Modelling the Security of Recognition-Based Graphical Password Schemes

Rosanne English

Submitted in fulfilment of the requirements for the Degree of Doctor of Philosophy

School of Computing Science
College of Science and Engineering
University of Glasgow
September 2012

Declaration

I declare that this thesis was composed by myself and that the work contained therein is my own, except where explicitly stated otherwise in the text.

(*Rosanne English*)

Abstract

Recognition-based graphical passwords are a suggested alternative authentication mechanism which have received substantial attention in research literature. The literature often presents new schemes, usability studies or propose countermeasures for specific attacks. Whilst this is beneficial, it does not allow for consistent comparison of the security of recognition-based graphical password schemes.

This thesis contributes a proposed solution to this problem. Presented in this thesis are models for estimating the number of attacks required before success for four aspects of the security of a recognition-based graphical password scheme. This includes two types of guessing attacks and two types of observation attacks. These models combine to provide an overall metric of the security of recognition-based graphical password schemes.

Attacks to be incorporated into the metric were established by reviewing the literature and establishing the scope and context. The literature review allowed extraction of the variables of a recognition-based graphical password scheme which represent the scheme.

The first aspect examined was that of guessing attacks. The first guessing attack considered was random guessing, the model for this aspect was an adaption of the frequently reported mathematical model. The second guessing attack was a newly proposed attack which prioritised images from more popular semantic categories e.g. animals. The model for this attack was constructed as a further adaption of the random guessing model based on the success rates for the attack which were established by simulations which incorporated user selected images.

The observability attacks modelled were shoulder surfing and frequency attacks. The observability attack models were constructed by simulation of the attacks for a wide range of potential configurations of the recognition-based graphical password schemes. A mathematical model was fitted to the resulting data.

The final metric combined these models and was evaluated against a list of metric requirements established from relevant literature. The metric results in a consistent, repeatable, and quantitative method for comparing recognition-based graphical password schemes. It can be directly applied to a subset of schemes which allows their security levels to be compared in a way not possible previously.

Also presented are details on how the metric could be extended to incorporate other recognition-based graphical password schemes. The approach detailed also allows the possibility of extension to incorporate different attack types and authentication contexts. The metric allows appropriate selection of a recognition-based scheme and contributes to a detailed analysis of the security aspects of recognition-based graphical passwords.

Acknowledgements

I'd like to take this opportunity to thank my first supervisor, Dr. Ron Poet, for giving me the freedom to explore and research in my own way. This has made me a stronger, better researcher. I'd also like to thank my second supervisor, Dr. Karen Renaud, for her unwavering belief in me and her continuous support and understanding.

I'd like to thank all those who have shared an office with me for their understanding and companionship through the years. Heather, Wendy, Melissa and Stuart. It's a hard road and I'm grateful to have had supportive friends to have traveled it with.

I'd like to thank Anne Marie, who was always there with tea and sympathy and never any hint of an "I told you so". I couldn't ask for a better friend, I'm so grateful I sat next to you all those years ago in ExCos! I'd like to thank Stuart, who has been many things, a rock, a shoulder to cry on, and an unbiased perspective. I can't thank you enough. I'd also like to thank my family for their love, support and understanding.

A special thank you to those who reviewed this thesis and provided constructive criticism and feedback on my work over the years. In particular I'd like to thank Tim, Karen, and Ron. To the school and EPSRC I'd like to express my gratitude for my allocation of funding, without which this PhD would not have been possible.

Contents

List of Tables

List of Figures

Associated Publications

As a result of this work, there were three associated peer-reviewed publications. These were as follows:

- "Towards a Metric for Recognition-Based Graphical Password Security" , 5th International Conference on Network and System Security (NSS) , Sept. 2011
- "Measuring the Revised Guessability of Graphical Passwords" , 5th International Conference on Network and System Security (NSS) , Sept. 2011
- "The Effectiveness of Intersection Attack Countermeasures for Graphical Passwords", TrustCom, 2012 11th International Conference on Trust, Security and Privacy in Computing and Communications, June 2012

Chapter 1

Introduction

This chapter provides the motivation for this work, an introduction to user authentication, and the thesis statement.

1.1 Motivation

Motivation for this thesis primarily evolved from the following quote by Herley *et al.* who noted that "in the absence of tools to measure the economic losses and the effectiveness of new technological proposals, we expect the adoption of password alternatives to continue to be difficult to justify" [42]. This thesis focuses on contributing to the measurement of the security of one type of alternative authentication, recognition-based graphical passwords (RBGPs). RBGPs will be discussed in detail in Section 2.3, but first the process of user authentication is considered.

1.2 User Authentication

User authentication (which will be referred to as authentication henceforth) is the process during which a (human) user proves they are who they claim to be. This is achieved by a distinctive characteristic. This characteristic can differentiate one individual from another [80, Page 3]. These characteristics can be called authentication factors and are said to fall into three categories: things you know (knowledge-based authentication, e.g. a password), things you have (token-based authentication, e.g. a card or key), and things you are (physical biometrics e.g. finger prints, or retina scans) [73]. Rejman-Greene expands this by two further factors, authentication by geographical location and authentication by behavioural biometrics [67]. Geographical location is when a user provides evidence of being in a physical location to authenticate. Behavioural biometrics are similar to physical biometrics, but are based on user's behaviour. The behaviour is distinctive and unique enough to be used for authentication e.g. keystroke dynamics [36]. Brainard *et al.* also propose authentication by someone you know, called "fourth factor authentication" [8], in this situation another user electronically "vouches" for the user attempting to authenticate. This work focuses on knowledge-based authentication.

1.2.1 Passwords

Passwords are currently a frequently deployed knowledge-based authentication mechanism. One need only consider the number of passwords they have to see how pervasive they are. For the purposes of this work, an "alphanumerical password" is a password which can consist of any combination of characters from the printable ASCII set. Upper and lower case characters are considered distinct. Such a password can also be referred to as "text-based". In password authentication, a user registers a secret word (their password) to be associated with their claimed identity.

Research on user password behaviour has highlighted a number of problems. Seminal work on password behaviour was conducted by Klein [49] who collected the unix passwd files of 15,000 users. Klein tried to break the hashed passwords (to obtain their plaintext equivalent) using a number of dictionary attacks. Dictionaries were constructed based on usernames and account numbers, character sequences, numbers, place names, common names, uncommon names, myths and legends references, Shakespeare references, science fiction references, film titles, actors, and bible references amongst others. The total dictionary size was 62727 words. Klein had 4 DECstation 3100 machines which could check approximately 750 passwords per second giving a total peak processing power of 3000 tests per seconds (since not all machines were always available). Twelve CPU months using the setup described resulted in 25% of passwords being cracked with 21% guessed in the first week and 2.7% within 15 minutes. Klein's experiment showed that users were selecting English words from common use, which made the passwords easier to guess.

Since Klein's experiment, there has been further research into the issues with password use and the mechanisms employed by users to cope with these issues. Evidence has been reported regarding password forgetting, password re-use, writing passwords down, and password sharing. Adams and Sasse collected evidence of all these issues. Adams and Sasse [2] performed an examination of password habits of users. The authors report the results of a web-based questionnaire which obtained quantitative and qualitative data on user behaviors and perceptions relating to password systems. There were 139 participants, approximately half of whom were from a telecommunications organisation, and the remainder of participants were from other organisations. The questionnaire was followed by 30 in-depth interviews with a cross section of users from two organisations. The responses collected provided evidence of users having multiple passwords which resulted in password re-use, password modification, writing passwords down and password sharing. All of these behaviours potentially reduce security since it is easier to guess and capture the passwords.

There has been additional evidence that users forget passwords, as shown by Florencio and Herley [30] who conducted research to obtain quantitative information on users' password habits. In this research they created a piece of software which measured user web-based password behaviour. The software obtained over half a million users and Florencio and Herley's paper reports results on the average number of passwords, average number of accounts (per user), how many passwords they type per day, how often passwords are shared amongst sites and

how often they are forgotten. The program also obtained data on the password strength, types and lengths of passwords and how they varied by site. In this study, 2149 out of 50100 users requested a password reset over three months. The data gathered per client assumed one user per machine and so the results reported may overestimate the number of passwords etc. for a single user.

Evidence is also provided in research by Yan *et al.* regarding users forgetting passwords [105]. Yan *et al.* recruited 288 first year University students to use a system which required password authentication using different password composition policies. Each participant was assigned to one of three groups. Each group had different forms of password composition advice [105]. Only 6 users required their password to be reset, indicating a low frequency of forgetting passwords. However, these passwords were frequently used and, as highlighted by Adams and Sasse, it is light use passwords which are most often forgotten [2].

Komanduri *et al.* [51] examined the effect of password composition policies on user password behaviour by conducting a two-part online study. The study asked users to create a password (conforming to a randomly selected password policy) and fill out a survey and enter the password again. Participants were then asked to return a few days later and enter the password and fill in another survey. In the study 31% of participants wrote down the password created and 11.1% of participants forgot their password. In the survey Komanduri *et al.* also collected evidence of password re-use. 34.6% of the 5000 participants admitted to password re-use, and 17.7% admitted to modified re-use (where a previous password was manipulated by e.g. addition of a number). Both types of re-use were also evidenced in Adams and Sasse's work (reported in [2]) .

Inglesant and Sasse also established evidence of users writing passwords down by collecting information from employees a University and a financial services company. Information was collected using password diaries and interviews [46]. Users were 15 members of University staff, the remaining participants were from a financial services organisation. From the financial company 12 members of a security team and 5 HR staff were recruited. In total 32 participants were recruited. Nine of the 15 users of organisation A admitted to writing down their passwords. Whilst none from organisation B reported this, organistation B's policy allowed modification of prior passwords (where organisation A did not) and this may have contributed to memorability. It is also possible that the employees of organisation B did not admit to writing down passwords when they did.

Evidence of password re-use was collected by Dhamija and Perrig [20] who interviewed 30 participants to examine password behaviour. The results reported that users had 10-50 accounts of various forms where password authentication was required and users had from one to seven unique passwords. The number of unique passwords appears less than the number of accounts reported, however these values have a large range and it is difficult without further detail to deduce password re-use.

Brown [10] also conducted a survey of 218 students' password habits and gathered further evidence of password re-use. Password systems included systems such as e-mail access, security codes for alarms, copier machine PINs, mobile phone PINs and ATM PINs in addition to online passwords and other computer

passwords. The mean number of password systems was 8.18 (with a standard deviation of 2.18 and a range of 3-20), while the mean number of unique passwords was 4.45 (with a standard deviation of 1.63 and a range of 1-11).

Gaw and Felten [35] also provide evidence of a high number of passwords per user and password re-use. The study asked 49 student participants to report their use of passwords, counting the number of passwords for online accounts. Participants were offered two approaches. The first approach was to authenticate for any sites they were already registered with on a pre-composed list (139 websites grouped into 12 categories). The second approach was to recall as many websites as they could and authenticate for them. The second approach excluded sites already covered in the first and the participants were allowed memory aids (they were told to use "any tools that will help you recall your passwords.). The participants counted the number of passwords and repeated passwords and reported them. The results showed no significant difference in password recall where memory aids were used. There was a mean of 4.67 passwords (with a minimum of 1, maximum of 11 and standard deviation of 4.67) where users were asked to use the pre-composed list. Users then came up with a mean of an additional 7.86 passwords (with a minimum of 1, maximum of 24 and standard deviation of 7.86). In the first attempt users reported a mean of 3.06 passwords which were re-used (standard deviation of 2.19, minimum of 0 and a maximum 11) and the second approach resulted in a further mean of 3.76 passwords which were re-used (with a minimum of 0, maximum of 25 and standard deviation of 3.96). It is unclear why the values for passwords recalled with and without the pre-composed list were separated.

Notoatmodjo and Thomborson [60] examined how users mentally group their passwords and showed that password re-use was more limited in passwords with a perceived high importance. Notoatmodjo and Thomborson surveyed and interviewed 26 university students. Participants were asked to describe their passwords using length, perceived security level and difficulty of recall. As noted by the authors, results showed insufficient evidence for correlations between perceived security level and length or length and difficulty of recall. However, there was some evidence for significant correlation between perceived security and difficulty of recall. The authors measured the number of password re-use occurrences and obtained evidence that the increase in number of password re-use occurrences is related to the increase in the number of accounts. Whilst the authors noted there was some subjectivity in the results arising from the users perceptions, they showed that perceived "high importance" password groups (which were identified by the users using a 5 point Likert scale where one extreme of the scale was an unimportant account and the other was very important) had less passwords in them. There was a mean of 1.84 unique passwords in the high importance group compared to a mean of 2.78 unique passwords in the low importance group. In total there were 68 passwords in the high importance groups, of which 43 (63%) were assigned passwords which were unique, 25 (37%) were re-used. Of the 253 accounts in the low importance groups, only 82 (32%) were given unique passwords and 171 (68%) were given re-used passwords.

The study provided by Florencio and Herley [30] discussed earlier also provided evidence of password re-use. The reported results include an average of 6.5

unique passwords for each client which was used over an average of 3.9 websites. Each client had 25 accounts which required authentication (thus each password was re-used an average of 3.85 times) and passwords were entered eight times daily. Over two months, an average password was eventually used at approximately six websites, with the first four sites being visited within the first week. In contrast to the report of 6 out of 275 passwords requiring a reset by Yan *et al.* [105] (which equates to approximately 2.18%), Florencio and Herley report 2149 out of 50,100 Yahoo! users forgot their passwords over three months giving a rate of forgetting as approximately 4.28%.

Whilst the highlighted literature shows variation in the quantitative values reported there is evidence for a number of common password coping mechanisms which are employed by users. From this literature the following coping mechanisms have been identified:

- use of easily remembered passwords
- writing passwords down
- password sharing
- password re-use (including password modification)

1.2.2 Graphical Passwords

It is due to these issues that research into alternative authentication has been established. One proposed alternative knowledge-based authentication mechanism is that of graphical passwords, thought to have originated from Blonder's patent in 1996 [6]. In a graphical password, the user selects, draws or identifies part of an image (or images) as their secret instead of constructing a text-based password. The user is then challenged to recognise their image(s) from a collection of other images, reproduce their image, or recognise a specific point in an image to successfully authenticate.

This research focuses on recognition-based graphical passwords (RBGPs). For a RBGP, instead of selecting a password, the user selects a number of images called their "passimages" [12]. Instead of being challenged to recall a password, the challenge for a RBGP consists of a screen which presents a grid of images, containing at least one of the user's passimages and a number of other images called "distractor" images. To successfully authenticate, the user must recognise and select their passimage from the collection of distractor images. This process can be repeated multiple times where each screen contains a different passimage from the user's set of passimages (their passimage set). This whole process forms a complete challenge session. This is discussed in further detail in Chapter 2, but in the subsection which follows a summary of the research on RBGP memorability and usability is provided.

1.3 Graphical Passwords Memorability and Usability

Graphical passwords have been proposed to address the perceived deficiencies of alphanumerical passwords. Experiments such as those conducted by Standing [82] provided evidence that recognition of pictures was higher than recognition of English words. This is referred to as the *picture superiority effect* [82] .

Memorability has been examined for recognition-based schemes by Valentine [93], Davis *et al.* [15], and Dhamija and Perrig [20]. Valentine examined the memorability of the PassFaces scheme. 77 participants were asked to use the Passfaces scheme under one of three conditions. The first condition involved authenticating each working day for two weeks. The second condition involved authenticating approximately seven days after registration. The final condition involved authenticating one month after registration. The first two conditions had 29 participants allocated and the third had 19 participants. Various levels of successful authentication were reported. The first condition participants successfully authenticated for 99.98% of attempts. The second condition participants successfully authenticated for 83% of the first attempts, and 100% by the third attempt. The final condition reported 84% success rates for the first attempt at authentication, and 100% success rate by the third attempt.

Dhamija and Perrig examined the memorability of their Deja vu scheme [20] by conducting a user study with 20 participants over a period of one week. Participants were selected to reflect the general population (ten "novice" and ten "expert" computer users were selected). Participants were asked to perform authentication using the Deja Vu scheme, a PIN scheme and a password scheme. There were no unsuccessful authentications reported immediately after registration. One week later 90% of authentication attempts were successful for random art, 65% were successful with PIN authentication and 70% of authentication attempts were successful with password authentication. The Deja Vu scheme performed the best of the different types examined, however there was no statistical analysis of the significance of these results reported in the paper. The users were given all three types of authentication methods to carry out (graphical, PIN, and password) which could have potentially affected the results by increasing the cognitive load on the user. An alternative approach could have been an inbetween users design.

Davis *et al.* [15] examined the memorability of the Faces and Story schemes over four months. Participants were computer science and engineering students from two separate universities, and three separate classes resulting in a total of 154 users. In the study, users were randomly allocated either the Face scheme or the Story scheme. During the four month period, there were 2648 login attempts, of which 2271 (85.76%) were successful.

The usability of RBGP schemes has also been researched. For example, DeAngeli *et al.* [16] evaluate three VIP system configurations and a PIN system in terms of three aspects of usability; effectiveness, efficiency and user satisfaction. Effectiveness is measured by authentication success, efficiency is measured in reaction and entry times and user satisfaction was assessed by a Likert scale

questionnaire. Significance between the VIP schemes and the PIN system were reported using the χ^2 test. This demonstrated an increase in usability from PIN to a RBGP scheme.

Also, Brostoff and Sasse [9] examined whether PassFaces were more usable than passwords. 34 users participated in a study using the PassFaces system over a 10 week period in which recall rates for passwords were compared to that of PassFaces. The results showed significant improvement through application of an ANOVA test comparing the error rate for logins with $p = 0.001$. One possible issue is that the users had to use both a password and a PassFaces set, which may have increased the cognitive load and impacted results. This is an issue with the within groups approach taken. This could potentially have been reduced if the group was split into two and one group performed authentication using passwords for a period of time then PassFaces and the other group performed authentication the other way around. Despite this potential limitation, the work provides evidence for an increased usability of a RBGP scheme compared to a password scheme.

The graphical password concept is still relatively new (thought to have originated from Blonder's patent in 1996 [6]) and commercial applications are limited, but do exist. Confident Technologies[1] offer a range of "Image-based authentication and verification products" which include graphical authentication solutions for mobile authentication, web-based authentication and an image-based CAPTCHA alternative.

GrIDsure[2] provide a one time password solution by allowing the user to select a pattern on a grid of squares. To authenticate, the user is shown a grid of numbers (a random number in each grid location), from which the user extracts their new passcode by entering the numbers corresponding to their grid sequence. Id-Arts Ltd[3] provides a commercial enterprise implementation of the Passfaces scheme where the passimages are images of human faces.

"PassLogix"[4] is a commercially available implementation of a cued-recall scheme. In this scheme, the user is presented with a scene which has a number of objects in it. The password is then a sequence of actions performed on those objects. For example, presented with a kitchen and a number of jars and ingredients, the password may be a jar filled with a subset of those ingredients presented. Microsoft is also purported to be developing a graphical authentication mechanism for the Windows 8 operating system[5].

As discussed above, research has contributed evidence for usability and memorability of graphical passwords. Despite this, as Herley *et al.* note, alternative authentication mechanisms have not been widely adapted as alternatives to passwords [42]. One potential reason for this is as noted by Herley *et al.* who commented that without tools to measure security, it will be difficult to justify adoption of alternative authentication. As will be demonstrated in Chapter 2,

[1] `http://www.confidenttechnologies.com/products`

[2] `http://www.gridsure.com/`

[3] `http://www.realuser.com/`

[4] `http://www.passlogix.com/site/`

[5] `http://blogs.msdn.com/b/b8/archive/2011/12/16/signing-in-with-a-picture-password.aspx`

the security of RBGPs is lacking in consistent and comparable security analysis. For this reason, this research focuses on the construction of a metric to measure the resistance to attacks for RBGPs. This is reflected in the thesis statement which is presented in the next section.

1.4 Thesis Statement

The thesis statement was established as follows:

The security of a recognition-based graphical password scheme can be quantifiably measured in terms of resistance to observation and guessing attacks.

The thesis statement was further refined into five objectives as follows, each of which is addressed separately in this thesis.

Objective 1

Identify potential attacks (where the aim of the attacker is to impersonate a user and to achieve a false positive authentication) and examine current recognition-based schemes in terms of resistance to these attacks.

Objective 2

Identify a list of requirements from current security metric literature against which the metric will be assessed.

Objective 3

Establish measurements of the guessability (how easily a user's passimage set can be guessed) of a RBGP scheme by means of a mathematical model which estimates the attacks required before success for each identified guessing attack.

Objective 4

Establish measurements of the observability (how easily a user's passimage set can be observed) of a RBGP scheme by means of a mathematical model which estimates the attacks required before success for each identified observation attack.

Objective 5

Combine the measurements established into a comprehensive metric which meets the requirements identified by Objective 2.

1.5 Thesis Contributions and Publications

The main contributions of this work are summarised as follows:

- Threat model for RBGP schemes - A threat model for RBGPs was constructed from a literature review. This threat model incorporates the security aspects of guessability, observability and recordability established by De Angeli *et al.* [17] (Chapter 2).

- Construction and analysis of a new guessing attack - for RBGP schemes which permit user selection of passimages which can be categorised according to their content, a semantic ordered guessing attack (SOGA) was proposed. This attack prioritises guessing images from more popular categories in a challenge screen. The attack demonstrated a higher probability of success compared to random guessing. (Chapter 4)

- Models of attack success - models were established for guessing attacks and observation against RBGPs which allow calculation of an estimated number of attacks before success for a given RBGP scheme. (Chapters 6 and 7)

- Security metric - this allows comparison of the security of RBGP schemes in a way which is repeatable, reproducible, quantitative, objective and extensible. This is presented in Chapter 8. As noted by Henning, "a metric that is meaningful and relevant today may be less relevant tomorrow" [41]. However, the metric was designed to be extensible to allow it to be adapted to different contexts. This is discussed further in Chapter 9.

A selection of the work presented in this thesis has been peer-reviewed and published in academic conference proceedings as follows:

- "Towards a Metric for Recognition-Based Graphical Password Security" , 5th International Conference on Network and System Security (NSS) , Sept. 2011

- "Measuring the Revised Guessability of Graphical Passwords" , 5th International Conference on Network and System Security (NSS) , Sept. 2011

- "The Effectiveness of Intersection Attack Countermeasures for Graphical Passwords", TrustCom, 2012 11th International Conference on Trust, Security and Privacy in Computing and Communications, June 2012

1.6 Research Methods

A number of different research and analysis methods were utilised in this research. For details on the approaches and techniques employed, please refer to Appendix B

1.7 Overview of Thesis

The structure of this thesis can be separated into three areas; background, data gathering and modelling, and final results and conclusions. The background

information is covered in chapters 2 and 3. Chapter 2 provides background information on graphical passwords and addresses Objective 1 by presenting the results of a review of literature relating to the security of RBGPs. This culminates in a threat model. Chapter 3 presents a review of related security metric research, culminating in a list of requirements for the security metric presented in this thesis which addresses Objective 2. Chapter 5 presents a preliminary attempt at measuring the security.

The data gathering and analysis is covered in Chapters 4 , 6 and 7. Chapter 4 presents an examination of guessing attacks against RBGPs relating to Objective 3. Objective 4 is addressed by Chapter 6 which examines a selection of observation attacks. A model relating to the observation attacks is presented in Chapter 7, also related to Objective 4.

The final results and conclusions are presented in Chapters 8 and 9. The final objective is addressed in Chapter 8, where the final metric (a culmination of the measures presented till this point) is presented and evaluated against the requirements established by Objective 2. The concluding chapter is presented in Chapter 9 which discusses the results of the thesis, the contributions to research and possible future work.

The first step in this research involved consideration of the relevant background for authentication and graphical passwords, this is presented in the following chapter.

Chapter 2

Graphical Passwords Literature Review and Security Analysis

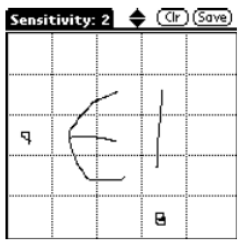
As noted by Biddle *et al.*, graphical password schemes can be split into three categories; recall, cued-recall, and recognition-based [5]. This chapter discusses each of these types of graphical password in Sections 2.1,2.2, and 2.3.

2.1 Recall Graphical Passwords

In a recall authentication scheme, the user is asked to draw an image upon registration. This image must then be replicated to provide subsequent authentication. The archetypal example for this category is the "Draw A Secret" (DAS) scheme, proposed by Jermyn *et al.* [48].

The DAS scheme presents the user with square grid of dimension G [48]. In the grid the user draws their graphical password. An example DAS password is shown in Figure 2.1. To authenticate the user must draw the same graphical password using the same order of pen strokes (a stroke or line drawn by the pen) and pen-up events (lifting the pen from the grid). The scheme records the strokes and pen-up events which then allows the division of drawings into equivalence classes. One DAS password is equivalent to another if they both have the same encoding i.e. they cross the same cells with the breaks between strokes occurring in the same places. This approach means that the DAS password must start in the same cell each time it is drawn. A DAS password is accepted if it is in the same equivalence class as the one stored.

Figure 2.1: DAS Grid



There have been a number of proposed extensions to the DAS scheme to increase the potential DAS password space and the password complexity . To increase the password space a grid selection approach was suggested by Thorpe and van Oorschot. In this approach a large fine-grained grid is presented to the user, from which they must select a smaller sub-grid in which to enter their password [89].

Dunphy *et al.* propose an extension to the DAS scheme by including a background image when users are creating their drawings, this is called Background DAS or BDAS [23]. In their work they report the results of a study involving 21 users who were randomly assigned to the control DAS group or the BDAS group. The experimenters allowed users to select the background images from a total of six alternatives. The aim was to examine the user choice distribution of the graphical passwords drawn using the background images compared to no background. Overall, an improvement in complexity of the drawings was achieved by using background images. Use of background images produced an increase in stroke count, password length and reduction of global symmetry and centering of drawings when compared to drawings created without a background. Significance was established using a one tail t-test with $t=2.948$ and $p<0.01$.

It could be argued that different background images may influence the number of strokes. This impacts complexity and potentially security through guessability. Using one image instead of the six options provided would have reduced the number of independent variables to the use of the BDAS scheme alone (and not the particular image) ensuring significance of difference was due to the BDAS scheme and not the image selected.

Another proposed improvement on the DAS scheme is presented by Gao *et al.* and is called YAGPS. Gao *et al.* claim the YAGPS scheme does not have restrictions on the position of a drawing for authentication associated with DAS i.e. the drawing need not be drawn in the same location on the grid [33]. This is achieved by using a neighborhood grid for encoding the strokes of the pen. No matter where the user starts on the grid, the positions surrounding the initial position are allocated labels of 1 through to 9 (excluding 5 which is used to denote pen up and pen down events). A similarity threshold is used to establish if the drawing is sufficiently matched to the stored password to allow successful authentication. The authors claim a potentially larger password space than the original DAS scheme, however the probable password space may not necessarily equal the theoretical password space. That is to say that users may be inclined to chose more simple passwords than the possible range as with alphanumerical passwords evidenced by Klein [49]. Also, making it possible to replicate the drawing anywhere on the grid whilst decreasing false negative authentications (where the user is incorrectly rejected), could also potentially increase false positives (where an attacker is incorrectly accepted).

A commercial example of a recall authentication mechanism is the Android pattern lock, which allows the user to lock their phone by drawing a pattern connecting dots on a grid, this is discussed by Shabtai *et al.* [78].

2.1.1 Security of Recall Graphical Passwords

In terms of security of recall graphical passwords, the password space of the DAS scheme has been examined. In the defining DAS paper, the authors consider the security of the scheme by calculating the theoretical password space. The authors define the length of a password to be the sum of the length of its component strokes, where a stroke is a line drawn on the grid starting with the pen being placed on the grid and ending with it being lifted from the grid. The grid is referenced by co-ordinate pairs, and the length of a stroke is the number of grids it passes through (i.e. the number of co-ordinate pairs). The total number of passwords (drawings) on a grid of dimension G given a maximum length L_{max} is defined as the number of passwords of each length L from 1 to L_{max}. A shorter password of length $L - l$ can have l strokes added to it to give a password of length L. Thus the number of passwords of a given length L is defined as the number of passwords of length $L-l$ multiplied by the number of strokes of length l. The number of passwords of length $L = 0$ on a grid of dimension G is defined as 1. The number of passwords of length L a grid of dimension G (denoted $P(L, G)$) is then defined recursively as the sum of all passwords of a length $L - l$ multiplied by the number of strokes of length l for each value of l from 1 to L i.e. $P(L, G) = \sum_{l=1}^{L} P(L - l, G)N(l, G)$

The number of strokes of length l on a grid of dimension G (denoted $N(l, G)$) is defined as the number of strokes of length l ending in each cell on the grid. For each cell the number of strokes of length 1 is defined as 1. The number of strokes of length l ending in cell (x, y) is then defined recursively as the number of strokes of length $l - 1$ ending in the immediately surrounding cells. This allows a calculation of the number of passwords of length L since the authors have defined $P(L - l, G)$ and $N(l, G)$ recursively.

The authors compare the password space of alphanumerical passwords with a given number of characters to the number of DAS passwords with the same number of strokes. For example, on grid of dimension $G = 5$ with $L_{max} = 8$, the log_2 of the number of passwords of length 8 is 38, where a password of length 8 with an alphabet of length 26 has entropy of $8 * log_2 26 = 8 * 4.70 = 37.6$ which is marginally less than 38.

To counteract the potentially smaller theoretical password space of the DAS scheme, Thorpe and van Oorschot suggest a grid selection before drawing the password in order to increase the password space [89]. The user selects a square on the grid, which then zooms in and displays a second grid in which they must draw their password.

Thorpe *et al.* [88] also construct a graphical dictionary to attack the DAS scheme. They consider the "memorable space" of graphical passwords (the whole set of possible passwords less passwords which are potentially forgettable). The authors postulate that, from psychological studies, people are better at recalling symmetrical images and thus would be inclined to use symmetrical images as passwords. This allows the authors to restrict the password space they are considering one smaller than the theoretical password space presented by Jermyn *et al.*. The memorable space is then further restricted by proposing that users would choose symmetry around the horizontal or vertical axis when drawing their pass-

word. To support this reasoning, the authors examine the examples presented in the original DAS paper [48] and these exhibit the symmetry discussed.

Van Oorschot and Thorpe extend the work reported in [88] by defining classes of possible graphical passwords [94]. The authors present a model for the complexity of a password. This model is based on complexity properties such as the length of the password, the symmetry of the password and the number of components i.e. the number of "visually distinct parts of the graphical password" [94]. DAS graphical passwords are then split into classes based on these complexity measures. It is from these classes that the authors then propose an attacker might build a dictionary attack. The attack is based on a multi-class graphical dictionary which consists of graphical passwords belonging to each class identified, with an increasing number of components. A dictionary attack would thus try all graphical passwords from each class with one component, then two components and so forth. Examples of the classes include class one probable passwords, which exhibit mirror symmetry about a vertical or horizontal axis in its components.

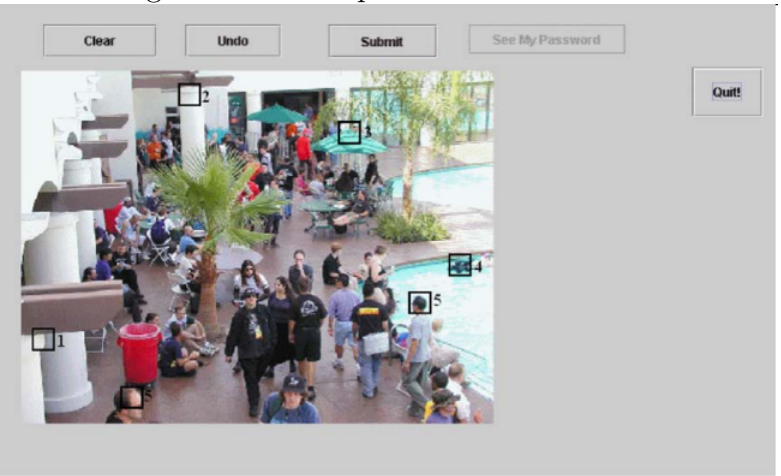
In addition to the examination of the password space and possibility of dictionary attacks, shoulder surfing for the DAS scheme has been examined by Zakaria *et al.* [106]. Zakaria *et al.* proposed and evaluated three approaches to countering shoulder surfing attacks on the DAS scheme; decoy strokes, disappearing strokes and line snaking. In the decoy strokes approach, fake strokes were drawn at the same time as the user's genuine strokes. In the disappearing strokes approach, after a stroke was completed (signified by a pen-up event) the stroke disappeared from the display. Line snaking involved a similar approach as the disappearing stroke approach, but this time strokes disappeared from the screen as they were drawn.

The first user study reported an evaluation of the efficiency of these countermeasures. The study involved 68 students, with 17 in each of the groups DAS (the control group), decoy strokes, disappearing strokes, and line snaking. Participants were asked to view an authentication session and attempt to replicate the DAS password (this included recalling the order and direction of strokes). Decoy strokes did not show a statistically significant improvement on the control setting. However disappearing strokes and line snaking showed an equivalently significant improvement on the control setting.

In the second study, usability of the disappearing strokes and line snaking approaches were examined by considering the number of authentication attempts required before success and the time taken to authenticate. Line snaking took significantly longer and significantly more attempts to successfully authenticate compared to both the DAS control and the disappearing strokes. This indicates challenges with regards to usability of the disappearing strokes countermeasure. The disappearing strokes countermeasure also took more login attempts and longer than the DAS scheme alone. Again, this shows possible issues with the usability of this approach.

Analysis of security of recall scheme has been consistent as the work reported considers the complexity of recall graphical passwords and the implications for guessing attacks. Shoulder surfing attacks have also been considered. There could be room for further examination in this instance, for example by examination of

Figure 2.2: Example Click Points Screen



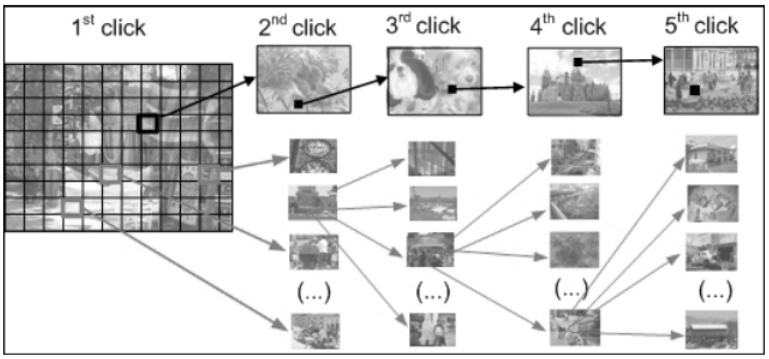
how easily guessed the drawings for a known user are.

2.2 Cued-Recall Graphical Passwords

In a cued-recall scheme the user registers by selecting a number of memorable points, "passpoints", from a provided image as their password. The PassPoints scheme proposed by Wiedenbeck *et al.* [100] demonstrates this type of graphical password scheme well. An example of an image and passpoints is shown in Figure 2.2. To authenticate the user selects the points previously indicated as their passpoints.

The PassPoints approach was further refined by Chiasson *et al.*'s Cued Click Points (CCP) scheme [14], shown in Figure 2.3. In this scheme, users are prompted to click on one point on an image for a number of images. The images in the sequence are dependent on the point selected in the prior image. This provides an alternative to remembering a number of points on one image which was noted by participants in Wiedenbeck *et al.'s* PassPoints scheme as being easier than multiple points on one image [100].

Figure 2.3: Cued-Click Points Sequence Example



2.2.1 Security of Cued-Recall Graphical Passwords

Research on the security of the cued-recall scheme "PassPoints" has primarily been conducted by by Thorpe *et al.* [90], Salehi-Abari *et al.* [74] and van Oorschot [95]. In the papers analysing the security, the authors demonstrate that the PassPoints scheme is subject to "hot spots", locations in the pictures which users are more likely to select as their passpoints.

Thorpe and van Oorschot implement human-seeded attacks on the PassPoints scheme and report the existence of hot spots in many images [90]. Human-seeded attacks were constructed by clustering observed passpoints of 43 users who were asked to select 5 distinct points on 32 to 40 separate images. The extracted hotspots were then used as areas for guessing passpoints of different users. The attacks successfully guessed 36% of user passpoints using a dictionary of size 2^{31} for user passpoints in one image and 20% with a dictionary of size 2^{33} for a second image. Also implemented is a purely automated attack based on image processing techniques, e.g. corner detection, which is reported to guess up to 30% of user passpoints using a dictionary of size 2^{35} for some cases, but under 3% in other cases. Confidence intervals are provided for the user selections which seeded the human-seeded attacks, which contributes evidence to identifying that the highlighted passpoints are hotspots.

Salehi-Abari *et al.* [74] report and evaluate different methods of automated attacks on the PassPoints scheme. The attacks employ a model of visual attention of a user which identifies areas of an image which are visually distinct from their surrounding areas. The authors propose that users will not only select passpoints in these regions, but that passpoints will be selected from distinct regions in a specific "click order". The automated attacks presented achieved success rates comparable with the human-seeded attacks reported by Thorpe and Oorschot [90] . The images and data sets used for testing were those used by Thorpe and Oorschot [90]. Dictionaries were composed of the likely passpoints established by the algorithm which combined the visual model and click order. Attacks were automated based on these points. Using one click order provided a dictionary size of 2^{33} and resulted in 21.1% of passpoints being successfully attacked (c.f. 20% using the human-seeded approach reported by Thorpe and Oorschot [90]) and 27.5% of passpoints being attacked successfully for the other image (c.f. 36% using the human-seeded approach reported by Thorpe and Oorschot [90]). Alternative approaches to establishing click orders resulted in an increased number of passpoints being successfully attacked. Though the effectiveness of the different approaches are not compared statistically in the paper, the work contributes further evidence to the hotspots issue.

Chiasson *et al.*'s Cued Click Points scheme [14] may also be susceptible to hot points. This is examined by van Oorschot *et al.* [95] who perform automated attacks on cued-click point schemes. The results report that their graph-based algorithm recovered 7% to 16% of passwords for two images where the full password space was 2^{43} using a dictionary of size 2^{26}. When they increased the size of the dictionary to 2^{35} entries the results were substantially improved to discovering 48% to 54% of passwords. Both these sets of figures are larger than the success rates of human seeded hot spot attacks described by Thorpe and Van Oorschot

in [90] which reported results between 1% and 9% on the same dataset with 2^{35} guesses.

Further work was also completed on the PassPoints scheme by LeBlanc *et al.* [53]. The authors presented a study in which participants were asked to look at the images used for the PassPoints scheme. This gaze data was then examined to determine any similarity between the gaze data and the hotspots identified in a prior study by Chaisson *et al.* [14]. The gaze data was transformed into potential passpoints by a number of different methods e.g heat map inspection. However, as noted by the authors themselves, the similarity between the gaze data and the passpoints data was minimal. However, this still presents an approach which could be used to improve a guessing attack.

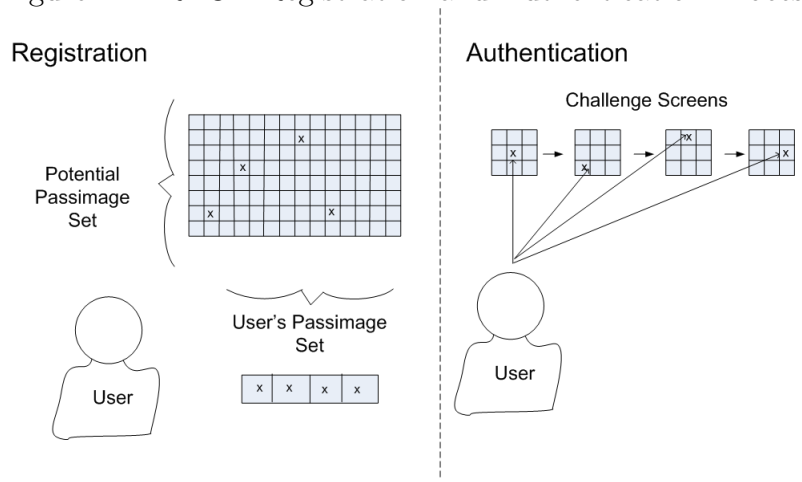
In an attempt to reduce hotspots in cued-recall schemes where selection is achieved by gazing at the desired point using eye tracking (as with [53]), Bulling *et al.* [11] introduce the concept of "saliency masks". These masks aim to cover the points on the picture which could be susceptible to hotspots. This encourages users to select other points on the image as their passpoints. There were two stages in the evaluation of this proposed countermeasure. The first stage was to collect a number of "gaze-based passwords" for three schemes; PIN (a PIN pad is presented to the user), gaze passpoints with saliency masks, and gaze passpoints without saliency masks. Four users provided five gaze passwords each. This provided a total of seven PINs, 7 with saliency masks and 6 without saliency masks.

The next stage was a study involving 12 users asked each participant to attack five gaze-based passpoints. This was achieved by showing the image and a video of the eyes of a user performing a genuine authentication and asking the participants to attack. Participants could view the video of the genuine authentication as many times as they liked. Once the participant believed they had the right point located, they attempted to authenticate. Participants were given three attempts per image.

Results showed a significantly higher attack success rate for the PIN scheme compared to the scheme with no saliency masks. This was not unexpected, as there is less complexity in the image of a PIN pad. Saliency masks showed significantly less successful attacks than the image alone, this suggests that saliency masks help to reduce the success rate of shoulder surfing attacks. However, as shown in the paper, the gaze detection threshold has a significant impact on the number of successful login attempts (attacks). Thus the use of the eye-tracking itself could impact on the success of the attacks. It would be beneficial to examine the efficacy of saliency masks in a normal passpoints approach where the impact of the "gaze-based" approach could be removed. This could contribute to a reduction in hotspots.

It can be seen from this research that examination of the security of cued-recall graphical passwords is consistent as each analysis considers the hotspots problem. This allows comparison of the security of cued-recall schemes in terms of the hotspots analysis. Cued-recall schemes are often similar (with less variation than configurations available for recognition-based schemes) since the common variables which can change are the number of challenges, the images used and the number of passpoints. In comparison RBGPs have more potential variations

Figure 2.4: RBGP Registration and Authentication Process



in the number of distractors, how distractors are selected and so forth. It is for this reason RBGP schemes were selected as the focus for this work.

The state of research on recognition-based graphical passwords security will be discussed in the remainder of this chapter.

2.3 Recognition Based Graphical Passwords

In the registration stage for a recognition-based graphical password (RBGP) scheme, the user is presented with a set of images from which they select a number of passimages to be used to authenticate. Alternatively, they upload their own images. In an authentication session (after providing their username) the user is presented with a number of screens which contain a grid of images. Each screen contains at least one passimage and a number of alternative non-passimages called distractors. The user must select their passimage from the screen, repeating the process for each of the challenge screens. Upon successful selection of the user's passimage from each screen, the user is authenticated. This process is demonstrated in Figure 2.4 where the registration process a user selecting a subset of the potential passimage set to be their passimage set is depicted. In the authentication process the user is presented with a number of challenge screens with a passimage on each screen. The user selects the correct passimages to authenticate successfully.

The archetypal example of a recognition-based scheme is the PassFaces scheme created by Id-Arts Ltd [1]. In this scheme the user authenticates by selecting the faces allocated to them upon registration as their passimage set (or pass set) from a collection of eight alternative distractor faces, consisting of four challenge screens. An example challenge screen is shown in Figure 2.5.

Other examples of recognition-based schemes include the Deja vu scheme by Dhamija and Perrig [20] and the Story scheme by Davis *et al.* [15]. In the Deja vu scheme the set of passimages used is randomly generated art. In the Story scheme proposed by Davis *et al.* [15] the user selects a password as a sequence of unique images selected by the user to make a "story" from a larger set of images.

Figure 2.5: PassFaces Challenge Screen (obtained from [1])

The images used represent every day objects e.g. food, animals, children, and scenic landscapes.

The security of recall and cued-recall graphical passwords have been considered in terms of guessability in a consistent manner (by examining potential password space and bias in user selections). In contrast, analysis of the security of RBGPs has been inconsistent. For example, one approach to calculating the entropy (hence guessability) of a RBGP is proposed by Hlywa [43], whilst a different approach to measuring guessability is reported by DeAngeli *et al.* [17] and Dhamija and Perrig [20]. This thesis focuses on the analysis of RBGP schemes due to the inconsistency of work in this area to date.

2.3.1 RBGP Definitions

For clarification, the definitions of aspects related to RBGPs which will be used frequently throughout the thesis are provided as follows:

- **passimage** - A passimage is an image selected from the set of all possible images by the user to be used as their authentication factor.
- **distractor** - An image shown on a challenge screen which is not a passimage for the user.
- **challenge screen** - When a user authenticates, they are presented with multiple grids of images which includes a passimage and a number of distractor images. Each grid is called a challenge screen.
- **challenge session** - An authentication session which consists of a number of challenge screens from which the user must select their passimages.
- **passimage set** - The set of images which comprise the user's selection of passimages.

2.3.2 RBGP Configurations

To establish the possible threats and attacks for a RBGP scheme, first a list of aspects which contribute to the configuration of these schemes needed to be established. This was extracted from the information gathered in the literature review. The different aspects contributing to the configuration of RBGPs extracted from the literature review covered all aspects of the schemes reviewed. The aspects were established as follows:

- Image Types
  - Abstract e.g. Deja Vu presented by Dhamija and Perrig [20]
  - Disguised e.g. Use Your Illusion presented by Hayashi *et al.* [39]
  - User uploaded e.g. the scheme presented by Tullis and Tedesco [91]
  - photographic images e.g. the Story scheme presented by Davis *et al.* which used categories such as transport [15]
  - user created/drawn e.g. the Doodles scheme presented by Poet and Renaud [64] and the mikons scheme presented by Renaud [70]
- Image Source
  - User supplied e.g. personal photographs as for Tullis and Tedesco [91]
  - User selected e.g. the Story scheme presented by Davis *et al.* [15]
  - Assigned to user e.g. the VIP variants presented by De Angeli *et al.* [17]
- Distractor Selection - how the distractors are selected for a given passimage
  - using a similarity measure e.g. the Doodles scheme by Poet and Renaud [64] and the scheme presented by Tullis and Tedesco [92]
  - random selection where distractor images are randomly selected by the system e.g. ImagePass by Mihajlov *et al.* [56]
  - random selection from images belonging to categories other than that to which the passimage belongs e.g. the VIP1, VIP2, and VIP3 schemes by De Angeli *et al.* [16]
  - random selection from images belonging to distinct categories other than that to which the passimage belongs e.g. Moncur and Leplatre [58].
- Challenge Screen Set Up
  - Number of challenge screens per challenge session
  - Number of passimages per challenge screen
  - Number of distractors per screen

- Strike Out Policy -users are permitted a limited number of unsuccessful authentication sessions before being locked out of the system. e.g. Davis *et al.* [15], and Tullis and Tedeco [91]
- Use of constant distractors for a given users' passimage(s) e.g. Dhamija and Perrig [20] and Tullis and Tedeco [91]
- Image selection is disguised - either by not highlighting the image selected or not clicking directly on the image e.g. convex hull click presented by Wiedenbeck *et al.* [101]
- Keyboard entry is allowed for selection of passimages e.g. the PassImages scheme by Charrau *et al.* [12]

More precisely, each RBGP scheme was examined for details on the different configurations. The results indicated the important aspects as being the number of passimages (denoted p for this work), the number of challenge screens (s), the number of distractors per challenge screen (d), the number of constant distractors per passimage (c), and whether the images were assigned to the user or not. 19 RBGP schemes were identified. However, Hasegawa *et al.* [37] presented no specific details of configurations in their paper. Also, Hoanca and Mock reference the PassFaces scheme, but discuss selection of points and not overall images and so is more akin to a cued-recall scheme. Therefore these two schemes are not included in the summary. This left 17 schemes which could be defined by the configurations which are summarised in Table 2.1. Where the information was not available from the paper, "NA" is used to denote not available. If the number of passimages selected by the user was determined by the user and not the system, p is used. If a range or maximum number of passimages was provided, this is provided. Occasionally (e.g. Everitt *et al.* [28]) the authors note that the number of passimages or challenge screens can vary, where this is the case the example configuration provided in the paper has been used.

It can be seen from Table 2.1 that RBGP schemes can be split into two groups. One approach consists of a single challenge screen with multiple passimages presented on this screen. Nine of the 17 RBGP schemes presented only one challenge screen. This approach can be further refined by the passimage selection being restricted to a specific order, or order being irrelevant. For example consider three passimages, one of a fox, one of a dog and one of a rabbit. If order is important, they must be selected fox, dog, rabbit to authenticate successfully. Selecting dog, rabbit, fox will result in an unsuccessful authentication. If order is unimportant then both attempts will be successful. Five of the schemes identified in Table 2.1 have ordered selection with one challenge screen (Story, Moncur, Komanduri, VIP1 and PassImages). Three have unordered selection with one challenge screen (Deja Vu, Use Your Illusion and Tullis). This leaves one notably different approach, PassImages, presented by Charrau *et al.* where a total of six passimages have to be selected from four challenge screens in the correct order.

The remaining eight schemes presented represent the group of schemes which present a single passimage on multiple challenge screens. All eight of these approaches have no order restriction.

In addition to the other factors (e.g. number of passimages etc.) the type of images used and whether the passimage set is assigned to the user is also a factor in the configuration of RBGP schemes. Of the 17 schemes identified six use photographs of objects, three use photographs of faces, four use personal photographs, two use drawings or doodles, one uses random art, and the last uses icon pictures. Images can also be assigned to the user (five schemes), selected by the user (six schemes) or uploaded by the user (six schemes).

2.4 Threat Model for Recognition-Based Graphical Passwords

Potter proposes that security can be assessed in terms of the likelihood of successful attacks [65] and so the next step in analysing the security of a recognition-based graphical password (RBGP) scheme was to construct a threat model.

In this work the definitions of Anderson [**?**] will be used when constructing the threat model. For this work a vulnerability is defined as a flaw in the RBGP mechanism or the user's interaction with it which results in a potential attack. An attack exploits a vulnerability to gain unauthorised access to a system (through authentication).

Once these definitions were established, it was necessary to identify areas of potential threats. DeAngeli *et al.* propose that security of authentication mechanisms can be judged in terms of three aspects; guessability, observability, and recordability [17]. Definitions of each are as follows:

1. guessability: the probability an attacker can guess the user's password (graphical or otherwise)

2. observability: the probability of an attacker being able to observe the authentication process

3. recordability: the ease with which a user can record the user's password (graphical or otherwise)

Renaud extends these areas to include analysability and resistability in [69]. Analysability refers to implementation details of the software itself e.g. bugs in the code which could be exploited. Resistability refers to "auxiliary attempts to secure the system", an example provided is a three strikes policy where the user is locked out after three unsuccessful authentication attempts. This research focuses on measuring the security of attacking the user's passimage set and thus not auxiliary security controls or attacks which by-pass the mechanism itself.

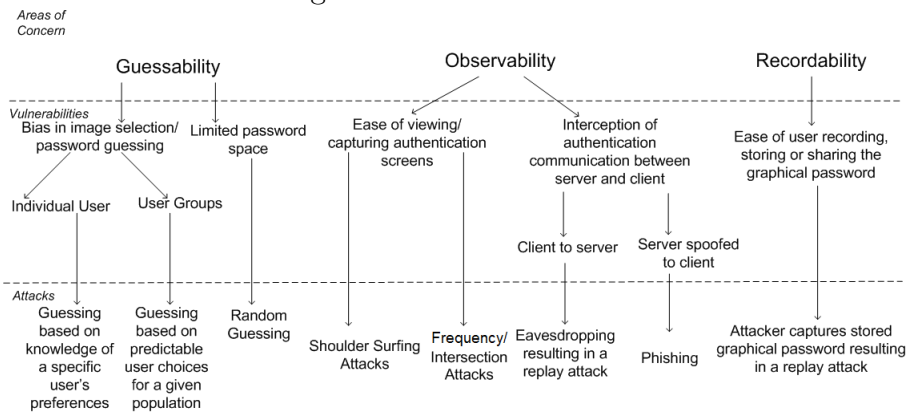
The three aspects of guessability, observability, and recordability are used to evaluate the security of recognition-based graphical passwords in this chapter. Resistability and analysability are considered outside the scope as they relate to attacking the system rather than the secret for example through software bugs which are specific to an implementation.

Presented in Figure 2.6 is the threat model for RBGP schemes, where the aim of the attacker is to obtain a user's passimages. This arose from the analysis of

RBGP Scheme	Passimages	Screens	Distractors	Constant Distractors	Image Assignment	Image Type	Order
PassFaces [1]	4	4	8	8	Assigned to user	Faces	No
Deja Vu [20]	5	1	20	NA	Selected by user	Random Art	No
Use Your Illusion [39]	3	1	24	24	Provided by user	Personal Photographs Obscured	No
Faces [15]	4	4	8	8	Selected by user	Faces	No
Story [15]	4	1	5	5	Selected by user	Photographs of Objects	Yes
Doodles [63]	4	4	15	NA	Provided by user	User drawn doodles	No
ImagePass [56]	p=max of 12	1	12-p	12-p	Selected by user	Photographs of Objects	Yes
Awase-e [85]	9	4	8	0	Provided by user	Personal Photographs	No
Pering [61]	p	10	3	0	Provided by user	Personal Photographs	No
Everitt *et al.* [28]	5	5	8	8	Selected by user	Faces	No
Komanduri [50]	8	1	72	72	Assigned to user	Drawings	Yes
Moncur [58]	4	1	6	NA	Assigned to user	Photographs of Objects	Yes
Mikons [70]	4	4	15	15	Provided by user	Combination of icon-type images	No
VIP 1 and 2 [16]	4	1	6	0	Assigned to user	Photographs	Yes
VIP3 [16]	8	1 but only 4 passimages	12	0	Assigned to user	Photographs	No
PassImages [12]	6	4 but all 6 passimages to be selected	25- number of passimages on the screen	NA	Selected by user	Photographs of objects	Yes
Tullis [91]	8-20	1 with 2-5 passimages	15 less 2 to 5 passimages	NA	Provided by user	Personal photographs	No

Table 2.1: RBGP Configurations Summary

Figure 2.6: Threat Model



literature and the review is presented in the remainder of this chapter. The main attacks are categorised into the three areas of concern; guessability, observability and recordability. The vulnerabilities corresponding to each of these areas were identified and attacks are shown which exploit these vulnerabilities.

The following attacks are considered in the threat model (as shown in Figure 2.6)

- Random guessing
- Guessing based on knowledge of biases in choice of a general population
- Guessing based on knowledge of a specific user
- Shoulder surfing
- Intersection/Frequency attacks
- Eavesdropping resulting in a replay attack
- Phishing
- Replay of passimages captured by recording

Each aspect of the threat model is evaluated in turn based on current research in the following section.

2.4.1 Guessability

Random Guessing

One of the key aspects of security of a recognition-based graphical password (RBGP) is the probability of guessing the correct images for a complete challenge session. Researchers often report a chance of guessing (or guessability) as shown in Equation 2.1 where X is the number of images displayed on a challenge screen and n is the number of challenge screens.

$$\frac{1}{X^n} \tag{2.1}$$

For example, the Doodles scheme [63] has a guessability value of $\frac{1}{16^4}$ since it has four challenge screens and displays 15 distractors and a single passimage per screen. In Deja Vu [20] and Use Your Illusion [39] the probability of guessing one image correctly is reported as $\frac{1}{\binom{t}{p}}$. In this equation t is the total number of possible images, p is the number of passimages on a challenge screen and $\binom{t}{p}$ is the number of combinations of p passimages which can be selected from t images. These values are correct where only one challenge screen is presented and the user must select all the passimages from the challenge screen. If $p = 1$ this reduces to $\frac{1}{t}$.

These values relate to random guessing. If users are allowed to select their own passimages, this random guessing value could potentially overestimate the real probability of guessing due to a potential bias in user selection of passimages. This work considers only schemes which present one passimage per challenge screen, adaption of this work to incorporate multiple passimages on a single screen (such as in Deja Vu and Use Your Illusion) is discussed in Chapter 9. A summary of the scope is provided in Section 2.5.

In a different approach to measuring guessability, Hlywa *et al.* calculate a value for the entropy. Entropy is a measure often used to approximate the strength of an alphanumeric password. Entropy counts the number of guesses required by an attacker in a guessing attack if each guess except the last one is wrong [80, Page 63]. Entropy is a measure of the randomness of the password (or how difficult it is to guess) and is based on Shannon's work in information theory [79]. The equation for entropy is shown in Equation 2.2, where the probability of each letter in the alphabet is multiplied by the log_2 of the probability and summed. The sum is multiplied by -1 because probabilities are of the form $\frac{1}{x}$ which, when logged is negative. Multiplying by -1 makes the result positive.

$$H(X) = -\sum_{x \in X} p_i log p_i \tag{2.2}$$

Hlywa *et al.* calculate a value for the entropy as follows: $log_2(n^s)$ [43] where n is the number of images per screen, and s is the number of challenge screens. This approach is similar to that for an alphanumeric password where the log_2 of the password space is multiplied by the length of the password. Entropy is often used as an indication for how difficult an offline brute force attack would be and as such may not be directly applicable to RBGPs. The passimage is presented in each challenge screen and so the process of guessing is reduced to selection of one of the images in the challenge screen. This is because the number of challenges presented is known (i.e. the number of passimages required to authenticate) and that each challenge contains one passimage. With a password the attacker does not know the length and cannot be sure of the characters used to construct the password. Offline attacks are discussed further in Section 2.5.

Van Oorschot and Wan present the "TwoStep" authentication scheme which combines alphanumeric password entry and a graphical recognition-based scheme [96]. In terms of security, they discuss several theoretically possible attacks. Measuring the level of security of the TwoStep scheme is presented as a combination of the entropy of the alphanumeric password and the graphical passwords' entropy.

The entropy of the graphical password element is calculated as $r.log_2 t = log_2 t^r$ where r is the number of challenge screens, and t is the number of possible choices (calculated using binomial coefficients). This is the same approach as Hlywa *et al.* [43].

Guessing With Knowledge of Predictable Choices

Davis *et al.* examine how permitting user selection of their passimage set affects the guessability. Davis *et al.* [15] implement two recognition-based schemes, Face and Story. Face is based on the PassFaces scheme by Id Arts Ltd, but users are limited to choices from distinct categories. The Story scheme asks the user to select a sequence of images to construct a "story" password, where each image selected is from a distinct category. In both the schemes, the images are categorised into non-overlapping subsets of images. For example typical white male, white male model etc. for the Faces scheme, and cars, landscapes etc. for the Story scheme.

Davis *et al.* estimated the probability of a given set of passimages (which they refer to as a password) being selected from either scheme. Graphical password selection was restricted as images had to be selected from distinct categories, and only a subset of categories were presented to the user upon selection. Assumptions in their estimated probability included that choices of the later images in a user's password were influenced only by the choice of the immediately preceding images (due to the distinct categories restriction). Use of a maximum likelihood estimation allowed the authors to estimate the parameters in their model of probability using 80% of the password data collected from 154 users. The probability model using the values in the data set allowed the authors to establish an ordering of the passwords from most probable to least probable.

The remaining 20% of password data was then used for attacking, which prioritised more probable passimage sets. To attack a password, the set of all combinations of passwords was first reduced based on the categories presented to the users upon selection (removing any passwords which contained images from any categories not presented to the user). The set of all passwords was then ordered by calculation of the probabilities using the model established by Davis *et al.* [15]. The position of the passimage set in this ordered list was then the number of guesses required to correctly guess the user's password (passimage set).

The work by Davis *et al.* considers the probability of the whole passimage set being selected, and not the probability of images from individual categories being selected indicating an area which could benefit from further research. The bias could be exploited by selecting the image on a challenge screen from the "most likely" category. This is examined in Chapter 4.

Guessing for a Known User

In the "Use Your Illusion" scheme proposed by Hayashi *et al.* [39] (where images are degraded so only colours and shapes are recognisable) the authors discuss an "educated guess" attack. This is a form of social engineering where the attacker tries to guess the correct image based on previously acquired information about

the user. In the Use Your Illusion scheme, the authors conjecture that it would not be possible to pick out an individual's obvious choice of image due to the degradation of the image.

The work was extended by Hayashi *et al.* [40] where experiments which evaluated "individualized educated guess attacks" for user taken photographic passimages were conducted. A friend was defined for the purposes of the experiment as someone the participant had friend status with on Facebook, who met with each other at least twice a week and had known each other at least three months. The hypothesis tested in the experiment was :

"An attacker can make more accurate guesses about authentication images if the attacker possesses information about the user who chose them."

Attackers were given 10 attempts to guess the three authentication images of the target user. Eight out of 15 attackers correctly identified the target set of three images within 10 guesses. This approach reflects the set up of Hayashi *et al.*'s "Use Your Illusion" scheme [39] in which users must select their images from decoys presented on one challenge screen. This does not reflect the RBGP approach where one passimage is shown per challenge screen and multiple challenge screens are presented. If a three strikes policy was in place which restricted the number of guessing attacks, success would be reduced. Of the eight successful attackers, only three managed to perform a successful attack within three guesses.

2.4.2 Observability

Progressing to the second area of concern, the vulnerabilities associated with observability are considered.

Shoulder Surfing

As noted by Wiedenbeck *et al.*, shoulder surfing graphical passwords is the process of observing authentication sessions and noting the images selected to be used to impersonate the user at a later time [101]. In the "Use Your Illusion" scheme proposed by Hayashi *et al.* [39], users authenticate by selecting the degraded version of their images (a non-photo-realistic rendering algorithm which removes the majority of the image features, but retains some colours and shapes) from a set of challenge degraded images. The claim is that an observer would find it more difficult to capture the degraded image, but users can easily recognise it. The user recognition is established by a user study by Hayashi *et al.* [39], but no user study is reported relating to resistance to shoulder surfing. Instead the authors propose two countermeasures for shoulder surfing. Firstly, they propose allowing selection of passimages using keyboard entry. They ensure that the location of authentication images are not constant so the observer cannot memorise the key pressed for the location of the image. The second countermeasure is to avoid any indication on the screen which highlights which image has been selected (hence reducing likelihood of successful shoulder surfing).

Hasegawa *et al.* [37] present a method in which they combine low frequency components of a distractor image with high frequency components of the passimage. A discrete wavelet transform was applied to a passimage and a distractor image and the lowest frequency band of the distractor image (which contains the "average information of the picture") was combined with the higher frequency bands of the passimage. The authors propose the hypothesis that the user will be able to establish which picture has been combined with their passimage, but shoulder surfers will be unable to detect the high-frequency components of the passimage. The results showed that users were able to recognise the passimage at least half the time, whilst observers couldn't. The results also note the scheme doesn't work well for camera shoulder surfing.

Other attempts at providing counter measures for shoulder surfing include Hoanca and Mock [44] who proposed a camera-based eye tracking system to allow the user to select their graphical password by fixation of their eyes on the passimage. There are potential limitations of this approach. First, there is a possibility a user's eye may wander, making it difficult to select the passimage. Secondly, there could also be an issue with hardware availability, as the camera required might not always be available. It should be noted that this was a short paper which indicated potential work and thus no experimental evidence was presented.

Indirect selection of images is proposed as a countermeasure for shoulder surfing by Gao *et al.* [34] who proposed a scheme in which users selected their passimages in a specific order. To authenticate the users have to draw a path through their passimages presented in a grid in the correct order. A 20 participant user study over one week was reported which examined usability, however the authors performed no study to examine whether this approach is effective against shoulder surfing. In addition, the approach could result the possible reduction in password space. A path could be drawn which covers all the images presented, all that would remain is to establish the correct order.

Sreelatha *et al.* also propose an indirect selection method [81]. The authors propose that users select image pairs and "key positions" on a challenge screen. A challenge screen consists of a grid of images as normal, but the user must locate the passimages in the key position and (instead of selecting this image) must then select the corresponding pair. If there is only one passimage pair shown on screen, this may not reduce shoulder surfing. In this case the attacker views the image selected and repeats this, the position and identity of the key passimage is not required. If however there are multiple pairs of passimages on the challenge screen, this could reduce shoulder surfing.

Another possible limitation of this approach is the guessability factor, if a user were to select an obvious pairing (e.g. the cartoon characters Tom and Jerry) this could reveal key positions.

A study which performed an evaluation of the efficacy of shoulder surfing attacks was performed by Tari *et al.* who reported the results of a study in which users were asked to attempt to steal the password and passimages of a "victim" by shoulder surfing [87]. 20 participants were recruited to attempt to capture the passwords and passimages of the experimenter. The participants were provided with a notepad and pen and told to sit/stand wherever they thought best.

The study used four configurations for knowledge-based authentication schemes. These were PassFaces with mouse selection, PassFaces with keyboard selection, a strong alphanumeric password and a dictionary alphanumeric password. The PassFaces configuration used 5 challenge screens with 9 images displayed on each screen. The passwords had 5 characters with the aim of providing a comparable length. The characters and passimages had to be selected in the correct order, which may have made the PassFaces configurations more difficult and could overestimate the resistance to shoulder-surfing in cases where order is unimportant. If the same passimage set was used for both PassFaces configurations a potential limitation could be a learning effect which could potentially overestimate the success rate of whichever configuration was performed second.

The results indicated that the least susceptible configuration to shoulder surfing was PassFaces with keyboard selection (an average of 0.55 images from five were recalled in the correct order) and the other extreme was the most susceptible being a non-dictionary alphanumeric password (an average of 3.65 characters were recalled in the correct order). The Duncan's multiple range test statistic was applied to establish if there was a significant difference in performance between each configuration. There was no significant difference between a non-dictionary 5 character password and an ordered 5-passface set, but each other configuration was significantly different from the others. The conclusions of the work were that the significant differences in performance are due to the variation in setup (dictionary passwords, non-dictionary passwords, passimages with mouse selection, and passimages with keyboard selection). If the same passimages were used for both the keyboard and mouse selections there may have been a learning effect. This paper contributes evidence for the resistance of the authentication schemes to shoulder surfing. It provides a comparison of different configurations of RBGP schemes with dictionary and non-dictionary passwords.

Intersection/Frequency Attacks

An intersection attack, as defined by Dhamija and Perrig [20] (and discussed by Dunphy *et al.* [21], Hayashi *et al.* [39] and Poet and Renaud [64]) is an attack in which the attacker records multiple challenge screens and notes the images which are constant between two screens. Assuming the distractor images all change this would result in the passimage being identified. Takada *et al.* also identify a similar attack which they call a frequency attack [85]. In a frequency attack, the attacker notes multiple challenge screens and notes the frequency with which each image appears, they then select the image which occurs most frequently for any given screen. For this work a frequency attack will primarily be considered as an intersection can be thought of as a special case of a frequency attack. This can be seen by considering where multiple screens are recorded and if all the distractors change then the distractors appear with a frequency of 1 and the passimages appear with higher frequency. Dhamija and Perrig [20] successfully summarise general approaches to counter measures for this issue as:

- Use the same distractor images and passimages for each session.

- A small subset of distractor images could be shown for a given passimage each time that passimage is used in a challenge screen. The result being that the subset of distractors will occur with the same frequency as the passimage it is associated with. This would mean the attacker would need to make a selection from the images with the same frequency. Thus this approach only mitigates the attack.

- If a user fails any challenge screen, all subsequent screens display only distractor images, "dummy screens". The aim is to reduce the number of times an attacker sees the correct passimages for an intersection attack.

- Implement a limit on the number of incorrect authentications a user can perform, this stops an impersonator attempting to discover all of the images. (A "three strikes and you're out" approach).

The authors note that these solutions may impact memorability, as re-use of distractor images may result in users recognising distractor images and selecting them instead of their passimages [20]. However, maintaining the same distractors for a given passimage does ensure that an intersection attack is not possible since all images occur with equal frequency. The remaining three options minimise the potential for an intersection attack.

Hayashi *et al.* implement the first countermeasure in their Use Your Illusion Scheme [39] by using the same distractor images for each authentication session. Thus, the challenge sets are the same each time for any given image from the user's passimage set.

Takada *et al.* propose that including the possibility that a challenge screen contains no passimages stops intersection attacks [85]. One possible limitation of this approach is that it is likely that at least one challenge screen must have a passimage on it. This image could be attacked by a frequency attack. It is likely this approach mitigates the attack rather than stopping it completely. Takada *et al.* also propose a variation on maintaining constant distractors by using a set of "priority" distractors, which are not always used, but are given preference which would increase their frequency. This is a less strict version of using a subset of constant distractors.

Man in the Middle Attack

A man in the middle attack is an attack in which the user is led to believe they are authenticating legitimately, but they are actually authenticating to the attacker who has intercepted the communication. The attacker then uses the gathered credentials for authentication on the legitimate service. As noted by Biddle *et. al* [5], a phishing attack could use a man in the middle attack and be applied to a RBGP scheme. If the attacker is already in possession of the user's username, they can capture challenge screens by entering the username for the legitimate service. These screens can then be used to construct a phishing attack. If the attacker had no username, the attack would be carried out in real time. A man-in-the-middle attack would be performed and the user's username is sent to the attacker who then uses this in the legitimate site to obtain a valid challenge

screen. This is then relayed back to the targeted user. The process is repeated until full authentication has occurred.

Eavesdropping Resulting in Replay Attack

Application of a replay attack to a recognition-based graphical authentication scheme is similar to considering a replay attack on an alphanumeric password scheme. A man-in-the-middle attack is constructed and the data copied when a user performs the authentication process and sends the authentication data to the server. The data is then "replayed" to the server at another time, potentially resulting in a successful attack.

2.4.3 Recordability

Recordability refers to the ease with which a user can record their password or passimages. Passwords can be recorded by writing them down. Considering this issue, Dunphy *et al.* [22] examined the feasibility of recording PassFaces by description. A user study was performed in which 18 participants were asked to give verbal descriptions of 15 random face images. This resulted in a total of 6 descriptions per face. Participants were then asked to listen to descriptions of faces (not described by themselves) and select the image they thought the description related to. The reasoning behind this was that if people could describe the photo well enough, then they could potentially record their passfaces by description and hence share passwords or have them captured. One potential limitation of this approach is that people could use a mobile phone camera to capture the image, making recording by description redundant.

Dunphy's argument against this is that "sometimes digital devices will fail and description might be the only means of sharing a graphical password" [22]. Examples of these types of situations would have been of benefit at this point, as would an explanation of why pictures would have to be sent using network connections. This does not invalidate the results of the paper which indicate that describing PassFaces in general doesn't work well. Dunphy's examination was the only work found regarding recordability showing that work in this area has been limited. This could be attributed to the ease with which one can record using digital devices (e.g. taking a photograph on a mobile phone and storing the images). It could also be due to the lack of information as to what extent this could affect the security since it relies on user behavior which is difficult to control.

2.4.4 Summary

It can be seen from this examination that most research concentrates on one aspect of the security of RBGP schemes e.g. shoulder surfing or guessability, and often fails to provide a complete analysis covering all aspects. To provide a complete overview of analysis to date, the results of this review are summarised in Table 2.2 which presents a selection of RBGP schemes and summarises their security aspects in terms of guessability, observability and recordability.

Scheme/Authors	Guessability	Observability	Recordability
VIP ([17])	Guessability for schemes VIP1 and VIP2 is reported as $\frac{1}{x^n}$	VIP2 aims to reduce shoulder surfing by having a passimage set larger than the number of challenge screens	VIP was described as difficult to record or describe, however this did not consider cameras or screen captures.
Faces and Story ([15])	Guessing entropy was calculated.	Not reported	Not reported
PassFaces (Id Arts Ltd.)	PassFaces are assigned to the user, and so user choice has no application. Random guessability is not reported.	Not reported	Dunphy [22] showed that it was difficult to describe faces, however as for VIP this does not thoroughly consider the possibility of using a digital camera or screen capture to record the images.
Use Your Illusion [39]	Chance of guessing one image = $\frac{1}{\binom{n}{p}}$ where n is the number of images in the challenge set and p is the number of images in the passimage set. The probability of guessing within t attempts is reported as $\frac{t}{\binom{n}{p}}$. An educated guess attack is mitigated by applying a distortion the original images so that only the real user will recognise them.	Keyboard entry mitigates shoulder surfing whilst constant distractors for a given passimage to stop intersection attacks.	Not reported
Hasegawa *et al.* [37]	Not reported	Images are obscured to mitigate successful shoulder surfing.	Not reported
Komanduri & Hutchings [50]	Not reported	Keyboard selection of passimages to reduce shoulder surfing.	Not reported
Hoanca & Mock [44]	Password space reported, though the calculations are not explained.	No feedback to the user is given on which image has been selected.	Not reported
Deja Vu [20]	Guessability value of $\frac{1}{\binom{n}{m}}$ Passimage sets are selected from random art to reduce exploitation of user choice.	Shoulder surfing counter measures are hiding image selection and altering images so they are unrecognisable to the attacker. Intersection attack countermeasures are provided as follows: the challenge set is always the same, a subset of distractor images are constant within the challenge screen, use of dummy screens	Not reported

Table 2.2: RBGP Security Summary

2.5 Context and Scope

As a result of the literature review and analysis, the context and scope of this work is refined and presented as follows. The scope was restricted to allow the work to be completed in the time frame with the resources available. Future work includes extension of the scope, this is discussed further in Chapter 9.

2.5.1 Authentication Environment & 'Passimages'

The environment being considered is a local authentication environment which can be physically observed by the attacker. Since local authentication is the authentication environment under consideration, observation attacks which involve intercepting authentication communication between the client and server (e.g. man in the middle attacks) are excluded.

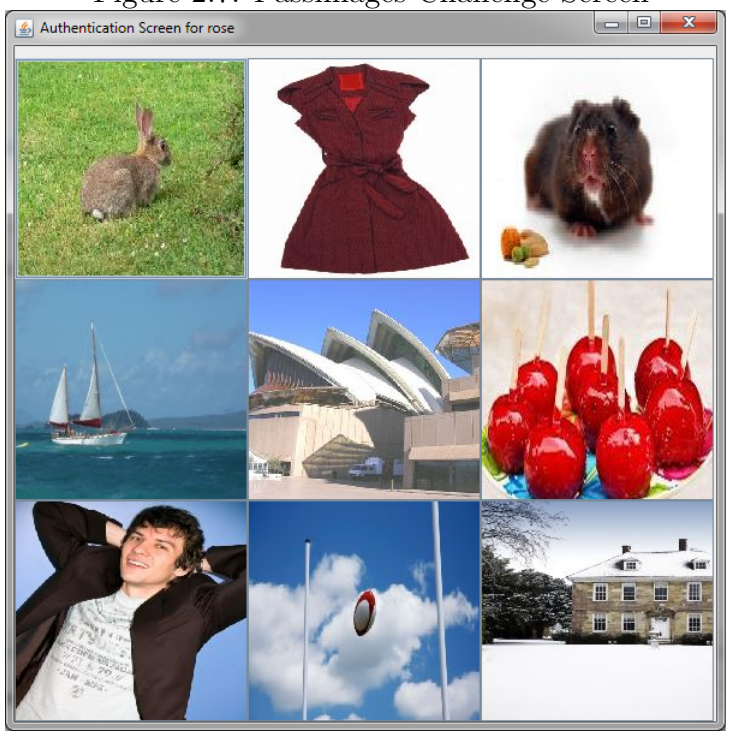
At each stage of the research it was necessary to apply hypotheses to a concrete implementation of a RBGP scheme. The passimages scheme (the term passimages was established by Charruau *et al.* [12]) was selected as it reflects a number of recognition-based schemes. It is similar to schemes proposed by Davis *et al.* [15] and the commercial application PassFaces [1], though these schemes use a variation of image content. Focusing on this configuration allowed the framework for a metric to be constructed and adaption to different contexts is discussed further in Chapter 9.

In the passimages scheme the following values are configurable: the number of challenge screens in a session, the number of distractors per challenge screen, the number of passimages in a user's passimage set, and the number of constant distractors. In the default configuration, the user is presented with four challenge screens comprising of nine images per screen, one of which is a passimage. An example challenge screen is shown in Figure 2.7.

In each configuration of the passimages scheme, the order of passimage selection is not considered. This is because of the 17 RBGP schemes presented in Table 2.1 only six (roughly 35%) used passimage sets which had to be selected in a specific order. Also Davis *et al.* recommended order be avoided as the images were successfully recognised, but order was often not [15]. Since order of selection is defined as outside the scope of this work the RBGP scheme type which employs multiple challenge screens is examined. This is because all ordered schemes identified (with the exception of Charrau *et al.* [12]) used a single screen. To extend the application of the metric presented in this work to incorporate all schemes identified in Table 2.1 further work is discussed in Chapter 9.

In addition to the selection of the number of challenge screens, choice of images and number of images per screen it was necessary to consider how distractor images were to be selected. It was decided that three different distractor selection algorithms would be used. The first was based on random selection of distractors (using ORDER BY RAND in SQL and the Random class in Java). Random selection is seen for example in the ImagePass system by Mihajlov *et al.* [56]. The second, modelled on a VIP1, VIP2, and VIP3 schemes by De Angeli *et al.* [16] selected distractors randomly from any category except the category that the passimages presented in the challenge session belonged to. The final

Figure 2.7: Passimages Challenge Screen



selection algorithm was an extension of the VIP1 algorithm where distractors were selected randomly from distinct categories excluding the category that the passimage belonged to. This algorithm was used by Moncur and Leplatre [58]. These reflect the majority of selection algorithms covered in literature at the time of writing.

Finally, the images to be used were established as photographs of objects the largest proportion of schemes identified (six of 17) used this image type. Six of the schemes were selected by the user, an equal number were provided by the user, and five were assigned to the user. User selected images are examined in this research as half of the photographs of object schemes use this approach. In total 144 images were collected for use in this work. Further details on the images are provided in Section 4.2.2.

2.5.2 Attacker Model

To put the attacks into context an attacker model is now discussed. This represents the attacker under consideration in this work, what information they have access to and their abilities. The attacker attacks the authentication stage where the users select their passimages. At this point it is assumed the attacker has already identified a target user and has obtained their username. The aim of the attacker is to authenticate as the targeted user by impersonating them. Thus they do not exploit any bugs in the implementation of the mechanism. To impersonate a user they must identify the user's passimage set. The attacker does this through guessing and observation attacks.

The attacker does not attempt offline guessing attacks such as dictionary

attacks. In this situation for a password, a dictionary attack works as follows. The attacker captures the hash of the password when it is sent from the client to the server. They then apply a hash function to a list of possible words (a dictionary) until a match for the hash captured is found. The password is then known to be the plain text which was hashed to result in the value captured. If one were to consider such an attack on a recognition-based graphical password scheme the form of communication sent from the client to the server when the user selects their image must be considered. There are a number of possibilities:

1. a hash of the image itself
2. an identifier for the image
3. a hash of the identifier for the image
4. a temporary identifier for the image (identified by Mihajlov [56])

Each of these approaches relies on some secret detail regarding the implementation of the scheme. The attack relies on the attacker copying this communication and establishing the connection between the information sent and the passimage. As the focus of this work is on local authentication and communication observations attacks are outside the scope, this attack is not considered any further. Since a RBGP scheme presents the attacker with the passimage on the challenge screen, an offline attack examining all possible images may not be optimal.

The attacker is unable to access user recorded prompts of their passimages. This effectively excludes all recordability attacks from the scope of this work. The reason for this is that it is unclear to what extent users may record their passimages and how easily an attacker may gain access to this information.

The attacker can continuously attempt authentication without being locked out of the system. This is because for this work limiting the number of authentication attempts (e.g. "three strikes and you're out") is considered as an auxiliary attempt to secure the system which are considered outside the scope of this work as they do not pertain to the mechanism itself.

In general for this work, the attacker can use information which could feasibly be leaked by or extracted from the RBGP interface. Thus, the attacker can capture as many challenge screens as required by observation. The attacker doesn't know if there are dummy screens employed, or if there are a number of constant distractors for a passimage. They don't know how many passimages a user has in their passimage set. They do know there's one passimage per challenge screen and hence how many distractors per screen.

In addition an assumption about the ability of the attacker to deduce the categories to which potential passimages belong is made. The assumption applies if images can be split into categories based on their semantic content. It is assumed the attacker can establish which semantic categories each image belongs to by examination of the content of the images. It is assumed the attacker establishes a distribution which is insignificantly different to that of the scheme. The attacker can also establish an order of bias of user choice for the categories (e.g. users will select animals more than scenery).

The attacker attempts each of the attack types separately. That is they do not combine attacks to increase the chance of success.

These details assume an optimal attacker. This assumption may not be realistic. The purpose of the metric is to provide a comparison of the potential security of multiple RBGP schemes. Assuming the optimal attacker may not be realistic, but provides consistent comparison. Fixing the level of the attacker allows us to focus on the metric and not the skill of the attacker.

2.5.3 Scope Summary

As detailed in this section, a number of details are considered with and outside the scope of this work. These are summarised as follows:

- The purpose of the attacker is to impersonate a targeted user.
- The attacker will not attempt offline brute force/dictionary attacks.
- The attacker does not have access to the communication between client and server and so cannot perform replay or phishing attacks.
- The attacker does not have access to user recorded graphical passwords.
- The RBGP schemes modelled have one passimage per screen and have no restrictions on order of selection.
- Attacks types are attempted separately and not combined.
- The attacker is unaware of the use of dummy screens or constant distractors.
- The attacker can deduce semantic categories and user selection biases where appropriate.
- The attacker can attempt to authenticate as many times as necessary.
- The attacker is assumed to be "optimal".
- Intersection attacks are defined as a special case of frequency attacks and so frequency attacks are considered for the final model.

In summary, this work considers guessing and observation attacks (exclusive of phishing and replay attacks) for RBGP schemes were one passimage is shown per challenge screen, where order of passimage selection is irrelevant. This configuration reflects a common configuration for a RBGP scheme e.g. PassFaces [1], Davis *et al.*'s Faces [15], and Pering *et al.* [61] though these schemes use a variation of image content.

2.6 Summary

This chapter has summarised current literature related to graphical passwords. The current state of security research relating to RBGPs has now been examined. In particular it has highlighted an inconsistent approach to measuring security levels of RBGPs. The first step in addressing this is to consider what requirements a security metric for RBGPs should have. Approaches to measuring security and metric requirements are considered in the next chapter.

Chapter 3

Measuring Security

This chapter aims to introduce some research regarding a high level approach to examining the security of authentication mechanisms in general. The potential qualities of the metric are then considered. The chapter concludes with a list of metric requirements, qualities against which the metric will be assessed.

3.1 Measuring Authentication Security

Attempts at examining the security of authentication mechanisms in general have been made. For example Renaud presents an overall evaluation of an authentication mechanism as a calculation of an "opportunity" measure of the attacker. This aims to reflect how much opportunity the attacker has. It is proposed in terms of a function of the guessability, observability, recordability and analysability divided by the resistability [69]. Values for the component parts are discussed on a high level, for example any password as strong as a 4 digit pin (which has guessability of 1 in 10,000 due to the 10,000 four digit combinations) is assigned a 0, any password which is weaker is assigned a "proportionally higher guessability figure". The paper is focused on the approach and details on how the proportion would be established are limited. Similarly, a value of 1 is assigned to recordability if the password is easily recorded, details are limited regarding at which point a password is easily recorded. The function adds the constituent parts and divides by the resistability. One potential limitation of adding the values is that a system could be deemed less secure if it has a poor level for one aspect (such as observability) but this aspect may not be important in a given context (e.g. if the system will only be accessed from a secure location where only the user is present). The resulting opportunity value could be skewed. This work contributes to the area by highlighting the areas of consideration for authentication security, and suggesting a high level approach to examining the security.

A similar approach to measuring security of authentication mechanisms is given by Mihajlov *et al.* in [55] and [57]. In these works the authors establish a quantitative evaluation of the quality of authentication mechanisms in terms of meeting their identified requirements for security and usability. The values assigned for the component parts are high level, similar to those presented by Renaud in [69]. The approach taken assesses the suitability in terms of quality

criteria (the requirements) which are: secrecy, abundance, revelation, privacy, and breakability.

Secrecy relates to the predictability of a key where the score is assigned based on "how many people find the key predictable" [55]. Abundance refers to the effective password space (the size of the collection of likely passwords), a value of low medium or high is identified based on the size. Calculation of this value is indicated as the use of a combination or a permutation calculation. Details on how this relates to high, medium or low levels of predictability presented in the work are not presented in the paper and thus assumptions regarding this cannot be made. Revelation measures the "disclosure level of the authentication key from a user and system perspective". This is further split into system revelation and user revelation and each of these aspects has a maximum value of 0.5. Privacy relates to the "amount of private details required by the authentication mechanism". The items of data considered are name, DOB, email and "additional data", each of which reduces the value of 1 by 0.25. The reasoning is that each of these items contributes to a possible identity theft, but this could equally depend on how securely the information is stored and who has access to this data. Breakability refers to the effort required to get access to an account. This is categorised into the types of attack e.g. brute-force, dictionary and key-logging, with each attack providing a deficiency maximum of 0.25. The attacks mentioned appear specific to passwords, and details on how to calculate the value for deficiency of each attack are not presented in the paper. A similar approach is taken to the usability aspects, but usability is outside the scope of this work and so shall not be discussed.

A single overall quality value is then defined as the Euclidean distance of the security criteria and the usability criteria where the individual squared values are summed and the square root is applied. The work is based on concepts proposed by Renaud in [68], where the qualities are considered in terms of a 3D space. The Euclidean distance is applied by Renaud and a similar approach is taken by Mihajlov *et al.* [55].

3.2 Security Metrics Background

Since a large proportion of this work relates to the construction of a measure or metric for the security of RBGP mechanisms, it is appropriate to consider relevant security metric research. This section aims to provide an overview of the attributes to be considered when constructing a metric. As a result a list of requirements to be applied to the metric proposed in this research will be established.

The metric qualities identified in this chapter are primarily based upon those qualities identified in the Workshop on Information Security System Scoring and Ranking, reported by Henning *et al.* [41]. As indicated by Henning, within the workshop there were multiple opinions on what constituted a "good" security metric. The qualities identified here are used as a benchmark to assess the final metric. It is not claimed that these qualities are exhaustive. However, many of the identified qualities are noted as important by other researchers and for this

reason it is proposed they are sufficient for evaluating the final metric.

Henning *et al.* [41], Wang [98] and Jansen [47] agree that the term "security metrics" is often used but with no clear singular interpretation or definition. Jansen also comments on the "misnomer" of using the word metric. He argues that metric implies that well established concepts from physics and other sciences apply equally to information technology, where in reality this is not the case. However, for the purpose of this research, the term metric will be used to denote a measurement of resistance to the attacks covered in this research.

Henning *et al.* categorised metrics (in the Workshop on Information Security System Scoring and Ranking, reported in [41]) into three areas: technical, operational and organisational. Technical metrics are used as comparison for technical objects such as algorithms and products. An example of a technical metric discussed by Vaughn *et al.* [97] is the "number of vulnerabilities of a program which can be detected with a scanner". Operational metrics are used to depict operational environments such as operating practices. An example of an operational metric discussed by Vaughn *et al.* [97] is an operational practice metric which measures the security practices of those who affect the information assurance policies, e.g. the number of users with a policy compliant password. Organisational metrics are related to processes of organisations. An example of an organisational metric discussed by Vaughn *et al.* [97] is process maturity metrics such as the software security engineering capability model (SSE-CMM).

Other taxonomies follow a similar approach with those proposed by Jansen [47] and Savola [77] each proposing a version of technical and organisational metrics. Savola divides metrics for information security management into three categories; management, operational and technical [77]. Management is the same as Henning's organisational definition, and definitions of operational and technical remain identical. Bohme notes that management metrics, e.g. return on security investment, are often used to establish where to spend money in terms of security [7]. This work differs as it aims to establish a measurement related to the ease with which a recognition-based graphical password system can be attacked by examining possible threats, attacks and related counter measures. The work presented in this thesis is related to the security of a specific system, therefore it falls under the category of technical as defined by Henning *et al.* [41].

While a large proportion of literature relates to business management or software metrics, there are a number of attributes of "good" security metrics discussed in related literature. These are discussed in this chapter to provide a method of evaluating the metric produced by this research. Some examples of proposed criteria for metrics were presented in the Workshop on Information Security System Scoring and Ranking by Henning *et al.* [41] and discussed by Jansen [47], Vaughn *et al.* [97], and Wang [98]. Each is discussed here in turn with specific attention to why they have been included or excluded from the final requirements.

In this work these qualities are categorised into two distinct groups. The first group contains attributes that, whilst useful, rely somewhat on the context in which they are applied and so these qualities are termed as *Context Dependent*. The remaining qualities are context independent and can be assessed with no consideration of context. The context dependent qualities are aims of this re-

search, but cannot always be easily assessed. This can be seen by considering an example such as effectiveness. Vaughn *et al.* note that effectiveness means that a metric can be quickly evaluated with minimal cost [97]. The problem arises when one tries to establish how quick is quick enough, and at what point is the cost low enough. A metric may be effective in one situation or context such as for local machine authentication, but not for mobile authentication. In contrast the context independent qualities can be evaluated to determine if they have been achieved and so are included explicitly in the metric requirements. The qualities are detailed in the following section. The chapter concludes in Section 3.4 by summarising the requirements established for evaluation of the metric proposed in this work.

3.3 Potential Qualities of a Security Metric

3.3.1 Context Dependent Qualities

Clear Scope

As highlighted by Vaughn *et al.* [97], scope dictates that the problem domain should be clearly identified. This is achieved throughout the thesis by the literature review and research hypothesis and is also disseminated in related publications ([25], [24], and [26]).

Sound foundation

Extending the scope quality, Vaughn *et al.* [97] state that sound foundation means that the metric should be based on a "well-defined model of the portion of the problem domain it describes". An important aspect of this work is to establish a mathematical model of the security of recognition-based graphical passwords (with respect to observation and guessing attacks) which is incorporated into a metric. This work aims to provide a well-defined model but, as with all qualities in the context dependent category, this quality is not easily assessed. It is unclear at what point the model is well enough defined. For this reason, whilst sound foundation is addressed in construction of the metric, it is not explicitly assessed as part of this research.

Process

Vaughn *et al.* [97] identify that the process for evaluation of the metric should be thoroughly defined, i.e. details of the information necessary to apply the metric should be provided. The definition of the final metric and instructions on how to apply it are provided in Chapter 8.

Relevance

Vaughn *et al.* [97] highlight that a metric must be relevant to the context in which it is being used. The authors say this can be achieved by being useful to decision

makers. Both Vaughn *et al.* and Jansen agree on this attribute. Relevance is context dependent and so it is not included in the requirements of the metric.

Effectiveness

Vaughn *et al.* [97] claim that for a metric to be useful, it should be possible to quickly evaluate the metric. This is also agreed by Jansen [47], but as with the preceding context-dependent qualities it would be difficult to ascertain at which point this quality is achieved, and so it is not included in the requirements which are directly assessed.

3.3.2 Context Independent Qualities

Repeatable

Jansen states that in order to be of value, a metric must be repeatable [47]. This means that a second assessment by the same evaluator provides an identical result. Vaughn *et al.* agree [97]. Thus due to agreement and the ability to evaluate this quality, this is included in the final requirements.

Reproducible

Jansen states that in order to be of value, a metric must be reproducible [47]. This means that second assessment by a different evaluator provides an identical result. Once more, Vaughn *et al.* [97] are in agreement with Jansen. Thus due to agreement and the ability to evaluate this quality, this is included in the final requirements.

Quantitative

Wang [98] notes that some proposed metrics are qualitative rather than quantitative and that this is not suitable, this is in agreement with Vaughn *et al.* [97]. Also, if the metric is quantitative it is then easier to make comparisons between RBGP schemes which is an essential aim of this work as Herley *et al.* indicate ability to compare authentication schemes is essential [42]. Thus due to agreement and the ability to evaluate this quality, this is included in the final requirements.

Objective

Objectivity ensures that an individual's perception has no influence on the result of the metric when evaluated, and links into reproducibility since if an individual's perception impacts their interpretation of the metric then this could result in a different evaluator obtaining a different result. Wang [98] notes that metrics are often subjective rather than objective and states this is not preferable, this is also noted by Vaughn *et al.* [97].

Dynamic

As noted by Vaughn *et al.* [97], dynamic metrics are those which can evolve over time whilst static metrics do not. This links to the repeatability and reproducability qualities and would mean that dynamic metrics would not be repeatable and reproducible since they could have different values at a different point in time.

Extensible

Extensibility of the metric is achieved by allowing new components to be added. Wang notes that there is often no time aspect associated with current metrics, essentially saying that something which may be secure today might not be tomorrow [98]. Extensibility of the metric will be established by making the metric easily adaptable. This means it will be possible to change the definitions of "secure" not only to match the situation, but also to match any new attacks or countermeasures.

Absolute/Relative

Absolute measures are independent of other measures whilst relative metrics are only meaningful within context. No preference is highlighted here by Vaughn *et al.* and so neither attribute will be included in the final requirements.

Direct/Indirect

Hasle *et al.* note that direct metrics are obtained by measurement of the property of interest [38]. Hasle *et al.* also note that indirect metrics are derived from a measurement of other properties which have a strong correlation with the property of interest [38]. Vaughn *et al.* claim that direct metrics are often preferable, but sometimes not possible [97]. Thus a direct measure is aimed for, but it is acknowledged that this may not be possible thus it shall not be included in the metric requirements.

3.4 Identified Metric Requirements

From the literature reviewed a number of attributes of good security metrics were recurrent and will be used to assess the finalised metric. The final metric should be:

- Repeatable - Multiple evaluations by the same evaluator result in the same end value.
- Reproducible - Multiple evaluations by different evaluators result in the same end value.
- Extensible (called dynamic in [97]) - The metric should be designed in such a way which makes it theoretically possible to extend the metric to incorporate further attacks .

- Objective - The metric should not be subjective, that is it should not be based upon the opinions of the evaluator but on the configuration of the RBGP under evaluation.
- Quantitative - The metric should be numerical in nature.

At this stage, the review of relevant background information has been presented. This culminated in a threat model which represents the areas of concern and a list of identified metric requirements. The next stage is to consider the individual areas of concern with the aim of establishing mathematical models. The first area is "guessability" which is addressed in the following chapter.

Chapter 4

Guessability Evaluations

Users often select easily remembered passwords which results in more successful guessing attacks (shown by Gaw and Felten [35] and Klein [49] for example). It reasonably follows that there is potential for user choice of passimages to reduce the passimage space and hence the security of RBGPs. Guessability is an aspect of authentication security which refers to the ease with which an attacker can guess the user's authentication secret. As with passwords, user choice of RBGPs could impact guessability. As indicated in the threat model (Figure 2.6 on page 37) there are three approaches to a guessing attack:

- Random guessing
- Guessing based on predictable user choices for a population of users (group bias)
- Guessing based on an individual user's preferences (individual bias)

In this chapter, each of these approaches is discussed in turn. The aim is to establish a model which estimates the number of attacks against a user's passimage set for a RBGP authentication scheme before success. The model could be established in a number of ways:

- Mathematical calculations of combinations
- Simulation
- Combination of user experiments and simulations

The first approach was taken for random guessability as user choice has no impact on the random guessability and the calculation is feasible. This is discussed in Section 4.1. Mathematical calculations of combinations or simulations alone were not appropriate for the second or third types of guessing attack as these aimed to exploit user bias, which could not realistically be predicted mathematically. Thus for the second and third types of guessing attacks, a combination of user studies and simulations were used, these are discussed in Sections 4.2 and 4.3.

4.1 Random Guessability

Assuming one passimage per challenge screen, x images per screen (i.e. the number of distractors per screen plus one passimage) and s challenge screens, the probability of correctly guessing the correct passimages for a challenge session is shown in Equation 4.1. This is calculated by multiplying the probability of one correctly guessing one screen, which is $\frac{1}{x}$ s times (where s is the number of screens). This provides the probability of correctly guessing each screen in succession.

$$\frac{1}{x^s} \tag{4.1}$$

The denominator of Equation 4.1 will be used directly to reflect the maximum number of random guessing attacks required before success, this is shown in Equation 4.2.

$$x^s \tag{4.2}$$

This reflects an estimate, and such an attack could be successful achieved with more or less attempts. Applying this to the passimages scheme (discussed in Chapter 2, Section 2.5.1) gives an example random guessability value as shown in Equation 4.3 .

$$9^4 = 6561 \tag{4.3}$$

4.2 Guessability for a User Group: General Population

User choice influences the security of passwords by making them susceptible to dictionary attacks. Thus user choice may also influence the security of recognition-based graphical passwords. This is highlighted by De Angeli *et al.* [16] who noted that the issue of predictability in user selected passimages still required evaluation at the time of writing. The research reported in this section aims to contribute evidence for a bias in user choice of images and considers the impact this could have on guessability.

Related works include that by Davis *et al.* [15] who examined the effect of user choice of images on the security of the Story and Face schemes. The authors estimated the probability of a given set of passimages (which they refer to as a password) being selected from either scheme. Password selection was restricted as images had to be selected from distinct categories (established by the experimenters), and only a subset of categories were presented to the user upon selection. To attack a password, the set of all combinations of passwords was first reduced based on the categories presented to the users upon selection (removing any passwords which contained images from any categories not presented to the user). The set of all passwords was then ordered by calculation of the probabilities using the model established by Davis *et al.* in [15]. The position of the passimage in this ordered list was then the number of guesses required to correctly guess

Figure 4.1: Example Passimage - Image: Maggie Smith / FreeDigitalPhotos.net

the user's password (passimage set).

It could be argued that the restriction of having only one image per category significantly increases the chance of guessing the passimage set due to the consequently reduced password space. The work also considers the probability of the whole passimage set being selected, and not the probability of images from individual categories being selected and this bias being exploited by selecting the image on a challenge screen from the "most likely" category. Accordingly there was scope to consider this type of attack. For the purposes of this research, this attack will be referred to as a semantic ordered guessing attack (SOGA). The attack is examined in this section with an aim to establish an estimate of the required number of attacks of this type against a user before success.

The passimages scheme, described in Chapter 2 was used in this research. An example passimage from the scheme is shown in Fig. 4.1. The scheme had a total of 144 potential images split into twelve semantic categories (this process will be discussed in Section 4.2.2) and three distractor selection algorithms were used as follows:

1. Random distractor selection - images other than the passimage were selected using a pseudo-random number generator.

2. Random distractor selection avoiding the passcategory - distractor images were randomly selected from all categories except the category to which the passimage belonged (the passcategory). Multiple distractors could be selected from the same category.

3. Random distractor selection from distinct categories avoiding the passcategory - distractor images were selected randomly from all categories except the passcategory, with at most one distractor selected from any given category.

4.2.1 Hypothesis

The research hypothesis of this experiment was as follows:

Users' choice of passimages will not be uniformly distributed between semantic

categories. This bias can be used to construct an attack which has a higher success rate than random guessing. The success rates of the attack are contingent on the algorithm used to select distractor images.

This was split into the following sub-hypotheses:

- H_1 User selections for passimages will be non-uniformly distributed between the semantic categories.
- H_2 Bias in user selections for passimages could be exploited to achieve more successful attacks than expected by random guessing.
- H_3 The distractor selection algorithm has a significant impact on the success rate of SOGAs.

To examine the research hypothesis, four stages were proposed:

1. Categorisation of the passimages into non-overlapping semantic categories.
2. Examination of the distribution of passimage choices between the categories to determine any user bias.
3. Exploitation of the bias as extrapolated from the second step by construction of an attack.
4. Analysis of attack success rates for different distractor selection algorithms to determine if there exists a correlation between the distractor selection algorithm and success rates of SOGAs.

The first two stages related to hypothesis H_1. The third stage related to hypothesis H_2 and the fourth related to H_3 .

4.2.2 Categorisation of Image Passwords

The first stage involved splitting the passimages into non-overlapping semantic categories based on the image content. For the passimages scheme there was considered to be little ambiguity over the image content and hence the images were categorised by the experiment conductor. This assumption may have influenced results, and this is discussed further in Section 4.2.7. 144 digital photographic images were split into twelve categories which were identified based on the main semantic content of the images. The categories were as follows:

- Food and Drink
- Cartoon and Fictional Characters
- Scenery
- Animals
- Faces and Body Parts

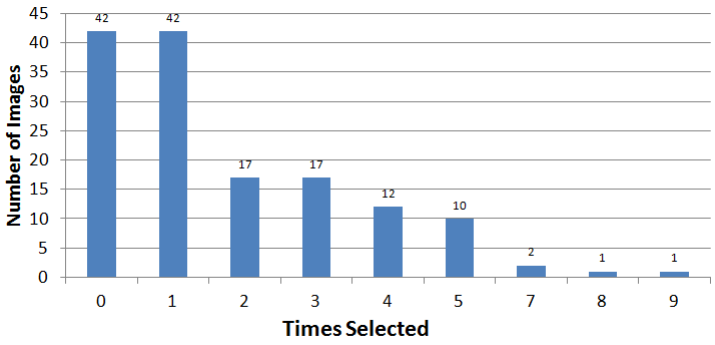


Figure 4.2: Passimages Image Selection Counts

- Transport
- Clothing
- Entertainment
- Trees, plants and flowers
- Skyscapes
- Buildings, tools and devices
- People

Each category had 12 images to remove potential bias in the selection due to more images being in any particular category. Example images are shown with their designated categories in Appendix A.

4.2.3 Passimages User Selection Results - H1 Analysis

A total of 64 individuals participated, primarily students and employees of the University of Glasgow. Each participant selected a passimage set of four images, resulting in 256 passimage selections. Of the 144 images, 42 images were not selected at all. This indicates potential for bias towards individual images, shown in Figure 4.2 which shows the number of images selected at each value of times selected (i.e. selected 0,1,2,... times).

The uneven distribution of user choice between semantic categories is shown in Figure 4.3, where error bars are also shown and were calculated as described in Appendix B Section B.2.3 (95% confidence intervals for proportions, details provided by Rumsey in [71, Page 207]). Food and drink was the most popular category with 37 selections; cartoon characters came a close second with 30 selections. The confidence interval for the category "People" could not be calculated as the condition $n\hat{p} \geq 10$ required to calculate the confidence interval was not met. The value of n was 256, the value of p was 0.01, resulting in a value of 2.56.

To prove the significance of the bias in user selection (H_1) it was necessary to establish a significant difference in the observed distribution of the image selections compared to a uniform distribution. The chi-square test was applied as

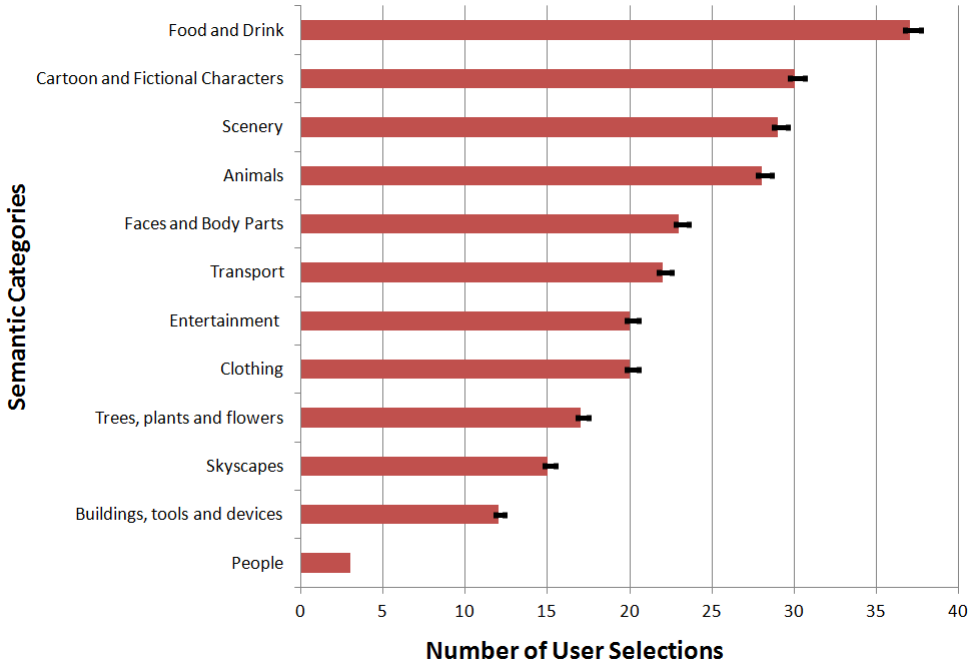


Figure 4.3: Passimages Semantic Categories Selection Distribution (with Error Bars)

this compares the probability of at least two samples of non-ordered categorical data to establish if they are statistically different from each other (see Howell, [45]). Table 4.1 shows the number of selections in each category and calculates the individual chi-square values ($\frac{O-E^2}{E}$ where O is the observed frequencies and E is the expected frequencies) to be summed to calculate the overall χ^2 value. The expected frequencies were calculated as the total number of selections (256) divided by the number of categories (12), giving an expected value of 21.33 selections in each category for a uniform distribution. The degrees of freedom is the number of independent variables in the final calculation less one (see Howell, [45, Page 521]). In this instance since there are 12 categories, minus one gives 11 degrees of freedom. Using a 0.05 significance level (giving a 95% confidence level) the value required to reject the null hypothesis was 19.68 (see Howell, [45, Page 697]). The individual values of $\frac{O-E^2}{E}$ sum to 42.78 which means the null hypothesis related to H_1 can be rejected. This contributes evidence of a significant bias in user selection of passimages.

4.2.4 Exploiting User Choice- Semantic Ordered Guessing Attack (SOGA) H2

To exploit the established bias in user choice of passimages, the attack proposed in this work is referred to as a "Semantic Ordered Guessing Attack" (SOGA). In this attack the attacker has knowledge of the categories from which passimages are most likely to be selected by users. It should be noted that a SOGA involves a human element in the categorisation of the images. The attacker could make the

Category	Observed Number of Selections	Expected Number of Selections	$\frac{(O-E)^2}{E}$
Food and Drink	37	21.33	11.50
Cartoon and Fictional Characters	30	21.33	3.52
Scenery	29	21.33	2.75
Animals	28	21.33	2.08
Faces and Body Parts	23	21.33	0.13
Transport	22	21.33	0.02
Clothing	20	21.33	0.08
Entertainment	20	21.33	0.08
Trees, plants and flowers	17	21.33	0.88
Skyscapes	15	21.33	1.88
Buildings, tools and devices	12	21.33	4.08
People	3	21.33	15.75

Table 4.1: Passimages Selection Analysis

assumption of what people would be most likely to choose given an authentication screen, or they could perform work to establish what kind of images users may prefer from a given image set. This information could be obtained in a similar method to that described in Section 4.2.2 by asking people to select images from the same group of images presented in registration. Alternatively, the attacker could make assumptions regarding the category bias and hence the "most likely" image.

To perform a SOGA, the attacker enters the username of the victim and is then presented with an authentication challenge screen which has a passimage from the user's passimage set and a selection of distractors. The attacker then attempts to authenticate by selecting the image from the most likely category given the challenge screen presented.

An example attack is illustrated by Figure 4.4 which represents a challenge screen where the attacker identifies the categories to which the images belong. The passimage has been highlighted by a square border for illustration only. The right side of Figure 4.4 shows the attacker ordering the categories from most likely (at the top) to the least likely (at the bottom). The attacker would then select the image from the category at the top of the list, in this case the attack is successful as the attacker picks the image of a steak.

To examine the significance of the success rate of SOGAs compared with random guessing (H_2), simulations of SOGAs were constructed by assuming the "perfect" attacker who has knowledge of the categories and bias of user choice within those categories. Each passimage selection was used to generate challenge screens using each of the distractor selection algorithms. A total of $256 * 3 = 768$ (256 selections multiplied by three distractor selection algorithms) challenge screens were generated and attacked. If the passimage for a screen was in the most likely category using the ordering established previously (when compared

Figure 4.4: SOGA Example

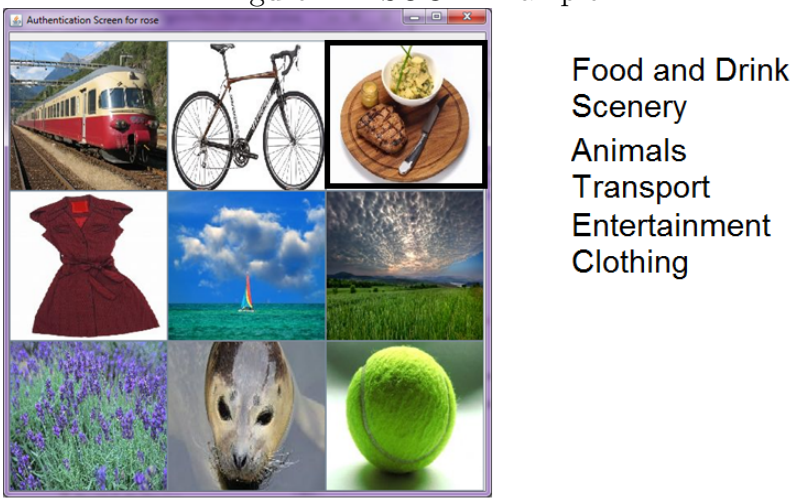
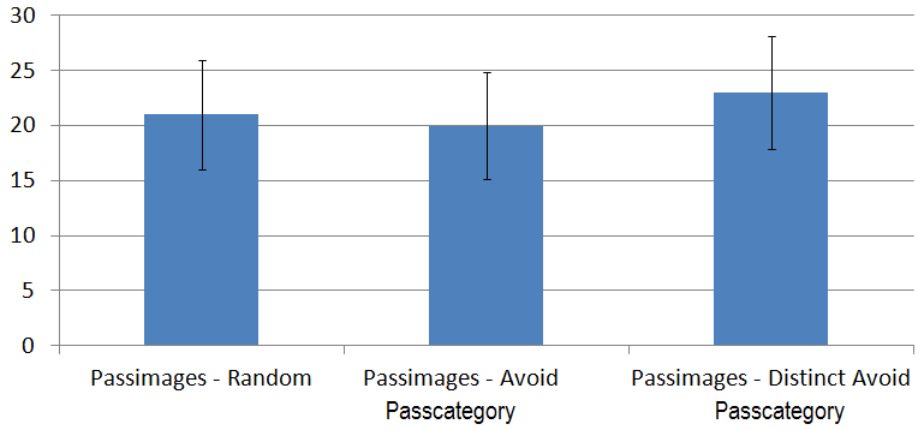


Figure 4.5: Percentage of Successful SOGA for Each Scheme Variation



to the categories of the other images on screen) then the attack was deemed successful.

The results of the SOGA for the passimages scheme are shown in Figure 4.5, with error bars calculated in the same way as for the user choice distribution. It can be seen from this figure that the most resistant distractor selection algorithm was random selection from categories other than the passimage category, in this case the attack was successful for $20\% \pm 4.9\%$ of the 256 attacks. The second most effective algorithm was random selection, which resulted in an attack success rate of $21\% \pm 4.99\%$. The worst resistance to attacks was with the distractor selection of distinct categories which resulted in an attack success rate of $23\% \pm 5.16\%$. This was as expected since a user bias in image selection towards more popular categories meant selection from the remaining categories for distractors (when a user has selected from a highly popular category) would mean the distractors will be from less popular categories providing optimal conditions for a SOGA.

The next step was to establish the significance of the success rates. To select an appropriate significance test, the distribution of the data needed to be established. The attack simulation data which resulted was a binomial distribution since it

	Successful Attacks	Failed Attacks	Total
Observed	53	203	256
Expected	28.44	227.55	256
Chi Squared Value	21.20	2.65	23.85

Table 4.2: SOGA Random Distractor Selection Chi-square Analysis

met the following criteria (as detailed by Rumsey in [71, page 135])

1. There were a fixed number of attacks (768).
2. Each attack had two possible outcomes, success or failure.
3. The probability of success (p) was the same for each attack ($\frac{1}{9}$, since nine images were shown per challenge screen).
4. The attacks were independent of each other i.e. success on one screen had no influence on the success or failure of subsequent attacks

H_2 stated that bias could be exploited to achieve a success rate significantly higher than that expected by chance. To establish this aim and since the data was binomial, it was appropriate to use a chi-square goodness-of-fit test to conclude if the observed frequencies of success and failure was significantly different from that expected by chance [45, page 142].

The number of expected successful attacks was calculated as $n * p$ where n was the number of attacks (256 for each distractor algorithms, 64 users selecting 4 passimages) and p was the probability of success on a given screen (which was $\frac{1}{9}$, the random probability of success where 9 was the total number of images on the challenge screen). The expected number of failed attacks was calculated as $n - (n * p) = 256 - (256 * \frac{1}{9}) = 227.56$, i.e. the total number of attacks less the number of expected successful attacks. The significance value was set at 0.05 to provide 95% confidence in the rejection of the null hypothesis related to H_2. Tables 4.2, 4.3, and 4.4 show the results with the chi-square values for each distractor selection algorithm. There was only one degree of freedom (since there were two categories, success or failure) and so using the table given by Howell in [45, Page 697] a chi-square value greater than 3.84 indicated significance. This allowed rejection of the null hypothesis related to H_2 for each of the distractor selection algorithms, showing SOGAs were more successful than random guessing for each distractor selection algorithm.

4.2.5 H3 Analysis - Establishing a Distractor Selection Contingency

The final hypotheses to be tested for significance was H_3. The aim of H_3 was to establish contingency between the algorithm for distractor selection and the

	Successful Attacks	Failed Attacks	Total
Observed	51	205	256
Expected	28.44	227.55	256
Chi Squared Value	17.88	2.23	20.11

Table 4.3: SOGA Avoiding Passcategory Distractor Selection Chi-square Analysis

	Successful Attacks	Failed Attacks	Total
Observed	58	198	256
Expected	28.44	227.55	256
Chi Squared Value	30.71	3.84	34.55

Table 4.4: SOGA Distinct Categories Avoiding Passcategory Distractor Selection Chi-square Analysis

success of the attacks. Due to this and the categorical nature of the data, it was appropriate to apply a contingency chi-square table provided by Howell on [45, Page 10, Figure 1.1]. The contingency table is shown in Table 4.5, where the expected values are shown in parenthesis next to the observed values. The expected values in a contingency table are calculated as $E_{ij} = \frac{R_i C_j}{N}$ where R_i is the total for the row of the related observed value, C_j is the total for the column of the related observed value and N is the total number of observations. The number of degrees of freedom are calculated as $(R-1)(C-1)$ where R is the number of rows (exclusive of headings and totals) and C is the number of columns (exclusive of headings and totals), thus Table 4.5 has 2 degrees of freedom. The total chi-square value was $\chi^2 = 0.61$, the value for significance (at 0.05 level, from Howell [45, Page 697]) is 5.99, and so the null hypothesis relating to H_3 could not be rejected. Thus, it is not clear if using a different distractor selection algorithm has a significant impact on the success rate of SOGAs.

Distractor Algorithm	Passes	$\frac{(O-E)^2}{E}$	Fails	$\frac{(O-E)^2}{E}$	Total
Random	53 (54)	0.019	203 (202)	0.005	256
Non-distinct Categories Avoiding the Passcategory	51 (54)	0.17	205 (202)	0.044	256
Distinct Categories Avoiding the Passcategory	58 (54)	0.30	198 (202)	0.079	256
Total	162		606		768

Table 4.5: SOGA Distractor Selection Algorithm Contingency Analysis

Distractor Selection Algorithm	*Percentage of Successful Attacks*	*Revised x Value*	*SOGA Value*	*Random Guessing Value*
Random	21%	4.76	513	6561
Passimages from categories other than the passimage category	20%	5.00	625	6561
Passimages from distinct categories other than the passimage category	23%	4.35	358	6561

Table 4.6: SOGA Results and Values Summary

4.2.6 Computing the Guessability

It has been shown by the work reported in this section that it is possible to construct a guessing attack based on the bias of user choice of passimages. This attack results in more success than would be expected by chance alone. The bias in user selection towards particular image categories was shown for the passimages scheme and was summarised in Figure 4.3. Success rates were examined for SOGAs with three distractor selection algorithms, and were shown to have statistically significant higher success rates than random guessing.

21% ±4.99% of passimage screens were successfully attacked where distractors were selected randomly (ignoring the semantic categories). 23% ±5.16% of passimage screens were successfully attacked where distractors were selected from distinct passimage categories (excluding the passimage category) and 20% ±4.90% of screens were successfully attacked where distractors were selected from passimage categories (excluding the passimage category). These figures relate to attacks on individual screens, and it should be noted that SOGAs are made more difficult by using multiple challenge screens.

A calculation of an ordered guessability value for RBGPs is now presented. This is achieved by first calculating a revised number of passimages per challenge screen and calculating the ordered guessability using Equation 4.1 with the revised number of passimages value. The steps are as follows:

1. Collect a sample of user selected images, a large sample is better.

2. Establish the bias in user choice by examining the categories of the passimages selected and ordering the categories from most to least popular.

3. Simulate screen generation and establish the percentage of screens from which the passimage is in the "most popular" category.

4. Calculate the revised number of images per screen by solving Equation 4.4 for x, the revised number of images per challenge screen, where p denotes

the percentage of successful attacks.

$$\frac{1}{x} = \frac{\mathrm{p}}{100} \tag{4.4}$$

5. Calculate the revised guessability as x^s where x is as calculated in Equation 4.4 and s is the number of challenge screens.

Using this approach, success rates and SOGA values are summarised in Table 4.6, where a comparison to the random guessing value is also shown. Passimage guessability varied between 10 and 18 times larger (rounding to the nearest integer) than predicted using the random guessing value for 9 images with 4 screens. It is acknowledged that steps 1-3 are the optimal solution in terms of calculating this for a realistic estimate of a specific scheme. However, in terms of assessing this value for the final metric it is recommended that the percentages obtained in this work are used as estimates. This limits the results, which is discussed further in the next section.

4.2.7 Limitations and Further Related Work

Limitations

There are some limitations of this experiment which are now discussed. The primary limitation of this work is that the percentages of success reported were specific to the users and passimages used in this experiment. It is proposed that if a more accurate rate were required for a different set of images with different categories the process of gathering the biases should be repeated. These biases could then be used in simulations to establish a more representative success rate.

There may also have been ambiguity in the image content resulting in a bias in the categorisation. To examine this further, a fellow researcher was asked to examine the semantic content of the images. They were asked to place the images into the correct category given the list of categories previously established in Section 4.2.2. Five of the twelve categories were exactly as established initially, however the remaining categories had differences in the distribution of the images resulting in a different number of images in these categories. This bias in categorisation could have had an effect on the success rates of SOGAs.

To determine if there was a significant statistical difference in the success of the SOGAs using the different category distribution, the simulations were performed again with the adjusted distribution. Chi-squared analysis was then applied to establish if the success rates were significantly different to that of the success rates for the original category distribution. Each of the three algorithms (random selection, non-distinct categories avoiding the passcategory and distinct categories avoiding the passcategory) had chi-square values less than the 3.84 value required for significance (using the table given by Howell on [45, Page 697] a value greater than 3.84 indicated significance for 0.05). The values were 0.0238, 0.881, and 0.357 respectively. Thus, the issue of differing distributions was not perceived to be a significant issue. However, there could also have been issues with the uneven distribution of images in the categories. If some categories had

less than 12 images it may not have been an equal comparison to the success rate when the original distribution had 12 images in it. To ensure no ambiguity, multiple examiners could have been used to establish the categorisation of the images. In particular, the food, animal, cartoon, clothing, and trees, plants and flowers provided the best categories for lack of confusion over the semantic category.

Another limitation of this work was that the attacks were performed against the same data used to populate the category biases. Ideally the data should have been separated into testing data and category bias data. This would have helped confirm the predictive power of the model. Also, attacks were performed against individual screens which does not reflect the success of attacking a user for a challenge session. Multiple challenge screens were accounted for by including the number of screens in the final calculation of the number of attacks required. This was to provide consistency with the random guessing value. An alternative approach would be to attack a user's whole passimage set and not individual challenge screens.

The attack algorithm was such that if a passimage was in the most popular category on a screen then the attack would be successful. This made the assumption of an optimal attacker. For this attack if there are other images in the same category then the attacker has a $\frac{1}{n}$ chance of success, where n is the number of images on the challenge screen in the most popular category.

It should also be said that distractor selection will likely not be the only potentially influential factor, though it was the only one explicitly examined in this work. To account for different factors the experiment should be repeated for multiple variables and examining the resulting success rates by use of a contingency table as conducted for the distractor algorithms examined in Table 4.5. Examples of variables could include the number of images on screen and the number of categories.

The attacker is assumed to have knowledge of the bias in user selection in this attack, this may not be a reasonable assumption. To examine this further it would be beneficial to perform an experiment in which participants were asked to categorise images. The level of agreement could then be evaluated. Participants could then be asked to rate the categories in order of which they believe would be most likely to be selected to the least likely. This could then be compared to the bias obtained and the differences evaluated.

Further Related Work and Discussion

In addition to the limitations discussed, related work by Hayashi *et al.* was published approximately six months after this work was completed. Hayashi *et al.* [40] considered educated guess attacks against the Use Your Illusion scheme where the users provide their own passimages. Educated guessing attacks are further subdivided into collective educated guess attacks (where guessing is based on the images thought to be more popular by a collection of users) and individualized educated guess attacks (where guessing is based on the images likely to be selected by an individual user). The authors performed two user studies to examine these attacks, and evaluated the use of the distortion of images approach taken in [39]

to mitigate these attacks. They presented five hypotheses, of which two were similar to this work:

1. "If a recognition-based authentication system lets users choose original, undistorted pictures as authentication images, an attacker can predict the images more accurately by using educated guesses than by guessing randomly. "

2. "Users tend to choose specific categories of images as their authentication images."

Differences between this work and that reported in [40] include that here an attack based on collective information about user biases was presented, where in [40] the authors asked participants with no knowledge of a user (or potential biases) to attack their images. Also, the images used were provided by the participants, and not selected from a provided set as in this work. Another difference is that participants were asked to guess three authentication images from a set of 27, this does not reflect the RBGP scheme under consideration in this work. In addition, the participants were asked to make their "10 best guesses", which also doesn't reflect the common format where guesses would be limited to one per screen.

For strangers' guesses (educated guess attacks) 3 of the 15 attackers correctly identified all three passimages from the set of 27 images shown within 10 guesses. These attackers reportedly selected the images which had similar properties, highlighting a connection between the images taken by the victim which was exploited, unlike the attack proposed in this work which takes commonly selected images and prioritises them for automated guessing. Hayashi *et al.* also independently identified the attack detailed in this work (though they do not call it a semantic ordered guessing attack). This contributes evidence of the relevance of this work.

It is also possible that the bias established could be for the image set used here only. This is unlikely as bias in user choice of images has been demonstrated elsewhere (e.g. [13],[15], and [40]). Ideally, to establish the success rate for a different set of images, a similar experiment should be conducted with a representative population and the passimages under consideration. An attack could be constructed which considers the bias to a specific image rather than a category. The attack would be adjusted slightly to establish a bias to individual images and then performed again using this ordering. If this were extended to a complete passimage set, this would be very similar to the work reported by Davis *et al.* in [15], the only difference would be in the ordering obtained by Davis by calculations but in this work it would be by ordering the frequency of selections. Another difference would be that Davis's work covers all possible combinations of passimages into passimage sets, where extending this work as described would not.

The SOGA will only work where images can be split into semantic categories, and could be avoided completely if images are assigned to users or all images are presented from the same category. However, assigning images from the same category as the user's passimage has resulted in an increased number of incorrect selections as reported in [16] where the VIP3 configuration presented challenge screens from one category to a user. This was termed "intra category error".

4.3 Guessability for Individual Users

The final aspect of guessability is related to bias in individual user choice of images. It is conjectured that if an attacker has knowledge of the user's preferences, they may be able to guess the user's passimages more successfully than expected by chance.

As discussed previously, Hayashi *et al.* also carried out experiments which evaluated "individualized educated guess attacks" for photographic passimages taken by the user [40]. Attackers were given "10 best guesses" to guess the three authentication images of the user based on personal knowledge of the user. Eight out of 15 attackers correctly identified the user's set of three passimages within 10 guesses.

To examine guessing for an individual user where the images are predetermined for the scheme and not uploaded by the user, a simple approach could be as follows. Users could be asked to volunteer a close friend or relative to guess the images they had selected for authentication. However, the approach would be flawed. The attacker's chance of guessing the correct images would increase if they were shown authentication screens with only nine images (one passimage and eight distractor images) compared to the complete set of potential passimages. Thus, the experiment could not accurately reflect an attacker trying to gain access to the system by posing as a legitimate user. In an attempt to better examine this a larger experiment was conducted of which known-user guessing was a part, this is reported in Chapter 6 Section 6.1.

4.4 Conclusion

The result of this evaluation was two guessing models to be used in the final metric, one model related to random guessing (an adjustment to an approach which has already been reported in other research) and the other related to guessing based on bias in passimage selection from semantic categories. The first model is as shown in Equation 4.5, where x denotes the number of images per challenge screen and s denotes the number of challenge screens in a challenge session.

$$guessing\,value = x^s \tag{4.5}$$

The second model related to semantic ordered guessing attacks. Depending on the distractor selection algorithm, different success rates were achieved for the SOGAs. The success rates of simulations for the image set and distractor selection algorithm were used to calculate an "ordered guessability" value by calculating a revised value for the number of images per screen by solving for x in Equation 4.6, where the percentage of successful attacks is denoted by p. This value was then used in the normal calculation for guessability as discussed in Section 4.1. The final equation for semantic ordered guessability is then as presented in Equation 4.7

$$\frac{1}{x} = \frac{p}{100} \tag{4.6}$$

$$semantic\,ordered\,guessability = (\frac{100}{p})^s \tag{4.7}$$

It is proposed that, if unable to obtain ordering data for the image set to be used for a given scheme, the estimates provided in Section 4.2.4 be used to calculate a SOGA value. The values for success are based on the ordering and selections of 64 individuals for the passimages scheme based on a collection of 144 images split equally into 12 categories. Due to this, one might argue that these results are not directly applicable to other schemes as they would have potentially different categories and user selections. Whilst this is true, it still provides an estimate and a basis of comparison. If one desired a more accurate value for percentage of success for their particular system, the program which simulates the SOGA could be updated to incorporate a collection of users choices and the image details with their corresponding categories. This could be used to establish an ordering for the categories and a simulation to establish a percentage of success could be carried out.

At this stage, the guessability of RBGPs has been examined with further research on the observability still to be completed. At this point it was decided that an initial elementary metric be established to examine the feasibility of the thesis statement. This is reported in the following chapter.

Chapter 5

Elementary Security Metric

5.1 Security Analysis and Results

At this stage it was decided that a preliminary measure of the level of security (incorporating analysis to date) be established to ensure the feasibility of the thesis statement. A heuristic approach was taken at this stage as this could be constructed based on knowledge already available. To construct a basic heuristic model for evaluating the security of RBGPs the proposed approach was as follows:

1. Examine the relevant literature and extract the possible countermeasures to the attacks presented in the threat model.

2. Abstract the identified countermeasures from specific implementations to general approaches.

3. Construct a series of "key questions" regarding countermeasures for each attack which determine a level of resistance to the attacks.

4. Construct flow charts which combine the key questions for each attack to establish overall levels of resistance. Flow charts were used as they were considered a simple way of combining the key questions in a way which allows a final value to be easily determined.

5. Combine the flow charts to provide a metric of security which reflects resistance to the attacks modelled. Resistance to attacks can also be dependent on the context in which the authentication mechanism is used, this is discussed further in Section 5.4.

Area of concern and a subset of related attacks from the threat model shown in Figure 2.6 (page 37) subject to the scope established in Section 2.5 (page 46) are considered in this metric. Specifically, semantic ordered guessing attacks, shoulder surfing attacks, and frequency attacks are considered in Sections 5.2 and 5.3.

5.2 User Generic Guessability: SOGAs

Chapter 4 discussed the impact of bias in user choice on guessability. The experiment involved collecting a number of user selections for passimages which belonged to distinct semantic categories. The number of selections in each category was then counted and using these frequencies, categories were ordered from most to least probable (to be selected). An attack was then launched by constructing a challenge screen for each passimage selection and then checking if the passimage was from the most likely category given the bias established. If the passimage was in the most likely category for the screen presented, then the attack was successful. Bias in user choice significantly increased the chance of success when compared to random guessing. From the study reported in Chapter 4, the following key questions were identified to be incorporated into the model for SOGAs:

- Are images assigned to the user?
- Are the images split into semantic categories by the authors who proposed the scheme? For example, the VIP schemes presented by De Angeli *et al.* [16].
- Is there only one distractor per category?

The evaluation of semantic ordered guessing is presented in Figure 5.1, a flow chart which combines the aforementioned key questions relating to the configuration of a RBGP scheme which coincide with the results of Chapter 4.

5.3 Observability

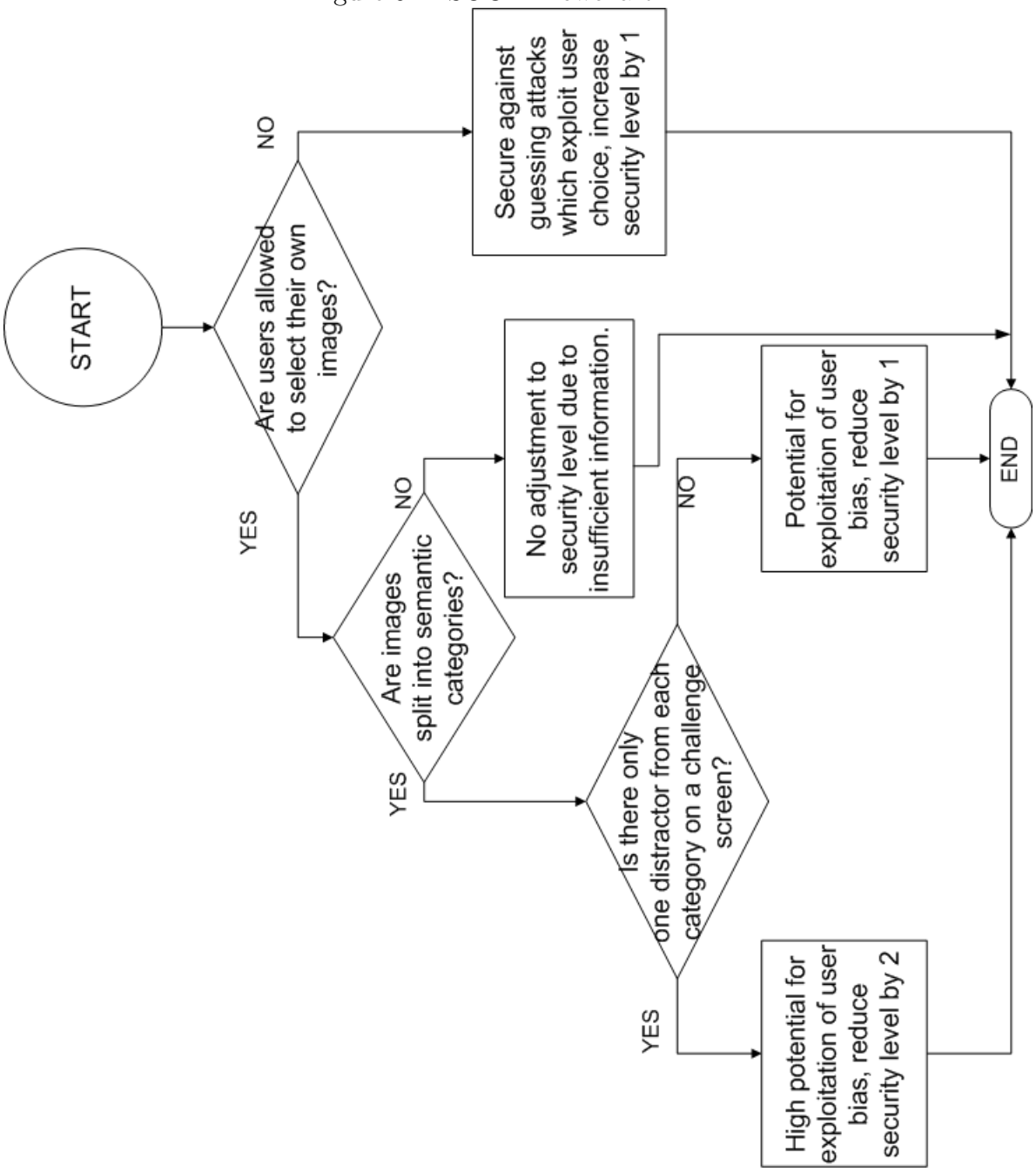
5.3.1 Shoulder Surfing

An examination of shoulder surfing literature was presented in Section 2.4.2. This section provides a summary of the results of the review. This allows a list of key questions to be established which are then incorporated into a flowchart modelling shoulder surfing.

On a high level, there are three approaches to mitigating shoulder surfing attacks. Tao and Adams propose that countermeasures for a shoulder surfing attack can be placed under two categories; using no indicators of passimage selection or disguising indicators of passimage selection [86]. This is extended here by inclusion of an additional approach taken by DeAngeli *et al.* in [17] and Dunphy *et al.* in [21], where the number of passimages in a user's passimage set (which DeAngeli *et al.* termed "key image portfolio") exceeds the number of challenge screens in a challenge session. This means that in any authentication session, a subset of the users passimage set is presented.

There are a number of implementations of countermeasures for shoulder surfing. Examples of schemes where no indicators of passimage selection are shown include the work by Komanduri and Hutchings (who allow keyboard selection of passimages [50]) and Wiedenbeck *et al.* (who allowed indirect selection by

Figure 5.1: SOGA Flowchart



a convex-hull click scheme [101]). Examples of schemes where image selection is disguised is presented in PassFaces where, upon selection of the passface, a "mask" is applied to all faces on the challenge screen [66] .

For the purposes of the shoulder surfing flow chart, the key questions were established as follows:

- Does the scheme provide details of whether an image is selected on a challenge screen?
- Does the scheme highlight the passimage on selection?
- Does the scheme disguise passimage selection?
- Does the scheme use more passimages than challenge screens?
- Does the scheme allow keyboard selection?

These questions were combined into the flowchart shown in Figure 5.2.

5.3.2 Frequency/Intersection Attack

An examination of literature related to frequency and intersection attacks was presented in Section 2.4.2. This section provides a summary of the results of the review which allows a list of key questions to be established, which are then incorporated into a flowchart modelling frequency attacks. The calculation for the number of attacks was not incorporated into the metric at this stage. This is because the heuristic approach is a preliminary attempt at constructing a metric to explore the feasibility and similar models for the remaining attacks had not yet been established. As noted by Takada *et al.*, a frequency attack is an attack in which the attacker records multiple challenge screens and notes the images which occur with the highest frequencies then select these images in an attack [85]. Dhamija and Perrig [20] successfully summarise general approaches to countermeasures for intersection (and hence frequency) attacks as:

- Use the same distractor images and passimages for each session.
- Repeat a small subset of distractor images for each passimage. This would result in an attacker recording the same frequencies for these distractors and the passimage and the attacker would be unable to tell which is the passimage. Thus, they would have to randomly select one of the images with the same frequency of occurrence.
- In any given challenge session, if a user selects a distractor on a challenge screen, subsequent screens only display distractor images - "dummy screens".
- Implement a limit on the number of incorrect authentications a user can perform, this stops an impersonator attempting to discover all of the images. (A "three strikes and you're out" approach which is classified under resistability for this work and so is outside the scope)

Figure 5.2: Shoulder Surfing Flowchart

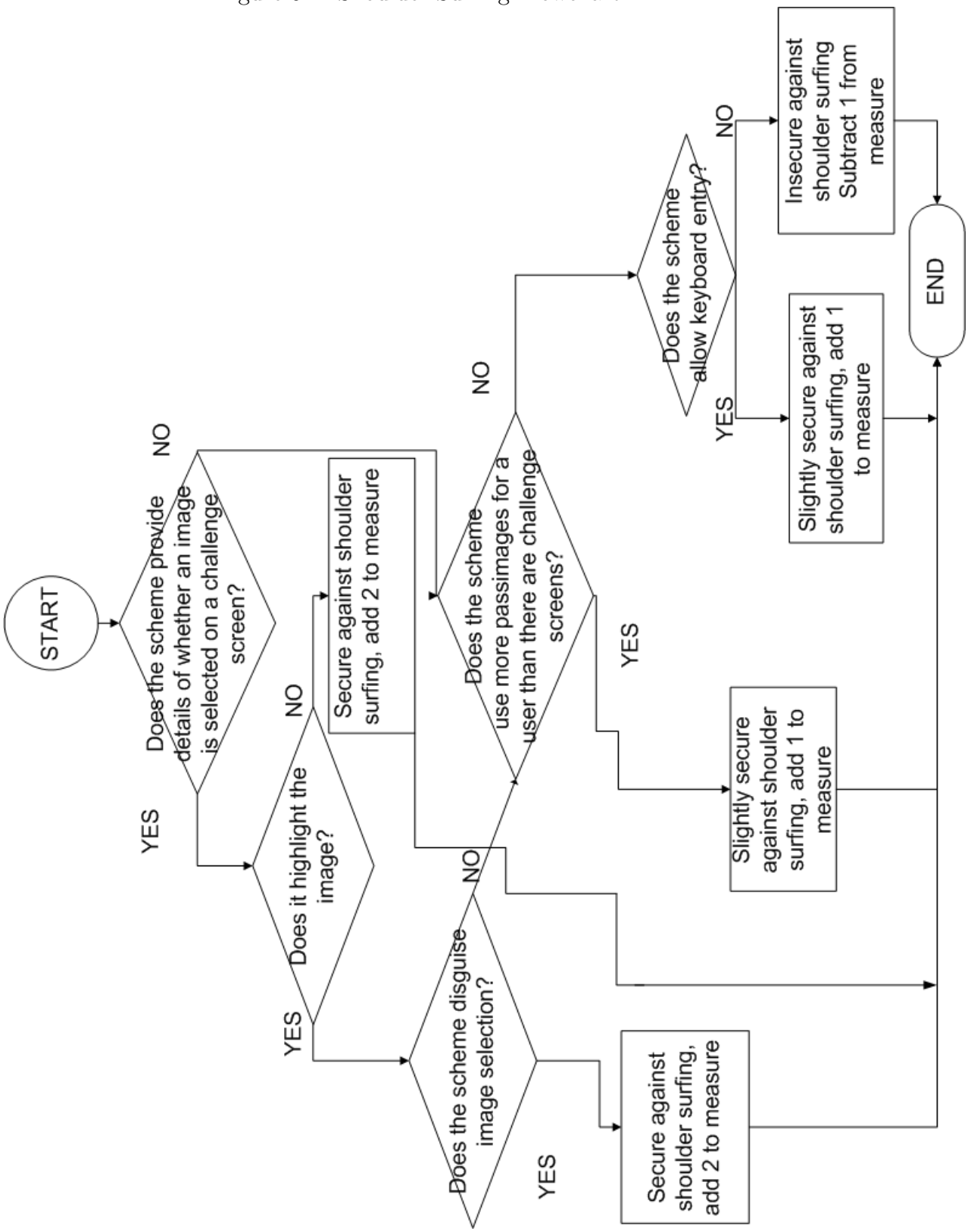
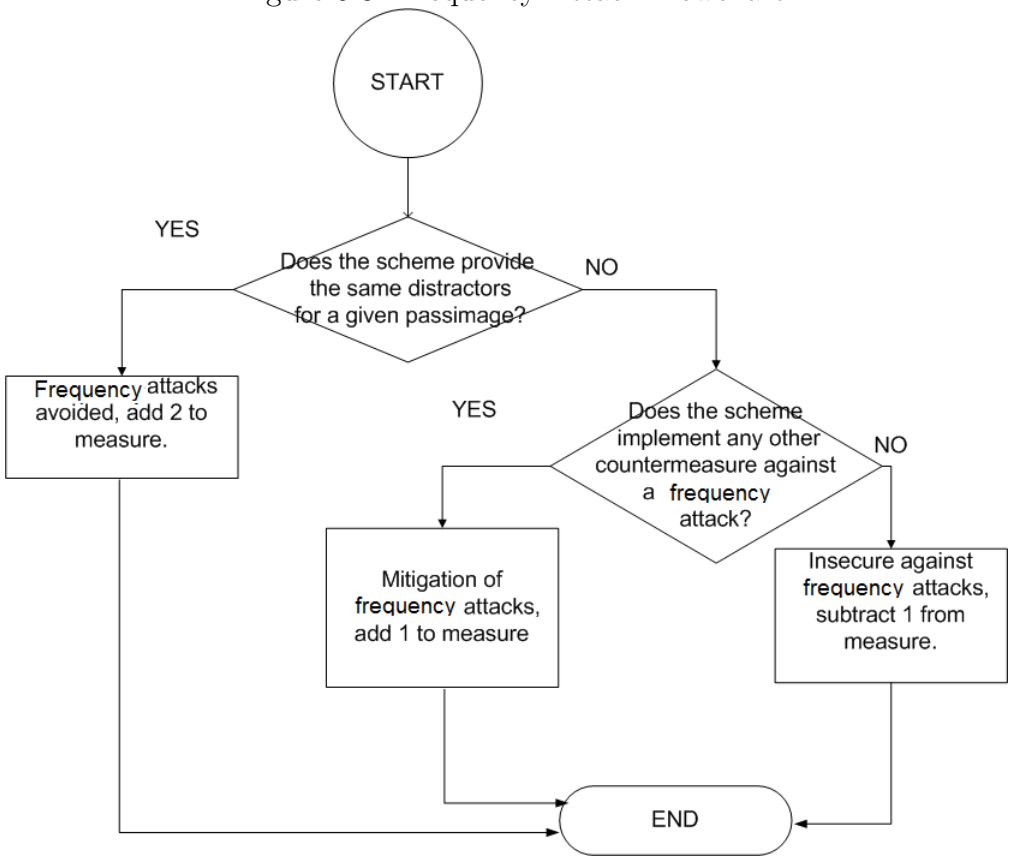


Figure 5.3: Frequency Attack Flowchart



Dhamija *et al.* note that these solutions are not perfect, as reuse of distractor images may result in users recognising distractors and selecting the wrong image [20]. However, maintaining the same distractors for a given passimage does ensure that intersection and frequency attacks are not possible, since each image appears with equal frequency. The remaining three options only serve to minimise the potential for an intersection or frequency attack.

Figure 5.3 shows the flowchart for frequency attacks which combines the following key questions:

- Does the scheme provide the same distractors for a given passimage?
- Does the scheme implement any other countermeasure against a frequency attack?

As with the shoulder surfing value, it was not possible to differentiate between the efficacy of different countermeasures, again indicating further work. Contributions to this area are presented in Chapter 7.

5.4 Heuristic Model for Security Evaluation

The heuristic model for security evaluation presented here arose from the combined flowcharts shown in Figures 5.1, 5.2, and 5.3 which covered semantic ordered guessing, shoulder surfing, and frequency attacks respectively. Each factor

provided one value of a 3-tuple, where the score for each value starts at 0 and is increased or decreased depending on the route taken through the appropriate flow chart. A tuple approach was considered appropriate instead of combining values (e.g. by summing the scores) or using a Euclidean metric since the interpretation of security is context sensitive.

For example, in the context of authentication in a home environment where no other individual is present, a negative shoulder surfing value would not be a concern. Thus, it would not be appropriate to reduce the overall security score due to this. Another approach could have been to weight the individual values before combining them. However, this does not remove the need to apply context. The resulting tuple represents the security of a RBGP scheme in terms of these aspects, that is {guessing value, shoulder surfing value, frequency value}. For example a scheme which is the most insecure in terms of the model presented would have a 3-tuple of {-2,-1,-1} whilst the most secure setup would result in a 3-tuple of {1,2,2}. Where insufficient information is available to define a score, the notation * is used.

In general, where countermeasures are implemented scores are increased by one, where no countermeasures are implemented, scores are decremented by one. In the shoulder surfing flow chart (Figure 5.2), there is the possibility of increasing the security value by two. This is due to the perceived significance (identified by the quantity of research in this area) of the threat caused by shoulder surfing resulting in the graded levels of counter measures available as described in Section 5.3.1. In addition, there are two instances where no adjustment to the security level is possible. The first case is related to images in a semantic ordered guessing attack (shown in Figure 5.1), where if images are not separated into semantic categories, no adjustment is made. In this case it is not possible to determine how successful a more specific ordered guessing attack, where images are given priority rather than categories, might be without further analysis. The second instance is when considering frequency attacks, where there are counter measures which mitigate (but do not remove the possibility of) frequency attacks.

5.5 Examples

5.5.1 Application To PassFaces Scheme

An example is provided by application to the PassFaces scheme [1]. From reviewing the white paper "The Science Behind PassFaces" [66] the following information regarding the configuration of the PassFaces scheme was extracted:

- The basic configuration includes assignment of four passfaces to a user and the use of four challenge screens, each showing one passface and eight distractor faces.
- The order of faces on the screen is random.
- No challenge screen contains faces from the other challenge screens. Distractor faces are similar in appearance to the passface. It is not clear how the similarity is established.

Scheme	Tuple
VIP	(2,1,*)
Faces/Story	(*,1,1)
Use Your Illusion	(*,1,1)
Deja Vu	(*,1,1)
PassFaces	(1,2,2)

Table 5.1: Popular Schemes Elementary Metric Values Summary Table

- Challenge screens are constant, i.e. the same distractors are selected each time for a given user's passfaces.
- Keyboard selection of a passface is permitted.
- A "mask" is applied to all faces on a challenge screen upon selection, this obscures the image selection.

From this information it was possible to rule out semantic ordered guessing attacks completely due to the random assignment of passfaces, which gives a secure semantic ordered guessability value of 1. In terms of shoulder surfing the mask application after selection meant image selection was disguised, resulting in a shoulder surfing security value of 2. If there was no masking, the use of keyboard selection would result in a shoulder surfing value of 1. However the highest of the two possibilities was considered appropriate as this represents the most secure configuration possible. Since the scheme provides the same distractors for a user's passface, the scheme has a frequency security value of 2. This results in a security 3-tuple of {1,2,2}.

5.5.2 Other Examples

Applying the same process to VIP, Faces/Story, Use Your Own Illusion and Deja Vu schemes resulted in the values as presented in Table 5.5.2, where * denotes a lack of sufficient information from the defining literature. It can be seen from this table that insufficient information is available for most schemes, and PassFaces provides the most secure configuration by this assessment.

5.6 Metric Evaluation and Conclusions

To evaluate the metric, each of the requirements established in Chapter 3 Section 3.4 (page 56) are considered in turn.

5.6.1 Repeatable

Repeatability, as previously discussed, requires that the metric produce the same result when applied multiple times by the same evaluator. Repeatability was evaluated by having an evaluator apply the metric twice to the schemes summarised

in Table 5.2. The schemes evaluated were a subset of those presented in Table 2.2 (page 45), selected as they represented the schemes with the most available information on their configuration. The schemes were evaluated based on the information provided to produce values for the tuple. The results are shown in Tables 5.3, 5.4, 5.5, 5.6, and 5.7 by the rows Participant 1 and Participant 1-repeat. Repeatability was achieved in all schemes except PassFaces. This was due to the fact that keyboard selection was allowed by the scheme, but it was unclear whether this constituted disguising selection. If this metric had not been an elementary attempt, this would have been repeated with multiple evaluators. However, this was to ensure feasibility of the overall aim of the thesis and so this was deemed sufficient.

5.6.2 Reproducible

To check reproducability, the details shown in Table 5.2 were provided to another researcher along with a Java program which asked the key questions relating to the setup of the scheme. As a result the program produced details of the metric values. This was then measured against the results achieved by the author to ensure reproducability.

There was an issue with reproducability for the PassFaces scheme in particular. This was due to multiple possible countermeasures for shoulder surfing and confusion upon interpretation of the questions proposed. This indicates that reproducability is not guaranteed by this approach and so the metric does not achieve the requirements established. The results are shown in Tables 5.3, 5.4, 5.5, 5.6, and 5.7 by the rows Participant 1 and Participant 2.

5.6.3 Quantitative

Whilst the resulting tuple is quantitative, it is very high level. As noted for similar attempts by Mihajlov *et al.* [55] and Renaud [68]. Whilst this is a useful starting point, it would be beneficial to establish a metric which provides an estimate of the number of attacks required before success. This could allow schemes to be compared in terms of security at a level of finer granularity.

5.6.4 Objective

Objectivity is achieved for the metric as it depends only on the configuration of the RBGP scheme, and not on the evaluator's perception of it. However, as noted for reproducability, the key question approach introduced an element of ambiguity due to the interpretation of the questions which may affect objectivity.

5.6.5 Extensible

The metric is designed in such a way that it is extensible. If further research is conducted, the results can be incorporated into the initial framework by further key questions to already established attacks. Also, addition of new attack flow charts which would result in additional elements in the tuple.

Scheme	Image Assignment	Image Highlighting and Passimage Set Size	Intersection Setup
VIP	Assigned to users, split into semantic categories	No details on highlighting. Number of passimages exceeds the number of challenge screens	No details
Faces/Story	Users select , split into semantic categories	No details on highlighting. Number of passimages exceeds the number of challenge screens	Constant set of distractors for given passimages
PassFaces	Assigned to users, split into semantic categories	Keyboard entry allowed. Images are highlighted by a border. No details on passimage set size.	Constant set of distractors for given passimages
Use Your Illusion	Users select	No details on highlighting. Number of passimages exceeds the number of challenge screens	Constant set of distractors for given passimages
Deja Vu	Users select	No details on highlighting. Number of passimages exceeds the number of challenge screens	Constant set of distractors for given passimages

Table 5.2: Popular Schemes Configuration Details Summary Table

Participant	Guessing Score	Shoulder Surfing Score	Frequency Score
Participant 1	1	1	*
Participant 1 -repeat	1	1	*
Participant 2	1	1	*

Table 5.3: VIP Scores

Participant	Guessing Score	Shoulder Surfing Score	Frequency Score
Participant 1	-1	1	2
Participant 1 -repeat	-1	1	2
Participant 2	-1	1	2

Table 5.4: Faces/Story Scores

Participant	Guessing Score	Shoulder Surfing Score	Frequency Score
Participant 1	1	1	2
Participant 1 -repeat	1	2	2
Participant 2	1	2	2

Table 5.5: PassFaces Scores

Participant	Guessing Score	Shoulder Surfing Score	Frequency Score
Participant 1	*	1	2
Participant 1 -repeat	*	1	2
Participant 2	*	1	2

Table 5.6: Use Your Illusion Scores

Participant	Guessing Score	Shoulder Surfing Score	Frequency Score
Participant 1	*	1	2
Participant 1 -repeat	*	1	2
Participant 2	*	1	2

Table 5.7: Deja Vu Scores

5.6.6 Conclusion

The aim of this chapter was to establish the feasibility of achieving the research thesis statement. The approach taken analysed potential attacks and countermeasures and as a result constructed a series of key questions in order to determine resistance to these attacks. The attacks included were semantic ordered guessing, shoulder surfing, and frequency attacks. Analysis of each aspect resulted in a flowchart which provided one score of a 3-tuple. The score for each factor started at 0 and increased or decreased depending on the route taken through the corresponding flowchart. The resulting tuple represented the security of a RBGP scheme in terms of these factors. The following issues were identified with the metric established when evaluated against the requirements:

1. Reproducablity was not guaranteed.
2. Values reported were quantitative, but too high level.
3. The frequency attacks value was limited as the efficacy of the different countermeasures was unclear.
4. Shoulder surfing efficacy of different countermeasures was unclear.
5. The flowchart approach introduced ambiguity.
6. The information presented to the second evaluator was organised to optimise use of the metric. This may have resulted in more success than would be expected if an evaluator had to read the documentation for the scheme and extract the relevant information.

The remainder of this dissertation aims to construct a different model which meets all the metric requirements and corrects the issues highlighted here. Specifically Chapters 6 and 7 aim to address points two, three, and four. Chapter 8 presents the final metric and evaluation.

Chapter 6

Observability Attacks

At this stage the feasibility of constructing a metric for RBGPs which models guessability and observability attacks has been confirmed. Several aspects of guessability have also been examined. It remains to examine and model observability attacks. To estimate the success rate of the observation attacks under consideration (shoulder surfing and intersection attacks) two approaches were identified for data gathering as follows:

1. User studies
2. Simulations and mathematical modelling

The first approach involved recruiting a group of users who worked in the same office/computer lab on a regular basis (approximately five days a week), and asking them to authenticate and attack each other. This approach was the first attempted, but was unsuccessful due to lack of participation in the experiment. This is discussed further in Section 6.1. Simulation and modelling was the final approach taken and the results are discussed in Sections 6.2 and 6.3. The final metric which incorporates these results is presented in Chapter 7.

6.1 User Studies for Observability Data

The first approach to gathering data on the success rates of observability attacks was a user study. In this study, the aim was to establish the frequency of different attacks (both guessing and observation), success rates of those attacks and the impact of countermeasures on their success rates. Guessing was included in this experiment with the aim of gathering data on known user guessing. This was approached by constructing an online experiment which allowed users to authenticate (using a RBGP scheme) and attack each other's passimages.

Adams and Sasse [2] highlighted that password mechanisms are an enabling task, they are an action performed to gain access. The main goal of authentication is to gain access to a service such as e-mail. It is due to this that performing experiments where authentication is the primary goal will not provide realistic results. This is also highlighted by Sasse *et al.* [76]. Thus the end goal (from a user's perspective) for this experiment was not authentication, but was use of a web-based forum which offered Java programming advice.

A web-based approach was taken as short-term use within a lab environment is not an accurate prediction of user behaviour (as argued by Beautement *et al.* [4]). A forum was selected since it would be relatively easy to implement and use of web-based forums was popular. Such a goal would aim to emulate a normal authentication situation. The web-based approach aimed to reduce the lab environment and reflect a normal environment. Another end goal option considered included a game scenario, however the time and effort required to implement this would have been excessive.

The forum used the passimages RBGP scheme (as described in Section 2.5.1). After two weeks using the forum, participants were asked to attack their classmate's passimages through an "attack mode". It was emphasised to participants that the attacks were artificial and would not grant access to the victim's account, but would count towards a running total of correct attacks. The participant at the end of the experiment with the most successful attacks would be awarded a prize in the form of a gift voucher. This was to incentivise attacks aiming to maximise participation. The high level aim of the experiment was to examine the nature of the attacks performed, the distribution between the attack types and their relative success. The design of the experiment was refined further by the hypothesis and variables presented in Section 6.1.1.

6.1.1 Hypotheses and Variables

Two hypotheses were established as follows:

- **H1** Attacks will not be evenly distributed between guessability and observability.
- **H2** Countermeasures for the identified attacks result in a significant reduction in the number of successful attacks.

To examine these hypotheses, users registered for the website and were allocated to one of four groups. An approximately even distribution between the groups was achieved by allocating the next participant to the next group and so forth. There was no control over who participated and when they registered, thus assignment to the groups was effectively random. The groups were as follows:

1. No Countermeasure Group (the control group) - the authentication configuration for this group involved no countermeasures and was intended to reflect the most insecure configuration for a RBGP scheme. In particular, it highlighted image selection (maximising potential for shoulder surfing attacks) and had no constant distractors for passimages (maximising potential for frequency attacks). The configuration used four passimages and four challenge screens.

2. Anti-Guessing Group: this configuration was designed to minimise potential for guessing attacks. The approach taken involved presenting the user with images from the same category as the passimage. This was to avoid a semantic ordered guessing attack. There was the possibility that the guessing attack could be more specific e.g. the user likes dogs. However, the

alternative anti-guessing approach was to assign images to users and this was dismissed as it would completely eradicate guessing attacks.

3. Anti-Observability Group: this configuration was designed to minimise observation attacks (specifically frequency and shoulder surfing attacks). The following countermeasures were implemented: the same challenge screens were used for each passimage (i.e. constant challenges, therefore no frequency attacks), no feedback was provided to the user when an image was selected on screen (to minimise shoulder surfing). Keyboard entry was not used as it was difficult to implement on the web, avoiding highlighting of the image was deemed sufficient.

4. Anti-Attack Group: this configuration combined the countermeasures from the anti-guessing group and the anti-observability group providing a comparatively attack resistant configuration.

The independent variables of the experiment were as follows:

- countermeasures against observability (the anti-observability configuration)
- countermeasures against guessability (the anti-guessability configuration)

The dependent variable was the number of successful attacks in each group where the countermeasures were implemented. The control group was set up with no resistance to any attack, establishing a base level of successful attacks to compare the success rates for the anti-observability and anti-guessing groups.

6.1.2 Forum Implementation Details

There were a number of aspects of implementation which required consideration, this section discusses these aspects. Masters in Information Technology students were invited to participate in the experiment. The group was selected as they worked together, knew each other, had a working knowledge of computers, and could potentially be authenticating frequently in front of each other in their computer lab. This was an important consideration because the potential for exploiting observability and guessability needed to be maximised. Observability aspects required close proximity and guessability required knowledge of the user.

It was possible that an individual might forget their passimages, in this case participants were asked to email the experimenter to reset their images. The email address from which a request was received was checked against the email address registered to stop participants attempting to reset other participant's images. There was also potential for a participant to cancel part way through an authentication session. If this happened incomplete data was recorded but not used. The data logged included an indicator of success or failure of authentication, the date, time and duration of the authentication process. Incorrectly selected images were not recorded as this data related to memorability and was outside the scope of this work.

To ensure a participant did not attempt to duplicate entries in the database for successful attacks the session variables relating to an attack were deleted.

Authentication Count	Number of Users
< 10	11
32	1
36	1

Table 6.1: Forum Authentication Sessions Summary

If the page was then reloaded the data was not re-inserted into the database. Attack data recorded was the same as the authentication data with the addition of the number of passimages correctly identified, the ID of the attacker and the ID of the victim. Upon completion of an attack, users were asked to fill in a short questionnaire relating to how the data which resulted in the attack was gathered. This questionnaire is presented in Appendix C.

To minimise the potential for one participant to create multiple fake accounts, an email address was required upon registration and was checked for validity. A valid email was required for the purpose of distributing a prize used to incentivise participation, so it was in the user's best interest to supply a real address. The prize was selected to be small enough to encourage participants, but not so large that they would go to excessive extremes to break or dupe the system. This was awarded for the most successful attacks. An alternative would have been to award the participant who launched the largest number of attacks, however this could have resulted in multiple "wrong" attacks on any one account where the attacker had no interest in trying to successfully authenticate. In the end there were no successful attacks, and so the prize was awarded randomly.

Another consideration was for the anti-attack group in which there was an issue with the number of images available in each category. If the user were to pick four images in one category there would be insufficient images for distractors. To remedy this, users in this group were restricted to one image per category. If sufficient images were gathered to remove this restriction, this could potentially have provided too much choice for users to select their images from. This could have potentially skewed the results (since users may select images shown first rather than traversing the whole set of potential passimages) more than restricting the users to one image per category.

6.1.3 Results

A total of 13 users participated in the experiment. Whilst the users agreed to participate, participation was desultory. As a result, the data gathered was also limited and insufficient to draw conclusions from. For this reason, an analysis of results would have no purpose. A summary of the activity is provided in Tables 6.1 and 6.2 which summarise the authentication sessions and attack sessions respectively.

An alternative approach would be to have smaller controlled lab-based studies for each attack type. However, there would still potentially be an issue with recruitment of participants. In addition, as noted by Salkind, controlling the experiment extensively could result in a loss of generalisability [75, Page 137].

Attack Type	Number of Attacks	Relationship	Successful Attacks
Random Guessing	13	Stranger	0
Random Guessing	1	Acquaintance	0
Knowledge of User Guess	1	Acquaintance	0
General User Bias	1	Acquaintance	0
General User Bias	1	Stranger	0

Table 6.2: Forum Attacks Summary

6.1.4 Experiment Limitations

In addition to the lack of data, there were a number of limitations in the experiment design, these are summarised as follows:

- Attack Mode- There was a possibility people did not use the "attack mode" to attack other users, and instead attacked them by attempting a normal authentication session as the victim. Participants would have no incentive to perform an attack outside the "attack" mode since it would not be noted as an attack and hence would not be counted towards the prize.

- Security awareness - There may have been limitations to the awareness of how to attack the system by the participants. The attacks could also have been influenced by the post-attack questionnaire which indicated potential attack methods. Thus not all attackers would be optimal as assumed in the attacker model.

- Variables- There were a number of variables which were not included in the experiment design, e.g. the number of distractors per screen, and the number of challenge screens.

- Varied levels of countermeasures- A number of counter measures were not examined, e.g. use of dummy screens. In addition, there are different numbers of constant distractors which can be used, these were not examined.

Due to these limitations and the lack of participation, the next approach was to consider simulations. This is discussed in Sections 6.2 and 6.3.

6.2 Shoulder Surfing Attack Simulations

In related work, Dunphy *et al.* examined the effect on security of using RBGPs on mobile devices [21] and performed simulations of shoulder surfing attacks. The key image portfolio approach (where the number of passimages exceeds the number of challenge screens) of the VIP system proposed by DeAngeli *et al.* [16] was extended to include a distractor image portfolio where passimages and distractor images are randomly selected from a larger fixed set of images (called key image portfolio and decoy image portfolios respectively).

In addition to the key image portfolio and the decoy image portfolios a ratio of the passimages in a challenge set to the passimages assigned to the user is kept

the same as the ratio of distractors shown in the challenge set to the distractors set from which the images are selected. For example, if the user's passimage set contains four passimages and one passimage is shown per challenge screen then this ratio is 4:4, i.e. 1:1. Thus, if there were 8 distractors and 4 screens, a total of 32 distractors per challenge set, the size of the distractor image portfolio is 32, since 32:32 is the same ratio as 1:1.

The shoulder surfing algorithm reported by Dunphy *et al.* was that an authentication session was observed with probability p of recalling the image. An attack is then performed, and if the attacker has recalled all the passimages shown in the challenge screen then the attack is successful. If not, then the process repeats until success. The average number of observations before successful attacks were reported and results showed an increase in the number of observations before success when the key image and distractor image portfolios were used with increasing number of passimages in the passimage portfolio (passimage set sizes of 6, 8 ,10, 12 and 14 were examined).

However, there were no details in the paper as to whether the results were significantly better than a control setting where no portfolio was used. In addition, there was no examination as to whether the results when increasing the passimage set size were significantly different. Since a portfolio was used for both distractors and passimages, it was not possible to tell from the results reported if the increase in observations before success was due to the passimage portfolio or the distractor image portfolio. The resistance to shoulder surfing is maintained from the key image portfolio approach as presented by the VIP scheme. To establish the significance of the use of a larger passimage set size and gather the raw data required to construct a model, further analysis was required.

The following sections discuss the high level simulation algorithms and the results of the simulations. For further details on the design of the simulation software, please refer to Appendix D.

6.2.1 Shoulder Surfing Algorithm

On a high level, the algorithm used in the simulations for this work was the same as that discussed by Dunphy *et al.* [21]: observe a session and record the images selected, then attempt login and repeat till success. To account for the recall rate, if the recall rate being simulated was 100% then all images viewed were added to a list of viewed images which was then used to attack. If the recall rate was less than 100%, a number of images from the session to be recalled was calculated as shown in Equation 6.1. If this value was greater than one, then it was rounded to the next whole integer and the list of images seen was reduced to this size. If this number was less than one, then multiple sessions would be required before a single image was recalled, and so a different approach was taken. In this case, the number of times an image had to be viewed before recalling it was calculated as shown in Equation 6.2. Each time an image was observed by an attacker, a count of the frequency shown was incremented. If this number reached the value calculated by Equation 6.2 then the image was kept in the attacker's list of recalled images, otherwise it was removed. The attack continued until the recalled image set had all the images presented in a challenge set.

$$numRecall = passpicsSeen * \frac{recallRate}{100} \tag{6.1}$$

$$numSessionsBeforeRecall = \frac{100}{recallRate} \tag{6.2}$$

Further details of the algorithm and implementation are provided in Appendix D. This includes an activity diagram showing the process of the shoulder surfing algorithm. Variables for the simulation were identified as follows:

- Experimental Variables
  - percentage of recall
  - number of passimages in users' set
  - number of challenge screens
- Constant variables
  - number of distractors per challenge screen (eight)
- Dependent Variable
  - number of sessions viewed before a successful attack is made.

Note that the number of distractors per challenge screen was not an experimental variable but a constant variable. This was because the simulation algorithm assumes that in a shoulder surfing attack the attacker views the correct image selection, thus distractor images are not noted. Another possible variable is the potential passimage set size, this was not examined since the algorithm assumes the correct passimage is selected and so the size of the set from which passimages are selected would not impact the success of the attack. The configuration of a RBGP scheme will now be referred to by $p-n-d$, where p is the size of the passimage set, n is the number of challenge screens in a challenge session and d is the number of distractors per screen.

6.2.2 Hypotheses

It was not feasible to model the different countermeasures presented in literature which claim to minimise shoulder surfing by simulation alone. To achieve this, individual user studies would need to be carried out to establish the efficacy of these countermeasures. Instead, to incorporate the variability of success of countermeasures, the recall percentage is included. Thus, if a user study is carried out, the success rate given the countermeasure employed can be used in the simulation. The other possible independent variables are the number of passimages and the number of challenge screens. As highlighted by Dunphy *et al.* [21] and DeAngeli *et al.* [16] increasing the size of the number of passimages in a user's passimage set could have an impact on the success rate of shoulder surfing attacks.

All simulations were run with eight distractors since changing the number of distractors would no effect on the success of the attack since it is assumed the attacker notes only the passimage. In reality, this assumption may be invalid as more images on a screen may make it harder for an attacker to observe which image is being selected. It should also be noted that the impact of increasing the number of challenge screens is examined. This is examined since an increased number of screens will mean an attacker collects more images in one session. For example, consider a passimage set of size 6 with 5 challenge screens. On first observation the attacker collects 5 passimages where if there were only 4 challenge screens only 4 images would be collected. This is further evidenced by considering the probability of observing the same challenge set for each situation. For five challenge screens, the probability is $\frac{1}{\binom{6}{5}} = \frac{1}{6}$ i.e. calculate the number of ways of selecting 5 passimages from 6, then the chance of getting the same screen twice is 1 divided by the number of possible screens. Similarly, with four challenge screens, the probability is $\frac{1}{\binom{6}{4}} = \frac{1}{15}$. Based on these identified potential independent variables, the hypotheses to be tested for the shoulder surfing simulations were as follows:

- **H1** - Increasing the size of the passimage set increases the number of sessions before a shoulder surfing attack is successful.
- **H2** - Increasing the number of challenge screens in a session reduces the number of sessions before a shoulder surfing attack is successful.
- **H3** - Increasing the memorability of the attacker reduces the number of sessions before an attack is successful. This is a check that the implemented algorithm is valid and reflects anticipated behaviour.

6.2.3 Results

The simulation program was run with different configurations shown in Tables 6.3, 6.4 and 6.5. The aim of these simulations was to establish the hypotheses set out in Section 6.2.2. A number of diagrams are used to represent the data in this section, further details of the types of analysis used are provided in Appendix B.

The first step was to establish the distribution of the simulation data which allows correct selection of statistical tests. Recall that data is a binomial distribution if it meets the following criteria (as detailed in [71, page 135])

1. There are a fixed number of trials
2. Each trial has two possible outcomes, success or failure
3. The probability of success (p) is the same for each trial
4. The trials are independent of each other

The distribution of the shoulder surfing attack simulations was not binomial because the second criterion was not met. A number of screens before success were recorded, not success or failure of an attack. The distribution was also not

Figure 6.1: Shoulder Surfing Simulations 6-4-8 100% Recall Histogram

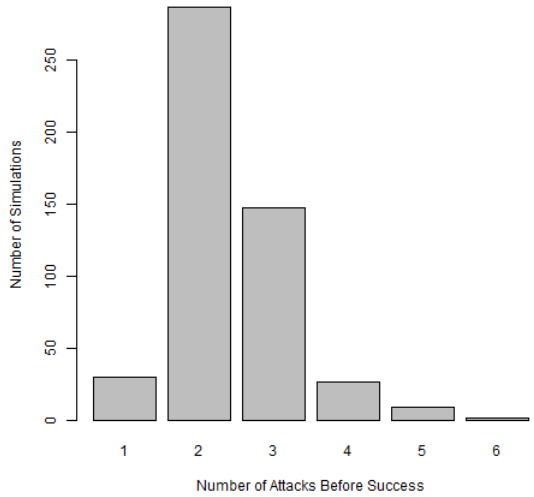
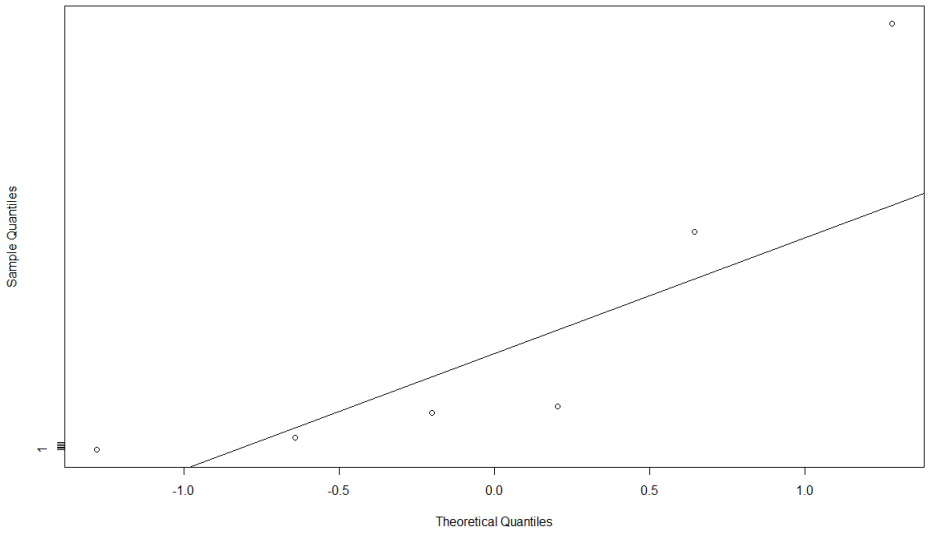


Figure 6.2: Shoulder Surfing Simulations 6-4-8 100% Recall Normality Plot



a normal distribution, this can be seen by the skew in the data presented in Figure 6.1 where it can be seen that the data is skewed to the right. In a normal probability plot, if the points lie very close to the straight line which represents the normal distribution, the data is normally distributed ([18, Page 115]). It can be seen from Figure 6.2, that this is not the case for the shoulder surfing data. This was assessed for other configurations with similar results.

For non-normal data the statistical measurements of mean and standard deviation are are inappropriate [18, Page 80]. The median is a more appropriate measure of spread of the data and the interquartile range is an appropriate measure which reflects variability of the data. More generally, standard parametric statistics are inappropriate for non-normal data [27]. Instead, robust methods are a more appropriate option as they are designed to perform well whether the data is normally distributed or not [27].

Configuration (Passimage Set Size)	Test Statistic	Critical Value	Null rejected?
5 v 6	6.04	1.96	Yes
5 v 7	10.23	1.97	Yes
5 v 8	16.44	1.97	Yes
5 v 9	30.35	1.96	Yes
5 v 10	25.13	1.97	Yes

Table 6.3: Shoulder Surfing Simulations H1 Summary Stats

For the purpose of hypothesis testing, the Yuen statistic with 20% trimmed mean and an alpha value of 0.05 was used. Details of this statistic are provided in Appendix B. The statistic was highlighted by Wilcox as yielding robust results (i.e. provide a more accurate statistic for non-normal data) which were better for smaller sample sizes than other approaches [104, page 157]. Wilcox [104] has written robust statistical functions for the statistical program "R"[1]. For this analysis, the R program was used. If the test statistic value was higher than the critical value (which is automatically calculated by the R program given the data input) then the null hypothesis was rejected (this is shown in [103, Page 252]). Each hypothesis is now considered in turn, applying the Yuen statistic to establish acceptance or rejection of each hypothesis.

H1- Increased Passimage Set Size

The null hypothesis related to H1 was rejected in all configurations examined as detailed in Table 6.3. This demonstrates that increasing the number of passimages in a passimage set significantly increases the number of attacks before success. This is further evidenced in Figure 6.3 where the median value of number of attacks required increases between each value for the passimage set size. In each configuration four challenge screens were used with eight distractors, and the passimage set size was varied to establish the effect on the number of attacks before success.

H2 - Increased Number of Challenge Screens

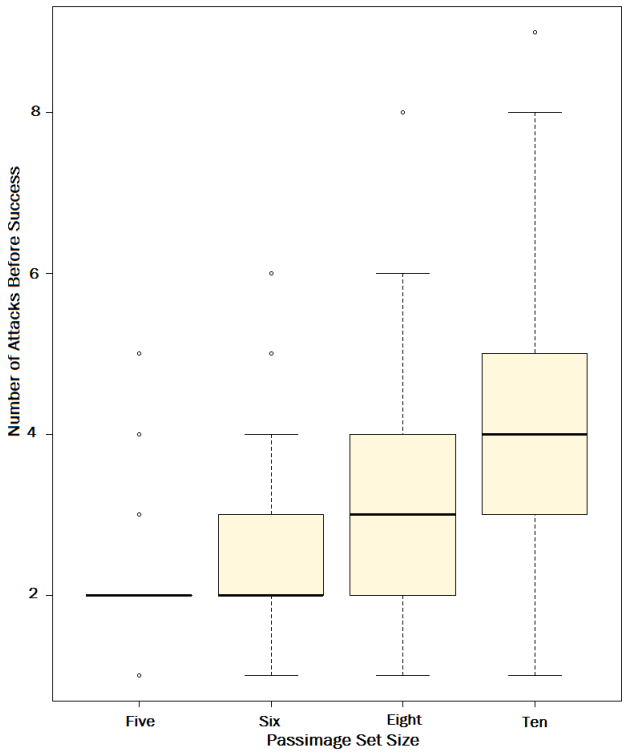
Where the number of passimages exceeded the number of challenge screens, the null hypothesis was rejected in all configurations examined. This is detailed in Table 6.4. This is further evidenced in Figure 6.4 where it can be seen that the median value of number of attacks required decreases between each value for the number of challenges. In each case, ten passimages were used with eight distractors and the experimental variable of number of challenge screens was varied from 4 to 9.

H3 - Increased Memorability (Algorithm Verification)

H3 was primarily verification of the implementation of the shoulder surfing algorithm (which was also tested using JUnit). If this null hypothesis was not

[1]Available at: `http://www.r-project.org/`

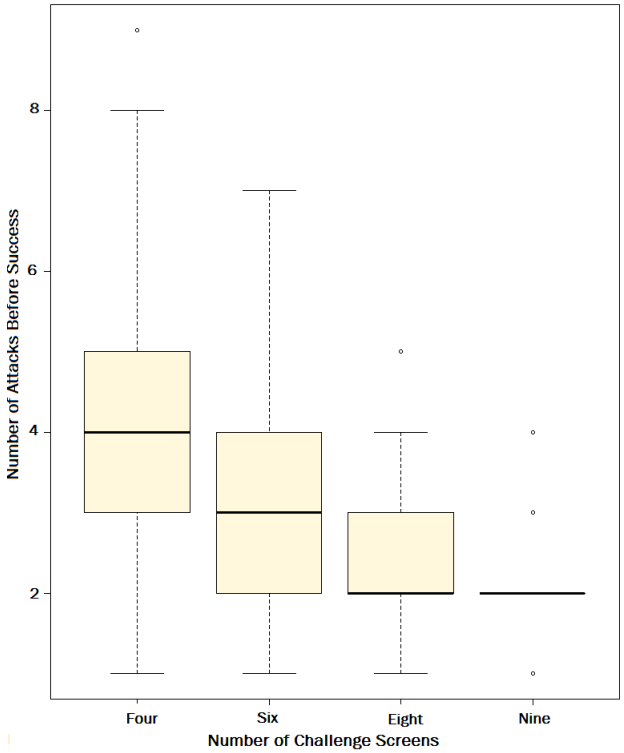
Figure 6.3: Shoulder Surfing H1 Boxplot



Configuration (No. of Challenge Screens)	Test Statistic	Critical Value	Null rejected?
4 v 5	5.09	1.96	Yes
4 v 6	9.41	1.96	Yes
4 v 7	16.41	1.96	Yes
4 v 8	21.47	1.97	Yes
4 v 9	27.72	1.97	Yes

Table 6.4: Shoulder Surfing Simulations H2 Summary Stats

Figure 6.4: Shoulder Surfing H2 Boxplot

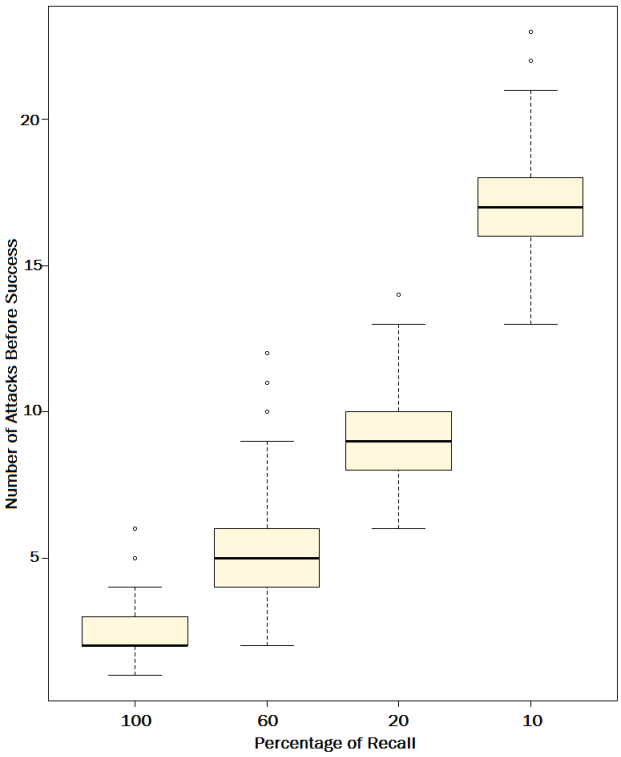


Configuration (Percentage of Recall)	Test Statistic	Critical Value	Null rejected?
100 v 10	166.23	1.97	Yes
100 v 20	99.33	1.96	Yes
100 v 30	50.99	1.97	Yes
100 v 60	7.69	1.97	Yes
100 v 90	0.85	1.96	No

Table 6.5: Shoulder Surfing Simulations H3 Summary Stats

rejected then there was a problem with the logic of the simulation. The null hypothesis was rejected in all configurations examined as detailed in Table 6.5 with the exception of 100% vs. 90%. This could be because there is an insignificant difference between one session required to attack with 100% recall compared to two sessions with 90% recall. Thus the hypothesis overall was accepted. This is further evidenced in Figure 6.5 where it can be seen that the median value of number of attacks required decreases as the value for the percentage of recall increases (right to left). In each case, four passimages and four challenge screens were used with eight distractors. The percentage of recall was varied in increments of 10% from 10% through to 100%. Figure 6.5 shows the data for recall rates of 10%, 20%, 60%, and 100% for comparison.

Figure 6.5: Shoulder Surfing H3 Boxplot



6.2.4 Shoulder Surfing Simulations Discussion

Shoulder surfing attacks were simulated, varying the independent variables (number of passimages, number of challenge screens, percentage of recall) to examine the impact on the dependent variable (number of attacks before success). Results showed increasing the number of passimages in a user's passimage set significantly increases the number of attacks before success, and increasing the number of challenge screens significantly decreases the number of attacks before success.

One possible limitation of this work is that it requires an estimate of the percentage of recall of the attacker. However this was designed as such to incorporate the varying impact of different counter measures. The impact of different countermeasures cannot easily be incorporated into a singular mathematical model due to the potential variation. Inclusion of the percentage of recall allows researchers to construct their own experiments to determine a success rate of the attack when their countermeasure is employed. This can be used in conjunction with different configurations for the number of passimages and challenge screens to estimate the overall attack success rate using the mathematical model presented in Chapter 7.

6.3 Frequency Attack Simulations

Unlike shoulder surfing, it was easier to incorporate countermeasures into the simulations of frequency attacks as modeling the countermeasures did not involve a human element. The approach involved creating a simulation of frequency attacks using each of five possible counter measures. The number of attacks

which had to be performed before a successful attack for each approach was noted and compared to a control simulation.

For the purposes of this work, a frequency attack is defined as follows. The attacker starts the challenge session which comprises of n challenge screens, each of which has one passimage from the user's passimage set (of size p) and a number of distractors (d). The attacker attacks each screen in turn by noting all the images on the screen (which is referred to as the challenge set). A count of the number of times each image has been viewed over all the attacks the attacker has launched against this set of passimages is then incremented. The attacker then attempts to pass the challenge set by selecting the image which has been viewed most frequently. Further details of the design and implementation of the frequency attack simulations is provided in Appendix D.

This is repeated for each challenge screen within the session. If each image selected corresponds to the passimages, then the attacker is successful and the process is complete. If the attacker selects a distractor for any one of the challenge screens, they must start a new attack. This repeats until the attacker is successful. Once the attacker has achieved successful authentication, the number of attacks before success is reported. The configuration of a recognition-based graphical password scheme is referred to by $p - n - d$, where p is the size of the passimage set, n is the number of challenge screens in a challenge session and d is the number of distractors per screen.

The dependent variable was the number of attacks before success. The independent variables for the simulations were identified as follows:

- The number of passimages
- The number of distractors per screen
- The number of challenge screens

As with shoulder surfing, the size of the potential passimage set was not included in the independent variables examined. This could have an impact if this set of images is used for the distractors. This is a potential limitation of this work, which could benefit from further examination. The distractor selection algorithm was also not examined. This was primarily due to the lack of evidence for significant impact of distractors selection on the success rate of SOGAs. This could potentially have been overlooked as a dependent variable since if a distractor selection algorithm has a preference for some distractors over others (hence increasing their count), there is potential for the number of attacks required to be increased. Further work to examine this could allow an improved model to be constructed.

The following sections discuss the high level simulation algorithms and the results of the simulations. For further details on the design of the simulation software, please refer to Appendix D.

6.3.1 The Countermeasures

Dhamija and Perrig [20] successfully summarise counter measures for intersection attacks (and hence frequency attacks) as follows:

- Use the same distractor images and passimages for each session.
- Repeat a small subset of distractor images for each passimage. This theoretically results in an attacker recording the same frequencies for these distractors and the passimage and the attacker would be unable to tell which is the passimage. Thus, they would have to randomly select one of the images with the same frequency of occurrence.
- In any given challenge session, if a user selects a distractor on a challenge screen, subsequent screens only display distractor images - "dummy screens".
- Implement a limit on the number of incorrect authentications a user can perform, this stops an impersonator attempting to discover all of the images. (A "three strikes and you're out" approach)

The first approach where the distractors are set for a given passimage will reduce an intersection attack (and also frequency attacks) to a random guessing attack. However, as Dhamija and Perrig note, re-use of distractor images may result in users recognising the distractors and selecting a distractor image instead of their passimage during authentication [20]. Dhamija and Perrig indicate that this requires further work to establish, however an experiment by Deffenbacher *et al.* [19] provides further evidence to support this opinion. Deffenbacher *et al.* conducted an experiment which assessed the amount of retroactive interference (difficulty of remembering old information due to acquisition of new information) for landscape images, line drawings of objects, faces and nouns. A recognition memory test was performed in which participants viewed target stimuli (the aforementioned image types and nouns) followed by distractor stimuli. After two minutes of performing an unrelated task, the participants were asked to recognise and distinguish targets from distractors. Two weeks subsequently, the same recognition task was performed.

The results showed no significant retroactive interference for line drawings of objects, but significant interference for faces and landscapes which resulted in distractor images being selected instead of targets. For the two week test, significant retroactive interference was established for all image types i.e. participants had difficulty distinguishing the targets from the distractors. This provides evidence for the claim that constant distractors could interfere with correct selection of passimages. However, further work would still be required to establish this concretely in the context of graphical authentication.

Due to the potential interference and the additional work required to implement constant challenge screens, it is feasible that other countermeasures may be used and so the efficacy of those countermeasures will be examined. In addition, as noted by Smith [80, Page 163], there are "different secrets for different uses". That is to say, there are different levels of security required for different environments. Thus it is feasible to consider that one might not wish to go to the effort of eradicating frequency attacks, but merely reduce the risk to an acceptable level. This research aimed to establish how effective the different mitigation countermeasures are to allow this to be incorporated into the mathematical model

of frequency attacks reported in the following chapter. Specifically the following countermeasures were examined:

- Repetition of a subset of distractors for a given passimage
- Use of "dummy" screens if an attacker selects the wrong image at any point within a session
- Using a passimage set which is larger than the number of challenge screens in a session (proposed by DeAngeli *et al.* [17] and extended by Dunphy *et al.* [21])

Additionally, there has been no claim that increasing the number of distractors shown per challenge screen or increasing the number of challenge screens would mitigate a frequency attack. Thus these were also examined for significance. The algorithm is reported in the next section, and hypotheses to be examined are reported in the following section.

6.3.2 Frequency Attack Algorithm

The collection of 144 images established earlier in the research was used in the simulation. The content of the images was unimportant as they were selected randomly for both passimage sets and distractors. The control configuration which included no counter measures against frequency attacks was as follows. The first step was to select a specified number of passimages from the potential passimage set. A subset of the user's passimage set is then selected. To create a challenge session, the number of challenge screens to be generated matched the number of images in the selected passimage subset. A specified number of distractors were then randomly selected from the remaining images (the complete collection, less the passimages for the current set of passimages) for each of the challenge screens required.

An attack on the set of passimages was then conducted as follows. A list of images seen by the attacker is created. For each challenge screen presented to the attacker, the images are either added to the list of viewed images, or the number of times they have been seen is incremented. To attack the screen, the attacker identifies the most viewed image on the screen (the image on the screen with the highest count in the list of viewed images) and selects that image as the passimage. If there are a number of images with equal frequencies, a random image is selected from this set. If the image is the passimage, a counter for the number of screens passed in that session is incremented. If at the end of the session the number of screens passed is equal to the number of challenge screens in a session the set of passimages was successfully attacked. The program then exits returning the number of attacks which were attempted before success (the dependent variable being examined in this research). For each experimental configuration ($p - s - d$ and any applicable countermeasure variables), this process was run one hundred times because Wilcox notes that the probability of a Type I error (where the null hypothesis is true, but is falsely rejected) is suitably minimised with a sample size of 100 observations [104, Page 154].

6.3.3 Hypotheses

The dependent variable being examined in this research was the number of attacks before success. The independent variables included the number of passimages in the user's passimage set (p), the number of challenge screens per session (s), the number of distractors per challenge screen (d), the number of distractors kept constant per passimage and the use of dummy screens. The hypotheses to test the relationships between the independent variables and dependent variable were established as follows:

- H1 It takes significantly more attacks before a successful frequency attack when there are a subset of distractors kept constant between challenge sessions.
- H2 It takes significantly more attacks before a successful frequency attack when the number of distractors kept constant is increased.
- H3 It takes significantly more attacks before a successful frequency attack when dummy screens are presented if one screen in a challenge set is failed.
- H4 It takes significantly more attacks before a successful frequency attack when a passimage set larger than the number of challenge screens in a session is used.
- H5 It takes significantly more attacks before a successful frequency attack when the number of challenge screens in a session is increased.
- H6 It takes significantly more attacks before a successful frequency attack when the number of distractors per challenge screen is increased.

For hypothesis testing, where an independent variable was altered, the remaining independent variables are kept constant so that the effect of only one independent variable was measured at any given time. The independent variable being examined is referred to as the experimental variable. The corresponding null hypotheses (referred to by the hypothesis number, with a subscript of 0 after e.g. the null hypothesis for H1 is $H1_0$) detail that there is no significant difference in the number of attacks before success. Control configurations had no countermeasures implemented and were used to compare to the other configurations where each countermeasure was implemented, or in the case of H2 a lesser number of constant distractors was used. A number of different variations were used to test each hypothesis, detailed as follows:

H1 Experimental Configurations

The simulation configurations run to examine H1 were as detailed in Table 6.6. Each configuration was run 100 times. The number of distractors used were selected as eight, nine and fifteen. This was to reflect common choices in recognition-based schemes. Eight distractors are used in passfaces [2], nine distractors are used in VIP [16] and fifteen distractors are used in the doodles scheme [62]. The last size provided a comparison of a larger distractor set.

[2] Available at : `http://www.realuser.com/`

p	s	d	c
4	4	8	0
4	4	8	1
4	4	8	2
4	4	8	3
4	4	9	0
4	4	9	1
4	4	9	2
4	4	9	3
4	4	15	0
4	4	15	1
4	4	15	2
4	4	15	3

Table 6.6: Frequency Attack H1 Configurations

p	s	d	c
4	4	8	1
4	4	8	2
4	4	8	3
4	4	9	1
4	4	9	2
4	4	9	3
4	4	15	1
4	4	15	2
4	4	15	3

Table 6.7: Frequency Attack H2 Configurations

For this hypothesis, the number of images in the passimage set was kept consistent (at four) as was the number of challenge screens. The number of distractors changed, but hypothesis testing always compared a control configuration with a corresponding configuration with only the experimental variable changed (in this case the number of distractors kept constant per passimage) . To test H1, configurations with the same values of p, s, and d were compared with different values for c. For example, the configuration 4-4-8 with 1 constant distractor was compared to the 4-4-8 configuration with no constant distractors.

H2 Experimental Configurations

The simulation configurations run to examine H2 were as detailed in Table 6.7. Each configuration was run 100 times. To test H2, configurations with the same values of p, s, and d were compared with increased values for c. For example, the configuration 4-4-8 with 3 constant distractors was compared to the 4-4-8 configuration with one constant distractor.

p	s	d	dummy screens
4	4	8	yes
4	4	8	no
4	4	9	yes
4	4	9	no
4	4	15	yes
4	4	15	no

Table 6.8: Frequency Attack H3 Configurations

p	s	d
4	4	8
4	4	9
6	4	8
6	4	9
8	4	8
8	4	9
12	4	8
12	4	9

Table 6.9: Frequency Attack H4 Configurations

H3 Experimental Configurations

The simulation configurations run to examine H3 were as detailed in Table 6.8. Each configuration was run 100 times. To test H3, configurations with the same values of p, s, and d using dummy screens were compared to those without dummy screens. For example, the 4-4-8 configuration which employed dummy screens was compared to the 4-4-8 configuration which didn't use dummy screens.

H4 Experimental Configurations

The simulation configurations run to examine H4 were as detailed in Table 6.9. Each configuration was run 100 times. To test H4, configurations with the same values of s and d were compared to configurations with larger values for p. For example, the configuration 6-4-8 was compared to the configuration 4-4-8. Values of p were selected to reflect 1.5 times the number of passimages (e.g. p=6 c.f. p=4), double the number of passimages (e.g. p=8 c.f. p=4) and three times the number of passimages (e.g. p=12 c.f. p=4).

H5 Experimental Configurations

The simulation configurations run to examine H5 were as detailed in Table 6.10. Each configuration was run 100 times. To test H5, configurations with the same values of p and d were compared to configurations with an increased number of challenge screens. For example the configuration 10-4-8 was compared with the configuration 10-5-8. The number of passimages was kept constant at ten. This value had to be large enough that the number of screens was always less than the

p	s	d
10	4	8
10	5	8
10	6	8
10	7	8

Table 6.10: Frequency Attack H5 Configurations

p	s	d
4	4	8
4	4	9
4	4	15

Table 6.11: Frequency Attack H6 Configurations

number of passimages in the set, but the number of screens could be increased.

H6 Experimental Configurations

The simulation configurations run to examine H6 were as detailed in Table 6.11. Each configuration was run 100 times. To test H6, configurations with the same values of p and s were compared to configurations with increased values for d. For example the configuration 4-4-8 was compared with the configuration 4-4-9.

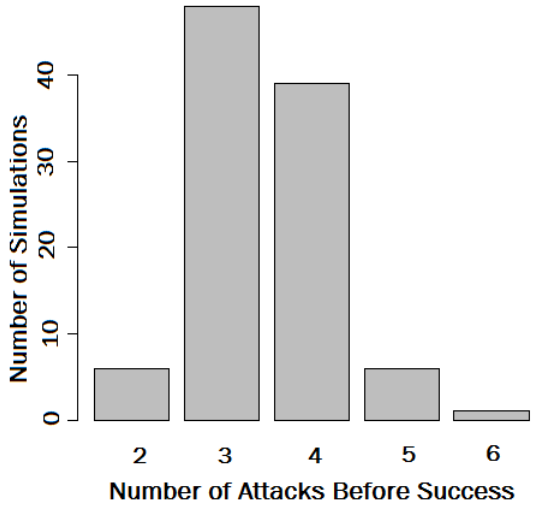
6.3.4 Results

The histogram showing the distribution of one hundred simulations for a control setting with four passimages, four challenge screens and eight distractors is shown in Figure 6.6. It can be seen from this figure that the distribution is skewed to the right, indicating that the use of standard deviation and other parametric statistics such as mean may not be appropriate [18, Page 80]. The frequency distributions for the countermeasures configurations also indicated asymmetric distributions. Examples for each of these are provided in Figures 6.7, 6.8 and 6.9 respectively. Some of these appear more skewed than others (e.g. Fig. 6.7). The asymmetric distribution was also confirmed using a normal probability plot (for the control configuration of 4-4-8), as shown in Figure 6.10. In a normal probability plot, if the points lie very close to the straight line which represents the normal distribution, the data is normally distributed ([18, Page 115]).

Due to the non-normal distribution of the results, it was decided that a statistical approach which was robust to outliers in data and skewed distributions should be taken. The Yuen statistic with 20% trimmed means and an alpha value of 0.05 was applied, as it is highlighted by Wilcox in [104, Page 157] that this approach yields robust results. As for the shoulder surfing analysis, analysis for the frequency attacks was performed using the statistical program "R" [3]. In the hypothesis testing, if the Yuen test statistic value was higher than the critical value (which is automatically calculated by the R program for the data input)

[3]Available at `http://www.r-project.org/`

Figure 6.6: Frequency Attacks Control 4-4-8 Histogram



Constant Distractors	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
0	2.00	3.00	3.00	3.48	4.00	6.00
1	2.00	7.00	13.00	17.76	23.00	85.00
2	2.00	27.00	57.50	83.45	110.00	670.00
3	2.00	72.75	178.50	264.90	348.20	1604.00

Table 6.12: Constant Distractors Summary Stats Table

then the null hypothesis was rejected (this is shown by Wilcox in [103, Page 252]). Each hypothesis is now considered in turn and the results reported.

H1 - Constant Distractor Subset: Results

A summary of the results for the use of constant distractor subsets is demonstrated by the boxplot in Figure 6.11. This figure demonstrates the effect of the number of distractors kept constant per passimage on the number of attacks required before success for values 0 (control), 1,2 and 3. The plot shows that the use of a number of constant distractors reduces the number of attacks required before success when compared to zero constant distractors.

In each case represented in the plot, four challenge screens with four passimages and eight distractors were used, the experimental variable (the number of constant distractors per passimage) was varied using values of zero, one, two and three. The values for minimum, first quartile, median, mean, third quartile and maximum for each configuration is given in Table 6.12. This tables shows that the minimum stays approximately equal in each case, but the values for median and the quartiles increase between each value for constant distractors. It should be noted that due to the skew of the distribution, mean is not an accurate measure of spread but is included for completeness.

Applying the Yuen statistic with 20% trimmed means, the null hypothesis $H1_0$ was rejected for each of the configurations used to test H1 with the test statistic

Figure 6.7: Frequency Attacks 4-4-8 Subset of One Constant Distractor Histogram

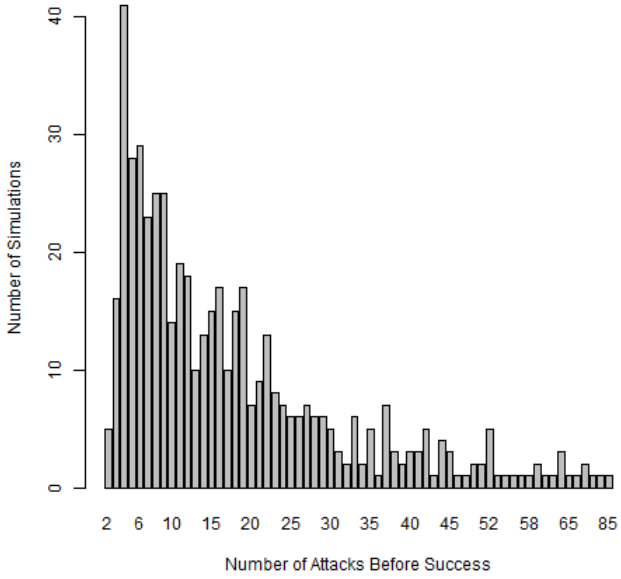


Figure 6.8: Frequency Attack 4-4-8 Dummy Screens Histogram

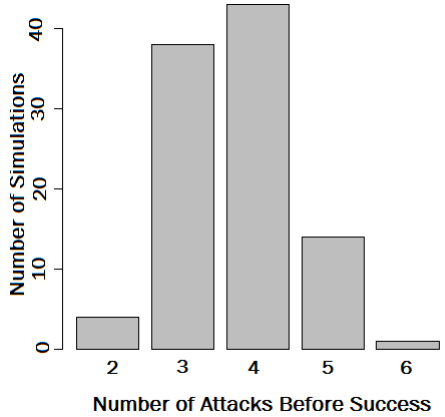


Figure 6.9: Frequency Attack 8-4-8 Larger Image Set Histogram

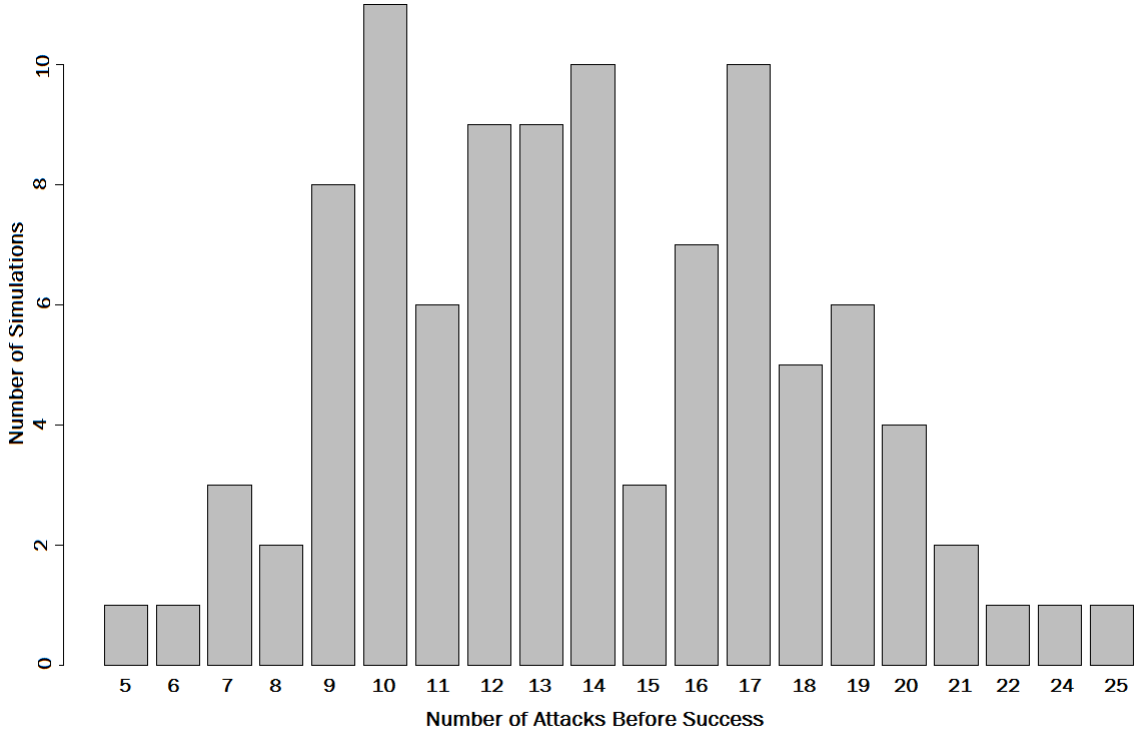


Figure 6.10: Frequency Attacks 4-4-8 Normal Probability Plot

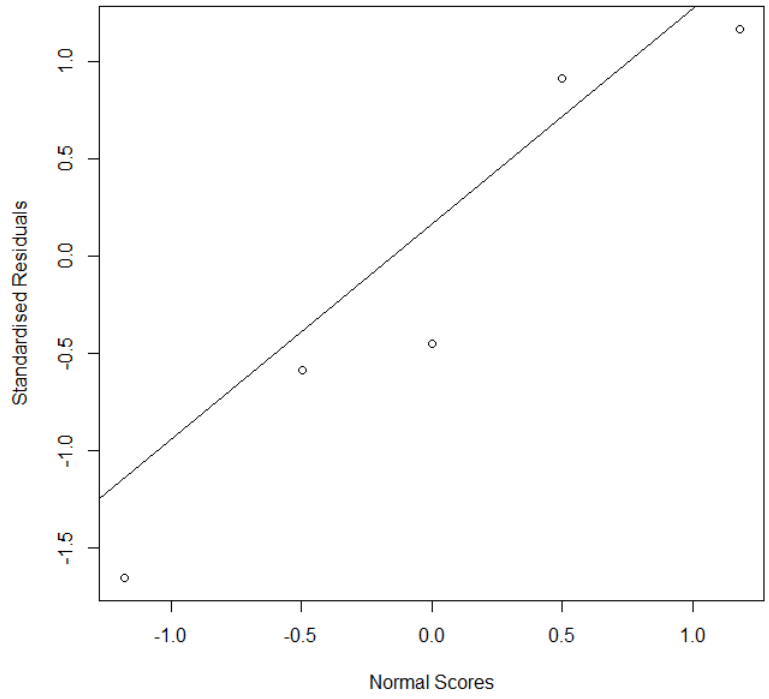
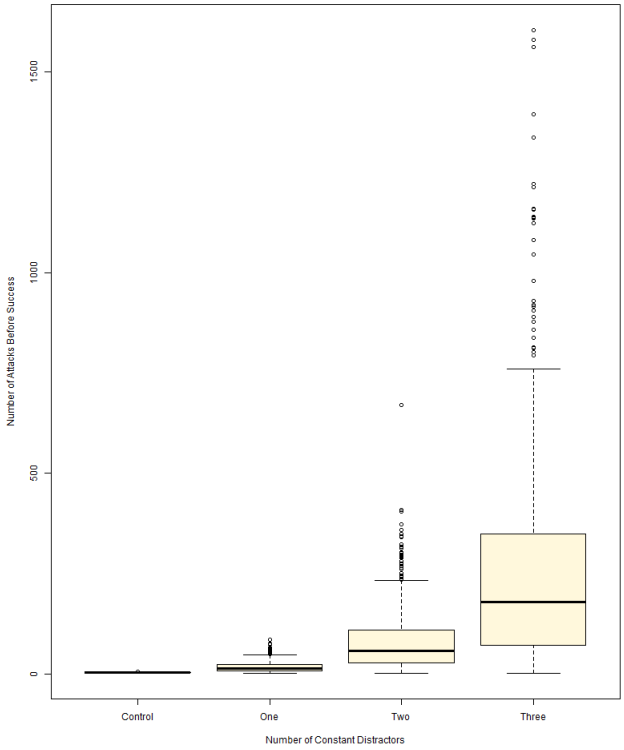


Figure 6.11: Frequency Attacks H1 Boxplot- Constant Distractors



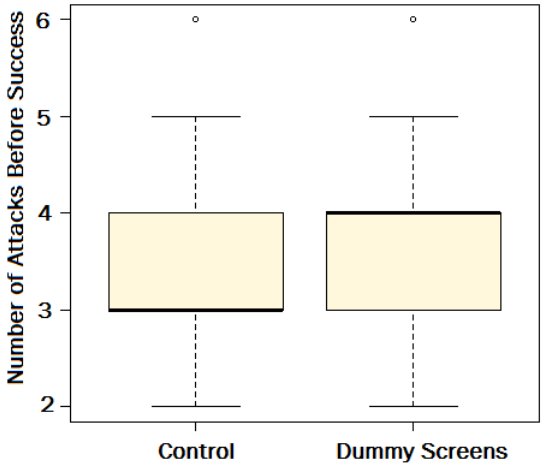
value ranging between 16.84 and 20.38 and the critical value of approximately 1.97 in each instance. Thus the number of attacks before success when a subset of distractors are kept constant is significantly more than that when no distractors are kept constant.

H2 - Increasing the Number of Distractors Kept Constant: Results

A summary of the results for the use of constant distractor subsets is demonstrated in Figure 6.11 when examining the second, third, and fourth boxes corresponding to one, two and three distractors kept constant respectively. It can be seen from this plot that increasing the number of constant distractors increases the number of attacks required before success. This is also shown in the statistics in Table 6.12 where the values for median increase substantially between each value. In each case here, four challenge screens with four passimages and eight distractors were used, the variable of interest (the number of constant distractors) was increased from one to two and then to three.

Applying the Yuen statistic with 20% trimmed means, the null hypothesis $H2_0$ was rejected for each of the setups used to test H2. The test statistic value ranged between 12.70 and 19.57 and the critical value was approximately 1.97 in each instance. This result is as expected from examination of the evidence shown in the boxplot in Figure 6.11. This means that increasing the number of distractors kept constant per passimage significantly increases the number of attacks before success.

Figure 6.12: Frequency Attacks H3 Boxplot - Use of Dummy Screens



Dummy Screens	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
No	2.00	3.00	3.00	3.48	4.00	6.00
Yes	2.00	3.00	4.00	3.70	4.00	6.00

Table 6.13: Dummy Screens Summary Stats Table

H3 - Dummy Screen Results

A summary of the results for the use of dummy screens (upon incorrect selection) is demonstrated in Figure 6.12. It can be seen from this plot that the use of dummy screens appears to have little effect on the overall number of attacks required before success. In each configuration four challenge screens with four passimages and eight distractors were used. The experimental variable (the use of dummy screens) was varied by either being used or not. When dummy screens were not used, this was the control configuration.

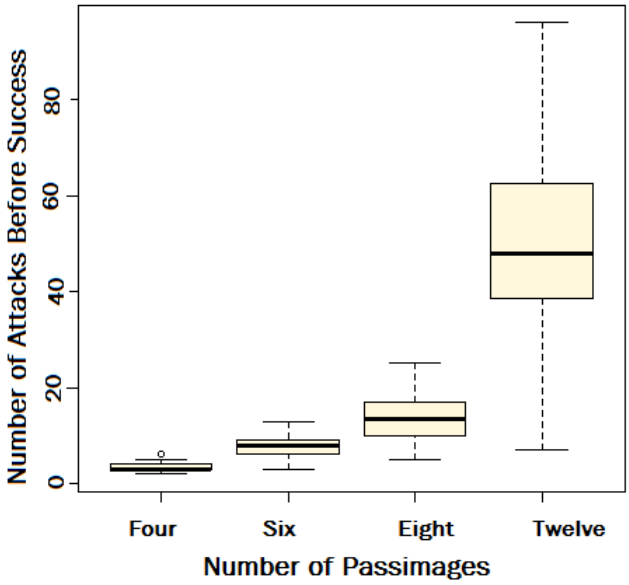
The values for minimum, first quartile, median, mean, third quartile and maximum for each set up is given in Table 6.13. One can see from Table 6.13 that the only value which changes between the control configurations (using no dummy screens) and the set up using dummy screens is the median, which changes by 0.22. Thus the use of dummy screens appears to show little evidence of an increase in the number of attacks required before success.

Applying the Yuen statistic with 20% trimmed means, confirmed that the null hypothesis $H3_0$ could not be rejected for each of the set ups used to test H3. The test statistic value ranged between 0.79 and 1.70 and the critical value was approximately 1.98 in each instance.

H4 - Larger Image Set Results

A summary of the results for the use of larger passimage sets (from which a subset is selected for any given authentication screen) is demonstrated in Figure 6.13. It can be seen from this plot that the use of a larger passimage set reduces the number of attacks required before success when compared to smaller sets. In each configuration, four challenge screens and eight distractors were used, the

Figure 6.13: Frequency Attacks H4 Boxplot - Use of a Larger Image Set



No. of Pas-sim-ages	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
Four	2.00	3.00	3.00	3.48	4.00	6.00
Six	3.00	6.00	8.00	7.79	9.00	13.00
Eight	5.00	10.00	13.50	13.85	17.00	25.00
Twelve	7.00	38.75	48.00	49.62	62.25	96.00

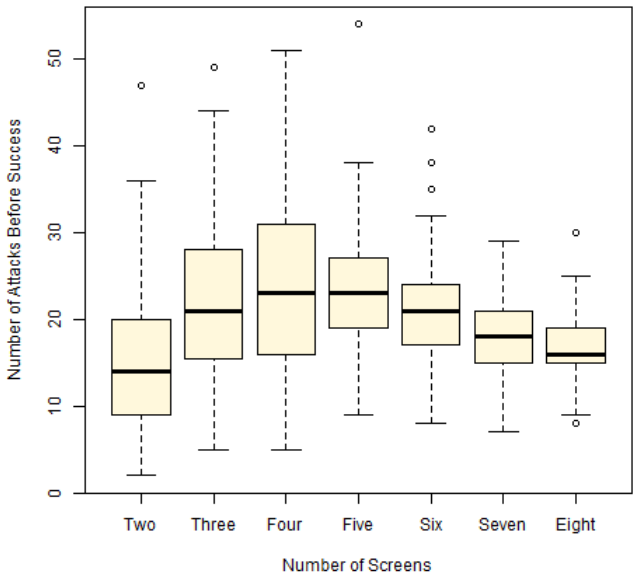
Table 6.14: Larger Passimage Set Summary Stats Table

experimental variable (the number of passimages in a set) was varied using values of four, six, eight and twelve.

There is an increase in the number of attacks when moving between lower values of passimage set sizes and larger values. This indicates that increasing the number of passimages has a significant effect on the number of attacks required. The values for minimum, first quartile, median, mean, third quartile and maximum for each set up is given in Table 6.14. As expected from the boxplot, it can be seen from Table 6.14 that there is a large jump in all the statistics between each of the number of passimages.

When applying the Yuen statistic with 20% trimmed means, the null hypothesis $H4_0$ was rejected for each of the configurations used to test H4. The test statistic value ranged between 15.89 and 24.17 and the critical value was approximately 2.00 in each instance. This is in line with the results shown in Table 6.14 and Figure 6.13 and means that there is a significant increase in the number of attacks required before success when the number of passimages in a user's passimage set is increased.

Figure 6.14: Frequency Attacks H5 Boxplot - Increased Number of Challenge Screens



H5 - Increasing the Number of Challenge Screens Results

A summary of the results for the use of an increased number of challenge screens is demonstrated in Figure 6.14. It can be seen from this plot that the use of a larger number of challenge screens reduces the number of attacks required before success when compared to a smaller number of screens. This was the case until the number of screens approaches the number of passimages. In each configuration, the number of passimages in the set was kept constant at 10. This is because the number of passimages in the set has to be larger than the number of challenge screens in each instance. Eight distractors were used and the variable of interest (the number of challenge screens) was varied using values of five, six and eight.

The values for minimum, first quartile, median, mean, third quartile and maximum for each configuration is reported in Table 6.15. These statistics demonstrate the increase in number of attacks before success until the number of challenge screens approaches the number of passimages. In the boxplot (Fig. 6.14) the median is consistent when using four and five screens. However as the number of screens approaches the number of passimages in the set this reduces. One reason for this could be that if the number of passimages is approximately equal to the number of screens then the attacker will see the passimages more frequently, making the attack more successful. Thus it is possible that it is not merely increasing the number of screens producing this effect, but that the value is close to the number of passimages.

Applying the Yuen statistic with 20% trimmed means, the null hypothesis $H5_0$ was rejected for four of the five configurations used to test H5. The test statistic value ranged between 0.29 and 5.87 and the critical value was approximately 1.99 in each instance. Where the test statistic was not significant was for the use of five challenge screens, the remaining results established a significant difference in the

No. of Challenge Screens	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
Two	2.00	9.00	14.00	15.73	20.00	47.00
Three	5.00	15.75	21.00	21.98	28.00	49.00
Four	5.00	16.00	23.00	24.09	31.00	51.00
Five	9.00	19.00	23.00	23.86	27.00	54.00
Six	8.00	17.00	21.00	20.84	24.00	42.00
Seven	7.00	15.00	18.00	18.27	21.00	29.00
Eight	8.00	15.00	16.00	16.63	19.00	30.00

Table 6.15: More Challenge Screens Summary Stats Table

No. of Distractors	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
Eight	2.00	3.00	3.00	3.48	4.00	6.00
Nine	2.00	3.00	4.00	3.91	4.00	7.00
Fifteen	3.00	5.00	6.00	5.64	6.00	8.00

Table 6.16: Increased Distractors Summary Stats Table

distributions. This is in line with the results shown in Table 6.15 and Figure 6.14. However, the statistic is a two-sided test meaning the result could significantly less or significantly more attacks before success. Thus using the boxplot it can be seen that increasing the challenge screens has a detrimental effect on the number of attacks required instead of a positive effect (i.e. it decreases the number of attacks instead of increasing them).

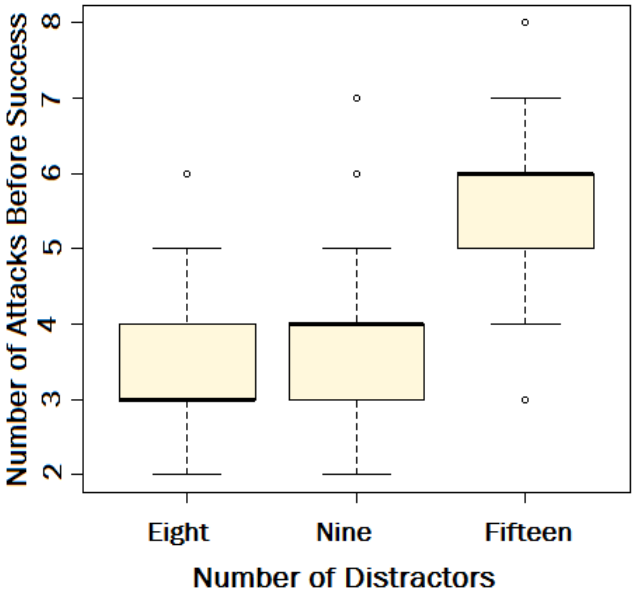
H6 - Increasing the Number of Distractors Results

A summary of the results for the use of an increased number of distractors in each challenge set is shown in Figure 6.15. It can be seen from this plot that the use of more distractors in each challenge screen increases the number of attacks required before success when compared to smaller numbers of distractors. In each configuration, four challenge screens and four passimages were used, the experimental variable (the number of distractors per screen) was varied using values of eight, nine and fifteen.

The values for minimum, first quartile, median, mean, third quartile and maximum for each configuration is given in Table 6.16. As expected from the boxplot in Figure 6.15, it can be seen from Table 6.16 that there is an increase in the median values between each of the variations. There is a larger increase of median attacks before success between the use of fifteen distractors and eight and nine. This is as expected as there is a larger difference in the number of distractors being compared.

When applying the Yuen statistic with 20% trimmed means, the null hypothesis $H6_0$ was rejected for each of the configurations used to test H6. The test statistic value ranged between 3.84 and 13.67 and the critical value was approximately 1.98 in each instance. This means that increasing the number of distractors per screen significantly increases the number of attacks required before

Figure 6.15: Frequency Attacks H6 Boxplot - Increased Number of Distractors



success.

6.3.5 Frequency Attacks Simulations Discussion

Frequency attacks were simulated varying a range of independent variables (the number of challenge screens, number of distractors, number of passimages, use of dummy screens, and the number of constant distractors). The effect on the number of attacks before success was examined. Whilst some of the previously identified countermeasures had a significant impact on the number of attacks required before success, others did not. In particular, the use of dummy screens (where only distractor images are shown if the user selects the incorrect image on any given challenge screen within a session) did not show significant results. Also, increasing the number of challenge screens increases the number of attacks required before success until the number of challenge screens approaches the size of the passimage set. Use of constant distractors increased the number of attacks required, as did use of a larger passimage set and increasing the number of distractors per challenge screen. Increasing the number of constant distractors used also significantly increased the number of attacks required.

Of the methods which achieved significant increases in the number of attacks before success, increasing the number of distractors per screen and the passimage set size and using a subset of constant distractors provided effective results. The most effective countermeasure was established as using a number of constant distractors per passimage. When comparing the median number of attacks before success to the control configuration (with four passimages, four challenge screens and eight distractors) using one constant distractor per passimage resulted in an approximate 5.17 times increase. Using two constant distractors resulted in an approximate 20.17 times increase and using three constant distractors resulted in an approximate 57.17 times increase compared to no constant distractors.

The second most successful countermeasure was using a larger passimage set. When compared to a control of four passimages, using six passimages gave approximately 2.67 times increase. Eight passimages gave approximately 4.5 times increase and twelve passimages gave an approximate increase of 16 times. The least effective of the significant countermeasures was using more distractors per screen, which when nine distractors was compared to eight resulted in approximately 1.3 times increase and fifteen distractors resulted in a 2 times increase compared to eight distractors.

6.4 Observability Data Collection Conclusion

In this chapter gathering data on observability attacks was approached in two ways; a user study and simulations. The user study approach provided insufficient data due to lack of participation. The final approach taken was that of simulations. This approach provided a better flexibility as further variables could be easily incorporated and data was gathered with relative ease. The simulation approach also provided an improvement on the user study approach as it allowed the attacker's ability to be set as constant, which meant that the results were not dependent on the ability of the attacker. Shoulder surfing attacks were shown to take a significantly larger number of attacks when the number of passimages in a user's passimage set exceeded the number of challenge screens. Also, increasing the number of challenge screens decreased the number of attacks required before success.

The efficacy of countermeasures for frequency attacks and the impact of independent variables were also established by the simulations. Now observability data has been established, and variables which impact the number of attacks required have been identified, it is now necessary to establish a model for these attacks. This is achieved by mathematical modelling and is reported in Chapter 7. The final metric and the corresponding evaluation are reported in Chapter 8. The final chapter presents conclusions and future work.

Chapter 7

Observability Models

The previous chapter helped identify aspects to be included in the models for shoulder surfing and frequency attacks. To construct these models mathematical modelling was applied to the results of the simulations detailed in Chapter 6. The modelling process and results are reported in Sections 7.1 and 7.2. Mathematical modelling was used instead of running the simulation each time. This is because running the simulation each time the metric was evaluated for a configuration may give slightly different results, resulting in the metric not meeting the reproducability and repeatability criteria.

7.1 Modeling Shoulder Surfing Attacks

To establish a mathematical model which would allow an estimate of the number of attacks before success, the approach taken was to run the simulations already established in Chapter 6 (for a variety of configurations) 500 times. Wilcox notes that the probability of a Type I error (where the null hypothesis is true, but is falsely rejected) is suitably minimised with a sample size of 100 observations [104, Page 154]. To ensure minimisation of the probability of a Type I error, 500 observations were simulated. The median value of each configuration was noted from the results and stored. These median values were then used in multiple regression (a mathematical modelling technique) to fit a model. This section discusses the results of this fitting process for shoulder surfing. As discussed in Chapter 6 Section 6.2, simulations were performed in two stages; a viewing session followed by an attack session. In a viewing session a challenge set is generated for the user and the attacker notes the passimages selected. However the set of passimages viewed is then reduced according to the recall rate of the attacker. The attack session is then generated. If the attacker has viewed and remembered all the passimages shown in this session, then the attack is successful.

7.1.1 Variables

The independent variable under consideration was the median number of attacks required against a targeted passimage set before successful authentication. As previously noted, the median value was used instead of the arithmetical mean as

Variable	Values
Number of Passimages (p)	{1,...,10}
Percentage of Recall (r)	{5,10,15,20,25,30,35,40,45,50,55,60,65,70,75,80,85,90,100}
Number of Challenge Screens (s)	{1,...,p}

Table 7.1: Shoulder Surfing Simulation Configurations

the distribution of the data was skewed. The independent variables were identified as:

- recall percentage (how much the attacker recalls of the images seen in viewing sessions, r)
- the number of passimages in the user's passimage set (p)
- the number of challenge screens in a challenge session (s)

The values used for the configurations of the simulations were as shown in Table 7.1. Note the caveat that the number of challenge screens had to be less than or equal to the number of passimages. Otherwise passimages would need to be repeated within a session.

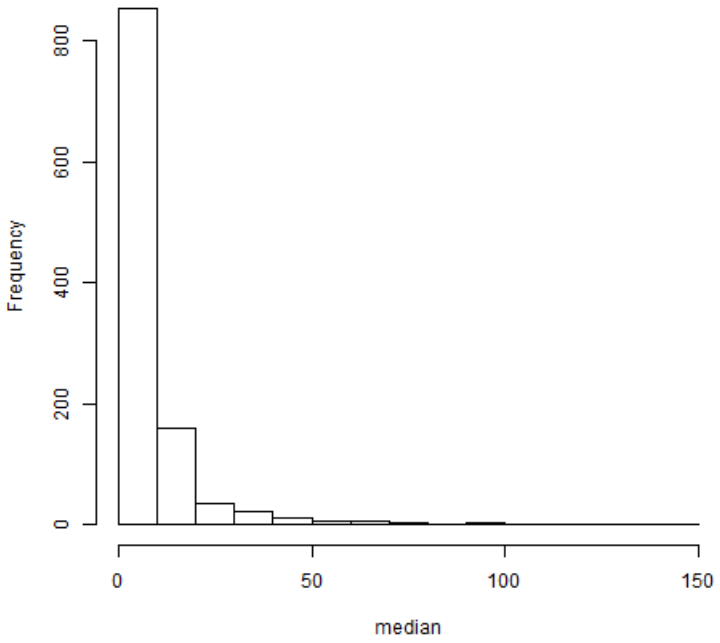
7.1.2 Initial Models

Since there were a number of independent variables to be examined, the data was multivariate. Multivariate data is more difficult to model than the situation where one has only one independent variable and one dependent variable. The approach taken in this work to fit a model to the data follows that proposed by Maindonald and Braun [54, Page 190] which is as follows:

- Examine the distribution of the dependent variables and the independent variable.
- Examination of the scatterplot matrix involving all the dependent variables, in particular look for evidence of non-linearity in the plots of these variables against each other and note any potential outliers.
- If there is evidence on non-linearity in some scatterplots, consider application of transformations to the data to produce more linear results which are easier to fit.
- If distributions are skewed, again consider transforms to establish a more symmetrical distribution.

The first step was an examination of the distributions of the variables. Since the dependent variable values were selected (as shown in Table 7.1), it was necessary to only examine the distribution of the dependent variable. This is shown

Figure 7.1: Distribution of Shoulder Surfing Median Number of Attacks



in Figure 7.1 from which it can be seen that the distribution was skewed to the right. DeVeaux states that such skewed data can benefit from square root and logarithm transformations [18, Page 56]. These transforms were applied to the data to see if it made it less skewed (and hence better for modelling). Applying the log_2 function to the data the resulting distribution is shown in Figure 7.2 which is noticeably less skewed than the original data. The square root of the median data was then examined and is shown in Figure 7.3. This transform of the distribution is still largely skewed to the right. Thus, the best result which was achieved by the log_2 transform and so log_2 of the median number of attacks was used in the modelling process as the dependent variable instead of the median number of attacks.

The next step was to examine the scatterplot matrix which shows scatterplots of each pair of variables in the data set. Further details on scatterplot matrices are provided in Appendix B. This is shown in Figure 7.4. To examine the plot comparing two variables, first find the row with the x-axis variable on the diagonal, then move to the right or left till the y-axis variable is above or below the diagonal. This plot then shows the scatterplot comparing the x-variable to the y-variable selected. For example, to examine the number of screens plotted against the median number of attacks then the plot in the second row from the top and the 4th column would be appropriate. It can be seen from the pairwise scatterplot that the relationship between the median number of attacks and the log_2 of the median and the relationships between the dependent variables (since the values were chosen in a specific way) are as expected. The plots under specific consideration are: each dependent variable against the median, and each

Figure 7.2: Distribution of Shoulder Surfing log_2 of Median Number of Attacks

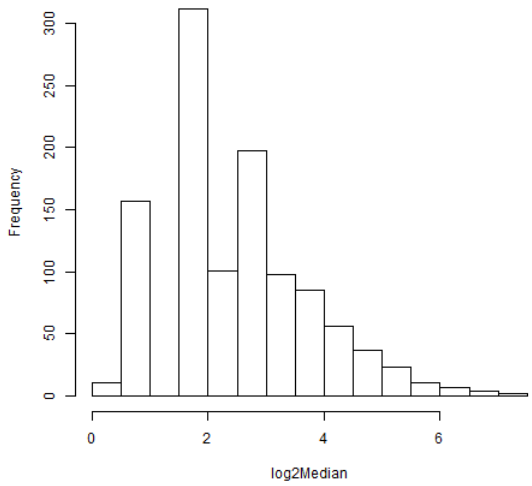


Figure 7.3: Distribution of Shoulder Surfing Square Root of Median Number of Attacks

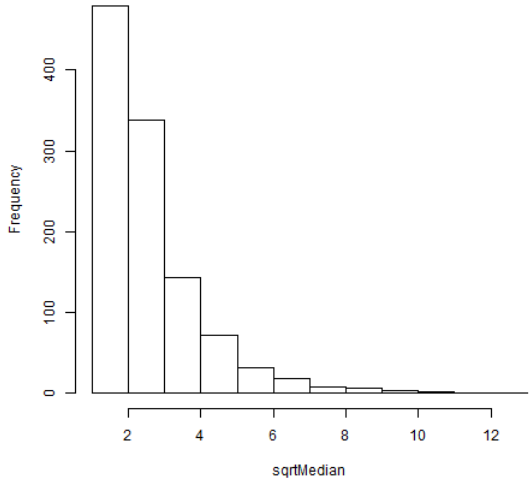
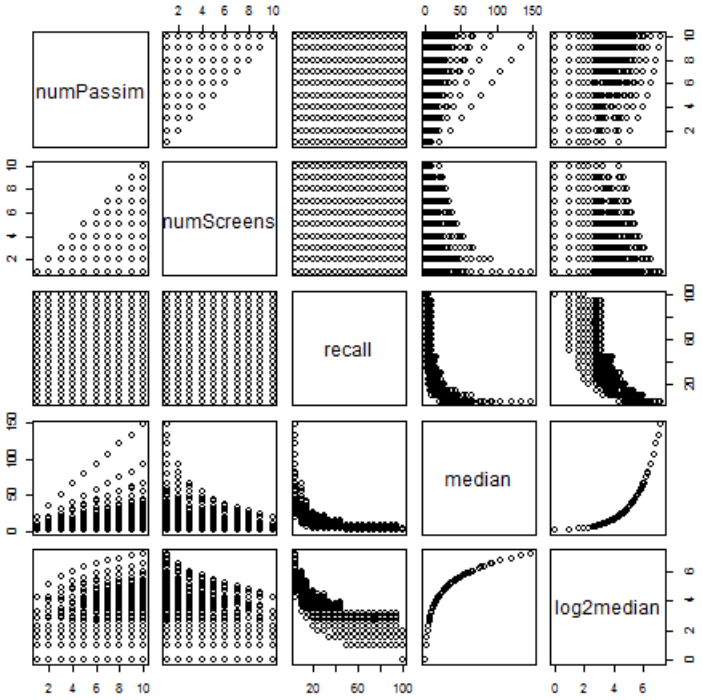


Figure 7.4: Shoulder Surfing Data Scatterplot Matrix



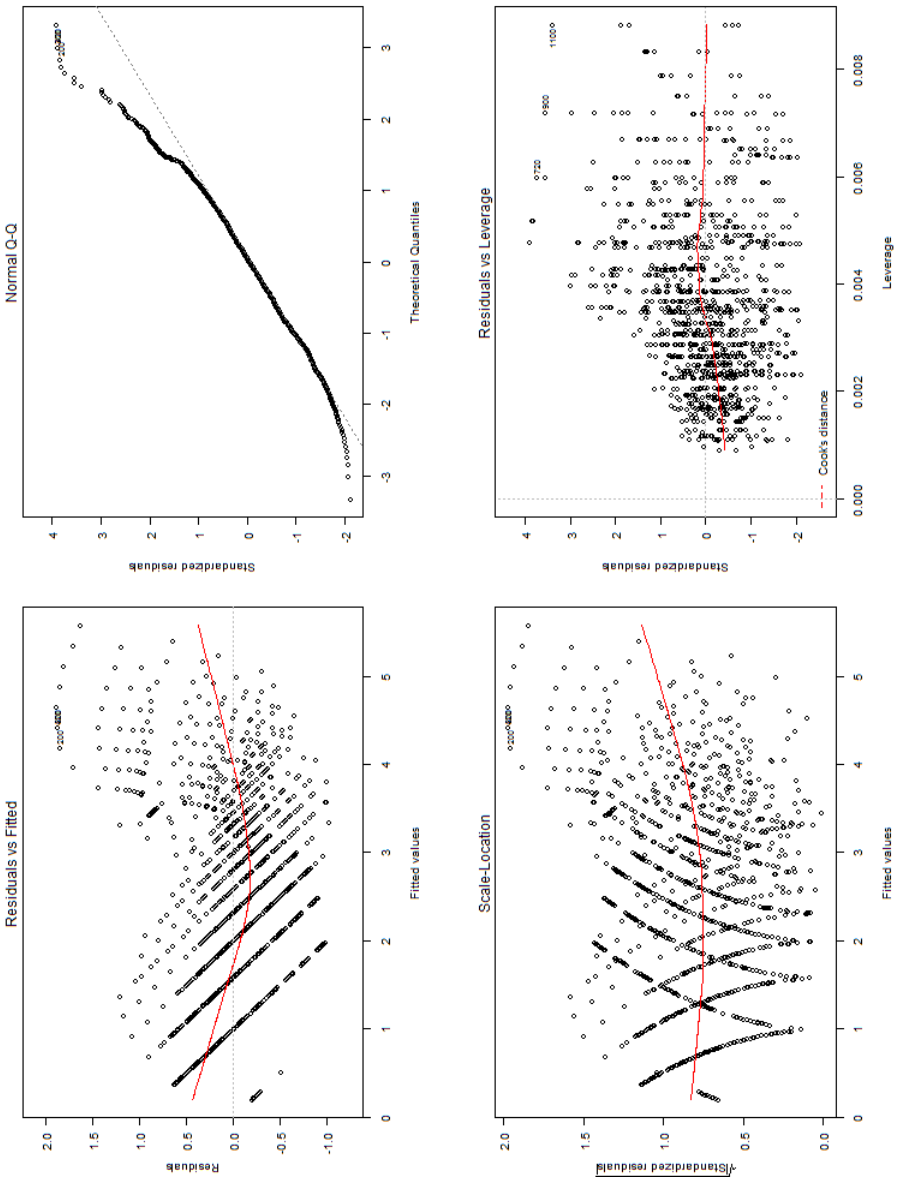
dependent variable against the log_2 of the median.

First the recall against the median was examined. A power shaped plot can be seen from the scatterplot in the fourth row and third column of Figure 7.4. When examining the recall against the log_2 of the median, a more linear correlation is achieved, which is preferable for modelling. Next, examining the number of screens against the median, a slight curve is noticeable, this again is reduced to a more linear relationship when examining the log_2 of the median. When comparing the number of passimages, both against the median and the log_2 of the median, approximately linear relationships are achieved. Models were fitted using the log_2 of the median number of attacks as the dependent variable due to the improved linearity in relation to the independent variables. The number of screens, number of passimages and percentage of recall were the independent variables. For ease of notation, these variables were denoted as s, p and r respectively. The dependent variable was denoted by $log_2 m$

Model 1

The first model examined (model 1, or M1) was as shown in Equation 7.1 which represents the model in terms of the dependent variable (log_2m) being a function of the independent variables. The co-efficient values established by the regression are not shown at this point to keep the model clearer. Instead i, j, and k represent the co-efficients and s represents the number of screens, r represents the recall

Figure 7.5: Shoulder Surfing Model 1 Diagnostic Plot



value and p represents the number of passimages.

$$log_2 m = is + jr + kp \tag{7.1}$$

The residual values (the observed values less the fitted values) provide an indication as to the fit of a model. In particular, the adjusted R^2 value for this model was calculated as 0.8458 which means that 84.58% of the variation in $log_2 m$ was accounted for by Model 1. The residual standard error also provides an indication of how well the model fits the data by providing a measure of the error in prediction. A smaller residual error means the model fits the data better than a larger value [31, Page 121]. The standard error for Model 1 was approximately 0.48 of an attack.

Diagnostic plots were constructed for this model and the results are shown in

Figure 7.5 where four plots are presented. These plots can be used to establish whether the data violates the underlying assumptions of multiple regression using the least squares method, which was employed here. The assumptions of least-squares regression are:

- linear relationship between the independent and dependent variables
- errors are independent
- errors have constant variation
- errors are normally distributed

[18, inside of the very back page]. If the assumptions are violated, robust regression which is robust to violations of the assumptions for least-squares regression should be used [32].

Starting with the top left hand corner plot, the residual values are plotted against the fitted values. This was examined for any pattern as this would indicate a violation of the assumptions of least squares regression that errors have constant variance [31, Page 29]. There appears to be a slight pattern in particular to the left of the plot, applying a polynomial model and possibly robust modelling may result in an improved fit.

To the right of the residuals vs. fitted plot is the Normal Q-Q plot, where if the residuals are normally distributed, the points should lie approximately on the line. Whilst this is the case for the majority of points, in particular to the right of the graph, the points stray from the line. Again, this indicates violation of the normally distributed errors assumption of the least squares method [31, Page 29].

The next plot (the Scale-Location plot) is produced to examine consistency of the variance between the residual and fitted values. It takes the square root of the absolute residuals and plots these against the fitted values. The greater the spread on the vertical axis, the less constant the variance which indicates violation of the assumption of constant variance. This particular plot suggests that the variance is not constant, so a better model and potentially robust regression may need to be applied.

The final plot (bottom right-hand corner) shows the residual values plotted against the leverage. An observation with an extreme value on a dependent variable is a point with high leverage. Such points can have a large effect on the estimate of the regression co-efficients. Also shown on this plot is a line representing Cook's distance, a measure which combines the information of leverage and residual of each observation. It shows residuals which have a large influence in determining the form of the regression line. In particular, points with Cook's distances that are greater than one may require further examination [54, Page 149]. In this plot a large number of points are above one, this indicates that a robust regression method may be more appropriate for establishing a model.

Model 2

As a result of examining the diagnostic plots for Model 1, the next model used robust modelling methods. The MASS package for robust regression in R was used, in particular the rlm (robust linear modelling) method was applied.

First, to establish a better model to use, the non-robust model was applied to establish the R^2 value for the second order polynomial $log_2 m = is + jr + kp + ls^2 + mr^2 + np^2$ where i, j, l, m, n represent the co-efficients. This aimed to establish whether the polynomial may provide a better fit. The results gave an R^2 adjusted value of 0.9142, improved from model 1 by 0.0684. A standard error of 0.36 was achieved, 0.12 less than model 1. Adding in cubic terms to give $log_2 m = is + jr + kp + ls^2 + mr^2 + np^2 + os^3 + r^3 + qp^3$ (i, j, k, l, m, n, q represent the co-efficients) again provided a better result, but only marginally (with an R^2 adjusted value of 0.9383, a difference of 0.0241) and so to keep the model simpler, the model was kept at the level of a second order polynomial.

The model now being clear, the robust regression was applied. Model 2 is shown in Equation 7.2 where i, j, k, l, m, and n are the co-efficients and s is the number of screens, r is the recall percentage, and p is the number of passimages. Since robust modeling was employed here, it is was not necessary to examine the diagnostic plots to establish whether the underlying assumptions of non-robust regression modelling were violated. The error for Model 2 was 0.37, just more than one third of an attack as the possible error and an improvement of 0.11 when compared to Model 1.

$$log_2 m = is + jr + kp + ls^2 + mr^2 + np^2 \tag{7.2}$$

7.1.3 Final Model

Models 1 and 2 had assumed an intersection point in the model. Since there is no situation in which zero attacks would be possible (an attacker will always need to perform at least one attack), the final model accounts for this. Model 3 is the same as Model 2, but has no intersection in the model. When using non-robust modelling methods, Model 3 resulted in an improved R^2 value of 0.93. However the residual error was 0.70 and s^2 was identified as having an insignificant affect on the fitted values. The co-efficient of s^2 was calculated at -0.001749 and had a p value of 0.632, larger than the desired significance level of 0.05. Significance was automatically calculated for the models by R, and all other independent variables had significant impact (i.e. had p values less than 0.05) on the models till this point. The robust version of Model 3 provided a standard error of 0.52, a better result than the non-robust equivalent of 0.70. However, in the non-robust Model 3 the results showed that the impact of s^2 was not significant. Thus the final model, Model 4 was as shown (with co-efficients) in Equation 7.3, the standard error was marginally less at 0.51, and the equation was simpler.

$$log_2 m = 1.3852p - 0.0824p^2 - 0.2143s - 0.0472r + 0.0002r^2 \tag{7.3}$$

Some examples of the predicted and actual values are shown in Table 7.2 (rounded to two decimal points) where it can be seen that some differences were very small, and others were larger. Overall of the 1100 configurations examined approximately 2.55% (28) of the estimated medians were at least twice as large as the observed medians and all were less than 2.5 times as large as the observed medians.

Number of Passimages	Number of Screens	Recall Percentage	Observed Median	Predicted Median	Difference
4	4	60	2	2.38	0.38
4	4	10	10	7.53	2.47
8	4	80	4	5.48	-1.48
9	9	100	1	2.21	-1.21
8	8	50	2	4.70	-2.70

Table 7.2: Shoulder Surfing Model Example Estimates and Observed Values

7.2 Modeling Frequency Attacks

This section reports attempts to establish a mathematical model which allows an estimation of the number of attacks before success for frequency attacks. The approach taken was to run the simulations already established in Chapter 6 for various configurations (discussed shortly) 100 times. The resulting data was used to fit the model. This was less than the number of observations simulated for shoulder surfing attacks as these simulations took longer to complete, but still met the requirements for minimising a Type I error as noted by Wilcox in [104, Page 154]. From the simulations, the median value was recorded and used in multiple regression to fit a model to the data. This section discusses the results of the modelling process. The approach taken was the same as for the shoulder surfing model, as discussed in Section 7.1.

7.2.1 Variables

The independent variables identified were as follows (with abbreviations for use in the model in brackets):

- number of passimages (p)
- number of challenge screens (s)
- number of distractors (d)
- number of distractor images kept constant per passimage (c)

The configurations used were all combinations (subject to the constraint that $p >= s$) of the values shown in Table 7.3. The number of constant distractors (c) used values from 1 through to the number of distractors divided by two (rounded to the nearest integer). This was to reduce the time taken to run the simulations which ran for weeks when using larger values of constant distractors. The configurations using 9 passimages, 15 distractors and more than three screens with a varied number of distractors kept constant were not run. These were attempted, but simulations ran for days (per iteration). With no way of predicting how long they would take to finish it was decided that sufficient data points had been gathered at this point. For each configuration ($p - s - d - c$), this process was run

Variable	Values
Number of Passimages (p)	3,4,5,6,7,8,9
Number of Challenge Screens (s)	1,2,3,4,5,6
Number of Distractors (d)	8,9,10,12,14,15
Number of Constant Distractors (c)	1,2,...,$\frac{d}{2}$

Table 7.3: Frequency Attack Simulation Configurations

one hundred times. The overall median for a given configuration was then used for curve fitting, which is discussed in the following section.

7.2.2 Initial Models

Since there were a number of independent variables (p, s, d, and c), the data was multivariate. As for shoulder surfing, the approach taken to establish an appropriate model followed that proposed by Maindonald and Braun [54, Page 190] .

The distribution of the independent variable (the median number of attacks) can be seen in Figure 7.6 where the data is heavily skewed to the right. To make this more symmetrical, a transform to the data can be applied. Since the log function often works for distributions skewed to the right as noted by DeVeaux [18, Page 57], the log_2 function was applied to the median number of attacks. The result is shown in Figure 7.7. This shows a more symmetrical distribution and so log_2 of the median was used as the dependent variable in the regression instead of the median number of attacks.

Next, the relationships between the different dependent variables were examined. This was achieved through plotting a pairwise scatterplot matrix (shown in Figure 7.8). This combines all the pairwise scatterplots of the variables. Notice the patterns in the relationships between the dependent variables, this is due to the selected values for each configuration. Thus the patterns exhibited were expected and no action was required to reduce apparent relationships between the independent variables.

When examining the scatterplots which plot the dependent variables against the median number of attacks it can be seen that in some cases (for example the number of constant distractors) the relationship is non-linear. This suggests transformation of the data may be required. However, if the scatterplot of the log_2 of the median values against the number of constant distractors shown in Figure 7.10 is examined, an approximately linear pattern is apparent. Note that the scatterplots display all the data. For clarification, an example where the number of constant distractors are changed, but the remaining variables are kept constant is shown in Figure 7.9 which can be compared with the log_2 transform applied as shown in 7.10. As shown in Figure 7.9, the shape follows a power thus

Figure 7.6: Frequency Attacks- Median Attacks Histogram

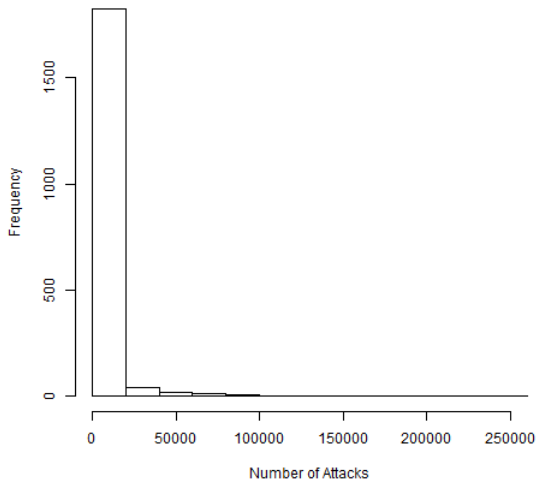
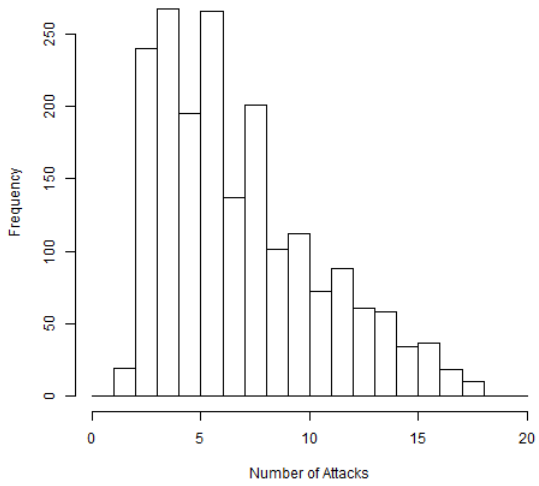


Figure 7.7: Frequency Attacks log_2 Median Attacks Histogram



taking the log provides a more linear result. Due to the increased linearity when using log_2 of the median, no further transformations were applied to the data.

To establish the frequency attacks model, the R programming language and statistical software was used to apply regression modeling to the data by using the "lm" function, which applies least squares regression [1]. Later the R programming language and statistical software was used to apply robust linear modelling using the "rlm" function.

Model 1

The first model, Model 1, is shown in Equation 7.4 (where i, j, k, l represent the co-efficients), which indicates in R to model a relationship between the dependent variable on the left and the independent variables on the right . The R^2 value calculates how much variation in the dependent variable is explained by the independent variables. For this fit the R^2 values was 0.81 and the residual standard error was 1.59. The value of 0.81 for R^2 shows approximately 81% of the variation in $log_2median$ is accounted for by the dependent variables. The residual standard error also provides an indication of how well the model fits the data. It does so by providing a measure of the error in prediction, thus a smaller residual error means the model fits the data better than a larger value as noted by Fox [31, Page 121]. The standard error is for this model was approximately 1.59 attacks, which appeared high given the context.

$$log_2median = ip + js + kc + ld \tag{7.4}$$

In Figure 7.11 four diagnostic plots for Model 1 are shown. The plots shown in Figure 7.11 are examined to establish whether the underlying assumptions of least-squares regression are violated. If the assumptions are violated, robust regression should be used as noted by Fox [32]. Starting at the top left hand corner, the residual values are plotted against the fitted values. Any pattern in this plot indicates a violation of the least squares regression assumption that errors have constant variance, as noted by Fox [31, Page 29]. In Figure 7.11 the normality and constant variance assumptions are violated as shown by the variation in clustering of the points in the residuals vs fitted which then fans out. This indicates non-constant variation and the normality of the residuals is violated as shown in the Q-Q plot where the points stray from the line representing the errors' distribution if they were normal.

Model 2

In an attempt to reduce the violations of constant variation and normality evidenced by plots for Model 1 in Figure 7.11, a model which included quadratic terms was examined. Model 2 is shown in Equation 7.5 and resulted in a marginally improved R^2 value of 0.81 and a marginally reduced residual standard error of 1.57. Looking at the diagnostic plots for Model 2 as shown in Figure

[1] `http://stat.ethz.ch/R-manual/R-devel/library/stats/html/lm.html` accessed 30/03/2012

Figure 7.8: Frequency Attacks Data Scatterplot Matrix

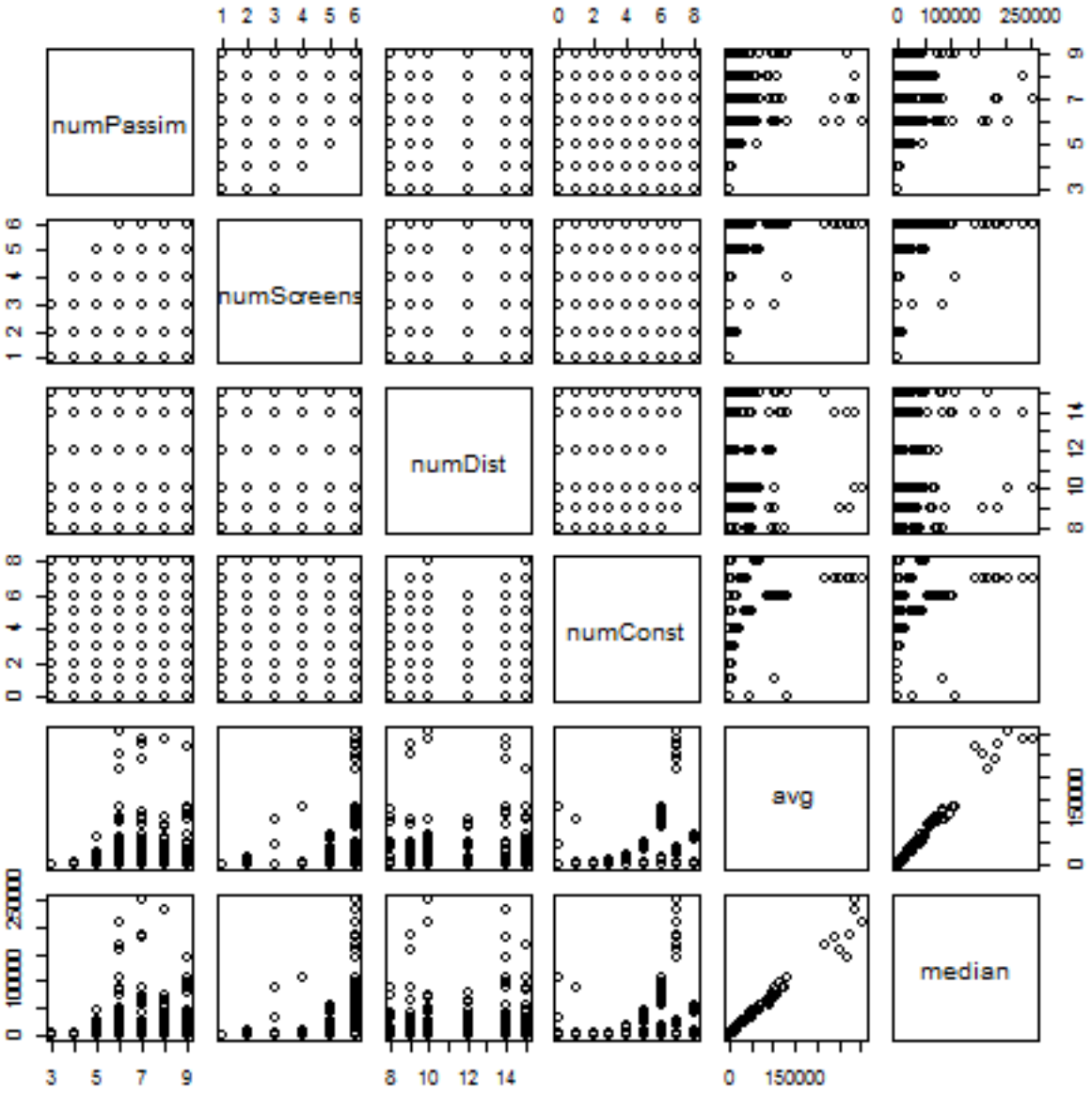


Figure 7.9: Frequency Attacks- Median Attacks vs Number of Constant Distractors Plot

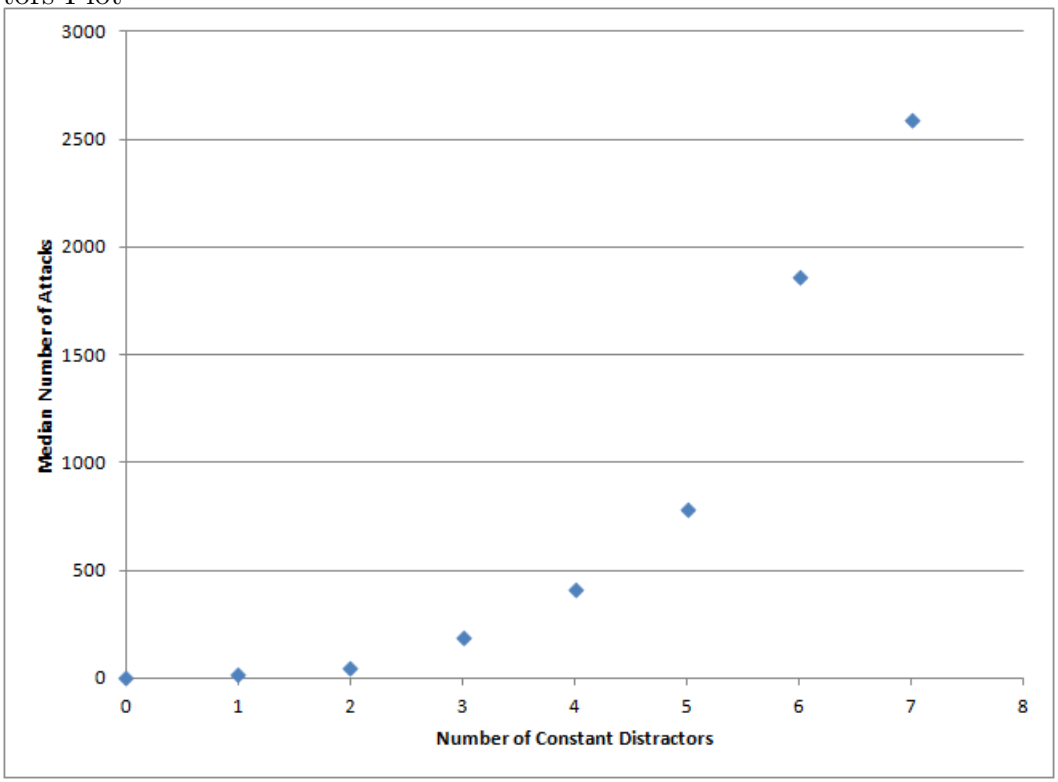


Figure 7.10: Frequency Attacks- log_2 Median Attacks vs Number of Constant Distractors Plot

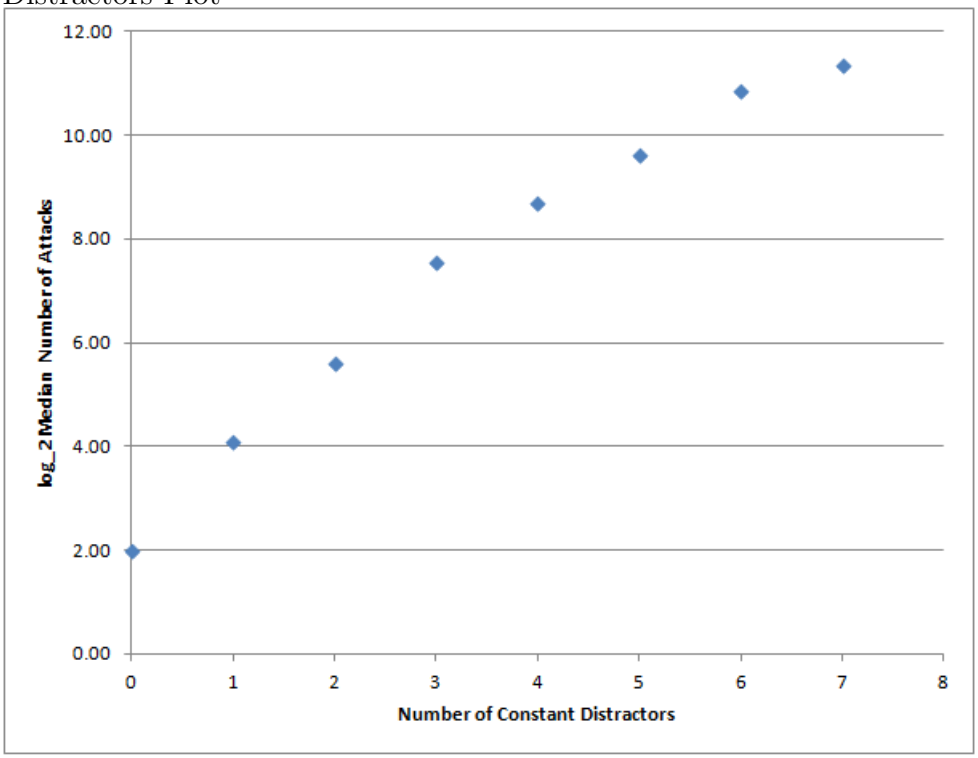
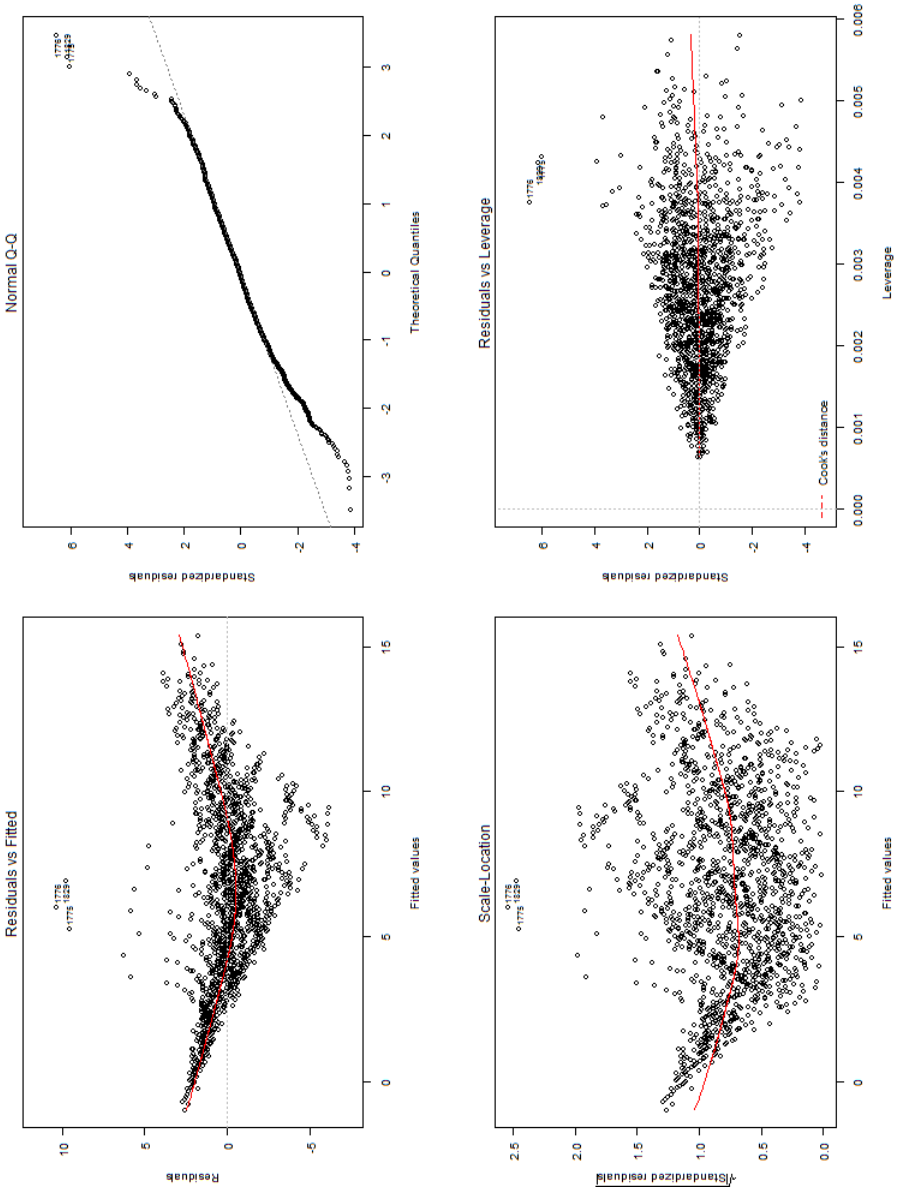


Figure 7.11: Frequency Attacks Model 1 Diagnostic Plot



7.12 (where i, j, k, l, m, n, q, and t represent the co-efficients) there was little improvement in terms of normality and constant variation. Thus robust methods appeared to be a better approach. In addition to this, the models used thus far had assumed an intersection point for the model, however this was removed as at no point will the number of attacks required reach zero.

Model 3

Model 3 removed the intersection requirement from Model 2 by 3 including "-1" on the right side of the independent variable. This indicated to the R program that there was no intersection in the model. This is shown in Model 3, represented by Equation 7.6 (where i, j, k, l, m, n, q, and t represent the co-efficients). Model 3 was applied with the least squares method to establish an improvement on the fit of the model using the R^2 value. This resulted in an R^2 value of 0.95, a large improvement on the previous models due to the removal of the intersection requirement. This meant that 95% of the variation in the log_2 median number of attacks was accounted for by Model 3. However the standard error increased to 1.76 at this point. Thus robust modelling was applied to reduce this and account for the violations in the underlying assumptions of least squares modelling. This results in the final model, presented in Section 7.2.3.

$$log_2 median = ip + js + kc + ld + mp^2 + ns^2 + qc^2 + td^2 \tag{7.5}$$

$$log_2 median = -1 + ip + js + kc + ld + mp^2 + ns^2 + qc^2 + td^2 \tag{7.6}$$

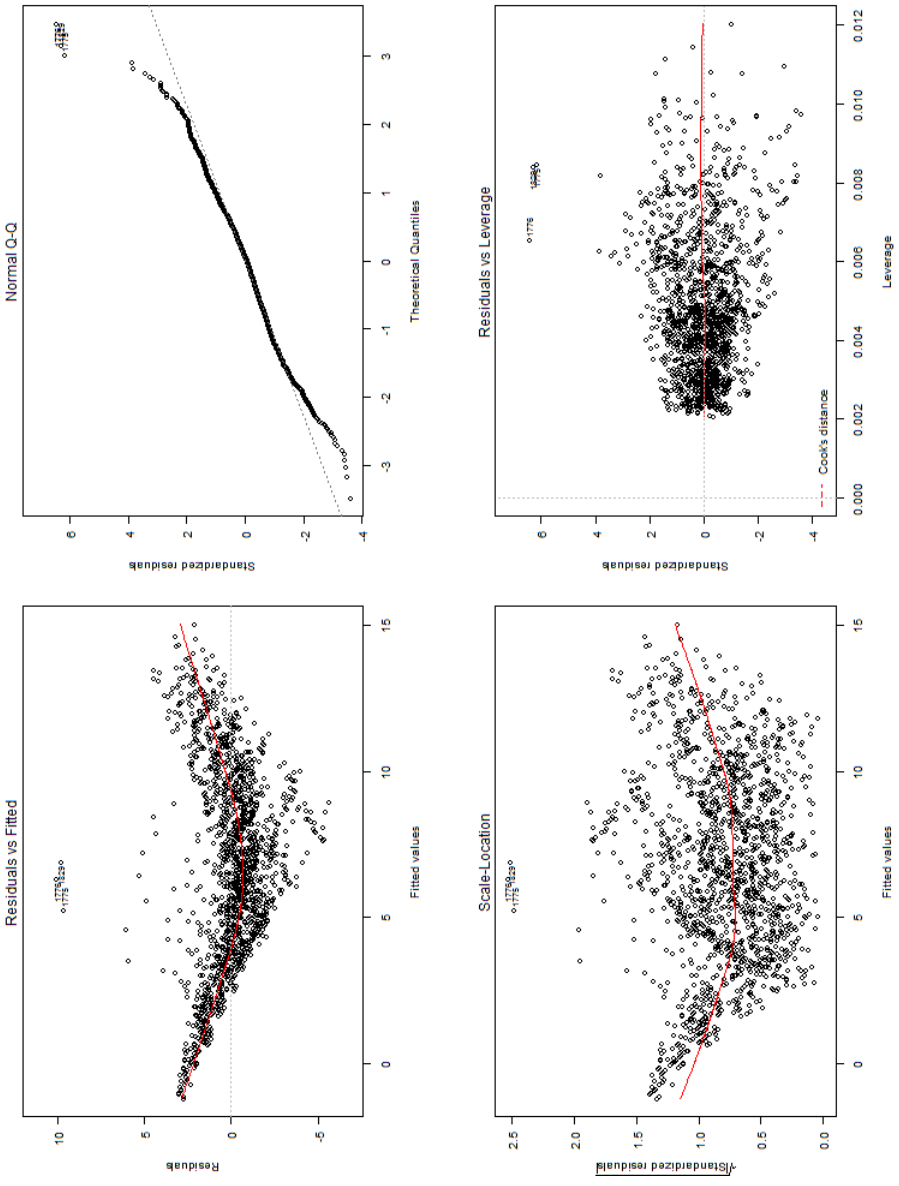
7.2.3 Final Frequency Attack Model

The final model resulted from robust regression using the "rlm" function from the MASS package for R. The relationships to be used for the model were as shown in Equation 7.6, but applying robust regression provided the lowest standard error of 1.3 . The final equation (including co-efficients) for the model was as shown in Equation 7.7, with the log_2 of the median value denoted by $log_2 m$.

$$log_2 m = 0.0156p + 1.6655s + 0.9497c - 0.5575d + 0.018p^2 + 0.0132s^2 - 0.0344c^2 + 0.0309d^2 \tag{7.7}$$

Some examples of the predicted and actual values are shown in Table 7.2.3 (rounded to two decimal points) where a variation in the difference between predicted and observed values can be seen. Whilst this may appear large, the model established provides an estimation of the number of attacks which need not be 100% accurate for prediction. It should be consistent and reasonable (evidenced by the examination of R^2 and standard error) such that one configuration is comparable to another. For example, it is possible to compare a scheme with a configuration of $4-4-8$ with one constant distractor to the same configuration, but using two constant distractors per passimage. In this comparison, the predicted number of attacks rises from 50.47 to 90.75, an increase of approximately 40 attacks. This would allow selection of an appropriate level of resistance. Of

Figure 7.12: Frequency Attacks Model 2 Diagnostic Plot



Number of Passimages	Number of Screens	Number of Distractors	Constant Distractors	Observed Median	Predicted Median	Difference
4	4	8	1	15.5	50.47	-34.97
4	4	8	2	84.5	90.75	-6.25
4	4	9	1	15	49.35	-34.35
6	4	9	3	177	199.65	-22.65

Table 7.4: Frequency Model Example Estimates and Observed Values

the 1916 configurations 59 of the expected values (approximately 3.08%) were at least 10 times as large as the observed median values. This is much larger than the shoulder surfing model, however there were more variables to model in this instance which could increase the difficulty of fitting the model.

7.3 Conclusion and Limitations

This chapter reported the results of modelling shoulder surfing and frequency attacks. The result was two models which can be used to estimate the number of attacks required for a given RBGP configuration. There are a number of potential limitations of this work which should be considered.

Firstly, as noted by DeVeaux [18, Page 198], it is necessary to be careful not to use the models for prediction, i.e. applying configuration values outside the values used in the simulations. This is because the model was based on the configurations used in the simulations, and values outside this could deviate substantially from the models. The models could be used outside the ranges, but care must be taken in interpretation of the prediction. Note that a prediction arises from the metric where configurations outside those upon which the models are based are used. An estimation is provided where configurations used were incorporated into the model. To minimise the need to apply values outside the configurations used, the simulations used configurations from literature to date and values either side. For example, 4 challenge screens are common, and simulations were run with 1 through to 10 screens.

Another potential issue is with the interpretation of the values resulting from these models. One must not consider the values reported as a concrete value of the number of attacks required in any given case. The values reported are estimates based on simulations, in reality other factors such as a combination of shoulder surfing, frequency and guessing attacks could be used which cannot be represented by these models. However, the purpose of this work was not 100% accuracy, but to provide an estimate which could be used to achieve a comparison of the security of different RBGP configurations.

The remaining step in construction of the metric was to combine the guessing and observability models into a complete model, and evaluate the result against the requirements set in Chapter 3. This is reported in Chapter 8.

Chapter 8

Security Metric and Evaluation

The approach to the final metric was to combine the estimate number of attacks required for each of the identified attacks. The attacks included in the final metric were as follows:

- Random guessing attacks
- Semantic ordered guessing attacks
- Shoulder surfing attacks
- Frequency attacks

Chapter 4 reported models for random guessing and semantic ordered guessing attacks. Chapter 7 reported the models for shoulder surfing and frequency attacks. This chapter considers the short comings of the initial metric, presents the final metric and evaluates it against the criteria set in Chapter 3 Section 3.4.

8.1 Comparison to Initial Metric

As detailed in Chapter 5, the previous approach to quantifying the security of a RBGP scheme was to establish a scoring system based on the resistance to the identified attacks. These attacks were shoulder surfing attacks, semantic ordered guessing attacks, and frequency attacks. The scoring system was presented in flowcharts. There were a number of issues with this approach, these are detailed as follows:

- The flowchart approach introduced ambiguity and affected reproducability.
- Random guessing attacks were not incorporated into the metric.
- The values assigned to the level of resistance were based upon the efficacy of countermeasures and had little relation to the security beyond this. Effectiveness of the countermeasures was primarily based on the literature review, with the exception of the SOGA values which were based on the attack simulations.

- The metric didn't incorporate the impact of different RBGP configurations, e.g. number of challenge screens, passimages and distractors, on the security levels .

As a result of these identified issues, the adjustments to the final metric were as follows:

- The flowchart approach was removed completely.
- Guessing attacks were separated into two distinct areas for measurement, SOGAs and random guessing attacks.
- Further investigation into the efficacy of various countermeasures was established by simulations.
- The security metric was altered to reflect the resistance to attacks by estimating of the number of attacks required on a user account before success.

With these adjustments in mind, the final metric is presented in the next section.

8.2 The Complete Finalised Metric

The final metric is a 4-tuple consisting of four estimated values of the number of attacks required before successful authentication. There is one estimate for each of the attacks; random guessing, semantic ordered guessing, shoulder surfing and frequency attacks. The metric is denoted as shown in Equation 8.1 where RG denotes the random guessing value, SOGA denotes the semantic ordered guessing attack value, SS denotes shoulder surfing value and FREQ denotes frequency attacks value. If for any of the attacks a countermeasure is implemented which means the attack is not possible, then a * is used to denote this. The calculation of each of the component parts is summarised in the following four subsections.

$$(RG, SOGA, SS, FREQ) \tag{8.1}$$

8.2.1 Random Guessing Value

The estimate of the number of random guessing attacks required before success is obtained from the calculation of the probability of success. This is commonly reported as $\frac{1}{x^s}$ where x is the number of images shown on a challenge screen (the number of distractors plus one passimage, $d+1$) and s is the number of challenge screens. The denominator of this calculation is used to provide an estimate of the number of random guessing attacks required before success, thus the RG value is calculated as shown in Equation 8.2

$$RG = (d+1)^s \tag{8.2}$$

8.2.2 Semantic Ordered Guessing Value

The calculation of the number of semantic ordered guessing attacks required before success relies on an estimate of the number of attacks which are successful for a given potential passimage set. This is calculated by performing simulations of the SOGA based on the category distribution of real user choices. For the purposes of this work, evaluation of the success rates of SOGAs was carried out for four configurations as detailed in Chapter 4. The following percentages of success were achieved:

21% of passimage screens were successfully attacked where distractors were selected randomly (ignoring the semantic categories). 23% of passimage screens were successfully attacked where distractors were selected from distinct passimage categories (excluding the passimage category). 20% of screens were successfully attacked where distractors were selected from passimage categories (excluding the passimage category). These success rates can be used as estimates for user selected passimage schemes where the images can be split into semantic categories.

Once the percentage of success has been estimated, one can calculate the estimated number of attacks as shown in Equation 8.3 where s denotes the number of challenge screens. If the passimages are assigned to the user, then this attack is not applicable and this is denoted by *.

$$(\frac{100}{successPercentage})^s \tag{8.3}$$

8.2.3 Shoulder Surfing Value

As for the semantic ordered guessing value, one must estimate the percentage of recall rate or success rate of an attacker given a specific shoulder surfing countermeasure. This can be done by performing an experiment to establish how successful shoulder surfing attacks are for the countermeasure implemented. Alternatively an estimated value of successful recall between 1 and 100% can be selected. Once the recall value has been established, the shoulder surfing value can be calculated as shown in Equation 8.4 where p denotes the number of passimages in a user's passimage set, s is the number of challenge screens in a session, and r is the percentage of recall. The modelling was based on log_2 of the median number of attacks and so the final equation includes a power of 2.

$$SS = 2^{1.3852p - 0.0824p^2 - 0.2143s - 0.0472r + 0.0002r^2} \tag{8.4}$$

8.2.4 Frequency Value

Unlike the previous two calculations, the frequency value relies primarily on the configuration of the RBGP scheme. This includes the number of distractors kept constant per passimage (denoted by c) in addition to the number of screens (s), the number of distractors per screen (d), and the passimage set size (p). The frequency value can be calculated as shown in Equation 8.5. The modelling was based on log_2 of the median number of attacks and so the final equation includes

a power of 2.

$$FREQ = 2^{0.0156p+1.6655s+0.9497c-0.5575d+0.018p^2+0.0132s^2-0.0344c^2+0.0309d^2} \quad (8.5)$$

This equation should only be used if the number of distractors kept constant per passimage is less than the number of distractors per challenge screen. If the challenge screens are constant then a frequency attack will be reduced to a random guessing attack. In this case, * denotes the attack is not applicable.

8.3 Evaluation of Final Metric Against Requirements

Each of the component parts of the 4-tuple metric have now been discussed. To determine if the thesis statement has been achieved, each of the requirements of the metric is discussed to establish whether they have been fulfilled.

8.3.1 Repeatability

The requirement for repeatability was such that if the metric was calculated for the same scheme repeatedly, the same result would be achieved. The metric is based upon calculations, and should result in the same values each time within the following potential limitations:

- There could be issues with rounding, in particular for the SOGA, SS, and FREQ values. To avoid this, the number of significant places is defined as 0 (to ensure a whole number, as less than one attack doesn't make literal sense). Rounding is defined as follows. One should round up where the remainder is 0.5 or above, and round down to the nearest integer where the remainder upon division by 1 is less than 0.5.

- Part of the SOGA value depends on a percentage estimated from simulations of the attack against user choices for a given passimage set, or is selected from a number of "suggested" percentages. If different percentages are used, different results will be obtained. However, if the same percentage is used the same values will be achieved.

- Similar to the SOGA calculation, the shoulder surfing value depends on estimation of the percentage of recall of the attacker. If different values are used here, then different results will be achieved. However the results will be repeatable assuming the same values are used.

Thus, whilst there are potential limitations due to rounding and establishing success rates for SOGA and SS, repeatability is achieved if the same estimations for percentage of success/recall are used and rounding is applied as described. To ensure repeatability, a program to establish the values could be written which would read in sample user choices, the categories and images and the number of successful shoulder surfing attacks. An empirical user study should be conducted to establish the success rates used in the program.

8.3.2 Reproducibility

The requirement of reproducibility was such that if different assessors were to carry out the evaluation, the result would be the same each time. This has the same limitations as repeatability-one must ensure the same rounding is applied and that the same percentages are used for SOGA and SS. Outside these caveats, the metric values are calculations from models and so reproducibility is achieved.

8.3.3 Extensibility

The requirement for extensibility was to ensure that the metric would be suitable for extension to include new attacks if more data were to become available. To add a new attack the approach taken can be one of two (or a combination of both). The first approach is to perform a simulation, the second approach involves an empirical study. Both are discussed below:

Simulation Approach

1. Implement a simulation of the RBGP scheme (challenge screen, distractor selection, images).
2. Implement an algorithm which simulates the attack.
3. Perform the simulation at least 100 times ([104, Page 154]) for multiple configurations and note the number of attacks before success.
4. Use the data produced from the simulations and perform statistical analysis to identify the significant dependent variables.
5. Perform multiple regression to establish an equation to estimate the number of attacks before success.

Empirical Study Approach

1. Implement the RBGP scheme with the configuration as required.
2. In your study, conduct the attacks against a challenge screen for user selected images (random images if the user choice doesn't influence the attack success).
3. Calculate the percentage of attacks which were successful.
4. Then calculate the metric value as $(\frac{100}{percentageOfSuccess})^s$.

The simulation approach can be taken where there is no human element to the attack. For example frequency attacks have no reliance on user choice or an element which varies depending on countermeasures which cannot be modeled by a simulation. Empirical approaches are required where there is a human element which cannot be simulated. Alternatively, a combination of the two approaches can be taken, as was the case with the SOGA evaluations in Chapter 4.

8.3.4 Objective

The requirement of objectivity was to ensure that the metric would not depend on a subjective measurement. Since the metric is based on calculations which depend on the configurations of RBGP schemes, this requirement is achieved.

8.3.5 Quantitative

The quantitative requirement was to ensure that the metric was not a heuristic evaluation (e.g. "high security" or "low security") but a value which could be related to the strength of resistance to the attacks identified. Since the metric is based upon calculations and results in an estimate of the number of each attack required, this requirement has been achieved.

8.4 Using the Metric

This section aims to consider the use of the metric by providing details on how to apply it to a RBGP scheme and the use of the metric in decision making.

8.4.1 Calculating the Values for the Tuple

To calculate the component values for the tuple the following approach is taken:

1. Examine the RBGP scheme to establish values for the configuration. This includes the following: number of passimages in a user's passimage set (p), number of challenge screens in a session (s), number of distractors shown per challenge screen (d), percentage of recall of the attacker for shoulder surfing (r), the number of distractors kept constant for each passimage (c), and the estimated success rate (as a percentage) of a semantic ordered guessing attack ($successPercentage$).

2. Establish if any of the attacks are not feasible for the scheme being examined. For example if images are assigned then a SOGA is not applicable. Alternatively, if challenge screens do not change distractors between sessions then a frequency attack is not possible. For any such attack, use a * in the appropriate place in the metric tuple to denote the attack is not applicable to the scheme.

3. For each of the attack types remaining use the appropriate configuration values (identified in step one) in the appropriate mathematical model (i.e. the equation) described in section 8.2.

4. Round each of the model values to the nearest whole integer.

5. Combine the values in the order (RG, SOGA, SS, FREQ) to obtain the final metric as applied to the scheme under consideration.

The metric as applied to the scheme under consideration can now be examined in terms of the security. This process is detailed in the following section.

8.4.2 Using the Metric to Examine a RBGP Scheme

Once the metric has been applied to a scheme, the result is an estimate of the number of attacks required for each of the following attacks: random guessing (RG), semantic ordered guessing (SOGA), shoulder surfing (SS), and frequency (FREQ). Using the metric one can determine the ease with which each individual attack could be successful, where a smaller value indicates an easier attack and a larger value indicates a harder attack.

The first two values in the metric provide estimates of the guessability of the scheme, if these numbers are small then the scheme could be vulnerable to these types of guessing attacks. The last two value in the metric provide estimates of the observability of the scheme. If these numbers are small then the scheme could be vulnerable to these types of observability attacks.

A typical RG value is $9^4 = 6561$, which is comparable with a PIN which provides a slightly higher guessability of $10^4 = 10000$. A number smaller than this indicates a lower level of security than that achieved by a PIN system. If the system allows a user to perform incorrect authentication attempts many times without locking the user out, then a random guessing value less than that of a PIN may be too low. If one were to assume an attempt rate of one authentication session per minute and no policy to lock the user out, then this would equate to approximately 4.6 days or 109 hours and 21 minutes. This could be longer or shorter depending on the time taken to attack, which could vary based on the number of machines used, and the speed of the attacker. If this number were to be deemed too small then it could be increased by adding extra distractors per challenge screen, or use of additional challenge screens. However, this could impact on the usability of the scheme by making it harder for the genuine user to authenticate due to an increased time to carry out the process. This could mean a trade-off between security and usability.

If a SOGA is feasible for the scheme under consideration, then such an attack will provide a lower estimated number of attacks than the same scheme's RG value. This is because a SOGA is at least as successful as a RG attack. A SOGA value for a PIN comparable system which uses nine images per challenge screen and (e.g. Faces [15]) gives a SOGA value of 514. If we assume again a rate of one attack per minute and no lock policy, this would result in 8.57 hours of attacking to successfully attack the system. This could be an important concern and would indicate use of either a lock out policy or a countermeasure for this type of attack. For example assigning distractor images from the same category as the passimage. However, this could also impact on the usability of the scheme by making it more difficult for the user to recall their passimage if all images are from the same scheme (intra-category error, as noted by DeAngeli *et al.* [17]). Once more, this could mean a trade-off between security and usability.

A shoulder surfing attack success rate can be estimated by the SS model. The difference with this model, compared to the others in the tuple, is that a user recall rate (r) is incorporated into the model. This is to reflect the large variety of counter measures possible for shoulder surfing. The model incorporates the impact of the recall rate and the other configuration variables (e.g. number of challenge screens, number of passimages, etc.) on the number of attacks before

success. If the scheme being examined employs no countermeasures for shoulder surfing which results in approximately 60% recall rate of the attacker (as shown in Tari *et al.*'s PassFaces implementation with mouse selection [87]) then a value of 8 is achieved. This means the attacker would in theory observe 8 authentication attempts before successfully attacking the system. If one were to use a countermeasure to mitigate the attack, a higher number of attacks would be required. However, this could impact the usability of the scheme. By making it harder for the shoulder surfer to establish which image is selected, the genuine user could also find it more difficult to ensure the correct image is selected. Once more this indicates a trade-off between security and usability.

A frequency attack success rate can be estimated by the FREQ model. If the challenge screens are kept constant for each challenge session, then a frequency attack is not feasible, providing optimal security against this type of attack. However, as noted by Deffenbacher *et al.* [19] this impacts on the memorability of the passimages, and users may select distractor images as they become more familiar with them. This would result in false negative authentication. If no counter measure was used for a PIN comparable scheme with 4 challenge screens, 8 distractors per screen, and 4 passimages, then a frequency attack value of 27 is estimated. This would mean approximately 27 authentication sessions would need to be observed, noting the images presented and their frequencies. Subsequently the passimage set would be attacked using the established frequencies. If we were to assume an increased time to note the frequencies, increasing the time per attack to 5 minutes (an arbitrary value selected merely to demonstrate a point, this could be changed) would equate to a total attack time of 135 minutes, or 2 hours and 15 minutes.

Overall one should consider the context in which the scheme is deployed. Whilst shoulder surfing may appear to be the largest vulnerability, if one were authenticating in a secure environment where no observation was possible then this may not be a concern. Thus, the metric allows the security of the scheme in question to be considered in terms of guessability and observability. Decisions on whether the levels reported are acceptable would be linked to context and consideration of the trade off between security and usability.

8.4.3 Using the Metric to Compare RBGP Schemes

In the previous section the use of the metric to examine the security of an individual scheme was discussed, this section aims to address how the metric could be used to compare the security of multiple schemes. For simplicity, this section considers the comparison between two schemes as an example. This could be easily extrapolated to examine more than two schemes.

To compare two schemes, scheme 1 and scheme 2, one should consider the values for each of the models within the tuple. Let us call the constituent tuple values of each scheme RG1, SOGA1, SS1, and FREQ1 for scheme 1 and RG2, SOGA2, SS2, FREQ2 for scheme 2. It is then possible to compare the values for each of the attacks e.g. RG1 can be compared to RG2 and so forth. Thus, if for example RG1 is larger than RG2 then we can deduce that scheme 1 is more resistant to random guessing attacks. Similarly for the remaining attacks.

In using this for decision making, for example to select an appropriate scheme, one should consider the context in which the scheme will be deployed. For example if observability is a key concern, but guessability less so then particular attention should be paid to the observability values. This may result in a situation where one scheme has a higher resistance for one attack and a lower resistance for the other whilst the scheme it is being compared to has the opposite resistance. For example, assume we are considering the observability as important and scheme 1 has a higher FREQ value and a lower SS value, whilst scheme 2 has a higher SS value and a lower FREQ value. In this case it may be more difficult to make a decision as to which scheme would be most appropriate. It is suggested that in such circumstances an examination of context to a finer granularity be made. In this example this would mean examining whether shoulder surfing or frequency attacks are more of a concern. If shoulder surfing is a higher concern in the context, then scheme 2 should be selected. Scheme 1 should be selected if frequency attacks are more of a concern. With guessability, as a SOGA is an adapted form of a random guessing attack it is suggested that if this attack is applicable, the scheme with a higher SOGA value should always be selected.

8.4.4 Use of Decision Making

The previous sections on evaluating the metric for a single scheme and using the metric as a basis for comparison suggest approaches which could reasonably be described as qualitative. The decision to be made when considering schemes is a benefit-risk trade-off and thus it could be appropriate to apply decision theory to establish which schemes are appropriate. The decision theory approach aims to provide a framework to allow rational choice between alternatives where the outcome of the choice is not completely known [59]. As detailed by North [59], a decision theory approach can proceed as follows. The approach involves assigning numerical values to the possible outcomes using a utility theory approach to establish preferences of outcomes. This can then be used to evaluate the outcomes by use of a utility function which mathematically assigns a number to each outcome. The next stage is to use probability theory to assign the likelihood of the possible outcomes. The utility of each of the decisions can be calculated using the values and probabilities assigned to the outcomes, this allows the decision maker to select the decision with the highest utility value as the best decision.

However, this approach could be difficult to apply in a real world situation. As shown by Fischhoff *et al.* [29] people perceived that current levels of risk of 30 activities and technologies (such as electric power, motorcycles, x-rays, and pesticides) were unacceptably high. The implications of this could be that those applying the metric might perceive the risk to be too high and thus assign more weight to any given aspect of the metric. From this, it could be suggested that using the metric to decide an acceptable level of risk could be inherently challenging due to the disparate nature of the theory of decision making and the observed behaviour as noted by Fischhoff *et al.* [29].

This topic is something which could be evaluated further, but is noted briefly here to provide a parallel between the suggested approach and decision theory.

Continuing in the previous approach, the following section aims to provide

some examples of the application to two individual schemes and then provides a comparison of these schemes.

8.5 Examples

This section aims to provide some example applications of the metric.

8.5.1 Application of Metric to PassFaces

In Chapter 5 the first metric proposed was applied to the PassFaces scheme. The application of the final metric to the PassFaces scheme is presented here. From reviewing the white paper [1] the following information on the configuration of the scheme was extracted:

- In general, four passfaces are assigned to a user and to authenticate users must identify their passfaces from four challenge screens each showing a passface and eight distractors.
- No challenge screen contains faces from the other screens in the session i.e. no distractors are repeated within a challenge session.
- The same distractors are used each time for a given passface.
- The option is provided to use keyboard selection of the passface from a challenge screen.
- A "mask" is applied to the faces after selection. However in the online demo images are highlighted upon selection.

Thus the configurations in applying the metric were as follows $s = p = 4$, $d = 8$, $c = 8$. Images are assigned and so a SOGA is not applicable, represented by *. Images appear highlighted upon selection potentially making shoulder surfing more successful as shown by Tari *et al.* [87] where approximately 60% of attacks were successful, thus this value is used for the recall rate of PassFaces. The resulting metric for PassFaces is then calculated as shown in Equation 8.6 where * represents that a frequency attack will be no better than random guessing since the number of distractors kept constant is equal to the number of distractors per screen.

$$(6561, *, 2, *) \tag{8.6}$$

From this result the weakest aspect of the security is shoulder surfing. If one were authenticating where the process could be viewed, then this could be an issue. The number of attacks required could be increased by doubling the number of passimages to 8, which results in a SS value of 7. It could be further increased by allowing keyboard entry, which results in a success rate of approximately 11% (again, shown by Tari *et al.* [87]) which results in a shoulder surfing value of 22.

[1] available at `http://www.realuser.com/published/TheScienceBehindPassfaces.pdf`

8.5.2 Application to Adapted VIP

Whilst the VIP scheme proposed by DeAngeli *et al.* has only one screen, it is adapted here to multiple challenge screens. This allows the metric to be applied to the scheme and provides an additional example. The metric is now applied to the adapted VIP1 scheme, reported in [16]. Since there are four passimages in a session $s = 4$ is used. From the defining paper, the configurations were as follows; with four passimages in a challenge session, $p = 10$, $d = 9$, $c = 0$. The shoulder surfing recall was estimated at 60% (as assumed for the PassFaces scheme) since there were no details on highlighting the images upon selection, but the images were selected on a touchscreen. A SOGA was not applicable to the adapted VIP1 since the images were randomly assigned to the users. There was no mention of maintaining constant distractors for passimages and so this was assumed to be 0. It should be noted that the random guessability value may underestimate the resistance as the calculations do not account for sequence, which is incorporated into the adapted VIP1 scheme. Also, location was maintained and thus there is potential for the shoulder surfing value to be overestimated as could be arguably easier to shoulder surf a passimage which stays in one position. The resulting metric is shown in Equation 8.7.

$$(10000, *, 6, 80) \tag{8.7}$$

8.5.3 Comparison

The purpose of this metric is to allow consistent comparison of the security of RBGP schemes. Using the metric to demonstrate this it is now possible to compare the security of the PassFaces scheme with the security of the adapted VIP1 scheme. It can be seen from the metrics reported in Equations 8.6 and 8.7 that the PassFaces scheme is more secure in terms of frequency attacks, but the adapted VIP1 scheme is more secure against random guessing and marginally more secure against shoulder surfing attacks due to the increased passimage set size. Both schemes are equally secure against SOGAs since passimages are assigned to users. In selecting an appropriate scheme, one would need to consider the context under which the mechanism would be used. For example, if shoulder surfing is not a concern then the PassFaces scheme may be a better fit.

8.5.4 Application to RBGP Schemes

In Table 2.1 on page 36 a list of the configurations of the RBGPs covered in literature to date were provided. Here this list is reduced to include only schemes which have one passimage per challenge screen which also excludes five of the six ordered schemes. Also excluding the PassImages scheme by Charrau *et al.* [12] (since the scheme uses multiple passimages over multiple screens) leaves a set of seven RBGP schemes which the metric can be directly applied to. Where there was no mention of constant distractors, a value of 0 is assumed. It is assumed that a percentage of success for SOGAs as 21% for all schemes to provide consistency and a shoulder surfing recall rate of 60% which was reported by Tari *et al.* in [87]

RBGP Scheme	Passimages	Screens	Distractors	Constant Distractors	Metric Value
PassFaces [1]	4	4	8	8	(6561,*,8,*)
Faces [15]	4	4	8	8	(6561,514,8,*)
Doodles [63]	4	4	15	0	(65536,514,7,56)
Awase-e [85]	9	4	8	0	(6561,514,24,51)
Pering [61]	10	10	3	0	(1048576,5995247,10,380253)
Everitt *et al.* [28]	5	5	8	8	(59049,2449,11,*)
Mikons [70]	4	4	15	15	(65536,514,7,*)

Table 8.1: RBGP Configurations Summary

as the success rate where images were highlighted. For Pering [61], there was no number of passimages specified. Since the number of passimages needs to be at least 10 (as there are 10 challenge screens) a value of $p = 10$ is used to apply the metric. In Table 8.1 the configurations and metric values are reported. It can be seen from this table in the row for Pering that the metric appears to fail for the SOGA value as it provides a much higher value than random guessing. This is because a SOGA success rate of 21% was assumed, which provides a higher revised number of images per screen than the actual number of images per screen. This provides evidence that further simulations for SOGAs would be required to improve accuracy.

8.6 Discussion - Context and Limitations of the Metric

As has been highlighted in conclusion and discussion sections in the component parts of the final metric, there are a number of limitations of the metric. These are summarised as follows:

- The final metric models are based primarily on simulations, and so the reality of attacks may be different. However, a large-scale user study was attempted and was unsuccessful in gathering sufficient data. Thus, this approach provided a reasonable alternative.
- The simulations performed do not consider the possibility of combining a number of attacks for optimal success, e.g. a shoulder surfing attack could be combined with a frequency attack, which could result in a larger success rate.
- The work primarily considers RBGP schemes with a predetermined set of

images (which was constant for the duration of the work) and does not consider user provided images.

- There are limitations of the SOGA work in that to get a practical estimate of the SOGA value one needs to conduct a user study to collect user choices passimages for their own potential passimages set. Estimates can be used from the percentages achieved for the schemes examined in Chapter 4, but this could be very different from an alternate potential passimage set and different users.

- Similar to the SOGA issue, the shoulder surfing value requires an estimate of the recall rate (or success rate) of a RBGP scheme in addition to the configuration. This is to account for the variability in countermeasures which cannot be simulated and require user studies to establish the efficacy of a given countermeasure.

- The metric is focused on a specific model of RBGP schemes. This could be extended to include the other possible models. This is discussed further in Chapter 9.

The next chapter concludes this thesis with a discussion of the contributions of this work, the research outcomes and future work considerations.

Chapter 9

Conclusions and Future Work

This chapter concludes the thesis by presenting a summary of the contributions, detailing how the thesis statement has been answered, and considering possible future work.

9.1 Contributions to Research

This thesis reported the construction of a model for the security of recognition-based graphical passwords. The overall model consisted of four smaller models which allow an estimation of the number of attacks required for the following attack types; random guessing, semantic ordered guessing, shoulder surfing, and frequency attacks. This provides a consistent, repeatable, reproducible, objective and quantitative method for comparing the security of recognition-based graphical password schemes. As indicated in Section 2.5, the RBGP scheme must meet a number of requirements before application of the metric, these are as follows:

- At most one passimage can be shown per challenge screen.
- The order of input of the graphical password should be irrelevant.

The contributions to research are as follows:

- Threat model for RBGP schemes - an in-depth examination of the security aspects of RBGPs in literature to date resulted in the extension of the guessability, observability and recordability categorisation (established by De Angeli *et al.* in [17]) to create a threat model.
- Construction and analysis of a new guessing attack - for RBGP schemes which permit user selection of passimages which can be categorised according to their content. A semantic ordered guessing attack (SOGA) prioritises guesses using images from more popular categories. The attack demonstrates a higher probability of success compared to random guessing.
- Mathematical model estimating the number of attacks required before success for a SOGA.

- Simulations of shoulder surfing attacks - These simulations demonstrated the efficacy of having a passimage set size which exceeds the number of challenge screens as a countermeasure for shoulder surfing. They also provided evidence that increasing the number of challenge screens decreases the number of attacks required before successful shoulder surfing.

- Simulations of frequency attacks - These simulations demonstrated the efficacy of countermeasures. This included evidence showing that dummy screens do not have a significant impact on the number of attacks before success when compared with no countermeasures. Also, increasing the number of challenge screens was shown to decrease the chance of success until the number of challenge screens approached the number of passimages. At this point the chance of success increased.

- Mathematical model to estimate the number of attacks required for a shoulder surfing attack to be successful -This incorporated a percentage of recall variable. The recall variable can be estimated or calculated from a user study which implements a countermeasure unrelated to the configuration of a RBGP (i.e. the number of screens, passimages in a user's passimage set etc.) e.g. obscuring the image selection. The shoulder surfing value can then be calculated which incorporates the RBGP configuration. This allows selection of an appropriate level of resistance as the variables can be altered until a required level of resistance is achieved.

- Mathematical model to estimate the number of attacks required for a frequency attack to be successful - This was based wholly upon the configuration of the RBGP scheme, including the number of distractors kept constant, the number of challenge screens, the number of distractors and the number of passimages in a user's passimage set. This means an estimate for the number attacks for a given configuration can be easily calculated.

- A security metric - this allows comparison of the security of RBGP schemes in a way which is repeatable, reproducible, quantitative, objective and extensible.

Some of these contributions have resulted in publications in peer-reviewed conference proceedings, namely [25], [24], and [26].

9.2 Achievement of Thesis Hypothesis Objectives

The thesis statement was as follows:

The security of a recognition-based graphical password scheme can be quantifiably measured in terms of resistance to observation and guessing attacks.

The thesis statement was further refined into five objectives as follows, each of which was addressed separately in this thesis.

Objective 1

Identify potential attacks (where the aim of the attacker is to impersonate a user and to achieve a false positive authentication) and examine current recognition-based schemes in terms of resistance to these attacks.

Objective 2

Identify a list of requirements from current security metric literature against which the metric will be assessed.

Objective 3

Establish measurements of the guessability (how easily a user's passimage set can be guessed) of a RBGP scheme by means of a mathematical model which estimates the attacks required before success for each identified guessing attack.

Objective 4

Establish measurements of the observability (how easily a user's passimage set can be observed) of a RBGP scheme by means of a mathematical model which estimates the attacks required before success for each identified observation attack.

Objective 5

Combine the measurements established into a comprehensive metric which meets the requirements identified by Objective 2.

9.2.1 Review

Objective 1 was addressed in Chapter 2 by examining relevant literature and condensing the information to form a threat model. The model extended the areas of concern identified by DeAngeli *et al.* (in [17]) to incorporate vulnerabilities and attacks related to these areas.

Objective 2 was addressed in Chapter 3 where literature regarding security metrics was reviewed and a number of key attributes were highlighted as important.

Objective 3 was addressed in Chapter 4 where the probability of randomly guessing the correct images was adjusted and a new guessing attack was constructed and analysed. As a result, a model based upon the random guessing value was reported.

Objective 4 was addressed in Chapters 6 and 7. Chapter 6 reported the construction of simulations which allowed the identification of dependent variables which affected the number of attacks before success for shoulder surfing and frequency attacks. This was extended in Chapter 7 which used the data generated from the simulations to fit a model using robust mathematical modelling methods.

Finally, objective 5 was addressed in Chapter 8 which combined the models established in previous chapters and evaluated the resulting metric against the criteria established by objective 2. Since these objectives have been met, it is concluded that this thesis has proved its research hypothesis.

9.3 Extension of Scope and Context

As detailed in Section 2.5, the scope and context of the metric was limited primarily to allow the work to be completed within the time allocated. It is recognised that this limits the direct applications of the work. This section discusses how the work could be adapted and the scope extended to provide a more encompassing metric. Specifically, the use of multiple passimages per challenge screen and order specific passimage sets is considered. This would allow the metric to be applied to all the schemes in Table 2.1 on page 36. Extending the context of authentication to include remote authentication (e.g. web-based authentication) is also discussed. This would allow an attacker to view communication between the client and server.

9.3.1 Incorporating Multiple Passimages per Challenge Screen

Each of the aspects of the metric are considered in turn:

Random Guessability

If there is the same number of passimages per challenge screen then the random guessability can be calculated as shown in Equation 9.1. In this equation x denotes the number of images in a challenge screen, p_s denotes the number of passimages per challenge screen and s denotes the number of challenge screens.

$$\binom{x}{p_s}^s \tag{9.1}$$

If there is a different number of passimages per challenge screen then the random guessability can be calculated as shown in Equation 9.2. In this equation p_i denotes the number of passimages for challenge screen i where $i = 1, .., s$. The only scheme found which uses this approach is the PassImages scheme by Charrau *et al.* [12].

$$\binom{x}{p_1} \times \binom{x}{p_2} \times ... \times \binom{x}{p_s} \tag{9.2}$$

SOGA

A SOGA could be adapted where multiple passimages are shown per screen by adjusting the algorithm to select all images in the most popular category. Subsequently the second most popular category and so forth until the number of

passimages on the screen is reached. The revised number of images per screen can then be calculated as previously shown in Chapter 4 and used in the appropriate random guessability equation (Equation 9.1 or 9.2) discussed above.

Frequency Attacks

A frequency attack can be adapted to multiple passimages per challenge screen by selecting the most frequently viewed image on the screen. Subsequently the next most frequently viewed and so forth until the number of passimages per screen has been reached. To model this, the simulations would need to be performed again to incorporate the number of passimages per screen as an independent variable. The same modelling approach (detailed in Chapter 7) could then be applied to obtain an equation for the number of attacks which incorporates multiple passimages per screen. This approach would assume the same number of passimages per challenge screen. If one considered the approach taken in the PassImages scheme by Charrau *et al.* [12], this would be potentially harder to model. The number of passimages per screen would need to be varied within a session and incorporated into the model.

Shoulder Surfing Attacks

The basic algorithm for shoulder surfing would not change if there were multiple passimages on a screen. However, the impact on the number of attacks required would need to be examined. In particular, the first hypothesis examined was that increasing the size of the passimage set increased the number of sessions before an attack was successful. This may not be the case if a larger number of passimages are shown on a challenge screen. The simulations would have to be performed again to incorporate multiple passimages per challenge screen. The results would then be modelled in a similar approach as described in Chapter 7.

9.3.2 Incorporating Order of Passimages

The schemes from the Table 2.1 on page 36 which use ordered passimage sets are Story [15], ImagePass [56], Moncur [58], Komanduri [50], VIP1 [17] and Passimages [12]. All but the Passimages scheme used only one challenge set. As for multiple passimages per challenge screen, each of the aspects of the metric are considered in turn:

Random Guessability

The random guessability for a passimage set where the order is important can be thought of as the number of permutations of size p from the challenge set. Where only one challenge set is used (as for 5 of the 6 schemes which required ordered selection) this can be calculated as shown in Equation 9.3. In this equation x denotes the number of images on the screen and p denotes the number of passimages to be selected.

$$\frac{x!}{(x-p)!} \tag{9.3}$$

Where multiple screens are used, the random guessability can be calculated as the number of permutations for each screen multiplied. This is because the probability is calculated as the probability of correctly guessing one screen, followed by the next and so forth. This is shown in Equation 9.4 where p_i denotes the number of passimages for challenge screen i where $i = 1, .., s$.

$$\frac{x!}{(x-p_1)!} \times \frac{x!}{(x-p_2)!} \times ... \times \frac{x!}{(x-p_s)!} \tag{9.4}$$

SOGA

The algorithm for a semantic ordered guessing attack could remain as for multiple passimages per challenge screen with a minor adaption. The simulations would need to be updated to incorporate order by calculating all the orders possible for the images in the top p categories, where p is the number of passimages on screen. The first order attempted would be the most to least likely categories, then permutations of the order. The revised x value could then be used in the relevant random guessing equation (Equation 9.3 if the same number of passimages are used on each screen or Equation 9.4 otherwise).

Frequency Attacks

Frequency attacks can be carried out in the same manner as described for multiple passimages per challenge screen with a minor adaption. If the number of passimages on the screen is equal to the total number of passimages in the user's passimage set then all passimages would appear the same number of times. If any distractors appear an equal number of times then the frequency attack should randomly select a subset of the correct size from these images. The simulations would need to be updated to incorporate order by selecting the most frequently observed image and so forth until the number of passimages on screen is reached. Permutations of this subset can then be attempted until success. After performing the simulations to incorporate the order, the data can be modelled as previously described in Chapter 7.

Shoulder Surfing Attacks

If it is assumed the attacker recalls the images and not the order, then the shoulder surfing simulation can be updated. Once the attacker recalls all the images, they must then attempt all permutations of order until success. This simulation could be run to gather the data and apply modelling as previously described in Chapter 7. Since Davis [15] provided evidence that users often remember the images, but not the order, this appears a reasonable assumption. If it is assumed the attacker can recall the order then the simulation and modelling can proceed as for multiple passimages per challenge screen.

9.3.3 Incorporating Web-based Authentication

Extending the metric to incorporate attacks which exploit remote authentication could be complex. If the attacker can view communications between the client and server and extrapolate the passimages from this communication, then a replay attack will be successful. For a phishing attack, assuming the attacker already has the username, then they can gather genuine authentication screens by attempting authentication. If the victim falls for the phishing attack then the attack will be successful. If this approach was taken the probability that a user will fall for a phishing attack would have to be modelled. This could be difficult and may not specific to RBGPs.

One possible approach would be to assume that if the attacker can perform a phishing attack then it will be successful. However this would depend on the same passimages being presented twice to the attacker. If the number of passimages is equal to the number of screens then the challenge sessions will have the same passimages. If the number of passimages exceeds the number of screens, then there is a probability that a different set of passimages will be presented to the attacker.

The number of subsets of passimages can be calculated as the number of combinations of s (the number of screens) passimages from the user's set of p passimages. This can be used to calculate the probability of getting two challenge sessions which use the same subset of passimages as shown in Equation 9.5

$$\frac{1}{\binom{p}{s}^2} \tag{9.5}$$

As a result, the value for phishing could be seen as 1 if the number of passimages in the user's passimage set is equal to the number of challenge screens. Otherwise it could be calculated as a maximum number of attempts before success as $\binom{p}{s}^2$. This could be less as the attacker could be presented with a subset of the collected images and correctly guess the remaining authentication images.

Taking a similar approach to a replay attack the assumption can be made that if a replay attack is possible, the attack will be successful in a maximum of $\binom{p}{s}^2$ attempts.

9.4 Metric Maintenance

This section aims to consider how the metric proposed in this work could be maintained and expanded in future. The metric could be extended to incorporate further attacks by first identifying potential threats. This process could be approached in a number of ways and is discussed in the following section.The subsequent section aims to present the remaining stages in identifying and adding a new model into the overall tuple metric. The final part of this section provides an overview of applying this process to a different authentication mechanism.

9.4.1 Threat Analysis

The threats established in this work were based on current research. This was achieved by identifying attacks already proposed in literature and the addition of a new type of attack, the SOGA. The SOGA was designed to reflect a dictionary attack on alphanumeric passwords. In the future, threats may also be identified by this process as new literature emerges on the topic. In addition NIST suggest the identification of vulnerabilities (which could be exploited by threats) based on the stage in the software in the software development life cycle the software is at. In summary, these were identified in [83] as follows:

- Not Yet Designed - At this stage the NIST recommend focusing on the planned security processes and procedures, requirements definitions and any existing security product analyses. For RBGPs the latter could include academic papers and this thesis.
- Implementation in Progress - At this stage, NIST recommend the analysis be more specific. For example, security design documents could be analysed.
- System is Operational - At this stage the NIST recommend analysis of the system security features and controls to identify vulnerabilities which could be exploited.

Another possible approach to identify vulnerabilities and threats could be to perform penetration testing on a RBGP system which has been deployed. This could help to assess the ability to withstand intentional attempts to by-pass the security mechanism [83]. Using these approaches to identify vulnerabilities the next stage would establish threats which could exploit the vulnerabilities by identifying potential attacks. Recall that the aim of the attack is to obtain the passimage set to perform successful authentication. Identification of attack types for a mechanism which is primarily academic at this stage could be difficult. This work has identified a number of possible attacks but others could potentially be identified by thought experiments or implementing the scheme and asking users to attack it. Once the attacks are established, models can be generated and added to the overall tuple metric. This process is discussed in the following section.

9.4.2 Extension of the Metric

The overall metric can be extended by addition of models for different attack types. The next step in this process would be to gather data to be used to generate a model for the attack. This was discussed briefly in Section 8.3.3 and will be discussed further here. The approach to establish a model varies slightly depending on whether the attack exploits user bias in image selection or not. Each approach is discussed separately as follows.

Modelling Attacks which Exploit User Bias

The first step in establishing a model for an identified attack is to gather data. If the attack aims to exploit a user bias in selection of passimages, this step will

involve gathering user selected passimages. This could be done in one of two ways: the users could be asked to provide their own images or the images could be gathered from different sources (e.g. `http://www.freeimages.co.uk/`) and combined into a collection from which users can select images as their passimage set. The end result will be a collection of user chosen passimage sets. At this point it is suggested that the data is split into two data sets, one to establish the suspected bias and the other to test the attack which exploits the bias. For example Davis *et al.* [15] split their data into 80% for establishing the bias and 20% to attempt the attack against. The attack against the data can then be performed by simulation.

To perform an attack simulation a RBGP scheme should be implemented. Details of implementing simulations are provided in Appendix D.1.5. The key responsibilities to be modelled in a RBGP simulation are generating challenge sessions, challenge screens and selecting passimage sets. Note that if the passimage set exceeds the number of challenge screens a subset of these images (equal to the number of challenge screens in a session) needs to be selected for a challenge session. If the passimage set size is equal to the number of challenge screens in a session then the whole set is used each time and a subset need not be selected.

Once the basics of a RBGP scheme have been implemented the next stage is to load the user selected passimage sets to be attacked and then implement an algorithm which simulates the attack. Recall that the attack should exploit the bias established by the remainder of the data set. The bias established from the portion of the user data will need to be incorporated into the algorithm. For details on the algorithms used for observability attacks, see Appendix D. The simulation should note the success rate in some way, be it writing out to the console or to a file. The final step is to incorporate the success rate into a model. This can be achieved by incorporating the success rate into the equation $(\frac{100}{percentageOfSuccess})^s$. Incorporating the percentage of success and the number of screens in a session provides an estimated number of attacks before success.

Attacks Independent of User Bias

If the attack is not based on user bias and can be modelled purely by simulation the following approach can be taken. As for the prior approach, a RBGP scheme and an algorithm to simulate the attack need to be implemented. Since the attack is not based on user bias, the simulation can randomly select a passimage set. The next step is to establish which variables of the RBGP configuration impact the number of attacks before success. This can be examined in an approach similar to that detailed for shoulder surfing and frequency attacks in Chapter 6.

The simulation should be run using multiple configurations to generate data. Data for a single configuration can then be compared to data for a different configuration where one independent variable has been altered. The aim is to establish if there is a significant difference between the two distributions, indicating that the variable altered has a significant impact on the number of attacks before success (the dependent variable). The significance can be established using statistics such as the Yuen Welch test or a t-test.

Once the variables which have a significant impact on the number of attacks

have been established, the next step is to generate the data to be used for modelling. This can be done by simulating attacks on a minimum of 100 passimage sets for a variety of RBGP configurations. The resulting number of attacks for a given configuration can then be averaged or the median taken (average should be used only if the data is normally distributed, otherwise median is appropriate). The values for all the configurations tested can then be combined into a data set which can be mathematically modelled using linear regression. The approach taken in this work to fit a model to the data follows that proposed by Maindonald and Braun [54, Page 190] which is as follows:

- Examine the distribution of the dependent variables and the independent variable.
- Examine the scatterplot matrix involving all the dependent variables, in particular look for evidence of non-linearity in the plots of these variables against each other and note any potential outliers.
- If there is evidence on non-linearity in some scatterplots, consider application of transformations to the data to produce more linear results which are easier to fit.
- If distributions are skewed, again consider transformations to establish a more symmetrical distribution.

Details of applying this approach for shoulder surfing and frequency attacks are provided in Chapter 7.

Incorporating the Model into the Metric

Once models have been established, either by the first or second approach, they can be incorporated into the overall metric by adding an extra column to the tuple. For example, suppose a known guessing model was established. The abbreviation for this could be KG and it could be added to the metric as follows: (RG,SOGA,SS,FREQ,KG). The order of the models within the tuple has no significance, but it could also be beneficial for usability to group together guessability attacks and observation attacks. In this example, this would result in the tuple (RG,SOGA,KG,SS,FREQ). The use of the metric could then proceed as presented in Chapter 8.

9.4.3 Adaption to Different Authentication Mechanisms

The approach detailed in this work could also be applied to different authentication mechanism. This section aims to provide a brief discussion as to how this may proceed. It is suggested that areas of concern identified by DeAngeli *et al.* of observability, guessability and recordability apply to all user authentication mechanisms [17]. Thus, the first stage would be to identify vulnerabilities and related threats and attacks as detailed in Section 9.4.1. Once the threats are identified the next stage would be to construct models relating to each attack type and combine them into a tuple, this would be a similar process to that described in Section 9.4.2.

9.5 Future Work

There is potential for future work in this area, this section aims to discuss some of these possible avenues. Whilst there are more directions which could be explored, this section discusses the three which are deemed the most important in more detail. This excludes the extension of the scope and incorporation of web authentication already discussed. After the important aspects, also presented is a list of other aspects.

9.5.1 Known User Guessing

The most important aspect which could benefit from further examination is the area of educated guessing, or known user guessing. This is where the attacker has knowledge of the user's interests and preferences which could potentially be used to successfully guess their passimage set with higher success than random guessing. This could be examined by means of a user study. In such a study a group of participants would be asked to select a passimage set from a collection of presented images. The participants would then be asked to nominate a least one friend to attack their graphical password. The nominated friend would then be presented with a challenge session for the participant's passimage set and asked to select the images they believe the participant has chosen. The success rate would be recorded. Each friend would then be asked to attack a randomly assigned stranger's passimage set and again the success rate would be recorded.

It would then be possible to examine if there was a statistically significant difference in the success rates of the stranger attacks and the known user attacks. The success rate could be incorporated into a guessability measure in a similar approach to the SOGA value. Post experiment questionnaires could determine why the users selected the images they did (i.e. if they reflect hobbies/interests of the participant/victim). Other variables which could be considered would be the content of the images (already examined briefly by Hayashi *et al.* [40] who examined the impact of obscuring the passimages), and different levels of knowledge of the user (e.g. colleague, acquaintance, friend, close friend).

9.5.2 SOGA Adjustments

There are a number of adjustments which would be beneficial if applied to the SOGA work. First, the evaluation of variables with respect to the success of SOGAs was limited. In particular the impact of the number of distractors on a challenge screen and the use of more passimages than challenge screens were not examined. The number of distractors could be increased and the simulations performed again to achieve percentages of success. Use of a larger number of potential passimages (the size of the set of images from which users could select their passimages) would require further user passimage selections to be collected. Subsets of the images could then be used in simulations to establish success rates for each value of potential passimage set sizes. Statistical analysis could then be applied to determine if there is any significant effect on the success rate when different potential passimage set sizes are used.

Secondly, due to the human element of user selection inherent in this attack, it would also be beneficial to repeat the experiment in its entirety with a different image set to confirm the results. The data should be split into two sets, one to establish the biases and one to test the attacks.

Finally, it could also be beneficial to perform the attacks against a complete passimage set, rather than one challenge screen or passimage. This could provide a more realistic estimate of the success rates rather than incorporating the number of screens by raising the adjusted number of images per screen to the power of the number of screens.

9.5.3 Incorporation of Potential Passimage Set Size and Distractor Selection

One variable which was not accounted for in the simulations and modelling of the shoulder surfing and frequency attacks was the potential passimages set size. This could be included in the simulations by generating a dummy image set of a given size and using this for the simulations. The effect could then be examined in a similar approach to those for the observability simulations reported in Chapter 6. The number of images could be varied while keeping the remaining variables constant and measuring the number of attacks required before success. The success rates could then be analysed to examine if the number of attacks required significantly increased or decreased when the potential passimage set size was increased.

Also, the distractor selection algorithm was not considered for frequency attacks and could potentially have an impact on the number of attacks required before success. If a distractor selection algorithm exhibits a preference to particular images (e.g. images from a specific category) then it is feasible that this could increase the number of attacks required if a given image is selected with higher frequency. This could be accounted for in a similar approach as for the observability simulations reported in Chapter 6 using the distractor selection algorithm as the independent variable of interest.

9.5.4 Other Potential Areas

For completeness, this section is concluded by a list of other possible avenues for further research:

- Authentication Context Recommendations- It could be beneficial to construct a framework of recommendations for selecting an appropriate RBGP mechanism. Renaud [69] has presented such a framework for web authentication, this could be adapted to RBGP schemes specifically to establish an appropriate configuration for a given context.

- Research could also be furthered by looking at the aspects of RBGP security which were deemed outside the scope of this work, i.e. recordability, resistability and analysability.

- There has been some research on the use of multiple graphical passwords, and that which has been reported (e.g. Everitt *et al.* [28] and Moncur and Leplatre [58]) has focused on the usability. It could be beneficial to examine the potential impact on the security. For example if a user supplies their own images, they could use the same images to authenticate for multiple services. This would have similar implications for security as re-use of passwords.

- Shoulder surfing could also benefit from further research. Currently, graphical password shoulder surfing papers often propose a new shoulder surfing countermeasure and examine the usability, but don't often examine the effectiveness of the countermeasure on shoulder surfing attacks success. This could be rectified by performing user studies asking participants to shoulder surf a user authenticating using different shoulder surfing countermeasures. The effect of image type could also be considered. For example, would it be easier to shoulder surf photographic images or faces?

- Interaction between attack types- This model reflects distinct attack algorithms, though it is possible an attacker may combine a shoulder surfing attack with a guessing or frequency attack. This could benefit from further examination, however modelling the impact of this could be difficult.

- Constant challenge screens impact on usability- The implementation of constant challenge screens stops frequency/intersection attacks. However, as evidenced by Deffenbacher *et al.* [19], there is an interference impact in recognition of target images when distractor images are also presented to the user. This could benefit from further examination, in particular an experiment specifically in the context of RBGP authentication should be carried out to confirm the results of Deffenbacher [19]. Also Deffenbacher *et al.* indicated varying levels of impact for different image types [19] , this could be incorporated into further research to determine an image type with the minimal interference which would allow optimal security against frequency attacks.

9.6 Discussion

As a result of this work a number of opinions with regards to lack of adoption of graphical passwords have been formed. These are discussed here. There are a number of issues which may be preventing adoption of graphical passwords. First, it is sometimes proposed that graphical passwords should replace the password as the authentication mechanism of choice. For example Suo *et al.* claim "graphical passwords have become a viable alternative to the traditional text-based passwords due to their superior ease of recall and potential large password space" [84]. It is my belief that we shouldn't be looking for a replacement, but a complement to passwords. In particular, due to the memorability of graphical passwords they may be more suitable to circumstances in which authentication is less frequent, as we often forget passwords which are used infrequently [3]. This

opinion is supported by Sasse *et al.* [76], and Wiedenbeck *et al.* [99]. Also the increase in time taken to authenticate using RBGPs may not be an issue for infrequently used systems, where it can be for frequently used systems (e.g. Brostoff and Sasse [9] where users took longer, and so authenticated less frequently).

Another issue with graphical passwords which could be influencing adoption is the amount of effort involved in implementation. There is much more to consider when employing a graphical authentication mechanism when compared to a password which ordinarily involves little more than a database entry. For RBGPs, one must consider (amongst other aspects) the configuration of the scheme, which images to use and how they will be gathered, the presentation of the images to the user, whether countermeasures for different forms of attack will be employed, whether the images will be assigned to the users or not, and how the distractor images will be selected. This takes more effort than required for a password.

Another issue which has become apparent from reviewing research, is that there is no standard approach to reporting RBGP schemes. Having a standardised way of reporting a RBGP scheme would allow easier more realistic comparison of configurations and potentially assist adoption. A good example of reporting a scheme is shown by Mihajlov *et al.* [56] where the image types, size of the passimage set and distractor set, how the information is sent from client to server and some of the details of how the database was structured are reported. Consistency allows a better more complete comparison of schemes and an easier application of the metric presented in this work.

The reason for lack of adoption could be due to a combination of the above issues, and could include others. Whatever method used, one must consider context when authenticating. For example, if the system is to be designed to be used in a space where no one would be able to view the session apart from the user wishing to authenticate, then perhaps shoulder surfing is not a concern. It is recommended that to select an appropriate scheme one should consider the context, and establish which threats will be an issue and what the requirements of the authentication mechanism are e.g. how memorable it needs to be, how often it is used, the media used to display the challenges. Another consideration with respect to context is that there are different passwords for different purposes as shown by Notoatmodjo *et al.* [60], where the authors demonstrate that users selected unique passwords for accounts which were perceived to be of "high importance". Implementing a highly secure mechanism for an account which is perceived by users as low priority may not be usable.

In spite of the potential issues highlighted, there exists a number of commercial authentication mechanisms using graphical elements. Notably, Android and Windows 8. Android pattern lock allows the user to lock their phone by drawing a pattern connecting dots on a grid, this is discussed by Shabtai *et al.* [78]. Windows 8 is reported to combine gestures and images for authentication [1]. It is due to this, in addition to the password problem and the lack of consistency in reporting the security of RBGP schemes that this work was identified. It has provided a consistent method of comparing the security of RBGP schemes which

[1] `http://blogs.msdn.com/b/b8/archive/2011/12/16/signing-in-with-a-picture-password.aspx`

could potentially be used if RBGP mechanisms are more widely adopted. The work is original because a metric which was objective and provided an estimate of the number of attacks required before success for multiple attacks had not been presented in research till this point.

To conclude I would like to add my own personal opinion on the future of graphical passwords, specifically RBGPs. I believe they will not become main stream. The main reason for this belief is the effort required to implement such a scheme. Implementing a password authentication mechanism is relatively straight forward and requires comparatively little resources. When employing RBGPs there is much more to consider in terms of configuring the mechanism. It would be much more effort to implement and also to maintain. The potential passimages would need to be gathered and stored. When compared to the storage space requirement for an alphanumeric password, the requirement of storing images could be considerably more. Another reason RBGPs may not become main stream is that users can be unaware of the impact of their coping mechanisms on the security of their passwords. This is not to suggest that passwords or people are "broken", but merely that the way people behave with passwords does not provide an optimal solution for either security or usability. If one were to consider the use of RBGPs in a system, I suggest that the appropriate context would be a system for which the user does not often authenticate due to the potential memorability benefits of RBGPs. I also believe RBGPs are best suited to a system for which a high security level is not required or at least that observability is not an issue. This is because shoulder surfing provided the lowest number of attacks in this work.

Bibliography

[1] PassFaces. http://www.realuser.com/ Last accessed 20.8.2012.

[2] Anne Adams and Martina Angela Sasse. Users Are Not The Enemy. *Communications of the ACM*, 42(12):41–46, 1999.

[3] Anne Adams, Martina Angela Sasse, and Peter Lunt. Making passwords secure and usable. In *People and Computers XII Proceedings of HCI'97*, pages 1–20, 1997.

[4] Adam Beautement. Gathering realistic authentication performance data through field trials. In *Usable Security Experiment Reports (USER) Workshop, Symposium On Usable Privacy and Security*, 2010.

[5] Robert Biddle, Sonia Chiasson, and Paul C. van Oorschot. Graphical Passwords: Learning from The First Generation. In *Technical Report TR-09-09, School of Computer Science, Carleton University*, pages 1–20. Carleton University, 2009.

[6] Greg Blonder. Graphical password. *US Patent 5,559,961*, 1996.

[7] Rainer Böhme and Thomas Nowey. Economic Security Metrics. In *Dependability Metrics*, chapter 15, pages 176–187. Springer Berlin / Heidelberg, 2008.

[8] John Brainard, Ari Juels, Ronald L. Rivest, and Michael Szydlo. Fourth-factor authentication: somebody you know. In *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS)*, pages 168–178, 2006.

[9] Sacha Brostoff and Martina Angela Sasse. Are Passfaces More Usable Than Passwords: A Field Trial Investigation. In *People and Computers XIV-Usability or Else: Proceedings of HCI*, pages 405–424, 2000.

[10] Alan S. Brown, Elisabeth Bracken, Sandy Zoccoli, and King Douglas. Generating and remembering passwords. *Applied Cognitive Psychology*, 18(6):641–651, September 2004.

[11] Andreas Bulling, Florian Alt, and Albrecht Schmidt. Increasing the Security of Gaze-Based Cued-Recall Graphical Passwords Using Saliency Masks. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, Austin, Texas, 2012. ACM.

[12] D. Charrau, Steven Furnell, and Paul Dowland. PassImages: An alternative method of user authentication. In *Proceedings of 4th Annual ISOneWorld Conference and Convention, Las Vegas, USA*, 2005.

[13] Sonia Chiasson, Elizabeth Stobert, and Alain Forget. Persuasive cued click-points: Design, implementation, and evaluation of a knowledge-based authentication mechanism. *IEEE Transactions on Dependable and Secure Computing*, 9(2):222–235, 2011.

[14] Sonia Chiasson, Paul C. van Oorschot, and Robert Biddle. Graphical Password Authentication Using Cued Click Points. In *Lecture Notes in Computer Science*, volume 4734 of *Lecture Notes in Computer Science*, pages 359–374. Springer Berlin Heidelberg, 2007.

[15] Darren Davis, Fabian Monrose, and Michael K Reiter. On User Choice in Graphical Password Schemes. In *Proceedings of the 13th conference on USENIX Security Symposium-Volume 13*, page 11. USENIX Association, 2004.

[16] Antonella De Angeli, Mike Coutts, Lynne Coventry, Graham Johnson, David Cameron, and Martin H. Fischer. VIP: a visual approach to user authentication. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, pages 316–323. ACM, 2002.

[17] Antonella De Angeli, Lynne Coventry, Graham Johnson, and Karen Renaud. Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems. *International Journal of Human-Computer Studies*, 63(1-2):128–152, 2005.

[18] Richard D. De Veaux, Paul F. Velleman, and David E. Bock. *Intro Stats*. Pearson Addison Wesley, second edition, 2006.

[19] Kenneth A Deffenbacher, Thomas H Carr, and John R Leu. Memory for Words, Pictures, and Faces: Retroactive Interference, Forgetting, and Reminiscence. *Journal of Experimental Psychology: Human Learning and Memory*, 7(4):299–305, 1981.

[20] Rachna Dhamija and Adrian Perrig. Deja vu: A User Study Using Images for Authentication. In *Proceedings of the 9th conference on USENIX Security Symposium-Volume 9*, pages 45–48. USENIX Association, 2000.

[21] Paul Dunphy, Andreas P. Heiner, and N. Asokan. A closer look at recognition-based graphical passwords on mobile devices. In *Proceedings of the Sixth Symposium on Usable Privacy and Security*, pages 1–12. ACM, 2010.

[22] Paul Dunphy, James Nicholson, and Patrick Olivier. Securing Passfaces for Description. *Proceedings of the 4th symposium on Usable privacy and security - SOUPS '08*, page 24, 2008.

[23] Paul Dunphy and Jeff Yan. Do Background Images Improve "Draw A Secret" Graphical Passwords? *Proceedings of the 14th ACM conference on Computer and communications security - CCS '07*, page 36, 2007.

[24] Rosanne English and Ron Poet. Measuring the Revised Guessability of Graphical Passwords. In *Network and System Security (NSS), 2011*, 2011.

[25] Rosanne English and Ron Poet. Towards a metric for recognition-based graphical password security. In *Network and System Security (NSS), 2011*, pages 239–243, 2011.

[26] Rosanne English and Ron Poet. The Effectiveness of Intersection Attack Countermeasures for Graphical Passwords. In *TrustCom 2012, 11th International Conference on Trust, Security and Privacy in Computing and Communications*, page To appear, 2012.

[27] David M Erceg-Hurn and Vikki M Mirosevich. Modern robust statistical methods: an easy way to maximize the accuracy and power of your research. *The American psychologist*, 63(7):591–601, October 2008.

[28] Katherine M. Everitt, Tanya Bragin, James Fogarty, and Tadayoshi Kohno. A comprehensive study of frequency, interference, and training of multiple graphical passwords. *Proceedings of the 27th international conference on Human factors in computing systems - CHI '09*, page 889, 2009.

[29] B Fischhoff, P Slovic, S Lichtenstein, S Read, and B Combs. How safe is safe enough? A psychometric study of attitudes towards technological risks and benefits. *Policy sciences*, 1978.

[30] Dinei Florencio and Cormac Herley. A large-scale study of web password habits. *Proceedings of the 16th international conference on World Wide Web - WWW '07*, page 657, 2007.

[31] John Fox. *An R and S-Plus Companion to Applied Regression*. Sage Publications, 2002.

[32] John Fox. Appendix - Robust Regression. In *An R and S Plus Companion to Applied Regression*. 2002.

[33] Haichang Gao, Xuewu Guo, Xiaoping Chen, Liming Wang, and Xiyang Liu. YAGP: Yet Another Graphical Password Strategy. *2008 Annual Computer Security Applications Conference (ACSAC)*, pages 121–129, December 2008.

[34] Haichang Gao, Zhongjie Ren, Xiuling Chang, Xiyang Liu, and Uwe Aickelin. A New Graphical Password Scheme Resistant to Shoulder-Surfing. *2010 International Conference on Cyberworlds*, pages 194–199, October 2010.

[35] Shirley Gaw and Edward W Felten. Password Management Strategies for Online Accounts. In *Second Symposium on Usable Privacy and Security (SOUPS)*, volume pp, pages 44–55, 2006.

[36] Romain Giot, Mohamad El-Abed, and Christophe Rosenberger. GREYC keystroke: A benchmark for keystroke dynamics biometric systems. In *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, pages 1–6. Ieee, September 2009.

[37] Madoka Hasegawa, Yuichi Tanaka, and Shigeo Kato. A study on an image synthesis method for graphical passwords. In *2009 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pages 643–646. Ieee, December 2009.

[38] H. Hasle, Yngve Kristiansen, Ketil Kintel, and Einar Snekkenes. Measuring resistance to social engineering. In *Information Security Practice and Experience*, pages 132–143, Singapore, 2005. Springer.

[39] Eiji Hayashi, Nicolas Christin, and Adrian Perrig. Use Your Illusion : Secure Authentication Usable Anywhere. In *Proceedings of the 4th symposium on Usable privacy and security (SOUPS '08)*, volume 337, pages 35–45, 2008.

[40] Eiji Hayashi, Jason Hong, and N. Christin. Security through a different kind of obscurity: evaluating distortion in graphical authentication schemes. In *Proceedings of the 2011 annual conference on Human factors in computing systems (CHI '11)*, pages 2055–2064, 2011.

[41] Ronda Henning and Et Al. Workshop on Information Security System Scoring and Ranking. In *Proc. of Workshop on Information Security System, Scoring and Ranking Information System Security Attribute Quantification or Ordering*. ACSA and MITRE, 2002.

[42] Cormac Herley and Paul C. van Oorschot. Passwords: If We're So Smart, Why Are We Still Using Them? In *in Proc. Financial Cryptography 2009*, pages 230–237. Panel Discussion from Financial Cryptography and Data Security 2009, Springer, 2009.

[43] Max Hlywa, Robert Biddle, and Andrew S. Patrick. Facing the facts about image type in recognition-based graphical passwords. In *Proceedings of the 27th Annual Computer Security Applications Conference*, volume 36, pages 149–158. ACM, 2011.

[44] Bogdan Hoanca and Kenrick Mock. Secure graphical password system for high traffic public areas. In *Proceedings of the 2006 symposium on Eye tracking research and applications - ETRA '06*, volume 1, page 35, New York, New York, USA, 2006. ACM Press.

[45] David C. Howell. *Statistical Methods for Psycholgy*. Wadsworth, Cengage Learning, 7th edition, 2010.

[46] Philip Inglesant and Martina Angela Sasse. The true cost of unusable password policies: password use in the wild. In *Proceedings of the 28th international conference on Human factors in computing systems*, pages 383–392, 2010.

[47] Wayne Jansen. Directions in security metrics research. Technical report, NIST Interagency/Internal Report (NISTIR) Number 7564, 2010.

[48] Ian Jermyn, Alain Mayer, Fabian Monrose, Michael K Reiter, and Aviel D. Rubin. The Design and Analysis of Graphical Passwords. In *Proceedings of the 8th conference on USENIX Security Symposium-Volume 8*. USENIX Association, 1999.

[49] Daniel V. Klein. "Foiling the cracker": A Survey of, and Improvements to, Password Security. In *Proceedings of the 2nd USENIX Security Workshop*, pages 5–14, 1990.

[50] Saranga Komanduri and Dugald R. Hutchings. Order and entropy in Picture Passwords. In *Proceedings of graphics interface 2008*, pages 115–122. Canadian Information Processing Society, 2008.

[51] Saranga Komanduri, Richard Shay, P.G. Kelley, M.L. Mazurek, Lujo Bauer, Nicolas Christin, L.F. Cranor, and Serge Egelman. Of Passwords and People: Measuring the Effect of Password-Composition Policies. In *CHI '11: Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, Vancouver, Canada, 2011.

[52] Averill M. Law and W. David Kelton. *Simulation Modelling and Analysis*. McGraw Hill, third edition, 2000.

[53] D. LeBlanc, Alain Forget, and Robert Biddle. Guessing click-based graphical passwords by eye tracking. In *Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on*, pages 197–204. IEEE, 2010.

[54] John Maindonald and W. John Bruan. *Data Analysis and Graphics Using R*. Cambridge Series in Statistical and Probabilistic Mathematics, third edition, 2010.

[55] Martin Mihajlov, Borka Jerman Blazic, and Saso Josimovski. Quantifying Usability and Security in Authentication. *2011 IEEE 35th Annual Computer Software and Applications Conference*, pages 626–629, July 2011.

[56] Martin Mihajlov, Borka Jerman-Blazic, and Marko Ilievski. ImagePass-Designing graphical authentication for security. In *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on*, pages 262–267. IEEE, 2011.

[57] Martin Mihajlov, Saso Josimovski, and B. Jerman-Blazic. A conceptual framework for evaluating usable security in authentication mechanisms-usability perspectives. In *Network and System Security (NSS), 2011 5th International Conference on*, pages 332–336. IEEE, 2011.

[58] Wendy Moncur and Gregory Leplâtre. Pictures at the ATM: exploring the usability of multiple graphical passwords. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 887–894. ACM, 2007.

[59] DW North. A tutorial introduction to decision theory. *... Science and Cybernetics, IEEE Transactions on*, (3), 1968.

[60] Gilbert Notoatmodjo and Clark Thomborson. Passwords and perceptions. In *Proceedings of the Seventh Australasian Conference on Information Security*, volume 98, pages 71–78, 2009.

[61] T. Pering, M. Sundar, J. Light, and R. Want. Photographic authentication through untrusted terminals. *IEEE Pervasive Computing*, 2(1):30–36, January 2003.

[62] Ron Poet and Karen Renaud. A Mechanism for Filtering Distractors for Graphical Passwords. In *13th Conference of the International Graphonomics Society*, page 14, 2007.

[63] Ron Poet and Karen Renaud. A Mechanism For Filtering Distractors for Doodle Passwords. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(5):1005–1029, 2009.

[64] Ron Poet and Karen Renaud. An Algorithm for Automatically Choosing Distractors for Recognition Based Authentication using Minimal Image Types. *The Ergonomics Open Journal*, 2(3):178–184, January 2010.

[65] Bruce Potter and Gary McGraw. Software Security Testing. *IEEE Security & Privacy Magazine*, 2(5):81–85, September 2004.

[66] Real User. The Science Behind Passfaces White Paper. Available at: http://www.realuser.com/enterprise/resources/white_papers.htm Last Accessed 20.08.2012.

[67] M. Rejman-Greene. Biometrics real identities for a virtual world. *BT Technology Journal*, 19(3):115–121, 2001.

[68] Karen Renaud. Quantifying the Quality of Web Authentication Mechanisms A Usability Perspective. *Journal of Web Engineering*, 3(2):95–123, 2004.

[69] Karen Renaud. A process for supporting risk-aware web authentication mechanism choice. *Reliability Engineering & System Safety*, 92(9):1204–1217, September 2007.

[70] Karen Renaud. Web Authentication Using Mikon Images. *2009 World Congress on Privacy, Security, Trust and the Management of e-Business*, pages 79–88, August 2009.

[71] Deborah Jean Rumsey. *Statistics for dummies*. Wiley, first edition, 2003.

[72] Deborah Jean Rumsey. *Probability for Dummies*. John Wiley and Sons, 2006.

[73] J Wyszynski S Carlton, J Taylor. Alternate authentication mechanisms. In *Proceedings of the 11th National Computer Security Conference*, 1988.

[74] Amirali Salehi-Abari, Julie Thorpe, and Paul C. van Oorschot. On Purely Automated Attacks and Click-Based Graphical Passwords. *2008 Annual Computer Security Applications Conference (ACSAC)*, pages 111–120, December 2008.

[75] Neil J. Salkind. *100 Questions (and Answers) About Research Methods*. Sage Publications, 2012.

[76] Martina Angela Sasse, Sacha Brostoff, and Dirk Weirich. Transforming the weakest linka human/computer interaction approach to usable and effective security. *BT technology journal*, 19(3):122–131, 2001.

[77] Reijo M. Savola. Towards a taxonomy for information security metrics. *Proceedings of the 2007 ACM workshop on Quality of protection - QoP '07*, page 28, 2007.

[78] Asaf Shabtai, Yuval Fledel, Uri Kanonov, Yuval Elovici, Shlomi Dolev, and Chanan Glezer. Google android: A comprehensive security assessment. *IEEE Security & Privacy*, (April):35–44, 2010.

[79] Claude Shannon. A Mathematical Theory of Communication. *The Bell Systems Technical Journal*, 27:379–423, 1948.

[80] Richard E. Smith. *Authentication From Passwords to Public Keys*. Addison-Wesley, 2002.

[81] M Sreelatha, M Shashi, and M Roop Teja. Intrusion Prevention by Image Based Authentication Techniques. In *IEEE-International Conference on Recent Trends in Information Technology*, pages 1239–1244, 2011.

[82] Lionel Standing. Learning 10000 pictures. *The Quarterly Journal of Experimental Psychology*, 25(2):207–222, May 1973.

[83] Gary Stoneburner, Alice Goguen, and Alexis Feringa. Risk Management Guide for Information Technology Systems. *NIST Special Publication*, 800-30(SP 800-30), 2002.

[84] Xiaoyuan Suo, Ying Zhu, and G Scott Owen. Analysis and Design of Graphical Password Techniques. *Advances in Visual Computing*, pages 741–749, 2006.

[85] Tetsuji Takada, T. Onuki, and H. Koike. Awase-e: Recognition-based image authentication scheme using users' personal photographs. In *Innovations in Information Technology, 2006*, pages 1–5. IEEE, 2006.

[86] Hai Tao and Carlisle Adams. Pass-Go: A proposal to improve the usability of graphical passwords. *International Journal of Network Security*, 7(2):273–292, 2008.

[87] Furkan Tari, A. Ant Ozok, and Stephen H. Holden. A Comparison of Perceived and Real Shoulder-surfing Risks between Alphanumeric and Graphical Passwords. In *Proceedings of the second symposium on Usable privacy and security - SOUPS '06*, page 56, 2006.

[88] Julie Thorpe and Paul C. van Oorschot. Graphical Dictionaries and the Memorable Space of Graphical Passwords. In *Proceedings of the 13th USENIX Security Symposium*, 2004.

[89] Julie Thorpe and Paul C. van Oorschot. Towards Secure Design Choices for Implementing Graphical Passwords. In *Proceedings of the 20th Annual Computer Security Applications Conference*, pages 50–60. IEEE, 2004.

[90] Julie Thorpe and Paul C. van Oorschot. Human-seeded attacks and exploiting hot-spots in graphical passwords. *Proceedings of 16th USENIX Security*, pages 103–118, 2007.

[91] Thomas S. Tullis and Donna P. Tedesco. Using personal photos as pictorial passwords. *CHI '05 extended abstracts on Human factors in computing systems - CHI '05*, page 1841, 2005.

[92] Thomas S. Tullis and DP Tedesco. Can users remember their pictorial passwords six years later. In *Proceedings of CHI 2011*, pages 1789–1794, 2011.

[93] Tim Valentine. Memory for Passfaces After a Long Delay, Technical Report. Technical report, London Goldsmith College, 1999.

[94] Paul C. van Oorschot and Julie Thorpe. On the Security of Graphical Password Schemes. In *Technical Report TR-05-11. Integration and extension of USENIX Security 2004 and ACSAC 2004 papers*. Carleton University, 2005.

[95] Paul C. van Oorschot, Julie Thorpe, and Amirali Salehi-Abari. Purely Automated Attacks on PassPoints-Style Graphical Passwords. *IEEE Transactions on Information Forensics and Security*, 5(3):393–405, 2010.

[96] Paul C. van Oorschot and Tao Wan. TwoStep: An Authentication Method Combining Text and Graphical Passwords. *E-Technologies: Innovation in an Open World*, (3):233–239, 2009.

[97] R.B. Vaughn Jr, Ronda Henning, and A. Siraj. Information Assurance Measures and Metrics - State of Practice and Proposed Taxonomy. In *Proceedings of the 36th Hawaii International Conference on System Sciences*. Published by the IEEE Computer Society, 2003.

[98] Andy Ju An Wang. Information security models and metrics. In *Proceedings of the 43rd annual Southeast regional conference*, volume 2, pages 178–184. ACM Press, 2005.

[99] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. Authentication Using Graphical Passwords: Effects of Tolerance and Image Choice. In *in Symposium on Usable Privacy and Security (SOUPS '05)*, 2005.

[100] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. PassPoints: Design and longitudinal evaluation of a graphical password system. *International Journal of Human-Computer Studies*, 63(1-2):102–127, July 2005.

[101] Susan Wiedenbeck, Jim Waters, Leonardo Sobrado, and Jean-Camille Birget. Design and Evaluation of a Shoulder-surfing Resistant Graphical Password Scheme. *Proceedings of the working conference on Advanced visual interfaces - AVI '06*, page 177, 2006.

[102] Rand R Wilcox. *Introduction to Robust Estimation and Hypothesis Testing.* 1997.

[103] Rand R Wilcox. *Basic Statistics: Understanding Conventional Methods and Modern Insights.* Oxford University Press, 2009.

[104] Rand R Wilcox. *Fundamentals of Modern Statistical Methods Substantially Improving Power and Accuracy.* Springer New York, 2010.

[105] Jeff Yan, Alan Blackwell, and Ross Anderson. Password memorability and security: Empirical results. *IEEE Security & Privacy Magazine*, 2(5):25–31, September 2004.

[106] NH Zakaria, David Griffiths, and Sacha Brostoff. Shoulder surfing defence for recall-based graphical passwords. In *Proceedings of the Seventh Symposium on Usable Privacy and Security*, pages 1–12, 2011.

Appendix A

Passimages Examples

Images were obtained from `http://www.freedigitalphotos.net/` and `http://www.freeimages.co.uk/`. An example for cartoon characters is not shown due to copyright, however this category contained images such as Mickey Mouse, Wylie Coyote and Homer Simpson.

Figure A.1: Food Category Passimage Example

Figure A.2: Transport Category Passimage Example

Figure A.3: Sport Category Passimage Example

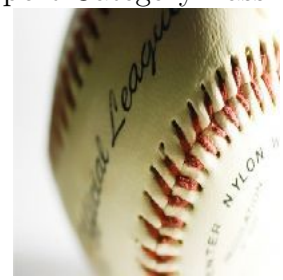


Figure A.4: Trees, plants and flowers Category Passimage Example

Figure A.5: Faces and body parts Category Passimage Example

Figure A.6: Buildings Category Passimage Example

Figure A.7: Clothing Category Passimage Example

Figure A.8: Scenery Category Passimage Example

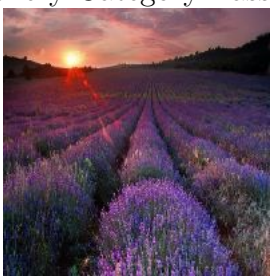


Figure A.9: Animals Category Passimage Example

Figure A.10: People Category Passimage Example

Figure A.11: Skyscape Category Passimage Example

Appendix B

Research Methods

B.1 Research Methods - Data Gathering

Data for this research was gathered in two ways; user studies and simulations. The design of user studies are reported where appropriate throughout the thesis. For the simulations, Law and Kelton [52] established a number of steps required to establish a comprehensive simulation study. These steps were applied to this research and are summarised as follows:

1. Formulate the problem and plan the study.
2. Construct a computer program and verify that it is correct - this was completed by JUnit testing.
3. Perform pilot runs.
4. Check the programmed model is valid - this was achieved by examination of the data resulting from pilot runs to ensure the program was running as expected. This was also confirmed with JUnit testing.
5. Design the experiments.
6. Run the simulations.
7. Analyse the output data.
8. Document, present and use the results.

The simulation process is documented in the appropriate sections throughout the thesis.

B.2 Research Methods - Data Analysis

B.2.1 Probabilities

At a number of points in the thesis basic probabilities are applied. This section briefly summarises the relevant probability rules (as detailed in [72, Chapter 2]).

- Given a sample space of possible outcomes of size n if each outcome is equally likely, the probability of event A occurring $P(A) = \frac{1}{n}$.
- Given the same sample space as above, the probability of event A *not* occurring is $1 - P(A)$.
- The probability multiplication rule states that the probability of two events occurring is the product of the probability of those two events occurring independently, i.e. $P(A \cap B) = P(A)P(B)$.

B.2.2 Combinations

Simple combinatorics were also used in this thesis. Given a set of size n, the number of ways of choosing a subset of size k is as shown in Equation B.1. Combinations were used instead of permutations as order was unimportant.

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{B.1}$$

B.2.3 Statistics

A number of statistical approaches were used to analyse data in this research. The selection of the appropriate statistic was obtained by reference to [45, Page 10] and other statistical references noted throughout the thesis. There were two types of data in this thesis, categorical and numerical. Categorical data is data which has at least one variable which has a fixed number of possible values, e.g. person's month of birth is one of twelve possibilities. Numerical data is data which has a range of numerical values, e.g. a person's height. Each data type required different statistical approaches. The approaches taken are summarised here.

Categorical Data Analysis

- Confidence Intervals for Proportions: to establish confidence intervals for proportions, the approach taken was as detailed in [71, Page 207]. The confidence interval formula for a proportion is $\hat{p} \pm Z\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$ where $\hat{p}$ is the sample proportion, n is the sample size and Z is the appropriate value for the desired confidence level (1.96 for a confidence level of 95% as used in this thesis). The sample proportion can be calculated as the number of people in the sample having the desired characteristic divided by the sample size. To use this calculation the following conditions must hold true: $n\hat{p} \geq 10$ and $n(1-\hat{p}) \geq 10$. This condition is called the "Success/Failure condition" and is used to ensure that the distribution used can be approximated by the normal distribution (thus allowing the calculations of confidence intervals as previously described). Further details of the calculation of the value of 10 can be seen in [18, Page 386].

- Chi-square Test for goodness of fit: This test examines the distribution of categorical counted data and compares it with a "null model", e.g. where the counts are equally distributed between each category to establish if they are significantly different. The chi-square test statistic is calculated as shown in Equation B.2 where O denotes the observed values and E denotes the expected values. The number of degrees of freedom is calculated as one less than the number of categories. The degrees of freedom value is then used to look up the critical value for the test statistic from a table of chi-square values. In this research, the chi-square table presented in [45, Page 697] was used. If the calculated chi-square value exceeded the critical value obtained from the table, then the observed distribution was significantly different from the expected (null) distribution.

$$\chi^2 = \Sigma \frac{(O-E)^2}{E} \tag{B.2}$$

  A significance value of p=0.05 is used throughout the thesis, this indicates with at least a 95% probability that the results are different from that expected by chance thus the results are significant.

- Chi-square Contingency Table: When there are two or more categorical variables under consideration, a chi-square contingency table can be applied to establish if one variable is contingent on another. A contingency table shows the distribution of one variable at each level of the other variable [45, Page 145]. The expected frequencies in the table represent the values expected if the two variables are independent and are calculated as follows. If E_{ij} is the expected frequency of the cell in row i, column j and row i has a total R_i, column j has a total of C_j and N is the total number of observations, then the expected frequency E_{ij} can be calculated as: $E_{ij} = \frac{R_i C_j}{N}$. From this point, the chi-square values are calculated as before, but summed over all cells in the contingency table. The chi-square test can then be applied as detailed above. Again a significance value of p=0.05 is used to provide a 95% confidence level.

Numerical Data Analysis

- Median - the median of data a measure of the half-way point, the value which separates the higher half of the data (ordered numerically) from the lower half of the data. It is an alternative measure of location, for normally distributed data the location measure often used is the arithmetic mean.

- Quartiles - Quartiles divide the top half of the data and the bottom of the data into halves again to obtain four quartiles. A quarter of the data is in the lower quartile, a quarter is in the upper quartile and the middle provides the remaining half of the data. The interquartile range provides a measure of the spread of the middle half of the data. This is calculated as upper quartile minus the lower quartile.

Figure B.1: Example Boxplot

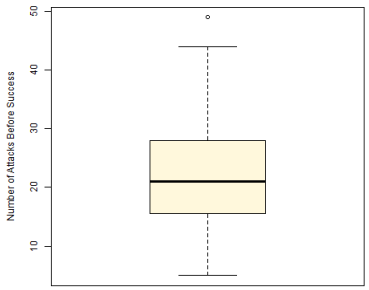
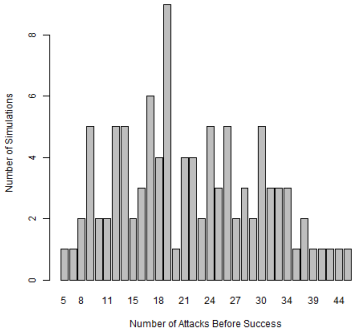


Figure B.2: Example Histogram

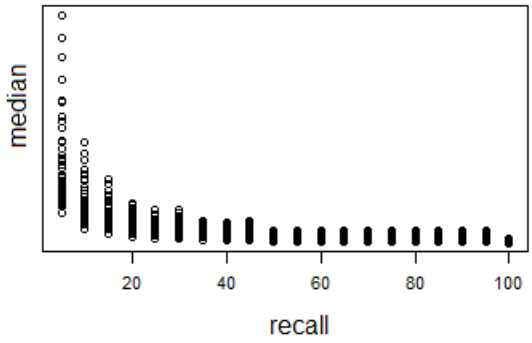


- Boxplots - A boxplot provides a graphical description of numerical data by using the maximum, upper quartile, median, lower quartile and minimum values. It allows groups of data to be easily compared [18, Page 77]. An example boxplot is shown in Figure B.1, where the top of the box represents the upper quartile, the lower edge of the box represents the lower quartile value, the horizontal line through the box represents the median value. The line above the top of the box represents the largest value from the upper quartile to 1.5 times the interquartile range. The line under the bottom of the box represents the smallest value from quartile 1 less 1.5 times the interquartile range. Any data values outside these "whiskers" are represented by circles.

- Histograms- a histogram plots counts of data which fall into "bins" of different values. The count of the data in each bin is represented by a rectangle of a height representing the count [18, Page 47]. An example histogram is shown in Figure B.2.

- Scatterplots - scatterplots plot two quantitative variables against each other and are useful for examining patterns, trends and relationships [18, Page 140]. An example scatterplot is shown in Figure B.3. A pattern which runs from the upper left to the lower right on a scatterplot is said to be negative, a pattern running in the other direction is positive. If the pattern runs in a (generally) straight form, the relationship between the two variables is said to be linear. The tighter the dots are clustered, the stronger the relationship between the two variables is said to be. In the example shown

Figure B.3: Example Scatterplot



in Figure B.3, the relationship between the two variables can be said to be strong, generally negative and appears to show an inverse relationship between the variables.

- Yuen-Welch test- The Yuen-Welch test for equality of trimmed means compares two groups of data (normally a control group and a treatment group [102, Chapter 5]) and is used to establish if the distributions are the same. If the distributions of the two groups are the same, there's no significant difference between them and the treated group is therefore not significantly different to the control group. If the data was normal a similar test would be a t-test, however this is a robust alternative suitable for use with non-normal data.

- Linear and robust linear modelling - A linear model is an equation of a straight line through data on a scatterplot [18, Page 167]. Where more than one independent variable affects the dependent variable, multiple regression, which models the impact of multiple variables on the independent variable compared to linear regression which models only one dependent variable, is required. This models the impact of all the independent variables on the dependent variable. Linear modelling applied in this work used the least squares approach (implemented by the statistical program and language "R") which essentially calculates the line for which the sum of the squared residual values (the observed minus the calculated values) is smallest and reports this as the best line of fit. However, there are underlying assumptions of this type of modelling, and if these are violated it is best to use a robust equivalent. In this thesis, the robust equivalent provided in R was used.

Appendix C

Attack Questionnaire

- How well do you know the person you have just attacked?
  - Close friend/relative
  - Friend
  - Acquaintance
  - Stranger
- How did you collect the username?
  - By observing multiple logins
  - By capturing their record of the username
  - By guessing the username through knowledge of the user
  - By guessing the username using variations of their name or similar
  - By using a username shown on the forum posts
  - Other - please provide details
- How did you collect the passimages?
  - By observing multiple logins and noting the images selected
  - By observing multiple logins and noting the images common between sessions
  - By capturing their record of the images selected
  - By guessing the pictures through knowledge of the users likes/dislikes
  - By guessing the images based on assumptions of what people in general might select
  - By randomly guessing/repeated attempts
  - Other - please provide details
- Any other comments?

Appendix D

Simulation Design

D.1 Requirements Gathering

D.1.1 Purpose

The purpose of the simulation software is to represent a RBGP scheme with a given configuration, construct a user's passimage set and allow frequency and shoulder surfing attacks to be emulated against that set. The RBGP scheme can have a varied number of distractor images d per challenge screen, a number of constant distractor images c, a number of challenge screens in a session s. A user of the scheme can have a number of passimages p. The combinations of these attributes represent the configuration of a RBGP scheme. If a frequency attack is being simulated dummy screens can also be used. This means upon incorrect selection of a distractor in an attempted attack the remaining challenge screens are set to contain only distractors, "dummy screens". If a shoulder surfing attack is being simulated an attacker has a percentage of recall, which reflects their ability to recall the passimages observed.

D.1.2 Scope

The system does not need to deal with user input. Configurations for a specific run of the simulation can be hard coded into the program. This is because the system was intended to generate data which will then later be used to fit a model and so only one user was considered. The system needs to be able to write out the results of the attacks for data analysis.

D.1.3 Assumptions

A number of assumptions were made in the design of the simulation software, these are summarised briefly here:

- The attacks will be made against complete sets of passimages to reflect the probability of successfully attacking a target user.
- For frequency attacks, use of constant distractors and dummy screens in a configuration were mutually exclusive.

- Each attack for a given configuration is run multiple times. The number of times run can be altered and the values for each attack or the median of all the runs for that configuration can be written to the results file.
- As with the rest of the thesis, one passimage per challenge screen is assumed.
- The frequency attack in particular assumes that if more than one image has the highest observed frequency then a random image from the set of images which have the same frequency is selected in the attack.
- The system created a set of 144 potential passimages and this value could not be changed.
- Distractors are selected randomly from the set of all potential passimages, less any constant distractors for the session and all the passimages in the user's passimage set.
- The number of constant distractors is always less than the number of distractors per challenge screen otherwise the attack would be reduced to a random guessing attack.
- If an attack works first time, then it is not based on a collection of frequencies and instead is due to randomly guessing the correct images. For this reason, the first attack is assumed never to be successful in an attempt to minimise the impact of randomly guessing the correct images.

D.1.4 Algorithms

Frequency Attack

In a frequency attack the attacker attacks a session by noting the images viewed and incrementing a count of the times it has been seen. To attack a screen, the attacker selects the image on the screen which he has seen most frequently. If the image selected is the passimage then the screen is passed. This is repeated for each screen in the challenge session. If all screens are passed, then the attack is successful. If not then the process is repeated until the attack is successful.

In the frequency attack simulations there was the possibility of two counter measures in addition to using a passimage set larger than the number of challenge screens in a session. The first countermeasure was the use of dummy screens. In this case, if the attacker selected a distractor for any screen in the challenge session, the remaining challenge screens consisted only of distractor images. The second countermeasure was the use of constant distractors. In this case, for each passimage in the user's passimage set there were a specified number of constant distractors. This meant that whenever a passimage was selected as part of a challenge screen the constant distractors associated with the passimage also appear as part of the challenge screen. The algorithm was as follows:

1. Select a subset of s passimages from the users passimage set.

2. Generate a challenge session for the passimages selected (i.e. generate s challenge screens with one of the passimages for each screen, and random and/or constant distractors).

3. While there is another challenge screen in the session:

   (a) For each image in the challenge screen increment the number of times it has been viewed or add to the list of viewed images if it hasn't been seen before.

   (b) Get the image which has been viewed the most times.

   (c) Check to see if the most viewed image in the screen is the passimage

      i. If the most viewed image is the passimage then increment the number of screens passed.

      ii. If the most viewed image isn't the passimage and dummy screens are being used, set the remaining challenge screens to dummy screens.

4. Once there are no more challenge screens, if all screens were passed then end the simulation. Otherwise return to step 2.

This process is represented by an activity diagram in Figure D.1.

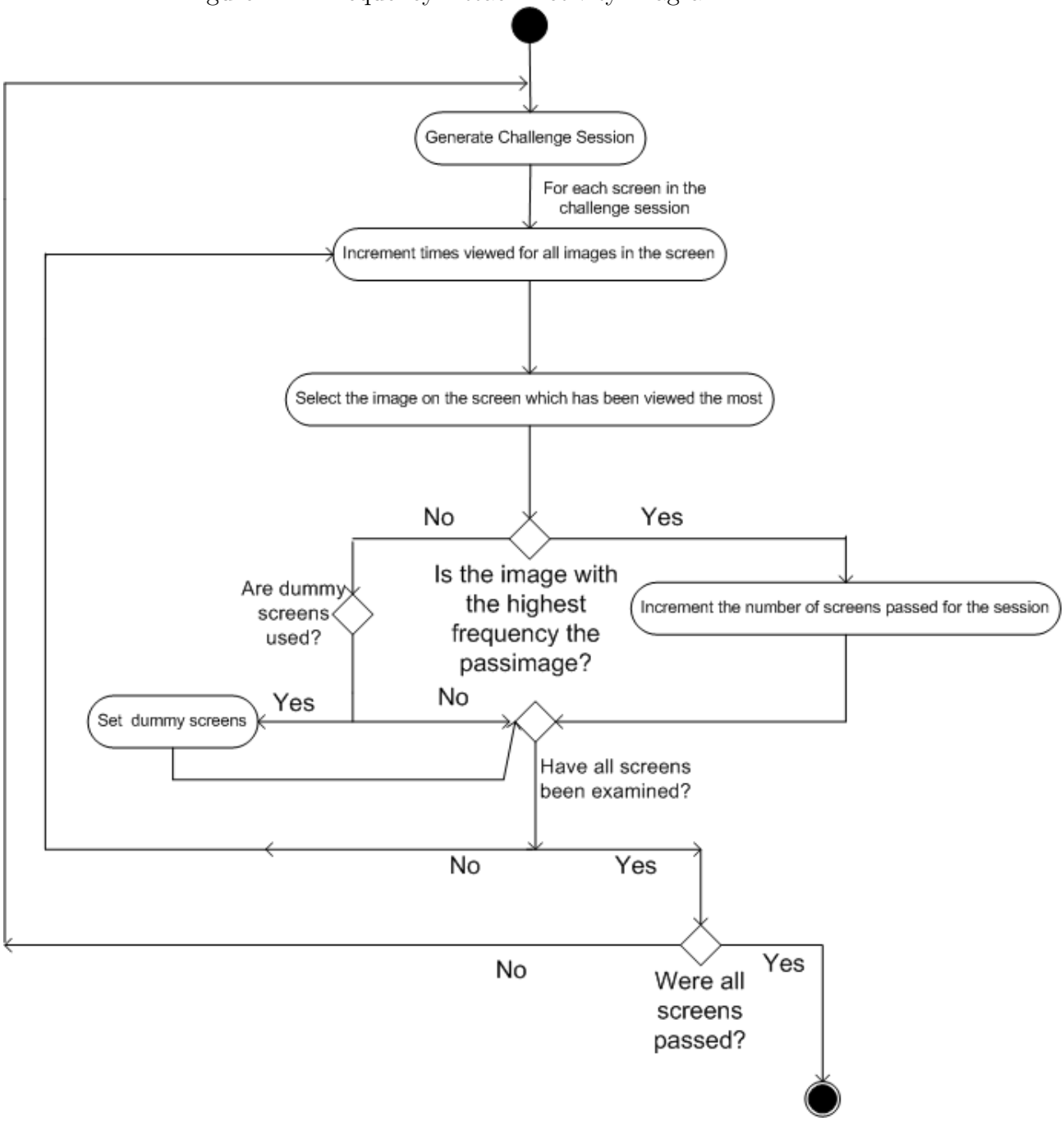
Shoulder Surfing Attack

The shoulder surfing attacks consist of two stages; viewing and attacking. The view and attack stages are performed in succession until the attacker has viewed all the passimages required to authenticate in an attack. If the number of passimages in the user's passimage set is equal to the number of challenge screens in a challenge session then the number of attacks required can be calculated as the number of screens per session divided by the number of screens multiplied by the probability of recall (percentage of recall divided by 100). If the number of passimages exceeds the number of challenge screens a different approach is taken. If the percentage of recall of the attacker is 100% then only one challenge session will need to be viewed to successfully attack. If the percentage of recall is less than 100% then the following calculation is used to calculate the number of sessions which need to be viewed. First we calculate the probability of recalling an image as the percentage of recall divided by 100. The number of images recalled per session can then be calculated as the number of screens multiplied by the probability of recall.

If the number of images recalled per session is greater than 1, the number of sessions required can then be calculated as the number of passimages divided by the number per session. This value is rounded up if it is not an integer since a fraction of a session does not make literal sense. The list of viewed images can then be reduced to this size by randomly removing excess passimages.

If the value of the number of images recalled per session is less than one then the number of views required per image needs to be calculated. We do not round to 1 each time as this could overestimate the recall of the attacker. For example

Figure D.1: Frequency Attack Activity Diagram



say the attacker has 2 images in his viewed list and he only recalls 40% of viewed images. This gives the number of images recalled as 0.8, rounding in this case would over estimate the recall. In other cases where we round down e.g. 2 images at 20% recall giving a value of 0.4 images recalled rounded down would mean no images would ever be recalled which we suggest is unrealistic. The number of viewed required per passimage is calculated as 100 divided by the percentage of recall. The viewed images list can then be reduced by removing images which have not been viewed enough times to recall.

The attack session is then performed. A challenge session for the target passimage set is constructed. If all the passimages selected are in the attacker's list of recalled passimages then the attack is successful and the simulation can be terminated. If not all passimages are in the attackers recalled list then the process starts once more. This algorithm can be represented as follows:

1. Generate a new challenge session for the target passimage set.
2. Add all the passimages in the challenge session to the attacker's viewed images list and increment the number of times each passimage has been seen.
3. Calculate the number of passimages recalled per session by taking the number of images in the attacker's viewed images list and multiplying it by the probability of recall (the percentage of recall divided by 100).
    (a) If this value is >=1 then round the number down if the remainder is less than 0.5 and up otherwise. The list of recalled images is then reduced to this size.
    (b) If this value is < 1 then calculate the number of times an image needs to be viewed until it is is recalled. This is calculated as 100 divided by the percentage of recall.
        i. The list of images viewed can then be reduced by removing images which haven't been seen frequently enough to be recalled.
4. Generate a new challenge session for the target passimage set.
5. If all the passimages presented in the challenge session are in the attacker's list of recalled images the attack is successful.
6. If the attack is not successful, return to step 1.

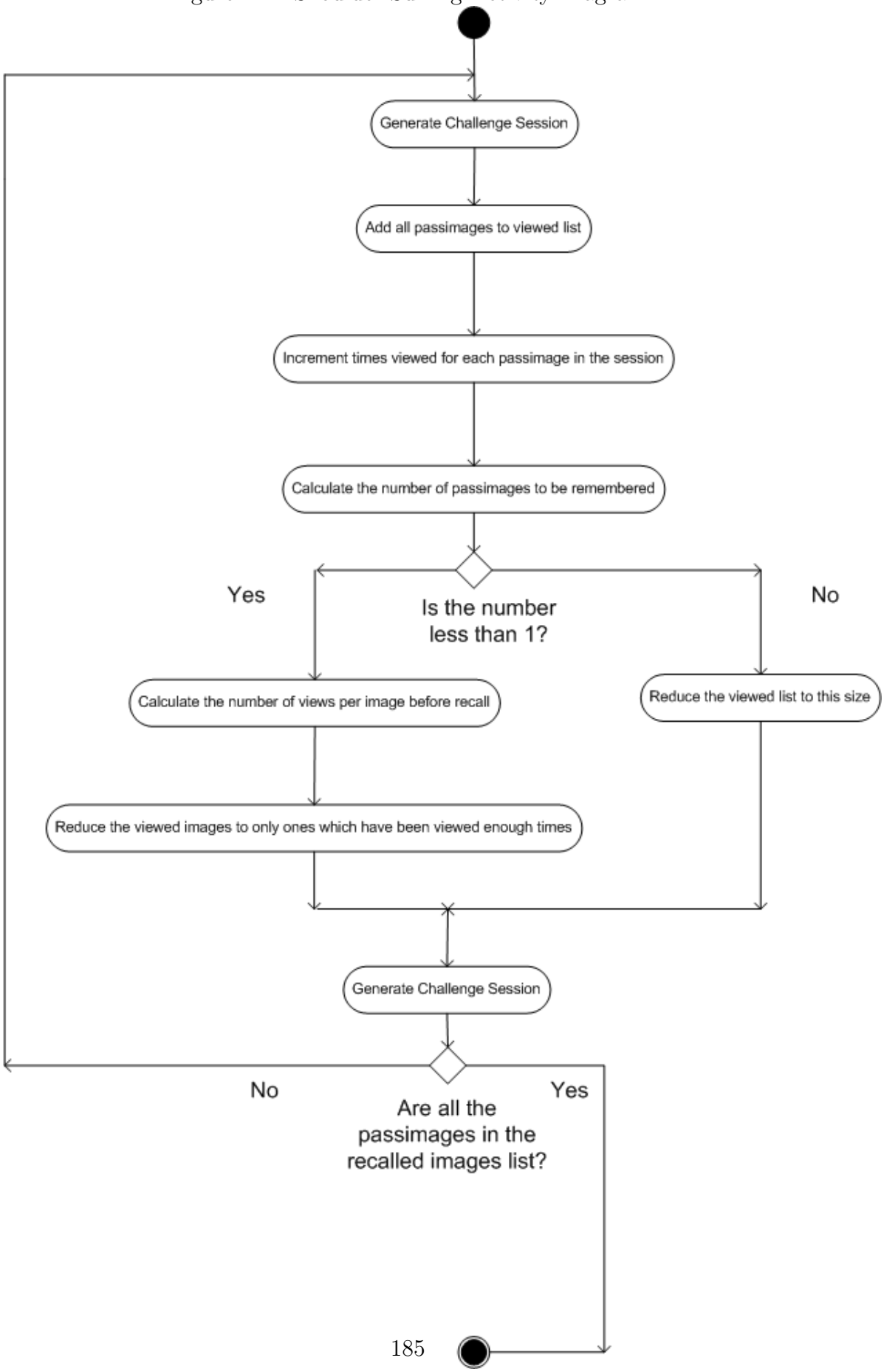
This process is represented by an activity diagram in Figure D.2.

D.1.5 Elements to Model

To model the RBGP scheme itself the following objects need to be modelled:

- User Passimage Set- a subset of p images from the potential images set
- Potential Images Set (all possible images to be used in the scheme)

Figure D.2: Shoulder Surfing Activity Diagram



- Image - to model individual images (distractors or passimages, i.e. all images in the potential images set)
- Challenge Session
- Challenge Screen

In addition the following processes need to be modelled:

- Distractor Selection
- Dummy screens

To model the attacks themselves the following processes need to be modelled:

- Construction of a passimage set
- The attack algorithms
- A function to get the image in a challenge screen which is the most frequently seen
- A function to check if the most frequently seen image on a challenge screen is the passimage on a screen (i.e. check two images are equal)
- A function to perform the calculation of the median value of a collection of results for a specific configuration. If the number of runs in the collection was even then the result would be the average of the two middle values. If the number of runs in the collection was odd then the result is the ((n + 1)/2)th item, where n is the number of runs.
- A function to write out the results data to a file

D.1.6 Class Diagrams

The class diagrams for the frequency attacks simulation and shoulder surfing attacks simulation are shown in Figures D.3 and D.4 respectively.

Figure D.3: Shoulder Surfing Simulation Class Diagram

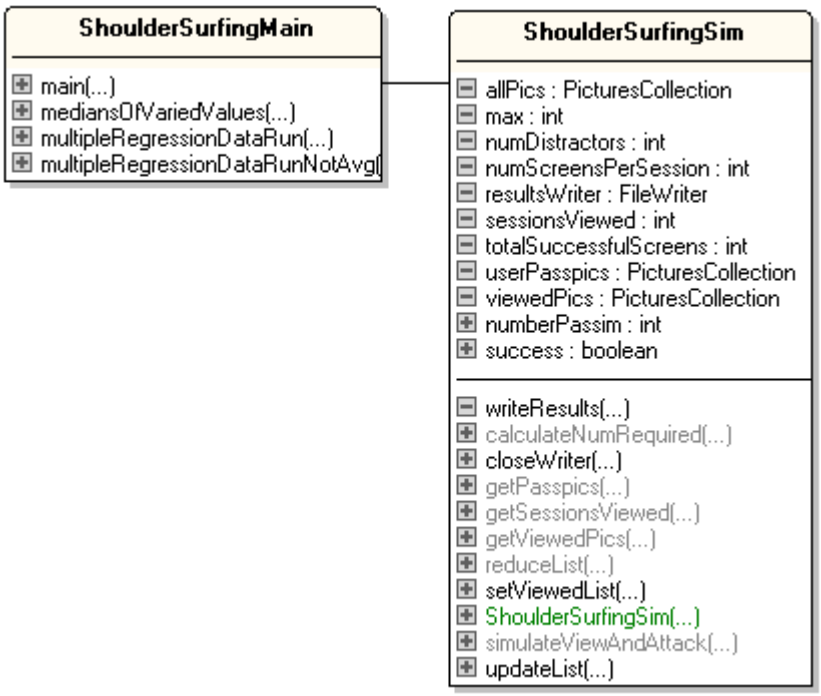


Figure D.4: Frequency Attacks Simulation Class Diagram