

In order to determine whether this idea had any chance of succeeding, we started off by posting an online survey. We advertised it via developer forums and to our respective institutions' postgraduate students. 198 programmers responded to our survey. The majority (179) had been programming for more than 3 years with the largest group (60) in the 5-10 year category.

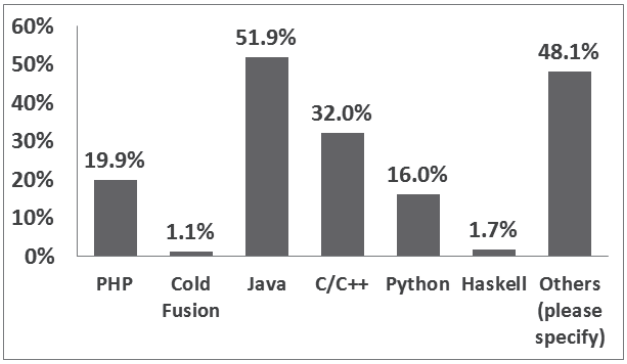

Figure 9: Which Programming Language did They Use

Figure 9 shows the distribution of programming languages used by the respondents. Some people mentioned C#, ASP, Javascript, PL/1, Perl and Assembler. The most commonly used language was Java. We provided a box for comments.

80% of the respondents agreed with the statement: "Every Programmer has his/her own programming style". Figure 10 presents the responses. This appears to confirm the findings that people develop personal styles [51]. Some comments from the respondents:

> *"programmers I knew all looked to add their own personalisation - it is their baby"*

> *"It's a mistake if a programmer doesn't have his/her own programming style as it is important to recognizing your own programs"*

> *"Programming is an expression of thoughts much like poetry. So a programmers individual style will be reflected in the piece of code that he/she develops. Bottom line is there can be several alternative solutions for a single problem, and different programmer may adopt different style. "*

> *"Yes, it's like writing where every author has his own writing style as well. "*

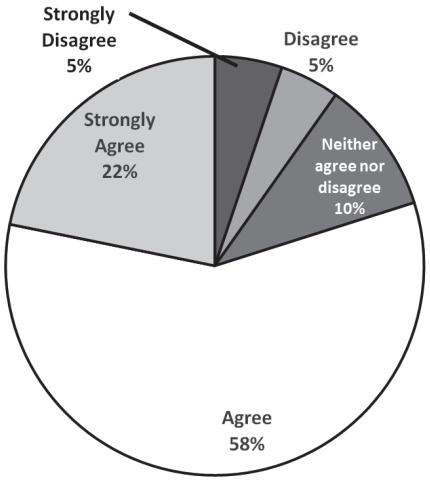

Figure 10: Every Programmer has his/her own programming style

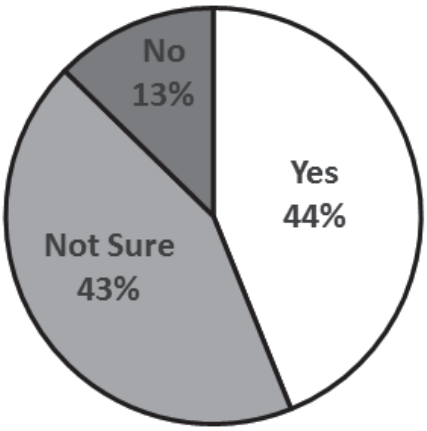

Figure 11: Could you Identify your own code from a group of code snippets 10 lines long?

44% felt sure they could identify their own code, with another 43% being unsure (Figure 11). One said: *"My own typing style is distinct (the whitespaces, the way I comment, variable naming, etc). I'm sure I can identify snippets of my own code among others"*. Only 20% felt they would be able to identify another programmer's code even if they knew the person well (Figure 12).

The survey responses convinced us that it would be beneficial to trial a scheme which tested whether (1) people could recognise their own code (recognisability) and (2) people could recognise each others' code (unpredictability).

5. EXPERIMENT

Our survey of programmers made it clear that while many of them felt they had a particular programming style, only 44% felt they would be able to identify their own code. Our findings had suggested that skill-based authentication demonstrated some promise, but it was clearly necessary to verify these soundness aspects empirically.

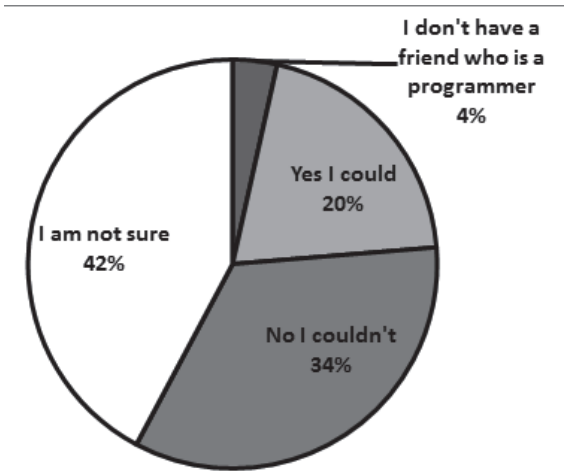

Figure 12: Could you identify a friend's code from a group of code snippets 10 lines long?

We carried out a proof of concept experiment into the use of "skill-based" authentication. The area of expertise we focused on was programming, since we possessed this skill ourselves and we worked in an environment that gave us access to a number of expert programmers. The aim was to design an authentication mechanism which would authenticate programmers based on their own programming skills, a genuine knowledge-based test. A recognition-based graphical authentication system which used snippets of code, instead of images, was implemented. We hoped to show that programmers would be able to recognise their own code, but that others, even those who are experts in the same language, would not easily be able to recognise the person's code snippet.

Our participants were 20 programmers, Masters students who had been together in the class for some 9 months and so knew each other fairly well. We asked them to provide five snippets of code in Java, since this was the most widely used language mentioned in our questionnaire. They were asked to avoid snippets containing comments. This constituted enrolment.

We then asked them to return a week later to see whether they could identify their own code from four challenge sets. Participants were required to identify their own code snippet from four challenge sets, each composed of 16 code snippets. Distractors, and targets, varied each time the user tried to authenticate since we had more code snippets than we needed for one authentication attempt. An example challenge set is shown in Figure 14.

To verify the two aspects identified as needing verification in Section 3, we tested recognisability (memorability) and predictability of the code snippets. Participants worked in parallel. For example, Participant A would authenticate while participant B watched. Then Participant B tried to replicate the attempt. Participant C, on the other hand, attempted to guess Participant A's code without

observing A authenticating. Hence every participant observed another authenticating and tried to replicate the attempt. They also tried to guess one other person's codes without observing them authenticating.

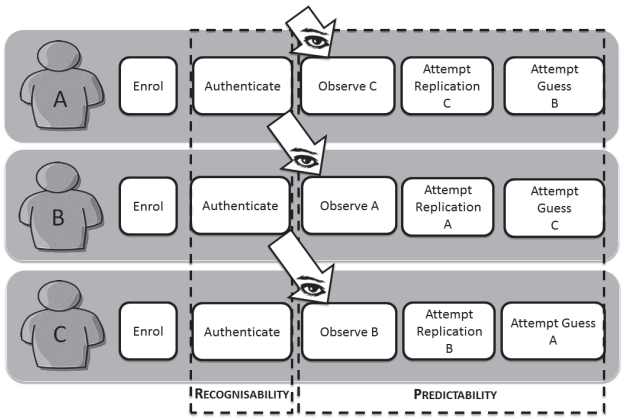

Figure 13: Participants Working in Pairs

5.1 Results

Recognisability: Identifying Their Own Code All participants were able to identify their code, some almost immediately, but some needing some time to examine the snippets in the challenge set. We did not record timings since these would not have supported analysis with so few participants. We asked the participants what particular aspect of the code made it memorable. Some of them stated that they identified their variables, others functionality or Java class names. One participant identified his secret sequence of images in less than a minute as the variables were expressed in his national language, whereas the others were in English.

Predictability: Guessing Another's Code None of the "attackers" managed to identify another's code images, both when they observed the authentication and when they just tried to guess it. This is probably due to the fact that the images and the distracters are varied so the attacker would need to identify the programmer's *style* and not one specific piece of code.

```
while (ss.moreStudents()) {
Student stud = ss.getNextStudent();
if (stud == null) break;
if (stud.degree.equals("PhD")) {
Staff.recordSupervision(stud.first,
        FIRST_SUP);
Staff.recordSupervision(stud.second,
        SECOND_SUP);
}
```

```
public class DB extends RSC {
String connString="jdbc:mysql://";
String username="";
String password="";
Connection conn=null;
String driverName=
    "org.gjt.mm.mysql.Driver";
boolean connError=false;
```

```
try{ Statement stmt =
conn.createStatement();
ResultSet rsRota =
stmt.executeQuery(query);
while (rsRota.next()) {
    count = rsRota.getInt("cnt");} }
 catch (Exception ee) {
System.out.println("Prob getting
count " + ee.getMessage());}
return count;
```

```
public String name;
    public String initials="";
    public String email;
    public int id;
    public boolean external = false;
    public int tally1st=0;
    public int tally2nd=0;
    public int numoffers=0;
```

```
if (isExtensionRequest) {
 int storeid =
storeExtensionRequest(studentNa
me,studentid,reason,courseworkNu
mber,courseid,getLecturerId(course
id));
sendEmailToStudent(from,student
Name,courseworkNumber,courseN
ame,storeid) ;
```

```
p2.setForeground(Color.black);
p2.setBackground(Color.white);
p2.setLayout(new BorderLayout());
JPanel p2small = new JPanel();
greenquestion = new
JLabel(questionicon);
        p2small.add(greenquestion)
;
```

```
JCheckBox sounds1 = new
JCheckBox("Play Lightning Sound");
JCheckBox sounds2 = new
JCheckBox("Play Star Sound");
JCheckBox sounds3 = new
JCheckBox("Play Absent Sound");
JCheckBox showStar = new
JCheckBox("Show Gold Star");
JCheckBox showFish = new
JCheckBox("Show Absence Button");
```

```
public class csvEntry {

String coursename;
String aenumber;
String matric;
String band;
String dayslate="0";
boolean instr=false;
```

```
String [] cmd = {"/bin/bash","-
c","sudo /usr/bin/gammu
deletesms 1 " +num};
java.lang.Process p =
Runtime.getRuntime().exec(cmd);
        p.waitFor();
```

```
try {
System.out.println("Sleeping for 5
minutes ...." + getTime());
synchronized(waitObject) {
        waitObject.wait(delay);
}}
 catch (Exception ee) {// do nowt
   ee.printStackTrace();
  System.exit(0);};
```

```
int posit2 =
s.substring(pos).indexOf("Status");
String fromnumber =
s.substring(pos+posit,pos+posit2);
fromnumber=fromnumber.replace(
"\"","");
fromnumber=fromnumber.replace(
"+","");
```

```
public static boolean bad(File f) {
for (int i=0; i<badfiles.size(); i++) {
if (badfiles.elementAt(i).
getAbsolutePath().equals
(f.getAbsolutePath())) return true;}
return false;
}
```

```
public static void main(String[] args)
{ Scanner userInputScanner = new
Scanner(System.in);
System.out.print("What's your
name? "); String userInputName =
userInputScanner.nextLine();
System.out.println("Hello " +
userInputName + "!");}
```

```
public boolean
monkeyTrouble(boolean aSmile,
boolean bSmile) {
if (aSmile && bSmile) { return true;
} if (!aSmile && !bSmile) { return
true; } return false;
```

```
public void paint(Graphics g) {
g.drawString("Click, drag, and type
in this window. ", 10, 20); } //
Handle mouse events public
boolean mouseDown(Event e, int x,
int y) {
showStatus(modifier_key_names(e.
modifiers) + "Mouse Down: [" + x +
"," + y + "]"); return true; }
```

```
public int bunnyEars(int bunnies) {
// Base case: if bunnies==0, just
return 0.  if (bunnies == 0) return 0;
// Recursive case: otherwise, make
a recursive call with bunnies-1 //
(towards the base case), and fix up
what it returns. return 2 +
bunnyEars(bunnies-1); }
```

Figure 14: An Example Challenge Set

Participant Comments We asked participants to express their opinions about the mechanism when the experiment concluded. All reported finding it easy to locate their own code snippets. 17 of the 20 believed it would be impossible for anyone else to identify their images. Some specific comments:

> *"The idea of having code images as passwords is unique and I believe holds a good future"*

> *"First time I used this mechanism was a bit difficult but gradually it became easy for me. Moreover I believe it is easier to remember images than text-based passwords."*

6. DISCUSSION

To end off this paper we return to the issue of authentication in general in order to position SNIPPET within the arena. To position authentication in terms of access control, consider that a person who wishes to access restricted information or services has to prove that they have the right to do so. This is a two-step process: *identification* followed by *authentication*, proof that the person claiming the identity does indeed own it.

The identifier needs to be unique but does not have to be secret. The most often used identifier is a username or email address, neither of which is necessarily secret.

NIST published a guideline for authentication in 1977, which argued that authenticators could fall into one of three categories: what you *know*, what you *are* and what you *hold* [52]. This model is simple and easy to understand but, in 2013, probably fails to capture the nuances of a rapidly changing authentication and identification arena. It is time to pose two pertinent questions:

1. Are the NIST categories still all-encompassing?
2. Do instances of the "big-three" authentication categories meet the soundness constraints?

6.1 Are the NIST categories still all-encompassing?

A number of new mechanisms have been proposed in the intervening years since NIST published their categories. Here we will provide a few examples, and show how/whether they fit into one of the already-proposed categories. This list is not exhaustive, but does provide a flavour of the research activity in the interim.

- NIST Categories

  - *What you know*: This field has moved on from the humble password. Later developments require a user to draw a picture [53–56]. The latest incarnation of this kind of mechanism is the sketch-based mechanism on the Android [57]. Others require users to remember positions within an image [33]. There has also been a great deal of work relying on people's memory of images [31, 58] or faces [59, 60] rather than an alphanumeric string.
  - *What you hold*: Traditional card ownership is moving to mobile phone ownership. Al Fairuz and Renaud [61] utilise the mobile phone channel to deliver a one-time password to authenticate transactions. Other examples are wearable keys [62] and RFID tags [63]. A relatively new addition to this category is the *embedded chip*. These can be used to gain access to controlled areas such as homes and offices, and grant access to electronic devices such as mobile phones. These chips have proved to be an emotive issue with privacy and bodily integrity concerns [64].
  - *What you are*: in this area much work has been done in the intervening years. Fingerprint readers have started to appear in products such as laptops (eg. IBM Thinkpad) and mobile phones (eg. iPhone 5S). There is also a growing body of research focusing on behavioural biometrics: authenticating people according to the way they use their device [65, 66].

- New Categories
  - *What skill you can demonstrate*: An example of this is the work by Tao and Adams, who propose an authentication mechanism inspired by the ancient Chinese board-game, Go [67], which relies on the user knowing how to play the Go game.
  - *Who you know*: Brainard [68] proposed a new kind of authentication, based on some*one* you know rather than some*thing* you know. This adds a social aspect to authentication, which has traditionally been a solo exercise.
  - *What you associate*: Smith proposes the use of word association to authenticate [27, 69]. This is fairly unique because every human reasons in a slightly different way. However, it is very time consuming.

Hence the original three categories have been augmented in the interim but it must be noted that these additional kinds of authentication have not been embraced by industry, probably being seen as novel and, as yet, unproven. Moreover, there is contention about whether some of the mechanisms mentioned above are indeed authenticators or actually identifiers. The following section will consider the second question above for the three traditional authentication categories.

6.2 Authentication Category Soundness

Non-disclosure, by means of secrecy, seems an obvious requirement when it comes to "what you know" mechanisms: users know that they ought to withold their passwords from others. If the secret is remembered and retained the legitimate user will always gain access, and imposters will be resisted. Unfortunately the newspapers abound with stories that prove that passwords are often not retained. There is a suggestion that humans find it difficult to keep secrets [70] and that revealing secrets is cathartic [71]. It must be acknowledged that in the secret-based academic literature the kind of secrets being referred to are those that people tend to be ashamed of, so these findings might not apply to keeping passwords secret. Still, there is a social element to password sharing that suggests that there is more to divulging password secrets than mere carelessness [72–74]. Hence increased availability compromises the non-disclosure of the mechanism.

For tokens, availability is ensured by keeping the token secure, i.e. close at hand. They can, unfortunately, be lost or stolen quite easily. Tokens are thus usually paired with knowledge or a biometric so that they can serve as authenticators. Given that the soundness of the token is so easily compromised, and the fact that they require a second factor in order to support authentication, we should perhaps refer to tokens as *private identifiers*. They are more secure than self-proferred usernames because their availability is somewhat restricted. Yet on their own they do not reliably authenticate the card holder, so they cannot realistically qualify as authenticators.

The third NIST category is the biometric. The most popular of these is the fingerprint, perhaps unsurprisingly since it is has the most established use in other contexts, such as law enforcement [75], and readers are relatively inexpensive. Much has been written recently about the use of fingerprints to protect mobile devices, and this has been made a major selling feature of the new iPhone 5S, but unfortunately it is the case that these digital fingerprint readers are not infallable [76]. The Chaos Computer Club spoofed the iPhone fingerprint biometric within a week of it being released, merely by copying a person's fingerprint onto a piece of paper. [77]

This highlights one of the biggest problems with biometrics: the fact that they are not secret. Many countries collect them when people travel there, users leave them all over their homes, desks, wherever they go. Having obtained the fingerprint, there are some who know how to create a fake finger which can fool a biometric reader [8]. This means that possession of the biometric does not automatically authenticate the user: there is a chance that the person presenting the biometric is an imposter. Even if the legitimate user is presenting the biometric it sometimes fails to authenticate the user since the matching process is not an exact science.

This leads us to the second requirement: *soundness*. A password challenge leads to a binary decision: match or no match. There are no grey areas inbetween. With a biometric, on the other hand, there is a matching process that leads to a confidence level: the biometric reader is seldom if ever going to deliver a 100% match between the stored template and the currently presented biometric. Soundness depends on a number of factors, ranging from the quality of the reader to slight changes in the biometric that happen quite naturally, perhaps as users age. Hence a ruling made when a biometric is presented is more in the nature of "eliminating reasonable doubt" rather than being able to rule definitively in one direction or another.

Given that they fail the soundness constraint, perhaps biometrics, too, should be referred to as *private identifiers*, once again stronger than a user name, but perhaps not entirely suited to use in an authentication context.

All the authenticators from the three original categories seem to have flaws but tokens and biometrics seem particularly problematic. This confirms the fact that authentication in the digital world is much harder than it seems at first glance.

In proposing mitigation, we have chosen to focus on the the most common authentication mechanism, "what you know" authentication. This is the mechanism most users are familiar with, and it is most accessible and usable, so this is where amelioration might deliver the greatest benefit. It seems that passwords fail because humans cannot remember them and because they are so easy to divulge. This makes them choose information-rich, and easily guessed, passwords or compromise their secret passwords by recording them. Weidenbeck argues:

> **"A better way to overcome the password problem is to develop password systems that reduce fundamental memory problems"** [78, p. 105]

Hence we should try to address the memorability issue. If there were a way to ease the password's memorial load and to make guessability probematical we might well strengthen the mechanism.

The research reported here seems a viable direction to take in terms of strengthening "what-you-know" authentication since it addresses memorability issues, and because skill artefacts are more unpredictable than passwords. However, it could reasonably be argued that soundness could be compromised when one programmer wishes to guess another's password, since they share the same skill set. Thus SNIPPET adds another dimension: action-planning memory, thus exploiting the generation effect [36]. An imposter does not only have to have the same skill set, they have to have the other part of the secret, the personal involvement with the production of the authenticator artefact, in order to be able to identify the correct target image. The authenticator artifact is the *result* of an expert deploying their skills. Our small pilot study has shown that, even amongst Java programmers who knew each other well, this second dimension helped to resist guessing attempts.

	User Authentication		
	Know-ledge-	**Object-based**	**ID-based**
Commonly Referred to as:	Password	Token	Biometric
Security Defense:	Closely kept	Closely held	Forge-resistant
Security Drawback:	Less secret with each use. Hard to remem-ber	Can be cloned	Impossible to replace. Not secret. No exact match
Soundness:	Yes if kept secret	Needs addi-tional knowl-edge or biomet-ric (must be kept secure)	Context & Bio-metric Depen-dent
Obscurity:	Secret	Possession	Possession
Unpredictability:	Secret	Unpre-dictable	Unpredictable
Matching:	Exact	Exact	Confidence Level
Memorability:	Not memo-rable if strong	Can be lost or stolen	N/A
Convenience:	Depends on strength	Can be lost	Very, except for false rejects and reader issues

Table 1: Extending O'Gorman's categorisation of authentication approaches [79, p. 7]

6.3 Summary

We have to consider whether the distinction between identification and authentication has blurred in our digital age. Biometrics have traditionally been an *identification* mechanism in the pre-IT world, and not used as *authentication* mechanisms. When one tries to use an identifier as an authenticator you come up against all the same problems you would for any unproven identity.

Tokens, as exemplified by bank cards and others of their kind, are also identification mechanisms. The holder of the card always has to proffer further proof that they do indeed have the right to hold the card: to verify their identity. The driver's licence has a biometric: the person's face, and in South Africa their fingerprint too. Thus the combination

of token and authenticator acts as convincing evidence that the holder of the card is entitled to claim the identity.

Only "what you know" mechanisms really keep the identifier completely separate from the authenticator. Given this desirable characteristic, it is definitely worth trying to bolster this mechanism to address its one big flaw: the memorial load imposed on users. A way to strengthen authentication is to remove need for the user to deploy coping mechanisms, i.e. to reduce the possibility that they will forget their authenticator. SNIPPET does this, by testing the expert's ability to identify artefacts that result from their practicing their trade, i.e. the evidence of their expert practice. From the evidence we have gathered this seems to be completely effortless, since it is encoded at a level in the brain that is not easily eroded.

Whereas this skill-based mechanism performed well, there is one issue that remains: the cost-benefit balance [22]. For the users the mechanism delivered a good cost-benefit balance since no effort was involved in recognising their own code snippets. They provided these snippets themselves, which gave the advantage of recognisability but since they were produced by skilled actions they were also less predictable than other schemes where users provided their own images [29]. Yet the manual selection of distractors, in order to ensure maximum strength, means that the system, as implemented, was not scalable. These images must be chosen carefully and should be purposely similar to the user's sequence of code snippets, in terms of programming language and perhaps the language used for the variables. In this way we could maximise the possibility that the distractors do not weaken the mechanism by making the target stand out. Clearly automatic generation of such distractors would be an interesting topic for further research.

Finally, even though users are less likely to forget their SNIPPET secrets, it is possible that this, in itself, will not deliver sufficient benefit to persuade organisations to expend the extra effort required to deploy SNIPPET.

7. CONCLUSION

In this paper we have argued that the common and garden password cannot reasonably be referred to as an instance of "knowledge-based authentication". Passwords are ideally meaningless and therefore more akin to nonsense than knowledge. We have also pointed out the flaws of the two most popular alternatives: tokens and biometrics, and concluded that they could perhaps be more aptly used as secure identifiers. They do not really satisfy the soundness constraint required of authenticators.

We have tested skill-based authentication, structured as a recognition-based graphical authentication mechanism. We found that it was possible successfully to test recognition of the artefacts resulting from the practice of skilled activities in an authentication setting. Moreover, such authentication appears to be both memorable and resistant to shoulder-surfing and guessing attacks. There is admittedly a problem related to scalability of the solution in an industrial setting and this is an area that merits further consideration. Certainly these preliminary findings suggest that further research is worthwhile.

REFERENCES

[1] F. J. Corbató, M. Merwin-Daggett, and R. Daley, "An Experimental Time-sharing System," in *Proceedings of the Spring Joint Computer Conference*. ACM, May 01-03 1962, pp. 335–344.

[2] B. Riddle, M. Miron, and J. Semo, "Passwords in use in a university timesharing environment," *Computers and Security*, vol. 8, no. 7, pp. 569–578, 1989.

[3] B. Schneier, "Two-factor authentication: Too little, too late." *Communications of the ACM*, vol. 48, no. 4, 2005.

[4] A. Conklin, G. Dietrich, and D. Walz, "Password-based authentication: a system perspective," in *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*. IEEE, 2004, pp. 10–pp.

[5] D. Florêncio, C. Herley, and B. Coskun, "Do strong web passwords accomplish anything?" in *Proceedings of the 2nd USENIX workshop on Hot topics in security*. USENIX Association, 2007, p. 10.

[6] Unattributed BBC News Item, "Apple sells 9 million of its new iPhone models," 23 September 2013, http://www.bbc.co.uk/news/business-24201526. Accessed 16 October, 2013.

[7] S. Pankanti, R. Bolle, and A. Jain, "Biometrics: The future of identification [guest eeditors' introduction]," *Computer*, vol. 33, no. 2, pp. 46–49, 2000.

[8] C. Barral and A. Tria, "Fake fingers in fingerprint recognition: Glycerin supersedes gelatin," in *Formal to Practical Security*. Springer, 2009, pp. 57–69.

[9] A. K. Karlson, A. Brush, and S. Schechter, "Can i borrow your phone?: understanding concerns when sharing mobile phones," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009, pp. 1647–1650.

[10] S. Landau, "Making sense from Snowden," *IEEE Security & Privacy Magazine*, no. 4, p. 5463, 2013, http://privacyink.org/html/MakingSense.pdf.

[11] HNBulletin, "Exclusive: Apple admits, 'iPhone 5s Fingerprint Database to be Shared with NSA'," 23 Sept 2013, hacker News Bulletin. [Online]. Available: http://hackersnewsbulletin.com/2013/09/apple-admits-iphone-5s-fingerprint-database-shared-nsa.html

[12] National Report, "Apple iPhone 5s fingerprint database to be shared with NSA," Sept 2013, hacker News Bulletin. [Online]. Available: http://nationalreport.net/apple-iphone-5s-fingerprint-database/

[13] Corbató, F.J., "On building systems that will fail," in *ACM Turing Award Lectures*. ACM, 1990.

[14] Federal Trade Commission, "Federal Trade Commission, Plaintiff, v. Wyndham Worldwide Corporation; Wyndham Hotel Group, LLC; Wyndham Hotels & Resorts, LLC; and Wyndham Hotel Management, Inc., Defendants (United States District Court for the District of Arizona)," August 2012. [Online]. Available: http://www.ftc.gov/os/caselist/1023142/120809wyndhamcmpt.pdf

[15] A. Adams and M. A. Sasse, "Users are not the enemy," *Comm. of the ACM*, pp. 40–46, 1999.

[16] G. Kreizman and A. Allan. (2006, November) Toolkit: Evaluating Enterprise Options for Managing Passwords. [Online]. Available: http://www.gartner.com/id=498322

[17] F. Cunha, J. J. Heckman, L. Lochner, and D. V. Masterov, "Interpreting the evidence on life cycle skill formation," *Handbook of the Economics of Education*, vol. 1, pp. 697–812, 2006.

[18] W. A. Rogers, "Assessing age-related differences in the long-term retention of skills," *Aging and skilled performance: Advances in theory and applications*, pp. 185–200, 1996.

[19] R. Smilkstein, "We're born to learn: Using the brain's natural learning process to create today's curriculum," 2003.

[20] L. Anderson, D. Krathwohl, P. Airasian, K. Cruikshank, R. Mayer, P. Pintrich, J. Raths, and M. Wittrock, *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives, Complete Edition*, L. Anderson and D. Krathwohl, Eds. Longman, 2001.

[21] US-CERT: Official Website of the Department of Homeland Security, "Security tip (st04-002) choosing and protecting passwords," May 2009. [Online]. Available: http://www.us-cert.gov/ncas/tips/ST04-002

[22] S. Kurzban, "Easily remembered passphrases: a better approach," *ACM SIGSAC Review*, vol. 3, no. 2-4, pp. 10–21, 1985.

[23] H. Crawford, "A framework for continuous, transparent authentication on mobile devices," Ph.D. dissertation, 2013.

[24] M. Perlmutter, "Age differences in adults' free recall, cued recall, and recognition," *Journal of Gerontology*, vol. 34, no. 4, pp. 533–539, 1979.

[25] L. Bäckman and L.-G. Nilsson, "Prerequisites for lack of age differences in memory performance," *Experimental Aging Research*, vol. 11, no. 2, pp. 67–73, 1985.

[26] K. Renaud, T. McBryan, and P. Siebert, "Password cueing with cue(ink)blots," in *IADIS Computer Graphics and Visualization 2008 (CGV 2008) Conference*, Amsterdam. The Netherlands, 24 - 26 July 2008, pp. 74–81.

[27] M. Zviran and W. J. Haga, "Cognitive passwords: the key to easy access control," *Computers & Security*, vol. 9, no. 8, pp. 723–736, 1990.

[28] D. Davis, F. Monrose, and M. Reiter, "On User Choice in Graphical Password Schemes," in *Proceedings of the 13th USENIX Security Symposium*, 2004, pp. 151–164.

[29] T. Pering, M. Sundar, J. Light, and R. Want, "Photographic authentication through untrusted terminals," *Pervasive Computing, IEEE*, vol. 2, no. 1, pp. 30–36, 2003.

[30] K. Renaud, "A visuo-biometric authentication mechanism for older users," in *People and Computers XIXThe Bigger Picture*. Springer, 2006, pp. 167–182.

[31] R. Dhamija and A. Perrig, "Déjà Vu: A User Study Using Images for Authentication," in *Proceedings of the 9th Conference on USENIX Security Symposium*, 2000.

[32] R. English and R. Poet, "Measuring the revised guessability of graphical passwords," in *Network and System Security (NSS), 2011 5th International Conference on*. IEEE, 2011, pp. 364–368.

[33] K. Renaud and A. De Angeli, "My password is here! An investigation into visuo-spatial authentication mechanisms," *Interacting with computers*, vol. 16, no. 6, pp. 1017–1041, 2004.

[34] S. Argamon, M. Koppel, J. Fine, and A. R. Shimoni, "Gender, genre, and writing style in formal written texts," *Text*, vol. 23, no. 3, pp. 321–346, 2003.

[35] D. Estival, T. Gaustad, S. B. Pham, W. Radford, and B. Hutchinson, "Author profiling for english emails," in *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, 2007, pp. 263–272.

[36] N. J. Slamecka and P. Graf, "The generation effect: Delineation of a phenomenon." *Journal of experimental Psychology: Human learning and Memory*, vol. 4, no. 6, p. 592, 1978.

[37] C. Fernyhough and J. Russell, "Distinguishing ones own voice from those of others: A function for private speech?" *International Journal of Behavioral Development*, vol. 20, no. 4, pp. 651–665, 1997.

[38] M. Longcamp, J.-L. Anton, M. Roth, J.-L. Velay *et al.*, "Visual presentation of single letters activates a premotor area involved in writing," *Neuroimage*, vol. 19, no. 4, pp. 1492–1500, 2003.

[39] M. Longcamp, T. Tanskanen, and R. Hari, "The imprint of action: motor cortex involvement in visual perception of handwritten letters," *Neuroimage*, vol. 33, no. 2, pp. 681–688, 2006.

[40] B. H. Repp and G. Knoblich, "Perceiving action identity how pianists recognize their own performances," *Psychological Science*, vol. 15, no. 9, pp. 604–609, 2004.

[41] M. Schleidt, "Personal odor and nonverbal communication," *Ethology and Sociobiology*, vol. 1, no. 3, pp. 225–231, 1980.

[42] D. Jokisch, I. Daum, and N. F. Troje, "Self recognition versus recognition of others by biological motion: Viewpoint-dependent effects," *Perception*, vol. 35, pp. 911–920, 2006.

[43] J. S. Anastasi and M. G. Rhodes, "Evidence for an own-age bias in face recognition." *North American Journal of Psychology*, 2006.

[44] J. Stahl, H. Wiese, and S. R. Schweinberger, "Expertise and own-race bias in face processing: an event-related potential study," *Neuroreport*, vol. 19, no. 5, pp. 583–587, 2008.

[45] D. B. Wright and B. Sladden, "An own gender bias and the importance of hair in face recognition," *Acta psychologica*, vol. 114, no. 1, pp. 101–114, 2003.

[46] K. Renaud and E. Olsen, "Dynahand: Observation-resistant recognition-based web authentication," *Technology and Society Magazine, IEEE*, vol. 26, no. 2, pp. 22–31, 2007.

[47] K. Renaud, "Web authentication using mikon images," in *Privacy, Security, Trust and the Management of e-Business, 2009. CONGRESS'09. World Congress on*. IEEE, 2009, pp. 79–88.

[48] K. Renaud and A. De Angeli, "Visual passwords: cure-all or snake-oil?" *Communications of the ACM*, vol. 52, no. 12, pp. 135–140, 2009.

[49] M. Gladwell, *Outliers: The Story of Success*. Penguin, 2009.

[50] F. Craik and E. Tulving, "Depth of Processing and the Retention of Words in Episodic Memory." *Journal of Experimental Psychology: General*, vol. 104, no. 3, pp. 268–294, September 1975.

[51] J. J. Heckman, "The economics, technology, and neuroscience of human capability formation," *Proceedings of the National Academy of Sciences*, vol. 104, no. 33, pp. 13 250–13 255, 2007.

[52] NIST, "Guidelines on Evaluation of Techniques for Automated Personal Identification," National Institute of Standards and Technology, Tech. Rep. FIPS-PUB-48, April 1977.

[53] I. Jermyn, A. Mayer, F. Monrose, M. Reiter, and A. Rubin, "The Design and Analysis of Graphical Passwords," in *Proceedings of the 8th USENIX Security Symposium*. Washington DC, 23-26 August 1999, pp. 1–14.

[54] D. Lin, P. Dunphy, P. Olivier, and J. Yan, "Graphical Passwords & Qualitative Spatial Relations," in *Proceedings of the 3rd Symposium on Usable Privacy and Security*. ACM, 18-20 July 2007, pp. 161–162.

[55] M. Oka, K. Kato, Y. Xu, L. Liang, and F. Wen, "Scribble-a-Secret: Similarity-based Password Authentication Using Sketches," in *Proceedings of the 19th International Conference on Pattern Recognition*. IEEE, 2008, pp. 1–4.

[56] R. Weiss and A. De Luca, "PassShapes: Utilizing Stroke Based Authentication to Increase Password Memorability," in *Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges*. ACM, 2008, pp. 383–392.

[57] T. Matthews, D. Vogts, and K. Naudé, "Sketch-based interfaces: drawings to data," in *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, ser. SAICSIT '13. New York, NY, USA: ACM, 2013, pp. 359–368. [Online]. Available: http://doi.acm.org/10.1145/2513456.2513482

[58] A. De Angeli, L. Coventry, G. Johnson, and K. Renaud, "Is a picture really worth a thousand words? exploring the feasibility of graphical authentication systems," *International Journal of Human-Computer Studies*, vol. 63, no. 1, pp. 128–152, 2005.

[59] The Passfaces Corporation, "The Science Behind Passfaces," *White Paper, June*, 2004.

[60] K. Renaud and J. Maguire, "Armchair authentication," in *BCS-HCI '09: Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology*. British Computer Society, Sep. 2009.

[61] M. Al Fairuz and K. Renaud, "Multi-channel, multi-level authentication for more secure ebanking." in *2010 Information Security for South Africa (ISSA 2010) Conference*, Johannesburg, South Africa, 2010.

[62] N. Matsushita, S. Tajima, Y. Ayatsuka, and J. Rekimoto, "Wearable key: Device for personalizing nearby environment," in *Wearable Computers, The Fourth International Symposium on*. IEEE, 2000, pp. 119–126.

[63] Y.-C. Lee, Y.-C. Hsieh, P.-S. You, and T.-C. Chen, "An improvement on rfid authentication protocol with privacy protection," in *Convergence and Hybrid Information Technology, 2008. ICCIT'08. Third International Conference on*, vol. 2. IEEE, 2008, pp. 569–573.

[64] K. R. Foster and J. Jaeger, "RFID inside," *Spectrum, IEEE*, vol. 44, no. 3, pp. 24–29, 2007.

[65] M. Shahzad, A. X. Liu, and A. Samuel, "Secure unlocking of mobile touch screen devices by simple gestures: you can see it but you can not do it," in *Proceedings of the 19th annual international conference on Mobile computing & networking*, ser. MobiCom '13. New York, NY, USA: ACM, 2013, pp. 39–50. [Online]. Available: http://doi.acm.org/10.1145/2500423.2500434

[66] E. Maiorana, P. Campisi, N. González-Carballo, and A. Neri, "Keystroke dynamics authentication for mobile phones," in *Proceedings of the 2011 ACM Symposium on Applied Computing*, ser. SAC '11. New York, NY, USA: ACM, 2011, pp. 21–26. [Online]. Available: http://doi.acm.org/10.1145/1982185.1982190

[67] H. Tao and C. Adams, "Pass-Go: A Proposal to Improve the Usability of Graphical Passwords," *International Journal of Network Security*, vol. 7, no. 2, pp. 273–292, 2008.

[68] J. Brainard, A. Juels, R. Rivest, M. Szydlo, and M. Yung, "Fourth-factor authentication: somebody you know," in *Conference on Computer and Communications Security: Proceedings of the 13 th ACM conference on Computer and communications security*, vol. 30, 2006, pp. 168–178.

[69] S. Smith, "Authenticating Users by Word Association," *Computers & Security*, vol. 6, no. 6, pp. 464–470, 1987.

[70] A. Wismeijer, "Secrets and subjective well-being: A clinical oxymoron," in *Emotion regulation and well-being*. Springer, 2011, pp. 307–323.

[71] A. E. Kelly, J. A. Klusas, R. T. von Weiss, and C. Kenny, "What is it about revealing secrets that is beneficial?" *Personality and Social Psychology Bulletin*, vol. 27, no. 6, pp. 651–665, 2001.

[72] J. Berg, "Surrogate decision making in the internet age," *The American Journal of Bioethics*, vol. 12, no. 10, pp. 28–33, 2012.

[73] M. Richtel, "Young, in love and sharing everything, including a password," *The New York Times*, 2012.

[74] A. Patrick, "Monitoring corporate password sharing using social network analysis," in *International Sunbelt Social Network Conference*, St Pete Beach, Florida, 22-27 January 2008.

[75] Anonymous, "DNA Fingerprinting: A powerful law-enforcement. Tool with serious social implications," *The Scientist*, vol. 3, no. 11, p. 10, 1989, 29 May.

[76] T. Van der Putte and J. Keuning, "Biometrical fingerprint recognition: don't get your fingers burned," in *Smart Card Research and Advanced Applications*. Springer, 2000, pp. 289–303.

[77] S. Gold, "Meeting the biometrics payment security challenge," *Biometric Technology Today*, vol. 2013, no. 10, pp. 5–8, 2013.

[78] S. Wiedenbeck, J. Waters, J. Birget, A. Brodskiy, and N. Memon, "PassPoints: Design and Longitudinal Evaluation of a Graphical Password System," *International Journal of Human-Computer Studies*, vol. 63, no. 1, pp. 102–127, 2005.

[79] L. O'Gorman, "Comparing passwords, tokens, and biometrics for user authentication," *Proceedings of the IEEE*, vol. 91, no. 12, pp. 2021–2040, 2003.