Violations of Good Security Practices in Graphical Passwords Schemes: Enterprise Constraints on Scheme-Design

Johannes Vorster[1], Barry Irwin[1] and Renier van Heerden[2,3]
[1]Rhodes University, South Africa
[2]Council for Scientific and Industrial Research, South Africa
[3]Nelson Mandela Metropolitan University, South Africa
JSVorster@gmail.com
B.Irwin@ru.ac.za
RVHeerden@csir.co.za

Abstract: During the past decade, the sophistication and maturity of Enterprise-level Information Security (EIS) Standards and Systems has increased significantly. This maturity, particularly in the handling of enterprise-wide capability models, has led to a set of standards – e.g. ISO/IEC 27001, NIST 800-53, ISO/IEC 27789 and CSA CCM – that propose controls applicable to the implementation of an Information Security Management System (ISMS). By nature, the academic community is fruitful in its endeavour to propose new password schemes; and Graphical Passwords (GPs) have had many proposals for schemes. In this paper, we explore the impact of good security standards and lessons-learnt over the past decade of EID as a model of constraint on GPs schemes. The paper focuses on a number of GP schemes and points out the various security constraints and limitations, if such schemes are to be implemented at the enterprise level. First, we use standards such as NIST 800-53, the Cloud Security Association's Cloud Control Matrix (CCM) v3 and others, to construct a subset of standards that a new authentication mechanism, such as GPs, should conform to. Next, we analyze various GP schemes and show the limitations of these schemes from an EIS perspective, given the mentioned standards. We show that some schemes are secure in their construction, but lack scalability to enterprise-wide implementations. We show that other schemes lack the ability to hash-store passwords. Yet other schemes have insecure session-password schemes. We furthermore show that some schemes claim to be implementable on top of existing password models, however, often that requires that non-hash passwords are available. According to the OWASP (Open Web Application Security Project) the number two global web-security issue is broken authentication and session management, trumped only by injection vulnerabilities. The paper therefore is relevant in the current security context and the global dialogue on improving security. This is the first attempt, to our knowledge, to analyze GP schemes using enterprise-level implementation constraints.

Keywords: graphical passwords, access management, information security management system

1. Introduction and background

With the advent of touch screens, and particularly the widespread incorporation of this technology in smartphones and tablets, alternative methods of authentication via these interfaces are now more readily available. One of the security challenges in this environment is the creation of authentication systems that are not only secure, but also easy to use. The user-friendliness of a system is directly related to how securely it is being used; users may find non-secure methods to store passwords, or even bypass or misuse the security mechanisms in cases where the system is not user-friendly. Because of the proliferation of passwords in modern society, users tend to select passwords that have some memorable characteristics such as dictionary words or sequences of numbers such as birthdays. Even simple patterns such as leading capitalization give attackers an advantage. By prioritizing often-used passwords, letter frequencies or using frequency based attacks, such password strategies become vulnerable to attack (van Heerden et al., 2009).

Graphical Passwords offer the opportunity to replace conventional text based passwords, which are often insecure (Florencio & Harley 2007). Biddle, Chiasson & van Oorschot (2012) points out that the use of graphical passwords not only improves usability compared to text based passwords, but also strengthens the overall password security.

2. Methodology

We initiated the study by identifying relevant standards. The approach followed started with international standards, identified standards used by specific industries, found standards indexes and identified which standards are regularly cross-referenced against each other.

This was followed by an analysis of each standard, to identify guidelines and controls that may be relevant to the implementation of an Alternative Authentication Scheme (AAuS). Although we are interested in Graphical Password Schemes (GPs) and constraints on such schemes, this analysis is also relevant to the implementation of any AAuS as a general concept. Finally, we used these controls and guidelines, and analyzed the impact that they may have on the implementation of an AAuS, and in particular GPs.

In this paper we present a summary of some of the analysis and findings. Due to space limitations it is not possible to present a complete overview of the analysis results, nor of the various GPs that were analyzed. An in-depth journal article is planned.

3. Enterprise security standards: Perspective

Information Security (IS) standards grew in number over the past decade. Initial standards were driven globally, to a large extent, by two organizations: The International Organization for Standardization (ISO), and - in the U.S. - the National Institute of Standards and Technology (NIST). Within ISO, the 27000 series of standards focuses on Information Security Standards and within NIST, the 800 series of standards play the same role.

A way to organize these standards is to see a specific standard, say NIST 800-53, as a program[1] what evolves the standard, then every few years a new version of that standard is published. Within the published standard the IS capabilities are defined in the form of various knowledge or focus Domains. For example, Risk Assessment is one of the domains within NIST 800-53. Then, usually the standard would define a set of objectives or requirements. These objectives are usually high-level directives such as achieving "Encrypted Transmissions". Within each such high-level objective the standards then define a set of security controls, or, rather, the standards propose a set of controls. It is up to the organization itself to define and implement the controls, specific to that organization. Figure 1 shows this progression in detail from the standards program down to the specific controls.

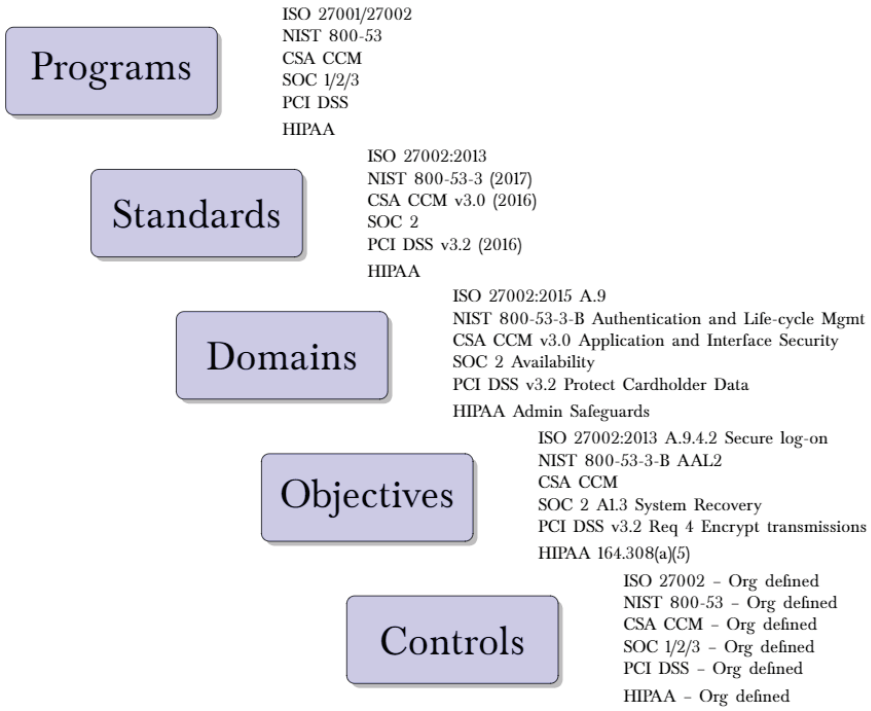


Figure 1: Structure of security standards

The number of standards, as well as the complexity of the standards themselves has grown. The complexity can be understood in the context of the growth of new technology such as wireless technologies, the emergence of mobile devices as high-end computational devices and new computational models such as Infrastructure-as-a-service (IaaS), Platforms-as-a-service (PaaS) and Software-as-a-service (SaaS). Apart from

[1] Program here has the context of a space program, not a software program.

these technological pressures on standards, there are also the emergence of industry specific standards that focus mostly on the global understanding of the risks associated with the protection of private data.

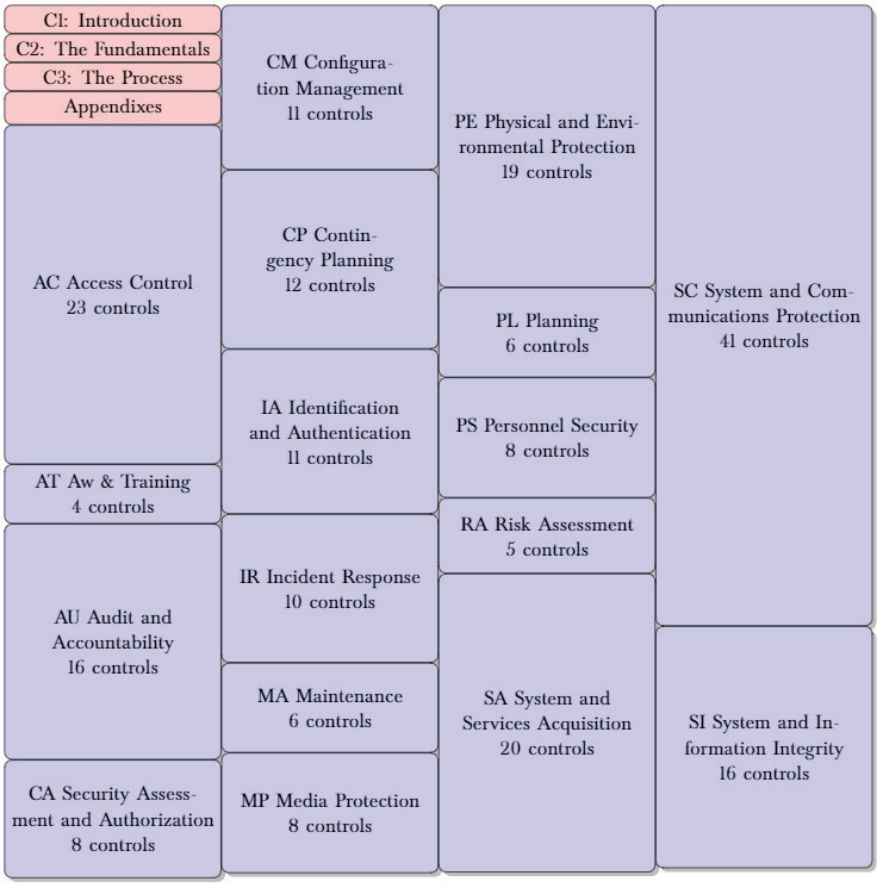


Figure 2: NIST 800-53 structure and controls

NIST and ISO are generic standards, but specific industries published their own standards as well. The Payment Card Industry (PCI) has published the Data Security Standards (DSS), globally known as the PCI DSS, to govern the enterprise ecosystems that handle credit card information. Similarly, the healthcare industry is publishing standards to protect customer's healthcare information – the Health Insurance Portability and Accountability Act (HIPAA) is an example. The attestation of compliance – measured by auditing firms has, at the same time, also become more focused, the American Institute of Certified Public Accountants (AICPA) created Service Organization Control (SOC) auditing standards. Lastly it is worth mentioning that cloud computing has now also generated a set of standards specific to that environment in the form of the Cloud Security Association's Cloud Control Matrix (CSA CCM).

4. Enterprise security standards: Controls

Each of the various security standards mentioned above has a specific breakdown of security domains, and each has a unique approach to specifying the requirements or controls that an organization should implement. The complexity and detail in these standards also vary significantly. **Table 1** shows some of the standards and the diversity in the number of domains and controls that they specify.

Table 1: Domains and controls for various security standards

Standards	Domains	Controls
NIST SP 800-53 (R4, R5 draft)	17 domains	224 Controls
ISO 27001/2 Standards for ISMS (2013)	14 domains	120 Controls
NIST SP 800-63 Digital Identity Guidelines (2017)	Guidelines	Guidelines
NIST Center for Internet Security – Critical Security Controls for Effective Cyber Defense (V 6.0)	20 control domains	96 Measurements
PCI DSS 3.0 Payment Card Industry Data Security Standards	12 requirements	77 sub-requirements

The Center for Internet Security published – Critical Security Controls for Effective Cyber Defense (V 6.0) (CIS CSC, 2015) – that acts as an implementation and measurement guide. It specifies measurements for the effective implementation of the controls. There is a large contrast between the level of details specified in the various standards and their controls. CIS CSC, for example, suggests a measurement of the number of attempts at gaining access to password files; a specification that can be considered as a very low level or detailed specification. In contrast, ISO 27001 and NIST 800-53 only dictates that a control should exist that measures security incidents, but the actual specifics of such a control is left to the implementation organization. Similarly, the PCI DSS specifies that password length should be at least 7 characters, and that dormant accounts should be deactivated within 90 days - both very detailed and specific controls.

A comparison between ISO 27002 (the detailed controls standard, and companion to ISO 27001) and NIST 800-53 R4 shows that, although both cover all the domains, there are some aspects where NIST is more thorough. Within the Access Control domain, for example, NIST specifies controls for session management; data mining protection and the use of a Reference Monitor; which is lacking in the current ISO standard.

Another important standard, often used by organizations for setting standards on Identity, Access and Entitlements Management (IAEM) is NIST 800-63. Its most recent release (2017) consists of three documents: NIST 800-63-A (2017) focuses on Enrollment and Identity Proofing; NIST 800-63-B(2019) looks at Authentication and Lifecycle Management and NIST 800-63-C(2017) addresses Federation and Assertions. These standards have enterprise implications for the implementation of federated identity models, single sign-on (SSO) and how an Alternative Authentication Scheme (AAuS) can be implemented at an enterprise level.

NIST 800-63-A (2017) specifies three levels of Identity Assurance (IAL). At level 1 the only proof is what the user says; at level 2 the user is present, and provides one piece of evidence, and at level 3 the user provides more substantial verified proof of identity. The IAL is formed by various sets of evidence based on what the user presents, how the evidence is validated and then verified. At IAL1 there is no evidence linking the applicant to a real identity of a known person. At IAL2 evidence for the link is provided, validated and verified. At IAL3, all the conditions for IAL2 is met, but a physical link is also established. Throughout all three levels NIST 800-63 documents, the same methodology is used, that is, to establish a level of assurance based on a very clearly defined set of criteria.

NIST 800-63-B(2017) establishes Authentication Assurance Levels (AAL), which is based on the types of authenticators used, such as something-you-know, also called knowledge-based-verification (KBV). AAL2 requires two-factor authentication through approved cryptographically secured protocols. AAL3 requires hardware based authentication as well as verification that the authenticator (the hardware) is impersonation resistant and a two-factor authentication.

NIST 800-63-C (2017) defines the Federated Assurance Level (FAL), based on how much evidence is available that the federated authentication can be trusted. An assertion of identity, called a bearer assertion, is provided by an Identity Provider (IdP) to the bearer, based on authentication provided to the IdP to which an AAL level can be assigned. The bearer can now use that assertion to access resources provided by a relying party (RP). The RP can verify the assertion (of identity and perhaps also authorization) because it is signed by the IdP using an asymmetric cryptographic key. At FAL1, the bearer assertion is signed by the IdP; at FAL2 the assertion is signed by the IdP and encrypted using the RP's public key; and at FAL3, the assertion includes that the bearer is a holder of a key – e. g. provided by a cryptographic device.

After an analysis of the various standards mentioned above, we have identified a set of requirements and constraints that apply to the implementation of an AAuS. In the remainder of this section we look at specific processes, policy statements and controls that apply to the successful implementation of an AAuS within a large organization.

4.1 User identification and enrolment processes

The establishment of the user's identity is largely ignored by the standards, other than stating that a user's identity must be unique within the system (PCI DSS 8.1.1, CSA CCM IVS-01, SOC 2 CC6.2). Of the standards under consideration, the NIST 800-63-A (2017) guidelines is by far the most sophisticated and thorough standard for the mitigation of identity theft. Most systems and processes assume that the identity at enrolment is sufficiently established and beyond reproach, but the level of confidence in that assertion is in

most cases at IAL2, because organizations will request identification documents (in countries where those are available) or do some form of background check. This part of the IAEM process is largely outside the scope of this paper.
The enrollment process needs to verify that the password was typed in correctly, and some standards specify the length of the password. PCI DSS control 8.2.3 require a password of at least 7 characters, both numbers and characters and makes provision for non-password AAuS by stating that such a mechanism must provide at least equivalent strength, but refers to NIST 800-63 for further guidelines. NIST 800-63-B (2017) has changed its stance on password lengths and now require 8-character length passwords, which do not have to expire when selected by the user. The other standards on ISMS reviewed (ISO 27001/2 and NIST 800-53) insist that the organizational policy define these numbers, but are non-prescriptive to the strength.

The transmission and storage of passwords must be encrypted using strong encryption (PCI DSS control 8.2.1). OWASP and RSA Labs (eg. Moriarty et.al, 2017) suggest that using salted hashes is the best approach to storage of credentials. The ability of an AAuS to evaluate passwords stored in salted hashed form will be a strong theme later. It is this requirement that also plays a significant role in biometric authentication, because it is not possible to store biometric information in salted hashed form, due to the non-precise measurements during enrolment and authentication. Thus, an intruder that steals the encrypted data can recover the original biometric data, thus compromising the link between the user-identity and the user as a physical entity. SOC 2 (cc 6.1) as well as ISO 27001 (A10) specify the use of strong cryptographic mechanisms for storing and transmitting password information, but again are not prescriptive on strength.

4.2 Password policy and authentication processes

The ISO 27001 standard specifies that an organization must have access control policies (A.9.4.1), secure logon procedures (A.9.4.2) and password management systems (A.9.4.3), but is in no way prescriptive to the quality of passwords and storage management principles. PCI DSS 8.2.4, is more specific, specifying a password change at least once every 90 days and that passwords should have a length of at least 7 characters. NIST 800-63 (2017) specifies password length of 8 characters if it is a user-selected password, but if it is a Credential Service Provider (CSP) generated password, then 6 character-length passwords are sufficient. NIST 800-63 (2017) also changed the time that passwords should be valid, and now, based on a body of research, changed this to no-expiry time. AAuS implementations have the potential to improve on password complexity. Figure 3 shows some statistics on GP schemes taken from Vorster (2014) that suggests that users may select GP passwords with lengths less than the expected number of characters in conventional passwords, but that the overall complexity may be higher due to the larger symbol-set, or potential symbols.

The standards also specify that tests should be done to verify that passwords are not re-used. The PCI DSS (control 8.2.5) specifies a 4-password history, though other standards are not so prescriptive.

The standards also require that a reset password, and a first-time password must be unique (no re-use of same password, say myOrg01 for many new users).

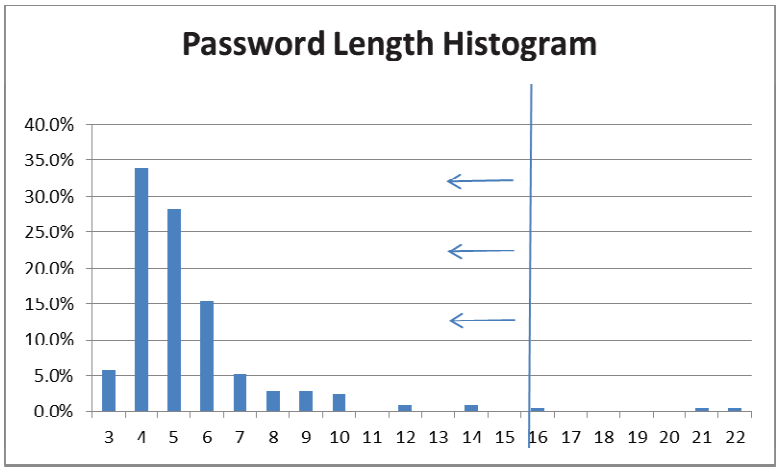


Figure 3: Measured password lengths for 5 graphical password schemes

As mentioned earlier, NIST 800-63-B defines the levels of assurance for authentication based on the number of factors, but also on the use of cryptographically signed keys from a crypto device. However, in most organizations single-factor authentication is sufficient (e. g. PCI DSS 8.2) when the user is under conditions of physical access control – that is, the user had to already have provided a what-you-have authentication check when entering the building. In some cases, such as when working from remote locations, or in specialized environments such as the Card Data Environment (CDE), specific industry standards require two-factor authentication (e. g. PCI DSS control 8.3).

4.3 Vendor and 3rd party system passwords

All default passwords must be changed prior to installation on networks (e. g. NIST 800-53 SA-4, PCI DSS 2.1). This requirement implies that any AAuS that is not in some sense password based, will have to co-exist in a password-based ecosystem. It will not be possible to completely implement an AAuS across all aspects of an organization's operations, due to the high degree of reliance on 3rd party authenticators throughout the various processes being used, and specified in the standards.

5. Graphical password schemes: General controls and constraints

During the past decade, many different graphical password systems have been proposed and developed. These can be categorized based on the mechanism used for recalling the password. The first category comprises systems based on recall, such as Draw-a-Secret (DAS) – see Figure 4 -- which asks the user to draw a simple picture as a password (Jermyn et.al 1999). This AAuS can use an encoding of the grids through which a line is drawn to encode a password. In this example the curve starts at (1), at grid (2,2) and can be encoded as a 5, next the curve moves to grid (2,3), a block that can be encoded as 9, etc. so that the cure shown can encode the password "59A654". The drawback with this encoding is that a slight shift in starting position can lead to crossing to different blocks, and thus encode a different password.

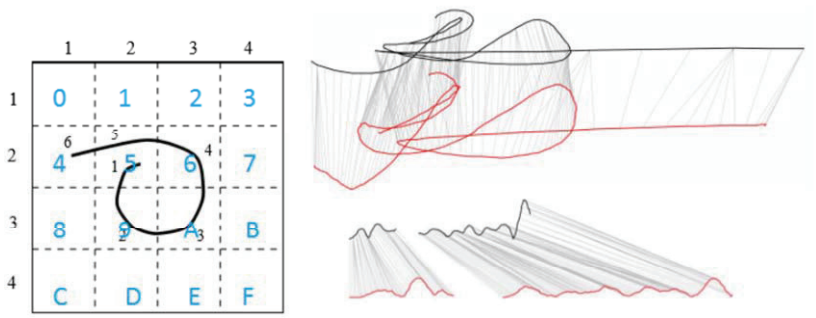


Figure 4: Draw-a-secret (left) and an example of dynamic time warping (right)

A Cued-recall graphical password scheme provides the user with a cue to help with recall. One way to do this is to provide an image on which the user can then draw the secret. This is what Dunphy and Yan (2007) proposed by adding a background image to DAS. This addition changes the user dynamic, but still has the same constraint as DAS.

Tao(2006) proposes Pass-Go, inspired by the Chinese game Go, as part of his Masters dissertation. Pass-Go is in many ways comparable to DAS (Tao and Adams, 2008). In DAS, the user selects cells, whereas in Pass-Go the user selects intersections. Pass-Go and DAS are very similar, but, because Pass-Go uses intersections, it is slightly more efficient at using the available interface space.

The most popular Graphical password scheme in use – measured by sheer number of users – is the Android Unlock Pattern (AUP) (Uellenbeck et al., 2013). The Android operating system (OS) implements a 3x3 version of Pass-GO, simplified to be usable on the small smart-phone screens. Uellenbeck computes that this method has a maximum of $2^{18,570}$ passwords. There are other limitations on this scheme from a standards perspective. Firstly, it only allows one class of characters (numbers, 0-9) which is already restrictive. Secondly, the password lengths are also a problem due to the re-use constraint – an artificial constraint that do not allow the re-use of an already used "character". One way to expand this method of authentication is by increasing the dimensions from 3x3 to 6x6, thus allowing for 36 "characters", the same size as an alpha-numeric symbol-set [{a-z}+{0-9} = 26+10 = 36].

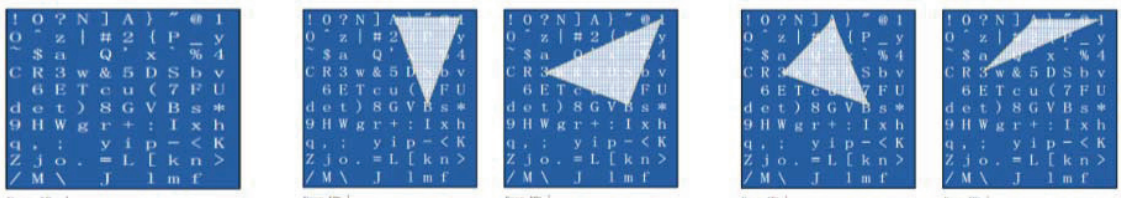
Everitt and McOwan (2003) proposed an alternative encoding, using time-based coordinates. Within this encoding scheme there are two models for automatic signature verification. The first is dynamic time warping (DTW), which uses the tempo of strokes as a biometric measure (Kholmatov and Yanikoglu, 2005). The second verification mechanism makes use of Hidden Markov Models (HMM) (Fischer et al., 2015). In Figure 4, the two signatures (right top) seem similar, however, when a DTW (right bottom) is applied, the signatures are very distinct. The Everitt-McOwan method yields false acceptance rates (FAR) of 4.4%, which is low for signature verification, but high as an authentication mechanism. That is, both these methodologies for draw-based authentication have flaws that make them unsuitable for enterprise authenticators. In addition, the Everitt-McOwan encoding cannot use salt-hash storage for the passwords, since a direct comparison is needed.

PassPoints was proposed and developed by Wiedenbeck et al (2005a). The user is presented with an image. And then asked to select several points on the image as the password – see Figure 5. Each point entered as part of the authentication process is compared to the corresponding point of the original enrolment set and must be within a certain tolerance level to that point. Jansen (2003) proposed a method whereby an image is overlaid with a grid to assist users. In this schema password patterns can be used that are dependent on the grid and not on the background image. Both methods show a weakness in that user-selection of points tend to be clustered around predictable points – see Figure 5 (right). However, because Jansen's method of encoding uses grid squares, the encoded sequence can be stored as a salted-hash, without compromising the security of AAuS.

Figure 5: PassPoints: Enrollment image (left) and hotspots (right)

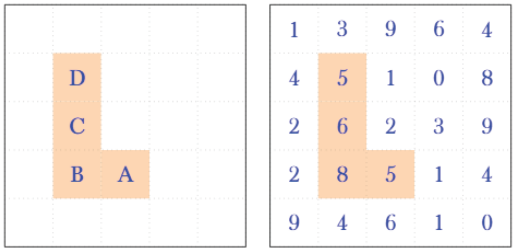
From an enterprise perspective this method of authentication has many difficult controls to overcome. Firstly, this method will work on high-resolution screens, but may not work that well on smaller screens, such as on mobile devices. It is also not clear how to implement this method for say SSO-based disk encryption, as is provided by many vendors. That is, if disk-level encryption is used, then a password is required prior to unlocking the disk, this password is linked to the organizational SSO provider via a filter installed in the GINA (Graphical Identification and Authentication) evaluation path.

Another class of Graphical Password Authenticators is Cognometric systems, that is, systems where the user has to recognize the correct image hidden within a set of dummies. PassFaces – proposed by Brostoff and Sasse (2000), and now commercialized by the Real User Corporation (http://www.realuser.com) – shows the user a set of faces from which the user must select the correct one; repeated for three rounds. Similar to PassFaces is Déjà Vu, also a Cognometric system, which uses abstract images rather than faces. However, it was found that memorability suffers with the use of abstract images. Dhamija & Perrig (2000) reported that the recall error-rates doubled for abstract images compared with faces. GrIDsure, proposed and implemented by Brostoffet al. (2010), is a GP scheme for OTP. In this scheme the user enrolls by selecting a pattern from a 5x5 grid – see Figure 7a. In this example the user selected grid-squares A,B,C,D; in that order. This is the user's PIN pattern. During authentication – Figure 7b – the user is presented with the same grid, this time randomly filled with numbers from 0 to 9. The user reads off the OTP following the same pattern used during enrolment. For this example, the OTP would be 5865. An implementation can be constructed using this method that is salt-hashed, and thus comply with the earlier mentioned constraints on password encryption.



a b c d e

Figure 6: S3Pass session passwords



(a) Enrollment (b) Authentication

Figure 7: GrIDsure OTP scheme

Because of its graphical and thus visible nature, GPs have been criticized as being vulnerable to shoulder-surfing, the practice of looking over a user's shoulder during authentication. To address the shoulder-surfing vulnerability of GPs, Zhao and Li (2007) proposed Scalable Shoulder-Surfing Resistant Textual-Graphical Password Authentication Scheme (S3PAS). The claim is that it "… seamlessly integrates both graphical and textual password schemes and provides nearly perfect resistance to shoulder-surfing, hidden-camera and spyware attacks." There is no enrollment; this proposal slots into existing conventional password models, but presents the user with an alternative way to enter parts of the password in such a way that the password itself remains hidden. The user is presented with an image of letters, numbers and symbols, randomly distributed – Figure 5a.

The user determines the OTP as follows. First find the first three letters of the user's password on the image (b), and select any letter from the triangle formed by these three letters. Next find letters 2,3 and 4 of the password (c) and again select a letter within the triangle formed by these three letters. The OTP is now the sequence of characters selected. If the user runs out of letters in the password, then wrap back to the first letter.

This seems like a workable, though slow, process. One way to speed this up is to use only the first 3 characters. Because the triangle size is random, the number of possible OTPs from the same password is huge, and even if a hacker observes this process repeatedly, it will be extremely difficult to deduce the original password - or so it seems. However, there are also security control constraints on this scheme. Because the system that generates the random first image (a) does not have access to the user-password, it is not possible to construct an image with all the letters form the user's password far apart. Consider a password "wET3" and how the OTP will be generated given image (a) above. In that case, because of the proximity of the letters on the image, the triangles are empty, save for the letters from the password itself. Thus, for the scheme to work in these cases, the authenticator must also accept letters from the password itself. Therefore, with a password of "wET3", the first OTP letter can be any one of "wET", let's pic T, the next triangle is formed from "ET3", and again we can re-use T, the third OTP character comes from the triangle formed by "T3w", wrapping back to the beginning of the password for the third letter. Again, the letter T can be used. In this scheme, no matter what the password, or how the letters are arranged, using the 3rd password letter 3 times, then the 6th letter 3 times, and so on, will always yield a successful authentication.

6. Graphical passwords: General discussion and road forward

In sections 3 and 4 of this paper the current international, industry and government standards and guidelines were analyzed for controls that may have an impact on the implementation of an AAuS. In the previous section, some GP schemes were analyzed and some vulnerabilities and implementation constraints were pointed out.
In this section these two viewpoints are considered, and potential future direction and research are proposed.

The standards are mostly policy driven and should be seen as a type of checklist to make sure that an organization has thought through all the aspects of building an ISMS. The standards are not prescriptive on the implementation, and thus there are no constraints from standards such as ISO 27001, NIST 800-53, SOC 2 or CSA CSS that restrict the implementation of an authenticator such as a GP.

From this research, we propose that there are three constraints on AAuS implementations in general and thus on GPs specifically:

1. The non-universality of an AAuS;
2. The attestation compliance for an AAuS;
3. The management of risk for an implementer-Organization.

6.1 The non-universality of an AAuS

An organization can implement an AAuS as part of its SSO configuration, or as a federated security model, but it cannot be a universal authentication mechanism. We mentioned earlier the disk-level encryption, where the vendors use the SSO credentials to unlock the disk-level encryption at boot-up. With this constraint, AAuS can only act as a secondary authenticator, or the disk-level encryption vendors must support the AAuS scheme.

In large organizations authentication solutions and key-management is then needed for BIOS access, disk-encryption, network-device passwords and VPN password. All these types of authenticators use conventional passwords, and thus an organization cannot implement only an AAuS as a universal solution.

This constraint, in our view, restricts an AAuS, and GPs to niche applications, one of which may be as a federated identity model for application-level access.

6.2 The attestation of compliance for an AAuS

The implementation of an AAuS will require an organization to construct various policies, and if formal compliance is sought -- such as ISO 27001 certification – then the auditing firm must accept these policies and controls as adequate.

6.3 The management of risk and effort for an implementer-organization

For an organization to implement an AAuS, and still comply with the mentioned standards will require a significant effort in the construction of policies, user training and user awareness. An AAuS may be proven to be more secure – because the password space is larger, for example – but risk-averse organizations and individual managers may consider the implementation of an AAuS as a mainstream authenticator as too risky, not just from a security perspective, but also from a cost-benefit perspective. As we have shown with S3PAS, even seeming improvements in a security scheme can have unintended consequences. It is not unreasonable that risk-averse organizations are unwilling to implement an AAuS.

7. Further work

An interesting insight that emerged from this work is how non-prescriptive the standards are, when viewed from a technology and solutions perspective. An interesting study would be to build a minimalist compliant policy document. That is, what is the smallest security policy that would be compliant to say ISO 27001. From the above analysis, it seems that the implementation of an AAuS, and specifically Graphical Passwords can be successfully used within a federated identity model as an authenticator for an Identity Provider. A detailed analysis of such an implementation and use of GPs is of practical value.

References

Biddle, R., Chiasson, S., and Van Oorschot, P.C. (2012). *Graphical passwords: Learning from the fist twelve years*. ACM Computing Surveys (CSUR), 44(4).

Brostoff, S. and Sasse, M. A. (2000). *Are passfaces more usable than passwords? A field trial investigation*. In People and Computers XIV—Usability or Else!, pages 405–424. Springer.

CIS CSC (2015), *A measurement Companion to the Center for Internet Security (CIS) Critical Security Controls (CSC) Version 6*. NIST (National Institute of Standards and Technology). October 2015.

Dhamija, R. and Perrig, A. (2000). *Deja vu-a user study: Using images for authentication*. In USENIX Security Symposium, volume 9, pages 4–4.

Dunphy, P. and Yan, J. (2007). *Do background images improve draw a secret graphical passwords*? In Proceedings of the 14th ACM conference on Computer and communications security, pages 36–47. ACM.

Everitt, R. A. and McOwan, P. W. (2003). *Java-based internet biometric authentication system*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(9):1166–1172.

Fischer, A., Diaz, M., Plamondon, R., and Ferrer, M. A. (2015). *Robust score normalization for dtw-based on-line signature verification*. In Document Analysis and Recognition (ICDAR), 2015 13thInternational Conference on, pages 241–245. IEEE.

Florêncio, D. and Herley, C. (2007). *A large-scale study of web password habits*. In Proceedings of the 16th international conference on World Wide Web, pages 657–666. ACM.

Jansen. W., Gavrilla, S., Korolev, V., Ayers, R. and Swanstrom, R. (2003). *Picture Password: A Visual Login Technique for Mobile Devices*. NIST Report: NISTIR 7030.

Jermyn, I., Mayer, A., Monrose, F., Reiter, M.K. and Rubin, A.D. (1999). *The design and analysis of graphical passwords*. In Proceedings of the 8th USENIX Security Symposium. Berkeley, CA, USA. USENIX Association.

Jones, A. (2015). Nedbank Security Policy. *Consulting security architect for the financial industry*. Private conversations.

Kholmatov, A. and Yanikoglu, B. (2005). *Identity authentication using improved online signature verification method*. Pattern recognition letters, 26(15), pp2400–2408.

Moriarty, K., Kaliski, B., and Rusch, A. (2017). *Password-Based Cryptography Specification Version 2.1*. Internet Engineering Task Force (IETF). RFC 8018. January 2017.

NIST SP 800-63 (2017). *Digital Identity Guidelines: Authentication and Lifcycle Management*. NIST (National Institute of Standards and Technology) Special Publication 800-63. Online: https://pages.nist.gov/800-63-3/sp800-63.html. https://doi.org/10.6028/NIST.SP.800-63-3. June 2017.

NIST SP 800-63-A (2017). *Digital Identity Guidelines: Enrollment and Identity Proofing Requirements*. NIST (National Institute of Standards and Technology) Special Publication 800-63-A. Online: https://pages.nist.gov/800-63-3/sp800-63a.html. https://doi.org/10.6028/NIST.SP.800-63a. June 2017.

NIST SP 800-63-B (2017). *Digital Identity Guidelines: Authentication and Lifecycle Management*. NIST (National Institute of Standards and Technology) Special Publication 800-63-B. Online: https://pages.nist.gov/800-63-3/sp800-63b.html. https://doi.org/10.6028/NIST.SP.800-63b. June 2017.

NIST SP 800-63-C (2017). *Digital Identity Guidelines: Authentication and Lifecycle Management*. NIST (National Institute of Standards and Technology) Special Publication 800-63-C. Online: https://pages.nist.gov/800-63-3/sp800-63c.html. https://doi.org/10.6028/NIST.SP.800-63c. June 2017.

PCI DSS (2016). *Requirements and Security Assessment Procedures, Version 3.2*, April 2016. Payment Card Industry Security Standards Council, Payment Card Industry Data Security Standard.

Tao, H. (2006). *Pass-go, a new graphical password scheme*. Master's thesis, University of Ottawa.

Tao, H. and Adams, C. (2008). *Pass-go: A proposal to improve the usability of graphical passwords*. International Journal of Network Security, 7(2):273–292.

Uellenbeck, S., Dürmuth, M., Wolf, C., and Holz, T. (2013). Q*uantifying the security of graphical passwords: The case of android unlock patterns*. In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, pages 161–172. ACM.

Van Heerden, R.P. and Vorster, J.S. (2009). *Statistical analysis of large password lists, used to optimize brute force attacks*. Proceedings of the 4th International Conference on Information Warfare and Security, Cape Town, South Africa, pp111-128.

Vorster, J.S. (2014). *A Framework for the Implementation of Graphical Passwords*. M.Sc Dissertation. University of Liverpool.

Vorster, J.S. and Van Heerden, R.P. (2015). *A Study of Perceptions of Graphical Passwords*. Journal of Information Warfare, 14(3).

Zhao, H. and Li, X. (2007). *S3pas: A scalable shoulder-surfing resistant textual-graphical pass-word authentication scheme*. In Advanced Information Networking and Applications Workshops, 2007,AINAW'07. 21st International Conference on, volume 2, pages 467–472. IEEE.