Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 11, November 2013, pg.166 – 174

RESEARCH ARTICLE

Evolution of Authentication Mechanisms

Ch. Krishna Prasad[1], A. Ramesh Babu[2]

[1]*CSE, Anurag Engineering College, JNTU, Hyderabad, Andhra Pradesh, India*

[2]*CSE, Anurag Engineering College, JNTU, Hyderabad, Andhra Pradesh, India*

[1] krishna_chkdd1978@yahoo.co.in; [2] akkivarapuramesh@gmail.com

Abstract— This paper presents an evolution of authentication mechanisms from textual passwords to graphical authentication mechanisms. For security concerns password must be protected from the unauthorized users. Authentication and authorization is necessary for essential data, this requires authentication mechanisms. An important goal for authentication is user has easy to memorize their passwords and recall passwords either textual authentication or graphical authentication.

Keywords— authentication; authorization; security; graphical passwords; textual passwords

I. INTRODUCTION

The authentication process can be described as three phases: identification, authentication, and authorization. Users must first make some claim of their identity, provide evidence to substantiate this claim, and if successfully authenticated by the system, access rights are granted to the user. We classify authentication mechanisms according to the following categories, primarily based on Renaud's model.

1)*Something you know (recall):* A secret is shared between the user and the sys-tem. Users must recall and correctly enter their secret to authenticate them- selves. Anyone who knows or guesses the secret will also be able to authenticate as the original user. Examples include passwords and PINs (Personal Identification Numbers).

2)*Something you recognize (recognition):* The user and the system share a secret. The system provides cues and the user must correctly recognize the secret. Anyone able to recognize the secret will be able to authenticate as the original user. Graphical passwords where users must recognize pre-selected images from a set of decoys fall into this category. Cued recall systems combine recall and recognition. Users must recognize the cue presented by the system and then use this cue to recall the secret shared with the system.

3)*Something you are (static biometrics):* Biometrics measures some unique physical characteristic of the user. These are more difficult to forge than the first two categories but introduce additional concerns. They may require specialized

Equipment, are difficult or impossible to change if compromised, and have potential privacy implications (e.g., they may make it difficult to create different identities for various purposes, and they enable organizations to cross-reference information about a user). Static biometrics includes fingerprint, iris, and facial scans, among others.

4)*Something you do (behavioral biometrics):* Some unique behavioral characteristic of the user can also be measured. Users authenticate by repeating the required action. Examples include handwritten signatures and keystroke dynamics.

5)*Something you have (tokens):* Users must carry a token to be presented for authentication. Anyone who gains access to the token will be able to authenticate as the original user. These are often combined with a PIN or password to offer some protection in case the token is lost or stolen. A smart card, i.e., a card with embedded microprocessor chip, is an example of a token used for authentication.

6)*Where you are (location-based authentication):* Location information can be used to determine if a user is attempting to authenticate from an approved location. This is typically used as a secondary check to identify suspicious login activities. Approved locations may be specific, such as a user's office, or more general, such as identifying the city or country of origin.
Authentication can be divided into three categories: Token based authentication, Biometric based authentication and Knowledge based authentication.

Token based techniques, such as key cards, bank cards and smart cards are widely used. Many token-based authentication systems also use knowledge based techniques to enhance security. For example, ATM cards are generally used together with a PIN number.

Biometric based authentication techniques, such as fingerprints, iris scan, or facial recognition, are not yet widely adopted. The major drawback of this approach is that such systems can be expensive, and the identification process can be slow and often unreliable. However, this type of technique provides the highest level of security.

Knowledge based techniques, is the most widely used authentication techniques and include both text-based and picture-based passwords

II. TEXTUAL PASSWORD AND PASSWORD PROBLEM

Despite the large number of options for authentication, text passwords remain the most common choice for several reasons. Text passwords are easy and inexpensive to implement, and are familiar to most users. Passwords allow users to authenticate themselves without violating their privacy, as biometrics could, since users can select passwords that do not contain personal information. And finally, passwords are portable since users simply have to recall them, as opposed to tokens which must be carried.

However, text passwords also have a number of the inadequacies from both security and usability viewpoints, such as being difficult to remember and being predictable if user-choice is allowed Passwords are only secure if they are difficult for attackers to guess, yet are only usable if users can remember them. The \password problem" is defined as the current situation where many passwords are either weak-and-memorable or secure- but-difficult-to-remember, despite the need for secure and memorable passwords. Systems sometimes provide on-screen advice on how to create more secure passwords (e.g., select something memorable that would be difficult for others to guess), give feedback about password choice (e.g., with a password strength meter), or force users to create passwords that comply with specific system-defined rules (e.g., the password must include both letters and numbers). Despite these strategies, users often select weak passwords that are predictable and are easy for attackers to guess. This occurs partially because users misunderstand the advice or requirements, underestimate the risks, and because limitations of human memory mean that they must employ coping mechanisms in order to reduce the burden of remembering so many passwords. These coping mechanisms may include reusing passwords across several accounts, using predictable alphanumeric combinations, or storing passwords in an easily accessible, insecure location .Although they have appeal- in characteristics, only limited success has been achieved through encouraging the use of passphrases (passwords are longer phrases) or mnemonic passwords (passwords are abbreviated from a longer word or phrase, for example by using the first letters of the words in a phrase, or including common character substitutions such as “I<3c@s" for \I love cats"). At least in their basic form, both suffer from predictability problems because users choose common character substitutions or well- known phrases. Such approaches also do not mitigate the problem of remembering which password corresponds to which account when users have multiple accounts.

Furthermore, phishing and other social engineering attacks on passwords have increased dramatically over the past few years since text passwords are easy for users to unintentionally reveal to attackers, complicating matters further.

A proposed solution to these password problems is to use password managers. One class of these managers maps easy to remember (weak, low-entropy) user passwords onto stronger passwords (more resistant to guessing attacks), and may also

generate site-specific passwords (protecting against some phishing attacks). Pass-word managers exist in different formats: stand-alone applications, browser plug-ins, and browser scripts.

We investigated two password managers .Our work shows that while the idea of password managers is promising, in their present form these systems have a number of usability problems that lead to decreased security. We conducted a user study of two browser plug-ins: PwdHash and Password Multiplier. We found that the most significant problems arose from users having inaccurate or incomplete mental models of the software. Our study revealed many interesting misunderstandings, such as users who reported that a task was easy even when they were unsuccessful at completing that task, and users who believed that their passwords were being strengthened when in fact they had failed to engage the appropriate protection mechanism. Such \dangerous errors" are especially concerning because they may have serious security consequences. Our findings also suggest that in the absence of additional education or other means of encouragement, ordinary users would be reluctant to opt-in to using these managers: users were uncomfortable with relinquishing control" of their passwords to a manager, did not feel that they needed the password managers, and did not believe that these password managers provided greater security.

Text passwords are a type of knowledge-based authentication, where users must prove knowledge of some secret. Graphical passwords are an alternative type of knowledge-based authentication. In graphical passwords, images or visual representations are used instead of alphanumeric characters. The premise behind graphical passwords is that humans have better memory for images than text, so this may be a way of devising more memorable passwords.

A. *Password Problem (Attack Models)*

Many strategies exist for attacking authentication systems. No system offers perfect security; therefore schemes must be evaluated according to their vulnerabilities. For a particular attack strategy, it is possible to compare the susceptibility of different schemes. In practice, the likelihood of such attacks cannot be accurately predicted since it is unknown what attackers may target next. We now identify several possible attack models for password systems.

1) *Dictionary Attack:* In a dictionary attack, a list of likely passwords is compiled based on knowledge or assumptions of typical user behaviour. Entries in the dictionary can be further prioritized to test passwords with higher probability of success first (if these probabilities can somehow be calculated or predicted), increasing chances of quickly finding a match. Dictionary attacks can lead to efficient password guessing because users are likely to select from a relatively small and predictable password space. Recent research suggests that dictionary attacks remain a serious on-going threat, although ex-act statistics are not widely available since most organizations do not reveal such breeches in security.

In an online dictionary attack, interaction is required with the live system; usually each password is entered in turn to see if login is successful. The success of this type of attack can be reduced by limiting the number of incorrect login attempts allowed by the system (before locking the system from all further login attempts) for a particular user account. However, in multi-account attacks, attackers may target many accounts on the system instead of a specific account, and for example try several guesses on each of many different accounts, increasing the chances of success on at least some accounts. Furthermore, there is a usability cost to locking accounts after a number of incorrect attempts since legitimate users who simply forgot their password may also be locked out; this can also be used to launch an effective denial-of-service (DoS) attack against users by purposefully entering incorrect passwords and locking out accounts. In an offline dictionary attack, attackers must first gain access to some verifiable text (such as the hash of user's password) and do not need to go through the live system to determine if a guess is correct. Schemes that are vulnerable to offline attacks are at a higher risk than those requiring online verification because work can be done behind the scenes and trial guesses can be processed much more quickly. Hashing and salting can be used to slow offline attacks.

Hashing encodes passwords using a one-way cryptographic hashing algorithm; only the result of the hashing operation is retained and stored by the system. To verify if a login attempt is successful, the system (or attacker) hashes the candidate password and compares the result with the stored password hash. Salting concatenates a string of characters to a password before hashing it for storage by the real system. This salt is user-specific and stored in a manner accessible to the system, along with the hashed password, so that it can be concatenated with the user's input password during login. The resulting string is hashed and compared for a match against the stored hash. This effectively forces attackers in an offline attack to compute the hash for each candidate password guess on a per-user basis. Password cracking tools, such as John the Ripper, are readily available to automate offline dictionary attacks (these tools or their dictionaries may also be modified for use in online dictionary attacks). John the Ripper takes hashed passwords and compares them to lists of potential passwords that it hashes in the same format as the passwords being examined, in an attempt to find matches. When matches are found, the program reports the plain text passwords to the attacker.

2) *Exhaustive (brute-force) Attack:* Exhaustive attacks can be executed in a similar manner to dictionary attacks, except that every possible password permutation is generated and used to attack the real passwords. In a more sophisticated attack, these permutations may also be prioritized in order of decreasing probability of being selected by users, if such probabilities are somehow predictable. Like dictionary attacks, exhaustive attacks can be launched either online or offline. The advantage to this type of attack is that with enough time and computing power, a match will be found (unless an online attack is detected and stopped before the list is exhausted), but with large password spaces it may not be feasible to search the entire space. In contrast to a dictionary attack, an exhaustive attack offers better coverage but requires more time or processing power.

3) *Shoulder-surfing:* Shoulder-surfing refers to attackers acquiring knowledge of a particular user's credentials through direct observation, or through external recording devices such as video cameras, while the legitimate user enters the information. Availability of high-resolution cameras with telephoto lenses and surveillance equipment make shoulder-surfing a real concern if attackers are targeting specific users and have access to the same geographic location as these users. This is especially problematic in public environments, but may not be as serious a threat in other more private environments.

4) *Phishing:* Phishing attacks involve tricking users into entering their credentials (username, password, credit card numbers, etc.) at a fraudulent website that is masquerading as a legitimate site. Users normally reach these phishing websites through spam email enticing users to click on an embedded link that directs them to a website designed to look like a site for which they have a legitimate account. When users attempt to log in, attackers record the user's credentials and subsequently use them for fraudulent purposes.

5)*Social Engineering*: Social engineering includes any technique used to trick people into divulging their credentials or private information to untrustworthy parties. Phishing is an example of social engineering using email and websites, but social engineering can also be done using other means, such through as phone calls claiming to be from the user's bank, credit card Company, or tech support. It is often easier to obtain a password or credentials from the legitimate user than trying to break into a system by other means.

6)*Malware:* Malware (i.e., malicious software) includes any unauthorized soft-ware that is installed without a user's informed consent. Such software has a malicious purpose, and can include viruses, worms, and ActiveX or JavaScript components. One category of malware is intended to gather confidential information, including user credentials, from the computer on which it is installed. For example, key-loggers record keyboard input, while mouse-loggers and screen scrapers capture mouse actions and the contents of screen memory, then either send this information back to the attacker or otherwise allow attackers to retrieve it.

III. GRAPHICAL PASSWORD

To overcome the problems and attacks from the unauthorized users, the graphical password authentication accepted. Graphical passwords can be grouped into five general categories based on the type of cognitive activity required to remember the password recall, recognition, cued recall, persuasive cued recall and prominent cued recall.

A. *Recall*

Graphical passwords requiring pure recall are most similar to text passwords because users must remember their password and reproduce it without any cues from the system. This is a difficult memory task and users sometimes devise ways of using the interface as a cue even though it is not intended as such. For example, we have evidence that users often include the name of the system as part of their text passwords.

1)Draw-A-Secret (DAS):

With DAS, users draw their password on a 2D grid using a stylus or mouse. The password is composed of the coordinates of the grid cells that the user passes through while drawing. A drawing can consist of one continuous pen stroke or several strokes. To log in, users repeat the same path through the grid cells. The theoretical password space is determined by the coarseness of the underlying 2D grid and the complexity of the images. A coarser grid helps with usability, while a finer grid increases the size of the password space. To date, the system has only been user tested through paper prototypes (but see also a similar system, Pass-Go, below), so it is difficult to get an accurate analysis of its usability or security. Nali and Thorpe asked 16 participants to draw 6”doodles" and 6 “logos" on 6x6 grids. These drawings were visually inspected for symmetry and number of pen strokes. They found that participants tended to raw symmetric images with few pen strokes and tended to place their drawing approximately in the center of the grid. This preliminary study has several limitations: users were not told that their drawings were \passwords",

users did not have to re- produce their drawings at any point, and data was collected on paper so users did not have to draw using the computer. Consequently, no usability data (login times, success rates, etc.) was collected for the scheme. Van Oorschot and Thorpe categorized DAS passwords into password classes based on characteristics such asymmetry and number of pen strokes. Using this classification, they show that a large number of passwords from the paper-based study and a subsequent study on a similar scheme (Pass-Go, discussed later in this section) fall within these predictable categories, which could help attackers identify candidate passwords with high probability of success and launch efficient dictionary attacks. The theoretical password space for DAS depends on the number of cells in the grid and the password length (calculated as the number of coordinate pairs defining the path of the password Since passwords are based on precise coordinates, DAS passwords may be hashed for storage (i.e., the system can use the hash of a password to verify a user-entered password). However, there is a many-to-one mapping from user-drawn passwords to system-encoded passwords (i.e., passwords in the theoretical password space); for example, all doodles drawn entirely within one grid square are equivalent to a dot. Although not discussed in the publications about DAS, we now consider other security characteristics. DAS would be susceptible to shoulder-surfing; an attacker would need to accurately observe only one login for the entire password to be revealed. Phishing and social engineering attacks may also be of concern since users may be able to describe their password by verbalizing the path through grid squares or by showing a sketch of the password. Although this would need to be verified through user testing, we suspect that DAS password attacks may be personalized to some extent; that is, someone familiar with the user may have a higher probability of guessing the user's password. For example, some users may choose to draw the initials of their name. As is the case for all recall-based schemes in this section, phishing attacks can easily be mounted. A phishing website simply has to copy the login page from the legitimate site, including the area for drawing the graphical password (a 5x5 grid in the case of DAS). Once users enter their username and password, this information can be utilized by attackers at the legitimate sites. Furthermore, all recall-based schemes in this section, including DAS, are vulnerable to malware attacks based on screen scrapers. They may also be susceptible to mouse-loggers, if an attacker is also able to identify the position of the password entry grid on the screen through other means. Recently, Dunphy and Yan added background images to DAS to encourage users to create more complex passwords. Their study compared the new BDAS with DAS using paper prototypes. It shows that the background image reduced the amount of symmetry and led to longer passwords that were similarly memorable to the weaker DAS passwords. They did not investigate whether the background images introduced other types of predictable behaviour such as targeting similar areas of the images or image-specific patterns.

2)*Passdoodle:* Passdoodle is similar to DAS, allowing users to create a freehand drawing as a pass- word, but without a visible grid. The use of additional characteristics such as pen colour, number of pen strokes, and drawing speed are suggested by the authors to add variability to the doodles. Goldberg report on a small paper-based prototype study of Passdoodle and found that users often remembered their final drawing, but they made mistakes in recalling the number, order, or direction of the pen strokes. In a lab study, 10 users created their doodle by tracing it with their finger on a touch screen. Users repeated the trace several times. This data was used as training for the recognition algorithm and it was found that similar input could be accurately interpreted as similar. No further usability or security analysis has been reported. Later, Govindarajulu and Madhvanath separately proposed a web-based pass- word manager where a “master doodle" was used instead of a master password. In their 10-participant user study, they collected Tamil language character samples using TabletPCs and PDAs. Using only one initial doodle as the master doodle, they used handwriting recognition techniques to evaluate whether the subsequent doodles were correct and reported 90% accuracy with one of the handwriting recognition techniques. All three Passdoodle studies focus on the users' ability to recall and reproduce their doodles and on the matching algorithms used to accurately identify similar entries. None of the studies look at usability metrics such as login times or success rates. During password creation, however, Passdoodles would likely require training of the recognition algorithm to build an accurate model of the password. Although no security analysis has been reported, we provide here a preliminary evaluation comments based on our understanding of the scheme. Shoulder-surfing would be possible with Passdoodle and accurately observing one login would be sufficient to learn the password. However, reproducing the drawing may be difficult and would depend on which measures (such as drawing speed) are used by the recognition algorithm. We expect that Passdoodle would be susceptible to the same types of predictability seen with DAS (symmetry and short passwords) and as such successful dictionary attacks may be possible. As with DAS, some users are likely to choose personally identifiable passwords that can be guessed by someone who knows the user. It would likely be difficult to accurately describe a Passdoodle password since there is no visible grid to act as a guide, although it may be possible to sketch and share such passwords. Passdoodle passwords (the drawings themselves) would likely need to be stored in a manner accessible to the system, as opposed to hashed, since the recognition algorithm must allow for various approximations of the original password.

3)*Pass-Go:* Tao's Pass-Go was named for the Chinese board game of Go which consisted of strategically placing tokens on the intersections of a grid. In Pass-Go, users draw their password on a grid, except that the intersections are used instead of grid squares. Visually, the user's movements are snapped to grid-lines and intersections so that the drawing is not impacted by small variations in the trace. Users can choose pen colours to increase the complexity of their drawing. Results of a large field study

showed that login success rates were acceptable (as determined by the study's authors) at 78%, but no login times were reported. Users chose more complex pass-words than with DAS, although a large number of passwords were symmetrical and would be susceptible to attack. The theoretical password space of Pass-Go is larger than for DAS, in part because of a finer grid (more squares), and also because Pass-Go allows for diagonal movement while DAS only permits horizontal and vertical movements. Pen colour was used as an additional parameter and the authors suggest using a finer grid to further increase the theoretical password space. Dictionary attacks may be less effective than DAS since it is reported that users selected longer passwords and used colour; both add variability to passwords. Interpreting other aspects of security, Pass-Go is similar to DAS in terms of shoulder-surfing, phishing, social engineering, and personalization. A similar scheme was proposed by Orozco. It uses a haptic input device that measures pen pressure while users draw their password. They suggest that this may help protect against shoulder-surfing since an observer would have difficulty distinguishing variances in pen pressure. Results of their user study, however, show that users applied very little pen pressure and hardly lifted the pen while drawing, so the use of haptics did not increase the difficulty of guessing passwords.

B. *Recognition*

Several theories exist to explain the difference between recognition and recall memory, based on whether these are two unique processes or whether they are similar and differ only in their retrieval difficulty. It is generally accepted, however, that recognition is an easier memory task than recall. In recognition-based graphical password systems, users typically memorize a portfolio of images during password creation and then must recognize their images from among decoys to log in. Humans have exceptional ability to recognize images previously seen, even if those images were viewed very briefly. Several recognition-based graphical password schemes have been proposed in recent years. The most prominent systems available in the literature are described below.

1)Deja Vu: In Deja Vu, users select and memorize a subset of images from a larger sample to create their portfolio. To log in, users must recognize images belonging to their pre-defined portfolio from a set of decoy images; in the test system, a panel of 25 images is displayed, 5 of which belong to the user's portfolio. Users must identify all of images from their portfolio and only one panel is displayed. Images of \random art" are used to make it more difficult for users to write down their password or share it with others by describing the images from their portfolio. The authors report that a fixed set of 10000 images is sufficient, but that \attractive" images should be hand-selected to increase the likelihood that images have similar probabilities of being selected by users. A 20-participant user study showed that although slower than traditional text passwords or PINs, users could more accurately remember their Deja Vu password one week after password creation. Users took an average of 45 seconds to create their password. They took an average of 32 seconds to log in immediately after password creation with 100% success rate, and then took an average of 36 seconds to log in a week later, achieving a 90% success rate at that time. This type of system is not suitable as a replacement for text passwords because with a reasonably sized set of images for usability, the theoretical password space is only comparable to 4 or 5 digit PIN. The authors claim that Deja Vu is resistant to dictionary attacks because few images in their user study were selected by more than one user; however, this claim has not been rigourously tested. Deja Vu is slightly more shoulder-surfing resistant than previously described schemes since only a portion of the user's portfolio is revealed during a login attempt. Several logins would need to be observed to identify all of the images in a user's portfolio. Participants in the user study found it difficult to describe the images in their portfolio and users who had the same image gave different descriptions from each other. This provides evidence that it may be difficult for an attacker to gather enough information from a social engineering attack to log in, at least if the attacker relies on the user to verbalize the password. Similarly, it is likely to be difficult to identify images belonging to a particular user based on knowing other information about that user; it is, however, possible that users select images that include their favourite colour. Screen scraping malware could record Deja Vu passwords; however, multiple logins would need to be observed before attackers learn all of the images in the user's portfolio. Phishing attacks are more difficult with recognition-based systems such as Deja Vu because the system must present the correct set of images to the user before password entry. This can be accomplished with a MITM attack where the phishing site relays information between the legitimate site and the user in real-time. In this case, the phishing site would get the user to enter a username, pass this information to the legitimate site, retrieve the panel of images from the legitimate site and display these to the user on the phishing site, then relay the user's selections to the legitimate site; thus the attacker gains access to the user's account on the legitimate site. This is a more sophisticated attack than phishing attacks for recall-based schemes, requiring more effort on the part of the attacker. A similar type of MITM attack can be launched against all of the recognition-based schemes discussed in this section. Furthermore, Deja Vu requires that identifiers for a user's portfolio images be stored in a manner accessible to the system so the correct images can be displayed during login. This means that passwords cannot be hashed for storage. This is true for all recognition-based systems described in this section.

2) *PassFaces / Faces*: In PassFaces users pre-select a set of images of human faces. During login, they are presented with a panel of candidate faces and have to select the face belonging to their set from among decoys. This process is repeated several

times with different panels, and users must perform each round correctly in order to successfully authenticate themselves. C. Story: Story was proposed by Davis as a comparison system for PassFaces. In Story, users first select a sequence of images for their portfolio. To log in, users are presented with one panel of images and must identify their portfolio images from among decoys. The images contained everyday objects, places, or people. Story also introduced a sequential component by requiring that users select their images in the correct order. To help with memorability, users were instructed to mentally construct a story to connect the images in their set. In the test system, a panel contained 9 images and a user's password consisted of a sequence of 4 images selected from within this panel. Story was user tested along with Faces as part of the same field study by Davis. They found that user choices in Story were more varied but still displayed exploitable patterns such as differences between male and female choices. Users had more difficulty remembering their Story passwords (85% success rate) and most frequently made ordering errors. Surveys with participants revealed that they were unlikely to have formulated a story as a memory aid, despite the designers' intentions, which may explain the high number of ordering errors (this might possibly be overcome with different instructions or further experience with the system). Passwords since the images in the password are in a specific sequence. Davis found that patterns in user choice existed in Story, indicating that it is likely possible to build an attack dictionary that accounts for these preferences. Also, since differences were seen between males and females, and it is likely that users choose images of things they like, a targeted attack may also succeed. Story is vulnerable to shoulder-surfing since the entire password is revealed with every login, especially if the mouse is used as an input device. With respect to social engineering, attackers would likely be more successful at getting users to verbalize their Story passwords than those of PassFaces or Deja Vu since a panel will include images of various everyday objects and scenes. Similarly to other recognition-based schemes, MITM attacks are possible and portfolio images must be stored in a manner accessible to the system. Story is also vulnerable to malware attacks using screen scraping software.

C. *Cued Recall*

In cued-recall systems, the system provides a cue to help trigger the user's memory of the password (or portion thereof). This feature is intended to reduce the memory load on users and is an easier memory recall task than pure recall. Tulving and Pearl-stone explain that items in human memory may be available but not accessible for retrieval. Their results show that previously inaccessible information in a pure recall situation can be retrieved with the aid of a retrieval cue. Ideally, the cue in an authentication system will be helpful only to legitimate users and not to attackers trying to crack a given password.

Several of the cued-recall graphical password schemes surveyed require that users remember specific details within the images (or 3D environment). This is a different memory task than simply recognizing the image as a whole. Hollingworth and Henderson show that people also retain accurate, detailed, visual memories of objects to which they previously attended in visual scenes; this suggests that users may be able to accurately remember specific parts of an image as their password if they initially focused on them. We now provide a survey of graphical password systems that employ cued-recall to facilitate password memory.

1)*3D graphical password:* Alsulaiman and El Saddik proposed a 3D scheme where users navigate a 3D world and perform a sequence of actions interpreted as their password. Much like the 2D graphical passwords in this section, the 3D environment acts as a cue to prompt users to perform their actions,

2) *Inkblot Authentication*: Inkblot Authentication uses images as a cue for text password entry. The system presents computer generated “inkblots" and users respond by entering text characters that match those earlier selected when the password was created. During password creation, users are shown a series of inkblots and asked to type in the first and last letter of the word/phrase that best describes the inkblot. These pairs of letters become the user's password. The inkblots are played, in order, as cues during login and users must enter each of their 2-character responses. The authors suggest that with time, users would memorize their password and would no longer need to rely on the inkblots as cues.

3)*Passlogix*: In his scheme, a system administrator prepares an image by defining the perimeter of objects within the image (”tap regions"), typically along the outlines of the objects in the scene. Users select a sequence of these pre-defined objects as their password by clicking on each object.

4)*PassPoints:* PassPoints is an extension of Blonder's click-based graphical passwords. During password creation, users are presented with an image and select a sequence of any 5 click-points (pixels) on this image by clicking on them with a mouse. During login, re-entry of the click-points must be accurate to within some system-specified tolerance and in the correct order

5)*Cued Click Points (CCP):* Cued Click Points was designed to reduce patterns and to reduce the usefulness of hotspots for attackers. Rather than five click-points on one image, CCP uses one click-point on five different images shown in sequence. The next image displayed is based on the location of the previously entered click-point, creating a path through an image set. Users select their images only to the extent that their click-point determines the next image. Creating a new password with different click-points results different image sequence.

6)*Persuasive Cued Click Points (PCCP):* By adding a persuasive feature to CCP, PCCP encourages users to select less predictable passwords, and makes it more difficult to select passwords where all five click-points are hotspots. Specifically, when users create a password, the images are slightly shaded except for a viewport. The viewport is positioned randomly, rather than specifically to avoid known hotspots, since such information might allow attackers to improve guesses and could lead to the formation of new hotspots. The viewport's size is intended to offer a variety of distinct points but still cover only an acceptably small fraction of all possible points. Users must select a click-point within this highlighted viewport and cannot click outside of the viewport, unless they press the shuffle button to randomly reposition the viewport. While users may shuffle as often as desired, this significantly slows password creation. The viewport and shuffle button appear only during password creation. During later password entry, the images are displayed normally, without shading or the viewport, and users may click anywhere in the image.

7)*Sound Signature in PCCP*: By adding sound signature to images easily recognize and recall passwords. Each click point can be store an image and a click point as sound. Each click point has a sound clip. But, here the password length can be increased. And also space for the password can be maximized

CONCLUSIONS

An important usability goal for knowledge-based authentication systems is to support users in selecting passwords of higher security, in the sense of being from an expanded effective security space. Textual passwords are higher usability than the graphical authentication mechanisms based on the user perspective in some of the scenarios. PCCP uses persuasion to influence user choice in click-based graphical passwords, encouraging users to select more random, and hence more difficult to guess, click-points. Overall, the current graphical password techniques are still in the early stages. Much more research and user studies are needed for graphical password techniques to achieve higher level of security and usefulness.

ACKNOWLEDGMENT

We thank all the faculty members of Department of CSE for valuable suggestions on the authentication mechanisms. We also thank the reviewers for valuable feedback.

REFERENCES

[1] A. S. Patrick, A. C. Long, and S. Flinn, "*HCI and Security Systems*," presented at CHI, Extended Abstracts Workshops). Ft. Lauderdale,Florida, USA.,2003.

[2] R. N. Shepard, "*Recognition memory for words, sentences, and pictures*" Journal of Verbal Learning and Verbal Behavior, vol. 6, pp. 156-163, 1967.

[3] J. Thorpe and P. C. v. Oorschot, "*Towards Secure Design Choices for Implementing Graphical Passwords*" in *Proceedings of the 20th Annual Computer Security Applications Conference*. Tucson, Arizona, 2004.

[4] I. Jermyn, A. Mayer, F. Monrose, M. K. Reiter, and A. D. Rubin, "*The Design and Analysis of Graphical Passwords*" in Proceedings of the 8th USENIX Security Symposium, 1999.

[5] W. Jansen, S. Gavrila, V. Korolev, R. Ayers, and R. Swanstrom, "*Picture Password: A Visual Login Technique for Mobile Devices*" National Institute of Standards and Technology Interagency Report NISTIR 7030, 2003.

[6] Davis, D., Monrose, F., and Reiter, M.K. *On User Choice in Graphical Password Schemes.* USENIX Security 2004.

[7] Wiedenbeck, S., Waters, J., Birget, J.C., Brodsky, A., and Memon, N. *Authentication Using Graphical Passwords: Effects of Tolerance And Image Choice.* SOUPS 2005.

[8] Chiasson, S., van Oorschot, P.C., Biddle R. *Graphical Password Authentication Using Cued Click-points.* ESORICS 2007.

[9] Cranor, L.F., S. Garfinkel. Security and Usability. O'Reilly Media, 2005.

[10] Chiasson, S., Elizabeth Stobert, Alain Forget,, Paul C. van Oorschot, Robert Biddle. *Persuasive cued click points: Design, implementation ans evaluation of a knowledge based authentication mechanism*. IEEE.,VOL. 9, NO. 2 ,2012.

[11] J. Padhye, V. Firoiu, and D. Towsley, "*A stochastic model of TCP Reno congestion avoidance and control*" Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.

[12] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 1997.

Authors' Profile:

First Author – Ch.Krishna Prasad, M.Tech. Associate Professor in the Department of Computer Science Engineering, Anurag Engineering College, Kodad, Andhra Pradesh. He is having more than 10 years of teaching experience. Areas of interests are Data Mining, Network security and Computer Networks.

Email address: krishna_chkdd1978@yahoo.co.in

Mobile No: +91-9848684256

Second Author – A.Rameh Babu, M.Tech. Assistant Professor in the Department of Computer Science Engineering, Anurag Engineering College, Kodad, Andhra Pradesh. He is having more than 5 years of teaching experience. Areas of interest are Computer Networks, Network Security and Cloud Computing.

Email Address: akkivarapuramesh@gmail.com

Mobile No: +91-9010910075

