REFERENCE ONLY

UNIVERSITY OF LONDON THESIS

Degree PhD Year 2005 Name of Author BROSTOFF, A.M.K.

COPYRIGHT

This is a thesis accepted for a Higher Degree of the University of London. It is an unpublished typescript and the copyright is held by the author. All persons consulting the thesis must read and abide by the Copyright Declaration below.

COPYRIGHT DECLARATION

I recognise that the copyright of the above-described thesis rests with the author and that no quotation from it or information derived from it may be published without the prior written consent of the author.

LOAN

Theses may not be lent to individuals, but the University Library may lend a copy to approved libraries within the United Kingdom, for consultation solely on the premises of those libraries. Application should be made to: The Theses Section, University of London Library, Senate House, Malet Street, London WC1E 7HU.

REPRODUCTION

University of London theses may not be reproduced without explicit written permission from the University of London Library. Enquiries should be addressed to the Theses Section of the Library. Regulations concerning reproduction vary according to the date of acceptance of the thesis and are listed below as guidelines.

A. Before 1962. Permission granted only upon the prior written consent of the author. (The University Library will provide addresses where possible).

B. 1962 - 1974. In many cases the author has agreed to permit copying upon completion of a Copyright Declaration.

C. 1975 - 1988. Most theses may be copied upon completion of a Copyright Declaration.

D. 1989 onwards. Most theses may be copied.

This thesis comes within category D.

☑ This copy has been deposited in the Library of UCL

☐ This copy has been deposited in the University of London Library, Senate House, Malet Street, London WC1E 7HU.

Improving password system effectiveness

Alexander Brostoff

A dissertation submitted in partial fulfilment
of the requirements for the degree of

Doctor of Philosophy
of the
University of London

Department of Computer Science
University College London

Submitted 30th September 2004

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Abstract

As computers reach more aspects of our everyday life, so too do the passwords that keep them secure. Coping with these passwords can be a problem for many individuals and organisations who have to deal with the consequences of passwords being forgotten, yet little is known of this issue. This thesis considers the effectiveness of password authentication systems for three groups of stakeholders including users, support staff, and system owners. The initial problem of how to create memorable but secure passwords is reconceptualised as how to improve password system effectiveness. Interview, questionnaire, and system log studies in BT, and experiments at UCL-CS confirm some basic hypotheses about key variables impacting performance, and show that other variables than the memorability of password content are also important which have hitherto not figured in security research and practice. Interventions based on these findings are proposed. Empirical evaluation suggests that the interventions proposed that 'redesign' the user but exclude other parts of the system would fail. Reason's (1990) Generic Error Modelling System (GEMS) is used as a basis for modelling password system performance at the level of individual users. GEMS and the Basic Elements of Production are used generalise these findings, and for the first time to model information security. This new model, "Elevation", is validated by expert review, and a modified version is presented.

Contents

Figures

Tables

Acknowledgements

This thesis is dedicated to my parents Daniel and Erica Brostoff. Special thanks are due also to my grandparents and my uncle and aunt, Jonathan and Deanna Brostoff. All of these people continue to inspire me; their wisdom, generosity and strength took me through my education and will carry me beyond it.

I'd like to thank my PhD supervisor Prof. M. Angela Sasse for recruiting me for this grand journey. Her guidance solved many difficult problems and her criticisms made this a better piece of work. She put her money where her mouth is. Peter Lunt, my second supervisor, helped me at the crucial beginnings.

I'd like to thank the many people at BT who've funded and made this PhD possible. Thanks are due to: Charles Brennan and Marek Rejman-Greene, my industrial supervisors; Simon Clayton and James MacKay, the kings of good cheer and password control; Jeff Prince and his gracious team in Cardiff; and the many BT people in Adastral Park, Cardiff, Portsmouth, Thurso and across the company who took precious time to talk with me, share data, make introductions, or fill in questionnaires.

Colleagues at the Ergonomics & HCI Unit, UCL, were a vital source of counsel and happy times. Thanks to Prof. John Long and the merry band: Rachel Benedyk, Jacky Cross, Wajih Djabri, Sub Chakraborty, James Middlemass, Becky Hill, Tony Lambie, and all the rest. Colleagues at Computer Science, UCL did the same excellent job. Thanks to John Dowell, Ismail Ismail for friendship and sanity, Simone Stumpf, Anna's Conniff and Bouch, Louise Sheeran, Piers O'Hanlon; Ben Southall, Richard Wheeldon and all those who worked on or used TACO; Nadav Zin, Gillian Wilson, Anne Adams for her recommendation and blazing the trail, Ian Brown, Simon Attfield for his visual design, Michael Tscholl, Jens Riegelsberger, John McCarthy, Peter Monthienvichienchai, the admin, finance and tech support teams, and my partner in crime fighting, Dirk Weirich.

Clare Holden and Mhairi Gibson brought a touch of class to the proceedings, and were always ready for tea and biscuits. Thanks to all the other friends, who listened to grumbles without complaint, then brought insight.

Finally I'd like to thank Rebecca Sear for sharing the highs and lows, and for doing it with long patience, warmth, and frequent good humour.

Chapter 1

Introduction

1.1 The problem and thesis scope

Many people have a password story to tell. More often than not, it is an enthusiastic description of the clever way that person has found of coping with them. This shows two things: that password systems (defined in section 3.1) affect many people in their day to day lives, and that this effect is problematic (else why would it need special coping strategies?). This is especially noticeable in the places where these problems all come together - computer helpdesks, where the individual users' password problems are shared by their organisation that must divert resources to sort them out. The extent of the problems, and their effect on individuals and organisations raise a number of research questions:

Research Question A.	What is the performance of password systems in actual use?
Research Question B.	How can the performance of password systems be measured?
Research Question C.	What are the causes of good or bad performance? And ultimately
Research Question D.	What interventions can be made to improve the performance of password systems?

These questions will begin to be addressed in this thesis.

This thesis is aimed at helping large organisations such as universities and corporations and the people they are composed of who are using passwords in the pursuit of the organisations' work ("*users*"). It is not aimed at solving the password problems of the organisations' customers, other private individuals, nor the personal (rather than work related) password problems of users in large organisations; though all these people will benefit from the research. By helping *users*, it is also possible to aid helpdesk workers / system administrators, and the high-level managers / owners of the organisation. This thesis aims to address needs of all these stakeholders. Should these stakeholders' needs conflict, it will prioritise the needs of the users.

The discipline of Human Computer Interaction (HCI) has several indices of work system performance, for example: users' satisfaction, learnability, fun, users' stress, cognitive workload, etc. This thesis employs the indices *time* and *errors* (see section 2.2.3), with *errors* frequently operationalised in this thesis as users' calls to a computer helpdesk to get a password reset, or users' requests for a password reminder. There are other valid foci for password performance research which are not pursued in this thesis: important issues relating to users mental models of computer security and

passwords, and related issues of perceived risks, corporate culture and social dynamics. However, the particular quantitative focus of this thesis has the benefit of simpler collection, analysis, and interpretation while maintaining the ability to deliver significant contributions.

In line with the goals of HCI (see section 2.2.2), the 4 research questions are reconceptualised as being part of an effort to:

Research Problem 1. *Reduce the costs* of password authentication systems to stakeholders, and to *improve the quality* with which password systems carry out the authentication task. And further to

Research Problem 2. *Start model building* that may eventually enable the findings of this research to be *generalised and validated,* so that they may be re-used for other classes and scales of work system in the domain of information security.

1.2 The research approach

The empirical research in this thesis amounts to a *summative* and a *formative evaluation* (see section 2.2.4) of password authentication with data from BT (the corporation sponsoring this PhD) and UCL. Part 1 of the research problem (see previous section) is addressed by measuring password system performance in BT and UCL, discovering the underlying causes, and then proposing interventions that would improve performance (see section 9.4): reducing stakeholder costs and/or improving task quality.

The problem was addressed using a range of standard Human-Computer Interaction evaluation techniques. Analytic techniques were employed, for example by analysing documentation such as recommendations of best practice for password mechanism configuration and use, and security policy documents. These techniques informed and directed the use of empirical methods: users' and system administrators' experiences were captured via interview and questionnaire, and their use of password mechanisms were observed through the use of system logs and diaries. This allowed the effects of different components of password best practice/received wisdom to be tested empirically.

This research has been done with participants from two different kinds of large organisations - each of interest on its own and complimentary. Corporate populations are likely to face a greater security and password problem, as corporate users will

more often process data belonging to and about other people. However, commercial necessity constrains both the availability of users for research in a corporate population, and the techniques that can be employed with them (for example, it was not possible to instrument a live business computer system to record performance data). Students are users in the comparatively flexible university environment. They are likely to be more easily available, and a wider set of techniques may be deployed with them. They exist in a different security context - as they process (and so defend) mostly their own data.

The thesis addresses part 2 of the research problem (see section 1.1) by presenting a model that abstracts from the particulars of password authentication in BT and UCL so that the findings can start to be applied more widely in the domain of information security to password problems in other organisations (see Chapter 11).

1.3 The thesis overview

A map of the thesis is given in Figure 1. Chapters 6,7, and 8 present the empirical findings, which tackle research problems a) and b) and collect the data for c). These Chapters perform a summative evaluation of password mechanisms in BT and UCL, and begin the process of formative evaluation. Chapter 9 tackles research problems c) and d), bringing together the results from the previous Chapters to diagnose the ineffectiveness of password mechanisms at BT and UCL, and suggest interventions that could improve performance. Chapters 6 through 9 can therefore be seen as answering Research Problem 1 (reducing the costs and improving the quality in password system performance). The final Chapters of the thesis are devoted to Research Problem 2 (modelling and validation). Chapter 10 begins to validate a subset of the interventions. Chapter 11 abstracts the findings of this thesis into a model intended to enable reuse of the findings in other large organisations.

The individual chapters are outlined below:

Chapter 2 introduces the academic disciplines that inform this thesis: Computer Security and Human Computer Interaction (HCI). The goals, fundamental principles, measures and techniques of these disciplines are outlined. The application of HCI's principal techniques to the research problems are briefly discussed.

Chapter 3 outlines the relevant properties and working of password authentication systems, including both its human and computer components. Outlines are given of how password strength is measured, the vulnerabilities of password systems, and common administrative responses to these vulnerabilities. Outlines are also given of the psychology of memory and human performance. It is noted that human error is of

great importance to computer security, and that the discipline of human computer interaction is ideally placed to study it. This chapter sets the scene, and gives the PhD its focus upon time and errors.

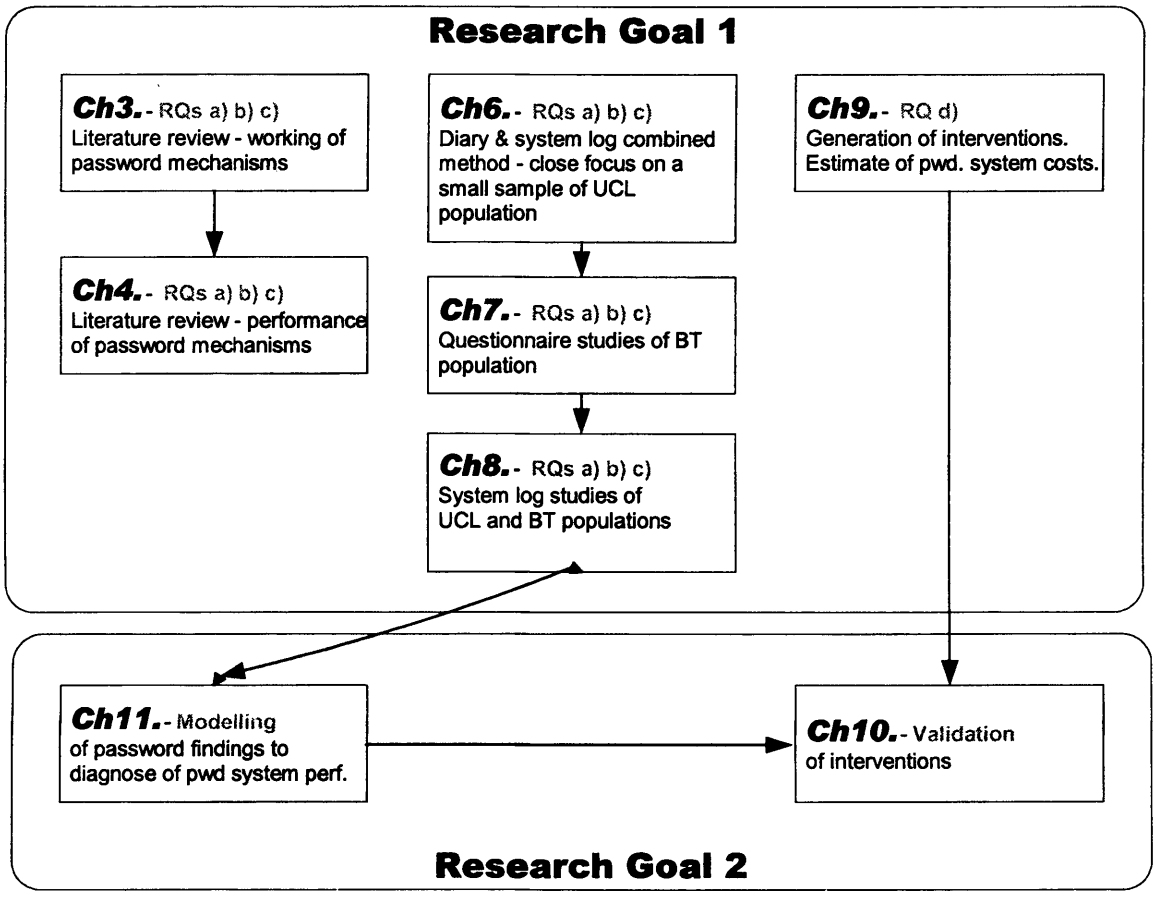


Figure 1 - Map of the thesis

Chapter 4 summarises what is already known about the performance of password authentication systems. An obvious intervention to improve authentication is to replace poor performing password mechanisms with rival technologies. Chapter 4 also outlines some of these alternative mechanisms. This review concludes that rival technologies achieve the same level of security by using more memorable content. However, the chapter also shows that with some modification, traditional passwords can also use more memorable content and achieve good security, and that passwords have significant strengths. The chapter ends by re-examining this PhD's initial problem (as given by the PhD's sponsor) in the light of the preceding chapters, and re-specifies it to be the one presented in section 1.1. This chapter sets the scope and direction of the work in the following chapters.

Chapter 5 is the methods chapter. A summary is given of the goals and methods of the studies described in the following chapters.

Chapter 6 begins to estimate the basic parameters of password use and performance in a real world setting. The number and type of password errors are estimated for six people with diverse roles in UCL (undergraduate, PhD student, staff member). This estimated performance is given context by data collection about the number of passwords owned by the participants, their password construction, and frequency of password use. The combination methodology is found to be powerful but too expensive, and improvements are suggested.

Chapter 7 pursues a methodology-questionnaire surveys-which enables data to be collected from the difficult to reach population of BT users. This data goes beyond the data collected from the UCL sample, allowing the beginnings of a diagnosis of performance. This data gives evidence that several of the management techniques employed to improve security actually reduce password system performance, and in so doing result in poorer security. However, the data does not allow the performance of passwords in the BT sample to be estimated.

Chapter 8 takes up this challenge, using system logs as the data source. Password performance is calculated for a UCL sample and three different BT information systems, thereby achieving one of this thesis' main objectives (answering Research Question A). Evidence is presented that BT has a greater problem with passwords than UCL. Several different analyses of these logs are made, that give further evidence that some of the management techniques employed to improve security actually reduce it (helping to answer Research Question C).

Chapter 9 brings together findings about the causes of password performance to suggest interventions for improving it (referring to a model presented in Chapter 11). Password costs are split into three categories, and estimates given for each type of cost in BT (similar data could not be collected for UCL). The aspect of performance primarily affected by each intervention is predicted (task quality or one of the three costs).

Chapter 10 puts together a hypothetical package of interventions from Chapter 9 that could be implemented in BT's economic and internal political circumstances: interventions which are aimed at changing the users behaviour rather than reallocating function to technology or changing management practices. This package of interventions is put through the beginnings of a validation process using focus groups from two different areas of the organisation. Evidence is collected that the interventions would work if used, but that a significant minority made up of relatively

powerful members of the organisation would not conform to the behaviour changes proposed in the package, and that the interventions would fail unless accompanied by changes to infrastructure and other aspects of security management.

Chapter 11 abstracts the thesis' findings about the causes of password problems. It presents a model that ties these findings to existing psychological theory-Reason's (1990) conception of human error.

Chapter 12 summarises the thesis, suggests further work and begins it, by starting development of a higher level model relating human error to wider aspects of computer security and highlighting initial directions for its development. This higher level model focusses on the management of organisations and is an interpretation of a model that has been successfully used in the domain of industrial safety. This new model begins to explain how the situation modelled in Chapter 11 could have developed, and how interventions based on this model could have been found to be relevant but not acceptable by BT users in Chapter 10.

1.4 The research contributions

This thesis will have two main substantive contributions:

- A body of empirical data measuring password system performance, collected from one corporate and one academic environment. This gives a starting place for comparisons between and within the organisations, and can serve as a foundation for model building and theorising. This data reveals unintended negative effects of security policies upon password system performance
- The presentation of a model (Chapters 10, 11) which is a first pass at explaining the data outlined above. This can be used as a starting place for further research which may lead to more general models (see 12.5.5 in Chapter 12), upon which proactive security review, retrospective investigation techniques, and security design methods can be based.

Parts of the research reported in this thesis have been published. The results in Chapter 7 were published as part of a journal article and book chapter (Sasse, Brostoff, & Weirich, 2001; Sasse, Brostoff, & Weirich, 2002) . Chapter 8 contains work that has previously been published in peer-reviewed conferences (e.g. Brostoff & Sasse, 2000, 2003), and the model in Chapter 12 has also been published in proceedings of a peer-reviewed conference (Brostoff & Sasse, 2001).

Chapter 2

Background /Security and HCI

This chapter will outline the fundamentals of the two disciplines that contain the background material to this PhD: Computer Security and Human-Computer Interaction (HCI). We will begin with computer security, outlining its goals, then its general techniques, followed by a more focussed description of the techniques used for authenticating users (including passwords and their major alternatives). Following that is an overview of how different attacks affect the problem of securing computers. The section on computer security concludes that the human factor in computer security needs to be considered to improve effectivenss of security systems.

The second half of the chapter will outline the fundamentals of HCI, a discipline considering the human and technology components of systems and optimising the interaction between them. The nature and concepts of HCI will first be outlined, then the discipline's goals, scope and measures. Finally, the techniques used in HCI and their application to researching computer passwords will be discussed.

2.1 Security

2.1.1 The problem of security

The fundamental concepts and models used to describe the security process are set down in international standards (BSI, 1996). These concepts will be referred to in the thesis, and are outlined here. Any organisation faces security *threats* (theft, fraud, etc) against their *assets* (customer data, secret product plans, reputation, etc.) which if realised will have a negative *impact* (destruction or permanent removal of the assets, etc.). *Vulnerabilities* to particular threats increase their opportunity for causing impacts - vulnerabilities increase the *risk* that impacts will occur. *Safeguards* are put into place to protect against impacts, and so reduce risks. *IT security policies* are rules, directives and practices that govern how assets are managed, protected (i.e. with safeguards) and distributed within an organisation. *Risk* of threats occuring, *vulnerability* to and *impact* of realised threats are all independent variables in security equation, which seeks to minimise risks by applying safeguards. However, safeguards have their own impact (initial cost and maintenance, inconvenience, etc.), and there is no sense in having them if the treatment is worse than the disease. Organisations therefore seek to minimise the impact of both security breaches and safeguards.

2.1.2 Goals (access control)

According to Garfinkel and Spafford (1996), computer security has several major principles that it strives to uphold:

1. ***Confidentiality*** Protecting information from being read or copied by people who are not authorised by the information's owner to read or copy it. This includes pieces of information from which the confidential information can be inferred.
2. ***Data integrity*** Protecting information including programs, backup tapes, file creation times, documentation, etc. from being deleted or altered without the permission of the information's owner.
3. ***Availability*** Ensuring that the computer services are not degraded or made unavailable without authorisation.

Garfinkel and Spafford note that a formal definition of computer security would require detailed explanations of risk assessment, asset valuation, policy formation, and a number of other topics beyond the scope of this thesis.

2.1.3 Techniques for security

The principles of information security are upheld using 3 main techniques, which are (Schneier, 2000):

Prevention. Stopping a security breach from happening, often by identifying vulnerabilities in a system and putting in safeguards. Examples of this technique include access control (passwords), firewalls, encryption. It is often impossible to completely prevent security breaches (Schneier, 2000).

Detection. Discovering that a security breach has occurred or is occurring (*detection*), identifying the nature of the attack (*localisation*), the identity and whereabouts (*identification*) and nature of the perpetrators (*assessment*). Examples of this technique include: intrusion detection systems, system logs, digital watermarking. Detection allows *Response*.

Response. Mitigating the consequences of the security breach, or deterring attacks - usually by punishment. Examples include: insurance and prosecution.

These three techniques are marshalled in a process of *risk analysis,* where threats and vulnerabilities are identified, and risks and impacts evaluated to decide on appropriate safeguards. Risk analysis is difficult and costly, and in practice often requires cursory analysis and blanket imposition of safeguards, when deeper analysis would allow tailored solutions (Cho & Ciechanowicz, 2001).

2.1.4 Techniques for authentication

The bedrock on which the principles and techniques are built is the ability to distinguish between authorised and unauthorised users. The process by which this occurs is called *user authentication.*

A person first declares her identity as a bona fide user, for example with a *user name.* There are 3 avenues that may be then taken to verify or *authenticate* her declared identity. They are: *knowledge based authentication*, where the user reveals a secret or secrets to the computer such as a password that only the real user should know; *token based authentication*, where the user presents some physical or digital object to the computer that should only be in the possession of the real user; and *biometric authentication*, where the user presents herself to the computer for examination, so it may recognise a characteristic of her phenotype or behaviour that is unique to herself. These avenues will be discussed in more depth below.

Knowledge-based authentication

The workings of different knowledge based authentication mechanims is described in sections 3.2.2 and 4.2. They all have in common the advantage that they usually do not require extra hardware than exists at every computer workstation, where the other authentication paradigms do. The objects of authentication are also more easily and cheaply changed if they are compromised than in the other paradigms. The most common type of user authentication system is the *password.*

Token-based authentication

In token based authentication, a physical or digital object is submitted to the computer for examination. The computer generally requires additional hardware and software to conduct the examination, which is extra cost not incurred in knowledge-based authentication. Tokens are designed to be unique and difficult to forge. However, this does not necessarily mean that it is impossible or economically infeasible to do so (Svigals, 1994). Moreover, because the tokens are so easy to pass on, take away, or mislay, they are usually used in combination with one of the other authentication paradigms. For example, a smart card may have a picture of its owner imprinted or stored in it or may require the use of a password. The use of token-based authentication may therefore increase the number of passwords, instead of decreasing them. Many token-based authentication systems therefore still involve password problems.

Biometric authentication

Biometrics are often heralded as the ultimate replacement of passwords (e.g. Murrer, 1999). However, biometrics technology is still relatively immature, and biometric authentication suffers from a number of weaknesses.

There are two kinds of biometric authentication devices: those that focus on the physical structures of the user such as retina scanning, hand geometry, face recognition, fingerprint reading, etc.; and those that focus on the behaviours of the user such as signature, voice print, keystroke dynamics, mousing dynamics, etc.

In general, *structural biometric* devices are expensive compared to passwords (Kim, 1995), while *behavioural biometrics* are unpopular with users because of their potential abuse as work monitoring devices (Deane, Barrelle, Henderson, & Mahar, 1995). For example, a mechanism that identified users by their keystroke dynamics could be used illicitly to monitor their productivity.

Another weakness in both structural and behaviourally based mechanisms is that the biometric presented to the device may appear to be different at each use due to environmental conditions, wear and tear on the device, or human factors - thus increasing the probability that legitimate users are denied access while imposters are let through. These two risks are related, so decreasing one increases the other, and different situations will call for one or the other to be minimised. Maintaining these devices (for example clearing grease and grime from fingerprint readers) adds to their operational expense.

Structural biometrics are based upon the user's physical characteristics. Consequently, one way for attackers to gain access is to use a copy of a legitimate user's characteristics. For example, copies may be left on the biometric reader as a result of login attempts. Some commodity fingerprint readers have been defeated by placing a bag of warm water on or breathing over the sensor to activate the most recently used fingerprint (Thalheim, Krissler, & Ziegler, 2002). People leave behind them many copies of their biometrics in much less defended places than biometric devices, for example, fingerprints left on a glass. This may be copied and used by attackers to pose as a legitimate user. While this may seem far-fetched at the moment, criminals will have much greater motivation to start dusting glasses for prints should structural biometrics become the norm rather than a rarity.

Another way for attackers to gain access is to steal a legitimate user's characteristics. Biometrics therefore lead to a greater threat of mutilation against the user (Garfinkel, 2000). Manufacturers attempt to reduce this threat technologically by building in "signs of life" detection, such as vascular throb (which also safeguard against copied

biometrics). However, these technologies have appeared to be fairly simple to defeat (Schneier, 2002). Moreover, the technical accuracy of this technology is irrelevant. Even if it were completely accurate, users would still have a legitimate fear because they would still face attack if the criminals did not know or did not believe that the technology was effective.

2.1.5 Nature of the attacker

An aspect of password security that is of prime importance is the *nature of the attacker*. It is intimately related with the utility of *strong password*s (Figure 2), and so is directly related to the effectiveness of password policy (see section 3.1.3, p.45), and the password recommendations given to users.

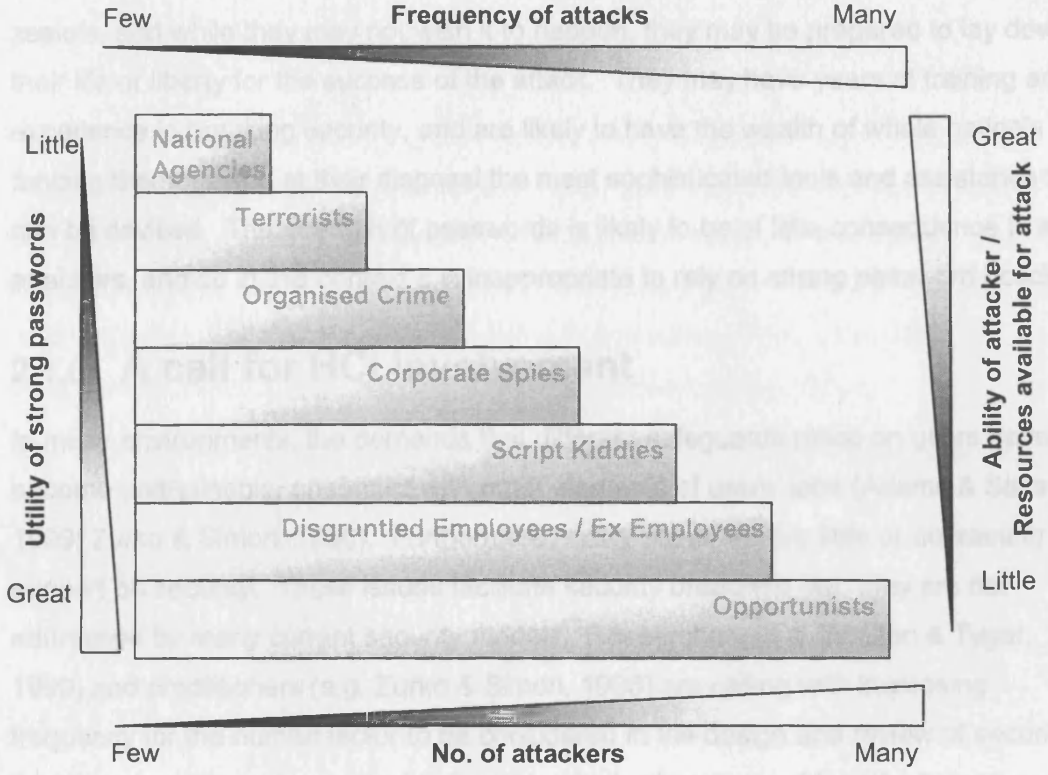


Figure 2 - Properties of different kinds of attacker and their relation to the utility of strong passwords

The people who attempt to break password security differ in important ways (Figure 2), including: their motivation, their abilities/skills, their resources and methods, and their number. These combine in predictable ways, and alter the frequency and power of attacks made against password mechanisms. Figure 2 is based on the computer security institute/FBI 2002 crime survey (Power, 2002), and shows different classes of

attacker, and how their characteristics vary and the implications for password based security.

At the bottom of the figure attackers may be characterised as: extremely numerous (millions worldwide, significant percentages of the constituents of any organisation), risk averse with little motivation (unwilling to tolerate much risk to their reputation or wallet) and so keen to avoid detection, unskilled, with almost no resources-budget of a few pounds, with little access to computerised tools. Their attacks are unlikely to be successful against password security, and if they were successful, would be unlikely to cause much damage. At this level, *strong password* policies could be relatively effective.

At the top end of the figure, the attackers are relatively few-merely hundreds to thousands of them worldwide. They are highly motivated, being professionals or zealots, and while they may not wish it to happen, they may be prepared to lay down their life or liberty for the success of the attack. They may have years of training and experience in breaking security, and are likely to have the wealth of whole nation's funding them, having at their disposal the most sophisticated tools and assistance that can be devised. The strength of passwords is likely to be of little consequence to such attackers, and so in this context it is inappropriate to rely on *strong password* policies.

2.1.6 A call for HCI involvement

In many environments, the demands that different safeguards place on users have become unattainable, or conflict with other elements of users' jobs (Adams & Sasse, 1999; Zurko & Simon, 1996). Furthermore, many users receive little or no training or support on security. These issues facilitate security breaches; yet, they are not addressed by many current security models. Researchers (e.g. Whitten & Tygar, 1999) and practitioners (e.g. Zurko & Simon, 1996) are calling with increasing frequency for the human factor to be considered in the design and review of security in the IT system life-cycle. A discipline exists which already considers the human aspects of computing - Human Computer Interaction (HCI), and the following section will review what HCI can contribute to improve password security.

2.2 HCI

2.2.1 Nature and concepts

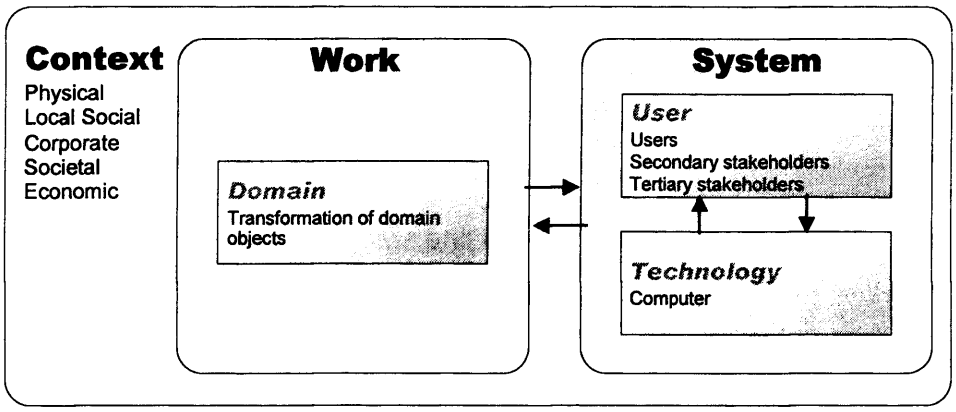


Figure 3 - Diagram of an Interactive Work System

HCI is concerned with optimising the performance of interactive work-systems. Interactive work-systems (henceforth termed as *systems*) are composed of *users* (people who operate computers) and *technology* (computers, mechanical or electronic components) that interact with each other to achieve a goal or do work, by performing certain tasks. The concern of the discipline is sometimes widened to consider stakeholders instead of users. There are three types of stakeholder:

- *Primary stakeholders* will be refered to as *users* - they operate computer components,
- *Secondary stakeholders* manage and maintain the system,
- *Tertiary stakeholders* own the system.

2.2.2 Goals & scope

The goal of the discipline is to reach an optimal balance between two criteria of system performance: the quality of the task performed - ***task quality*** (how good the product is, how fast it is achieved, etc.), ***and*** the cost of achieving that quality (to the user or other stakeholders, and the computer parts of the system) - ***stakeholder costs*** and technology costs. ***The balance between the cost and quality of work is described as the system's effectiveness***, with high quality and low costs being effective (Dowell & Long, 1998).

It has been argued that in its research aspect, HCI should seek to build validated theory and models that can make knowledge gained through practice more easy to re-

use, and give a better probability of successful design (cf. Dowell & Long, 1998). Dowell and Long (1998) advocate the generation of validated knowledge as engineering principles, which may be expressed in symbolic generalisations that function both as laws and definitions of solutions (p.128). The research in this thesis will be informed by Dowell and Long's (1998) conception, and begin the path to generalisation.

2.2.3 Performance indices

This section will justify the use of time and errors in this research as subjects of measurement. International standards (e.g. BSI, 1998) measure usability *summatively* (see section 2.2.5) in terms of the *effectiveness*, *efficiency* and *satisfaction* with which computing goals are achieved in a specific context of use. Dowell and Long's (1990) conception of effectiveness encompasses the international standard's conceptions of both effectiveness and efficiency, and can be extended to include satisfaction (if a purpose of the work system is identified as being to transform users from unsatisfied to satisfied).

In all HCI research and practice these measurements need to be described at a lower level before they can be made. One such description will be taken as an example: the MUSiC project (Macleod, Bowden, Bevan, & I., 1997). This description contains the following indices:

1. Task Effectiveness □(Quantity □Quality)/100 %
2. User Efficiency □Effectiveness/TaskTime
3. Human Efficiency □Effectiveness/Effort
4. Productive Time = *Task Time* remaining after unproductive periods have been removed - time a user spends progressing towards the task goals, irrespective of whether the goals are eventually achieved.
5. Productive Period = *Productive Time* expressed as a percentage of the *Task Time*.

Three of these five metrics involve time - showing that time is a core HCI measure of performance. Task effectiveness does not map easily into password system performance. The quality of authentication and password systems requires careful definition. On each use of a password mechanism a valid user may either be accepted as entirely valid or rejected as an impostor, which is an error by the system. Similarly, an impostor may be either accepted as a valid user (an error) or rejected. Errors are therefore intrinsic to Task Effectiveness.

The last two performance indices in the list above are defined using the action types listed below:

1. Unproductive actions = Help Actions, Search Actions, Snag Actions
2. Help Action = The user obtains information about the system, for example by telephoning the helpdesk.
3. Search Action = The user explores the structure of the system - displaying parts that are not currently accessed - without activating any of the parts that are presented.
4. Snag Action = The user or system performs an action that does not contribute directly or indirectly to the task output, and that cannot be categorised as a help or search action. There are three types of snag action - negating actions, cancelled actions, and rejected actions.
5. Negating Actions = User actions that completely cancel or negate previous user or technology actions. They always cause cancelled actions.
6. Cancelled Actions = User or technology actions that are completely negated by the user or the technology.
7. Rejected Actions = User actions that are rejected or 'ignored' by the technology, and that consequently have no effect.

Here the use of help facilities is explicitly defined as a performance index. Moreover, helpdesk uses are more likely to be logged and available to researchers than any of the other unproductive actions (see section 5.2.7). Hence the use of password helpdesks is a valid concern of HCI and this thesis.

Snag actions imply error - these actions are negated, cancelled or rejected because the actions themselves or the actions leading to them are wrong. Diagnosis of password system performance (which answers Research Question D) requires that the causes of these incorrect actions be identified, i.e what errors were made. Examining password systems' errors is therefore important to diagnose their performance, and is a valid concern of this thesis.

2.2.4 Why HCI and not Information Systems approach?

Researchers in the author's discipline have tended to study password system effectiveness in a purely technical manner, as detailed in section 4.1.2. While this is undoubtedly useful and important information, collecting it has not solved the problem of "poor" password system performance in operational environments. Researchers whose interests lie outside the narrow technical performance of password mechanisms

have become interested in the problem of computer security, and even researchers within the HCI & Computer Science tradition have begun to view password based security from a perspective that encompasses more than password content and memorability (e.g. Adams, Sasse, & Lunt, 1997).

However, a recent review concluded that there is little work that melds technical security issues with a wider perspective (Dhillon & Backhouse, 2001), particularly in the areas of key principles, theories and frameworks. Most of the research published to date has come from the information systems community, and has had little impact for practitioners (Dhillon & Backhouse, 1997). Information systems security research has also had little penetration into the traditional HCI community. Information systems research tends to employ theories and terminology from sociology and philosophy, leading to complex and holistic explanations quite different to the well-defined and pragmatic approach taken by the HCI community. It is also noted for its lack of a prescriptive knowledge (Dhillon & Backhouse, 1997), which makes it very difficult to apply. The combination of lack of applicability and its complexity has reduced its audience to *"a small niche of academic researchers"* (Dhillon & Backhouse, 1997). This thesis will therefore be concentrated on the HCI and Computer Science communities' efforts, will maintain a strong interest in a sociotechnical perspective of the problem, and aim to generate prescriptive knowledge.

2.2.5 HCI techniques and their application to the password problem

HCI practice has two main pursuits: *evaluation*, and *design/intervention*. These will be outlined below.

Evaluation

Evaluation is the prerequisite for design. There are two types: summative and formative. ***Summative evaluation*** is the measurement of a work system's effectiveness - enumeration of the system's task quality, and user and technology costs (see section 2.2.2). An example summative evaluation is: too many employees in BT are calling the helpdesk to have their passwords reset. ***Formative evaluation*** is the diagnosis of ineffectiveness - it uncovers the causes of poor system performance. An example formative evaluation is: there are so many password resets in the organisation because people forget their passwords.

Design / intervention techniques

Design/intervention is the specification of the interaction between the user and technology parts of the system, and extends to specifications of the users and technology themselves. Below aspects of design/intervention will be outlined and related to the problem of password system performance.

Allocation of function

Work is made up of tasks, which can be broken down into sub-tasks, and so on. *Allocation of function* is the part of HCI design/intervention where tasks are allocated to either the human or technology parts of the work system. For example, in a standard password authentication mechanism, the long-term storage of the users password is allocated to the user, though the user may sometimes choose to store it elsewhere. The task allocation can have significant consequences for work system performance. For example: if the user allocates long-term storage of her password to a sticky note, it is likely that she will be unable to gain access to much-needed computer resources if her sticky note should fail, for example by falling off the screen overnight and being removed by cleaners. Moreover, because of the properties of sticky notes (they can be read by anyone if they are on prominent display) by allocating long-term storage to a sticky note, she is greatly increasing the risk that someone would find out her password. The user is trading increased organisational risks of confidentiality and integrity breaches (for example against customer data - which will likely impact the organisation more than her) against a reduced personal risk of denial of service (for example forgetting her password when she needs to use the computer - which if realised would likely impact her more than the organisation). Allocation of function can therefore have profound consequences on system performance in general, and password system performance in particular. The theme of allocation of function will recur throughout this thesis.

Training

Training is a powerful intervention in the armoury of HCI. Through using training it is possible to enhance the human components of work systems, by equipping them with new behaviours and cognitive structures. Using this technique, it is possible to facilitate the allocation of functions to the human parts of work systems that are not well performed by the technology parts of the system. Training is therefore intimately related to allocation of function, and must be carefully balanced against the additional load that it implicitly places upon the user. There is a danger that the user is made to fit the technology at personal cost, when a change in the technology would achieve the

same results. For example, a common reaction to the problem of password memorability is to suggest users be given training in mnemonics so that they may better remember all their passwords (that the system allocates them to remember). Should users be burdened with using techniques that researchers describe as effortful and difficult to maintain (e.g.Schacter, 2001)? This thesis will argue that the answer is *no*, and that the current allocation of function within the system should be reconsidered.

Selection

Selection is also intimately related to allocation of function, where the properties of work system components are specified. When applied to password authentication systems, it is the specification of technology components that is the most relevant part of selection. It seems improbable that the problem of password memorability would be resolved by selection of personnel, for example with unusually powerful abilities of recall.

2.3 Summary

The goals of computer security (confidentiality, data integrity, availability) are upheld by technologies that distinguish authorised from unauthorised users. Complementing these safeguards are techniques used to detect and react to breaches so that their recurrence is lessened. Foremost among the safeguards is the password mechanism. While there are alternatives to passwords, it is concluded that they will remain important for many years.

People who attack computer systems have widely varying motivations, risk aversion, skill and resources, and that computer passwords can protect against the majority of these people, though not the few backed by powerful organisations. The discussion of security was concluded by noting that a call had gone out for more research about its human aspects.

The chapter continued by introducing the discipline of Human-Computer Interaction (HCI). Selection, training, and allocation of function are primary activities in HCI design/intervention. The main activities involved in design have been identified and related to password system performance and a conclusion made that selection is not approriate. Since this thesis is primarily an evaluation and diagnosis of password system performance, the discussion of HCI design shall be limited to this brief overview.

Chapter 3

The Password Interactive Worksystem

This chapter has three main sections, the first defines a password system. The second deals with the main technology component of the system: the password mechanism. The third section deals with the password system's human component.

In the second section, the history of the UNIX password mechanism is described, then its workings. The offline dictionary attack process is then described, which is usually considered to be the primary threat facing password mechanisms. Definitions are then given for the strength of passwords. Administrative policies are outlined that complement password mechanisms. Other attacks against password mechanisms are outlined, arguing that a focus on offline dictionary attacks may be detrimental to password system performance. Finally, a number of enhanced encryption password mechanisms are outlined.

In the third section of the chapter an overview is given of human memory and skilled performance. Different routes to accessing human memory are discussed. Theories of forgetting are presented with principles found to improve learning. A brief overview is given of the learning processes in human motor control that occur when a password is frequently used. Next is an outline of Reason's (1990) model of human error, using it to predict the amounts and types of errors that will occur in password use. The section closes with an assertion that human error is of great importance to computer security, and that the application of the HCI perspective and techniques is the most promising for improving the performance of the password authentication work system.

3.1 The password interactive worksystem

The interactive work system model (section 2.2.1) is applied to to password authentication in Figure 4, to provide a working definition of password systems. The goal of the password system is to transform users from not being authenticated to being authenticated. A subgoal of the work system is to transform users from a state where they cannot be authenticated to a state where they can be authenticated.

The password *system* is more than just the mechanism that assesses login attempts. It also contains the users the make the login attempts, who can make errors (see section 3.3.5 for a definition) during attempts and so affect the performance of the system. Also included are *secondary stakeholders* and the technology required to enable users to be authenticated by the password mechanism. For example-users cannot login until they have been enrolled with a password mechanism and had a password delivered to them through a secure channel. Finally, *tertiary stakeholders* and their technologies are included in the password system - they control the operating parameters of the system through policies and budgets.

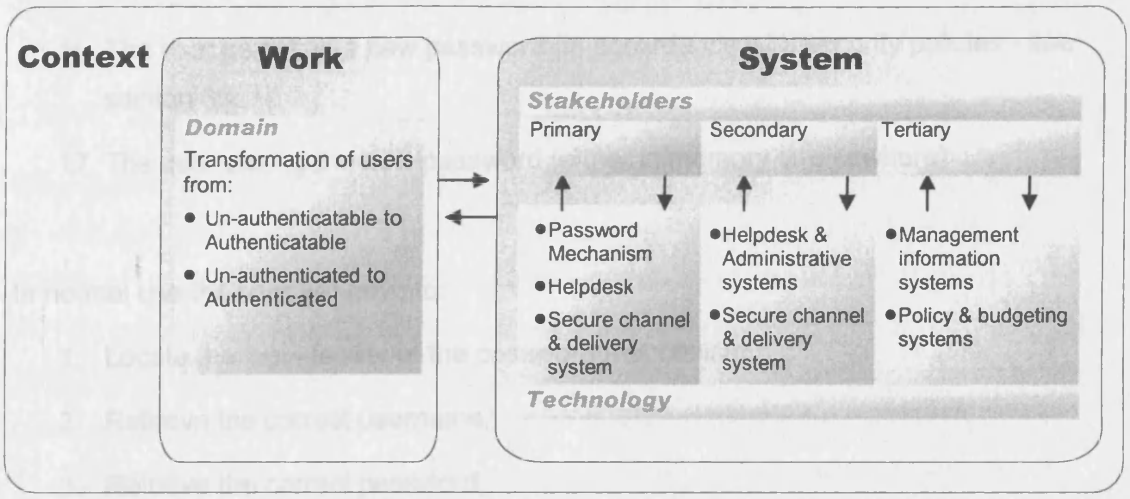


Figure 4 - The password interactive worksystem

The tasks performed by a password system include:

Enrolment/registration

1. Registering users with the password system support service - which may be automated or involve interacting with secondary stakeholders. From this point forwards the system support service will be called a *helpdesk*.
2. Generating credentials for the user to authenticate with a helpdesk.
3. The helpdesk storing these credentials.
4. Registering users with a password mechanism
5. Generating a user ID and password for the user to authenticate with the password mechanism
6. Distributing the user ID and password to the user in a secure fashion
7. Distributing the helpdesk credentials to the user
8. The user receiving the user ID and password, and helpdesk credentials
9. The user storing the user ID and password, and helpdesk credentials (in memory or elsewhere).
10. The user locating the login facility of the password mechanism
11. The user retrieving the correct username,
12. The user retrieving the correct password,
13. The user entering the correct username correctly into the password mechanism
14. The user entering the correct password correctly into the password mechanism

15. The user operating the facility to change their password on its first use
16. The user selecting a new password (in accordance with security policies - see section 3.2.7)
17. The user storing the new password (either in memory or elsewhere)

In normal use the user will have to:

1. Locate the login facility of the password mechanism
2. Retrieve the correct username,
3. Retrieve the correct password,
4. Enter the correct username correctly into the password mechanism
5. Enter the correct password correctly into the password mechanism
6. (if necessary store the user ID and password)

At regular intervals the user may have to:

1. Locate the password changing facility of the password system
2. Continue as from point 11 of Enrolment

If retrieval or entering of the correct user ID and password fail, then

1. The user must identify and contact the helpdesk
2. The user must retrieve helpdesk credentials to authenticate herself to the helpdesk.
3. The helpdesk must authenticate the user
4. The helpdesk must generate a new password for the user to authenticate with a password mechanism
5. Register the new password with the password mechanism
6. Distribute the new passwords to the user in a secure fashion
7. Continue as from point 8 of Enrolment.

3.2 The password mechanism

3.2.1 The Unix password

When UNIX was introduced in 1969/70, its original encryption algorithm was based on a version of the World War II Enigma device (Leong & Tham, 1991). However, its password mechanism was soon considered obsolete and replaced. In 1975, the PDP 11/70 came into service as a high-end computer that was state-of-the-art (cf. Microsoft, 1998). This computer took 1.2 milliseconds to encrypt a password (approximately 800 passwords a second; Leong & Tham, 1991). This was considered too quick at the time as it would make it too easy to use a computer to guess a password (see section 3.2.3). The new Data Encryption Standard algorithm (DES) was adopted circa 1977 (Leong & Tham, 1991). Using the new DES powered mechanism it took more than one second to encrypt a password on a PDP 11/70, and this was considered safe (Leong & Tham, 1991).

In 1988, the combination of Moore's Law and advances in algorithmics meant that passwords were being encrypted at the same speed that caused password mechanisms to be considered obsolete in 1977. In 1998 therefore the UNIX password mechanism became obsolete for the second time (cf. Leong & Tham, 1991).

The password mechanism of mainstream UNIX has not been changed since its second obsolescence, and during this time (according to Moore's Law - Intel, 2003), standard processors have increased in power a thousand times, have become massively more affordable and widely available. A cheap second hand personal computer (PC) with a low-end Pentium III processor can encrypt 50,000 UNIX passwords per second (Semjanov, 2002) - more than 60 times the speed that caused UNIX password mechanisms to turn obsolete for the first time.

3.2.2 How passwords work - the UNIX password mechanism

In general, passwords stored on computers are kept encrypted. This adds a layer of protection by helping to keep user's password secret in case an attacker gains limited access to the computer system. While the details of password authentication mechanisms (such as file names, hashing or encryption algorithms, etc.) vary in different application software and operating systems, they are broadly similar. The UNIX password mechanism will be examined to illustrate the workings of password systems more generally. Current version of Microsoft Windows (2000 Pro, XP home

and XP Pro) work in a similar way. Though the details of file names and locations, algorithms used, password length etc. are different, the principles are the same.

UNIX passwords are encrypted using the DES algorithm, 25 times in a row to produce *coded password data*. In UNIX a permutation is added during the process of encryption (Belgers, 1993), which is not present in Microsoft Windows (Shaffer, 2001). This permutation makes two people with the same password have different *coded password data*. Its function is to improve security by making it more difficult to compile a database which could be used to look up a person's password from its password file entry. There are 4,096 possible permutations, and one is chosen randomly for the user. This makes it 4,096 times as difficult to make the database - both in terms of time taken and storage space required. The chosen permutation is coded into two bytes called a *salt*.

In a *shadow password system*, the coded password data is replaced with * in the file available to normal users (Belgers, 1993); users with sufficient privileges (systems administrators) have access to the *shadowfile* - a second password file which contains the coded password data. This raises the bar for people trying to break the system's security, as they must gain administrator privileges to access the coded password data, and there are safeguards to prevent this.

It would take an unfeasible time to decrypt the password data using current technology; so much so that this decryption has been described as impossible (Belgers, 1993). Given the impossibility of decrypting the password file, how can a user log in? The user types her password, which is encrypted using the method outlined above (using the *salt* as it is found in her password file). If the output of the process matches the coded password data in the shadowfile, then she is considered to be authenticated and is given access to the system.

3.2.3 What is password cracking?

Most advice given to users about the construction of passwords is geared towards a very particular threat, the threat of an *offline dictionary attack* (which will be treated functionally equivalent to the related *brute force attack*). This threat has very significant consequences for the user, in that (all other things being equal) it requires password content that is likely to be difficult to remember.

An offline dictionary attack requires that the attacker has access to the target system's coded password data. This attack is said to be "offline" because it avoids interacting directly with the password authentication mechanism, instead acting on the information used by the authentication mechanism.

In essence, the attack is automated high-speed guessing. A dictionary attack tool has two parts. The first part is a substitute password mechanism, optimised for speed. The second part is a list of guesses, sometimes organised with the most probably successful guesses first. Since it became known that users frequently choose dictionary words as their passwords, this list of guesses has contained entire dictionaries. This part of the attack tool is known as the dictionary file, and has given its name to the attack. The attack can be extended by altering the dictionary words in ways that a user might (reversing them, appending digits or other characters, substituting numbers for visually similar letters, etc.) - a *hybrid* attack. This is slower as there are many alterations possible for each word in the dictionary. Slowest and giving the best guarantee of success (if the attacker can wait that long) is the *brute force* attack - where every possible combination of characters is tried as the password. These will all be referred to as dictionary attacks.

The speed with which guesses can be processed depends (1) on the characteristics of the password mechanism being attacked and (2) the calculating power of the computer used in the attack. Different password mechanisms, for example those of UNIX and Windows XP, have properties that result in dramatic difference in the speed with which coded password data can be created. The consequence is that different password mechanisms may require their users to have dramatically more or less complex passwords to achieve the same level of security.

A hypothetical *strong password* - that limits dictionary attacker to one guess per second - would force an attacker to expend 13.8 hours to run through a 50,000 word dictionary. The user of a hypothetical weaker password mechanism - that executes 50,000 guesses per second - would require users to remember passwords made of two words from the 50,000 word dictionary, to achieve the same level of security.

Having to recall two dictionary words as a password does not seem to be a heavy burden to place upon an user. However, the strength of the passwords used in the example are considered to be woefully inadequate by prominent security practitioners, who recommend baseline password strengths that can be measured in time-to-crack periods lasting millenia, rather than hours (e.g. Viega & McGraw, 2001). These recommendations are not usually well matched to the security required by the user or other stakeholders (e.g. Viega & McGraw, 2001). What impact do these requirements have on the complexity of password content?

For a normal password, this might entail an increase in complexity of approximately two orders of magnitude. For a *strong password*, an increase of five orders of magnitude might be required.

Desktop computers currently can make dictionary attacks on UNIX passwords at approximately single digit multiples of the rate given in the weaker password example above. However, much more powerful general-purpose computers exist in the world, which could be used in realistic scenarios (e.g. by large companies) to speed up dictionary attacks by orders of magnitude. It is also possible to design and construct computers specifically for the purpose of running the dictionary attacks. While this is unlikely in the majority of cases, it is considered realistic for organisations such as national intelligence agencies (Kedem & Ishihara, 1999). Such machines may further increase the rate at which passwords are cracked by orders of magnitude over large general-purpose computers. Kedem (1999) predicts that a $100,000 custom built computer could check 24,576,000,000 UNIX passwords a second.

An ***online*** *dictionary attack* submits guesses to the authentication mechanism and is less powerful than the *offline* attack, because it operates at the relatively slow pace of the authentication mechanism itself (although with multi-user systems this may still be several thousand attempts a second). Moreover, the mechanism can log unsuccessful attempts, and potentially raise an alert or contribute forensic evidence to subsequent investigations. Many password authentication mechanisms may also be configured to limit the number of guesses that can be made, for example in a *three strikes policy*.

If it is assumed that all illicit attempts to guess the password are online and that suitable safeguards are in place, then the feasibility of successful guessing of any particular password is very small, unless the cracker is in possession of detailed knowledge about the target user. Under this assumption, dictionary words can be considered to be safe password content, therefore the user may be permitted to select more memorable passwords, and so the password burden of the user is greatly reduced.

3.2.4 *The payoffs of password cracking*

How likely is a password cracking attack? One recent industry survey said that 21 percent of respondents suffered "attacks related to insecure passwords" (the least common of 7 attack types listed) (Briney, 2001). However, the proportion of dictionary attacks in this is not recorded.

Offline dictionary attacks are only possible in particular circumstances that are unlikely to prevail in a well-configured computer system (e.g. where the password file is restricted to access by system administrators), or in circumstances where the computer system would already have been thoroughly compromised by an attacker.

Let us assume that the attacker has gained administrator status on the victim's computer (by using a buffer overflow exploit for example), and so is able to read or write any data or execute any code on it. He already has the ability to destroy confidentiality, integrity, availability on the system with impunity, and therefore has no immediate need to break users' passwords to do these things.

Mandatory *strong password* content makes it more difficult for the cracker to impersonate any particular legitimate user. Impersonation gives the cracker the benefit of greater stealth, and the ability of "framing" someone "more convincingly" for illegal activities on the system than if he had not cracked the user's account.

The assumption that a dictionary attack is the key that lays bare a computer system is not warranted. The recommendations based on that assumption therefore offer only a slight improvement in security of the system on which they are enforced. Against this, we have to balance (1) the cost of the demands made on users in terms of mental and physical workload and anxiety, and (2) the cost of lost working time and running support mechanisms for the organisation.

3.2.5 Cryptographic strength of passwords

Many password policies state that users should choose *strong password*s - though it may be phrased differently. This section will explore what is meant by *strong password*.

There is a general agreement that password strength is something to do with its resistance to attack, which in most cases will be a form of guessing. However, the notion is operationalised in different ways, with different consequences. When measuring password strength, it is usually assumed that a dictionary attack is being carried out.

Time to crack. The advantage of this metric is that it uses familiar units (hours, minutes, days, etc.) that are intuitive to a wide range of people - and it has most practical relevance. The units are easy to compare in a meaningful way, for example contrast a password that has a strength of ten minutes to one with a strength of ten weeks. However, in practice it is difficult to generate a valid prediction with this metric. The time taken to crack a password depends on the speed that a computer can make guesses, so a valid time to crack prediction requires specification of the attacker's capability. As time progresses, computers get faster, and so password strength will diminish. Moreover, cracking time will depend on the organisation of the dictionary, which cannot be known in advance. For example: if a dictionary is organised alphabetically, then the password Apple will be cracked sooner than the password

Zebra. Dictionary items are often organised according to frequency of use as passwords and this will on average crack passwords faster than a dictionary that is organised alphabetically. However, a perfectly optimised dictionary will have the target password as the first item. Time-to-crack prediction therefore also needs a definition of dictionary content and ordering, which is difficult to predict accurately.

Character set. This has the advantage that it is also fairly easy to understand: the more character sets a password is made of (such as lowercase letters, uppercase letters, numbers, punctuation marks, or other symbols), the harder it will be to guess. However, users may circumvent password restrictions in regular ways, which can then be used to optimise attack dictionaries. One could therefore end up with a weaker password made out of numbers and letters than one made of numbers alone, thereby making this metric misleading. For example, there are relatively few male first names. Adding a number on the end to one (ie john1) might plausibly lead to a weaker password than one of equal length composed purely of numbers (ie 53207). There are about 50,000 possible passwords using the former technique (5000 male first names times ten numbers), and 100,000 using the latter (ten multiplied by itself five times).

Password length. This measure of password strength is founded on the principle that longer length will tend to lead to more alternatives. For example, there are very few words which contain only one letter, while there are many more which contain five letters. A longer password is therefore more difficult to guess, and therefore more secure. Password length is however more often discussed with reference to random combinations of letters, under the assumption of dictionary attack. Computer systems are often configured to enforce a minimum length for passwords.

Computer systems that enforce a minimum length are also likely to enforce a ban on dictionary words as password content. This leads to allowable passwords being able to contain only random combinations of letters-nonsense strings, perhaps not even pronounceable. Such combinations of letters have classically been considered so unmemorable that they are used to measure the performance of human memory, for example by using the Ebbinghaus paradigm (see Figure 6 on page 61).

Password entropy is the most complex measure of password strength. It is a combination of character set and password length - and its use implies a time-to-crack calculation. It is an attempt to measure exactly the amount of uncertainty in a password, or the number of possible combinations of characters that could make it. In the style of cryptography, the uncertainty of a password is quoted as the number of binary digits, or bits, required to write out of the possible number of combinations of characters in the password. For example, a one-letter password has 26 possible combinations, and 26 written in binary notation has five binary digits ("11010"). A one-

letter password therefore has 5 bits of uncertainty. Password entropy is useful for comparing the relative resistance of different passwords to dictionary attack, because it measures the number of possible guesses. However, it retains the problems of the time-to-crack measure of password strength - a better dictionary or even luck could greatly reduce the time taken to actually break a password even if it has high entropy. Moreover, in practice it would be difficult to calculate the entropy of a *particular* password-because this depends on what the attacker already knows about it, which is difficult for us to judge. Password restrictions are therefore usually set so that what attackers know about the password from examining the policy or mechanism still forces them to make a very large number of guesses.

Another weakness of password entropy is that it contains little information about the memorability of the password and the load it places upon its user and other stakeholders. It is therefore not useful for comparing the impact of different password system configurations on any of the stakeholders. It focuses on a narrow definition of resistance to guessing, whilst ignoring other important goals of security, such as resource availability.

So what is a *strong password*, and what does this mean to the research? The perfect password is one that is as random as possible, and as long as possible. An assumption that the password will be used on UNIX implies that it can be a maximum length of eight characters (larger passwords can be chosen, but only the first eight characters are registered). Using a character set of all 95 printable characters the strongest password has about 53 bits. At a rate of 100,000 attempts per second (which is the assumed speed of a Pentium III computer running at 866 MHz), a dictionary attack programme would take approximately two millennia to run through all possible combinations. Security practitioners do not consider this to be a strong password however. McGraw (2001) recommends at least 10 characters (64 bits), and states that 20 characters (approximately 128 bits),

"should provide adequate security for any use" (Viega & McGraw, 2001, p.631).

At a rate of 100,000 attempts per second, it would take 1025 millennia to run through all possible combinations of this strength of password.

3.2.6 Re-allocation of function: Better encryption for passwords

There are two related areas where improving encryption in password mechanisms will ultimately *allow users to choose simpler passwords*, and so should improve

authentication system performance: resisting a precomputed dictionary attack, and resisting an online dictionary attack.

A *precomputed dictionary attack* requires the attacker to first create a database of all possible passwords along with their encryptions. To carry out the attack, the attacker must steal the password file, and then can simply look up the password by matching its encrypted version to an entry in his database (Oechslin, 2003). To defeat this attack, it is necessary to make it infeasible to calculate all possible password encryptions. The original UNIX password 12 bit *salt* was designed with this intention, by forcing the attacker to compute several thousand encryptions per password, instead of only one (Oechslin, 2003). When UNIX password authentication was redesigned circa 1977, it took more than 1 second to encrypt a single password using computer hardware of the day (e.g. a PDP-11/70 mainframe computer Leong & Tham, 1991). On this type of machine a precomputed dictionary attack would not only have taken too long, but it would have been too expensive to store all the resulting data (Viega & McGraw, 2001). Since then, computing power has increased, and storage costs diminished by orders of magnitude (see section 3.2.1), making it feasible to precompute all UNIX passwords and store them in a database. The principle of password salt is still valid, however, and has been updated. UNIX now has the BIGCRYPT program (Garfinkel & Spafford, 1996), and FreeBSD has MD5crypt (Provos & Mazieres, 1999), both of which have far larger salts, which make it infeasible to store all possible password encryptions. However, traditional UNIX does not use BIGCRYPT by default: FreeBSD is a minority operating system, and Windows, which the huge majority of stakeholders work with, does not employ any kind of salt at all (Shaffer, 2001).

There are two further approaches, which also protect against both *online and offline dictionary attacks*. These approaches are: the *pepper* approach (e.g. Kedem & Ishihara, 1999; Manber, 1996 another way of forcing the attacker to perform more encryptions per login), and the computationally harder encryption approach (e.g. Provos & Mazieres, 1999). The latter will be outlined below.

Computationally harder encryption

This technique is already being employed in FreeBSD, which uses the MD5 hashing algorithm, for a double-digit slowdown compared to the standard UNIX password mechanism (see Table 1). The OpenBSD operating system uses a modified version of the *Blowfish* algorithm in the BCRYPT program to make password encryption arbitrarily time consuming (Provos & Mazieres, 1999), and so tunable while also avoiding the false positives drawback of Manber (1996) and Kedem & Ishihara's (1999) schemes (Provos & Mazieres, 1999).

In a previous section, it was suggested that 100,000 guesses per second was reasonable for an online dictionary attack - in fact it is exceeded by the modest computer this thesis is being written on. Consider if BCRYPT had been used in the password mechanism instead: the guesses could be restricted to one per second. This is an instant 100,000 times improvement in security, which can be used to simplify a stakeholder's password 100,000 times. A dictionary word with a digit appended might take six days to crack instead of six seconds. Two words concatenated might take 32 years to crack, rather than three hours. Two words and two symbols would now be as strong as the strongest possible UNIX password: a stakeholder could use the password *no-joy!* where before she would have to use *K;8l=dT&.* This mechanism has the advantage that it can be used to keep up with conventional increases in computing power-by configuring it regularly to take longer to encrypt passwords. Over the course of a workstation's life, this may only require relatively small increases in login times (e.g. from one second to five seconds), while ensuring that simple passwords can be chosen.

Table 1 - Approximate encryption speed for different password encryption types. Speed tested on a Pentium III 866 MHz, using John the Ripper 1.6 MMX dictionary attack software.

	Windows NT lan manager	UNIX DES	FREE BSD MD5
Passwords encrypted per second	781,000	102,000	2,000
Relative computational effort	0.13	1	50

3.2.7 Security policy standards

Security policy documents describe what security should do, and some of the general ways it should be accomplished. Guidance exists about information security policy best practice in the form of various national and international standards. Two of them are ISO 17799 (BSI, 2001), a recent international standard; and FIPS 112 (FIPS, 1985), which is primarily of historical interest. They are both partly general security policy documents, and partly guidance on how to make a specific security policy for the reader's own organisation and information systems. Their password recommendations are outlined in Table 2. Section 3.2.8 will describe the policy families that these

individual policy statements derive from, describing their function and assumptions. UCL's password policy is given in Table 2. It has not been possible to reproduce BT's security policy for comparison, as permission could not be obtained. However, Section 7.3 will summarise users' reports of some of these policies.

Table 2 - Local and international standards for password use

UCL	ISO 17799	FIPS 112 medium security	FIPS 112 high security
Ownership			
o Individual	o Use unique user IDs so that users can be linked to and made responsible for their actions. o The use of group IDs should only be permitted where they are suitable for the work carried out;	o Individual	o Individual
Disclosure			
o Not mentioned in policy.	o Group passwords should not be made known to users outside the group. o Individual passwords should not be revealed to other individuals. o Passwords should be written down only if these records are securely stored. o Where appropriate, paper and computer media should be stored in suitable locked cabinets and/or other forms of security furniture when not in use, o Passwords should never be stored on computer system in an unprotected form	o If passwords must be written, they should be protected in a manner that is consistent with the damage that could be caused by their compromise.	o If passwords must be written, they should be protected in a manner that is consistent with the damage that could be caused by their compromise.

UCL	ISO 17799	FIPS 112 medium security	FIPS 112 high security
Content			
o At least 7 characters long o at least three of the following different types of characters: lowercase, uppercase, numbers, symbols i.e. ! % ^ * ( ) _ + - = " ` : ; < > , ? | / \ @ $ & [ ] { } BUT NOT £ o Don't base your password on a dictionary word, proper name, personal details such as address, post code, phone number or department name o Don't use foreign language words o Don't use names of bands, asteroids, cartoons, movies, TV programs, swear words, Shakespearean or Monty Python characters, or science fiction jargon o Don't use any of the above with I's, L's and O's transposed with ones and zeros o Don't merely disguise a word by using repetition or reversal or by adding a number to the beginning or end of it o Don't use anybody else's user name or userid as a password	o Source: User selected where appropriate o Passwords should have a minimum of six characters. o Passwords should contain no consecutive identical characters. o Passwords should not contain only letters or only numbers o Passwords should be easy to remember o Passwords should not be easy to guess or based on easy to discover personal information	o Source: System generated and user selected o Length Range: 4-8 o Content: U.C. Letters (A-Z), L.C. Letters (a-z), and digits (0-9)	o Source: Automated password generator within the authentication system o Length Range: 6-8 o Content: Full 95 character set
Compartmentalisation			
o At user's discretion	o Passwords should not be reused within a twelvemonth. o If a user must use more than one	o classified passwords must not be used on terminals that are not authorized for data at the level of the password	o classified passwords must not be used on terminals that are not authorized for data at the level of the password

UCL	ISO 17799	FIPS 112 medium security	FIPS 112 high security
	password protected system, then the user should be allowed to use one single but high-quality password	o The presence of both single-level and multilevel terminals on a system may indicate the need for passwords at each security level.	o The presence of both single-level and multilevel terminals on a system may indicate the need for passwords at each security level.
Three strikes			
o Drop connection after 3 strikes	o It is recommended that login attempts should be limited to three. o Consider recording unsuccessful attempts; o Consider forcing a time delay before further log-on attempts are allowed or rejecting any further attempts without specific authorization; o Consider disconnecting data link connections;	o In some instances, it may be desirable to count the number of unsuccessful login attempts for each user ID and to base password expiration and user ID locking on the actual number of failed attempts. (Changing a password would reset the count for that user ID to zero.) o For example, the password could be identified as expired after 100 failed login attempts, and the user ID locked after 500.	o In some instances, it may be desirable to count the number of unsuccessful login attempts for each user ID and to base password expiration and user ID locking on the actual number of failed attempts. (Changing a password would reset the count for that user ID to zero.) o For example, the password could be identified as expired after 100 failed login attempts, and the user ID locked after 500.
Access rights reviewing / Password expiry / Password lifetime			
o Lifetime: 4 months	o Users' access rights should be reviewed and six-month periods, and every three months for users with privileged access rights.	o Lifetime: 6 months	o Lifetime: One month
Distribution			
o Via paper slip from the helpdesk on presentation of ID card.	o The use of third parties or unprotected (clear text) electronic mail messages should be avoided. o Users should acknowledge receipt of passwords.	o Terminal and special mailer	o Registered mail, receipt required; personal delivery, affidavit required
Time outs / screen locking			
o Not mentioned	o The policy should take into account the information security	o 10 minutes of terminal inactivity.	o 5 minutes of terminal inactivity.

UCL	ISO 17799	FIPS 112 medium security	FIPS 112 high security
	classifications, the corresponding risks and cultural aspects of the organisation. o Personal computers and computer terminals should not be left logged on when unattended and should be protected by key locks, passwords or other controls when not in use. o Limit the maximum and minimum time allowed for the log-on procedure. If exceeded, the system should terminate the log-on;		
Login feedback			
o Not display passwords on the screen when being entered; o Not provide help messages during the log-on procedure that would aid an unauthorized user; o If an error condition arises, the system should not indicate which part of the data is correct or incorrect	o Not display passwords on the screen when being entered; o Not display system or application identifiers until the log-on process has been successfully completed; o Not provide help messages during the log-on procedure that would aid an unauthorized user; o If an error condition arises, the system should not indicate which part of the data is correct or incorrect	o Non-printing keyboard and masked-printing keyboard	• Non-printing keyboards

3.2.8 Policies that harden password mechanisms

Though alternative and improved authentication technologies are available (see Sections 3.1.6, 4.2 and 4.3), they are not frequently employed. In their place, organisations prefer to mandate strict password policies. This allocates the security function away from the technology and onto the user. Individual password policies (such as ISO 17799's listed in section 3.2.7) can be grouped into families. Some of the policy families are as follows:

1. Ownership
2. No disclosure / no writing down
3. Strong password content
4. Compartmentalisation of passwords
5. Three strikes
6. Password expiry / access rights reviewing
7. Distribution
8. Time outs
9. Login feedback

Ownership

Most security policies mandate individual ownership of passwords as opposed to group ownership of passwords , mainly because in theory this creates an audit trail that can be tied to the individual user, and so makes her more accountable. Individiual passwords make it easier to detect and react to wrongdoing, or other security related events.

In practice, the effects of this policy may be less beneficial. Ownership of passwords has been found to have a significant negative effect on the acceptibility of security policy among users, and their compliance to it. Adams et al. (1997) found that when group passwords were applied in a situation where users believed individual passwords were appropriate, for example group passwords for individual e-mail, then users would try to subvert the safeguard. This was also true if individual passwords were applied in situations in which users believed group working was more appropriate.

In theory, a policy mandating group ownership of passwords has the effect of trading off two security goals against each other: improving availability at the expense of audit. Where there are group passwords, individuals within the group can help each other if a password is forgotten or confused and reduces the frequency with which helpdesk support is required compared to individual passwords. The effects of group versus individual password ownership on availability has not been systematically studied.

No Disclosure

Disclosure policies are designed to limit knowledge of passwords to the password's owner, i.e. the user. Only under special circumstances are people allowed to know passwords they do not own, such as when an administrator gives an account to a new user, or gives a new password on an existing account. Under these conditions, the passwords known to both parties are temporary, and must be changed by users on their first use.

Policies that forbid the writing down of passwords by users may be classed as disclosure policies, as writing a password on a post it note attached to your workstation is a clear example of disclosure. An outright ban risks being infeasible, as users may feel they are unable to comply and also do the work they are required to, as there is no backup store of the password for the user if her memory fails. If an outright ban is imposed, then users will have no alternative when they forget passwords but to use a helpdesk. Allowing the writing down of passwords should therefore reduce helpdesk use, but adds the risk that these passwords are available to an attacker.

Allowing passwords to be written down *if they are securely stored* should reduce helpdesk costs but requires an organisation to provide infrastructure suitable for secure storage. In the short term, this may cost more to implement than the savings received. Moreover, users may view the secure storage policy as unreasonable if they do not believe there is significant risk of attack or punishment. Adams & Sasse (1999) argue that negative perceptions of any particular security policy may weaken the user's respect for others, and result in a generally lower level of security conscious behaviour than is desirable.

Content

These policies require the user to choose a minimum strength of password content to make up for poor encryption and password mechanisms (see Sections 3.1.1 and 3.1.6). This policy has the disadvantage that stakeholders are required to choose difficult to remember password content.

This policy assumes that an identity theft will occur through the use of a dictionary attack. This implies that the attacker is willing to commit months of time to an attack (when other techniques are far quicker), and that he has access to the target system's password file. Access to this file implies access to files on the target computer system, and if the system employs shadow password files it implies the attacker already has the highest level of access to the system and the data it contains. If the attacker already has the highest level of privilege on a system their only major benefit

in cracking passwords is that they may be used to access other systems if there is weak compartmentalisation of passwords.

Compartmentalisation of passwords

Compartmentalisation of passwords comes in two forms. *Concurrent compartmentalisation of passwords* (the author's term) requires users to have different passwords on different systems, so that the compromise of a password on a particular system will not lead to the compromise of other systems.

This policy assumes that attackers will want to break into several computer systems in an organisation rather than target individual systems. It assumes that users will tend to use the same password on several systems if allowed to do so, and so a password used on a badly defended system could be stolen to allow entry into a better defended system. Furthermore, it assumes that the additional burden placed on users and the system more widely of having separate passwords is outweighed by the benefit of limiting the damage of a successful password-related attack.

Concurrent compartmentalisation of passwords is sometimes dovetailed with *password expiry* to become *historical compartmentalisation of passwords*, preventing users from having the same or a similar password to one they have had previously on that system. To enforce this, the authentication mechanism must record a *password history*, which acts as a blacklist. The list is usually of a fixed length, for example banning the last 6 passwords chosen for a system (but not the 7^{th}). On a system with this policy, if a user is determined to keep the same password after it expires she can do so by changing the password so many times that the password she wants drops out of the blacklist.

This policy is based on several assumptions that are questionable. It assumes that users recycle passwords - which is reasonable. However, it further assumes that the computer system's historical password store is defended as well as its current password store is. It further assumes that users would prefer to spend the effort of thinking of a new password over the effort of flushing out the password history so that they can pick the same password, or to having a repertoire of passwords that is slightly larger than is checked by the password history mechanism. It further assumes that the benefits of the additional security outweighs the extra costs to the user (and the password authenticationsystem, and the organisation) of the user's behavioural response to the password history mechanism. For example, if a user responds by keeping a password repertoire then she will have to expend more effort managing passwords than previously. This policy assumes there will be a net increase in security, as the users' behavioural response to the mechanism is not expected to lower

the overall security achieved. For example, it is assumed that users will not start to write down passwords in a diary so they can keep track of their new password repertoire. If users already keep a personal password repository, it is assumed that increasing the size of the repository to accommodate the repertoires will not make the users' repositories more vulnerable to attack.

Both types of compartmentalisation of passwords can help prevent attackers compiling lists of functioning passwords, but have the disadvantage of multiplying the amount of password content users have to remember.

Three strikes

The *three strikes* policy limits the number of online attempts at authentication, after which time some penalty is imposed. This may be a short delay before the next authentication can be attempted, but is more usually the locking of the account. The number of attempts is often set to three per session (hence the baseball metaphor - three strikes and you're out) though some security specialists recommend five per session, or more complicated arrangements (Viega & McGraw, 2001). This has the disadvantage that stakeholders are given few chances to correctly recall their unmemorable passwords with a penalty for failure ranging from minutes to hours of lost productivity as the users arrange to have their newly suspended computer accounts reinstated.

This policy also makes a number of questionable assumptions. It assumes that an identity theft will occur through an online password guessing attack. It assumes therefore that the account protected by the policy has a weak password, as it is extremely unlikely that online guessing could break a strong password in reasonable time. However, accounts protected by a three strikes policy are likely to have compulsory strong passwords as well, as both these policies are seen as necessary for high security. It is also assumed that authorised users can remember their passwords within 3 attempts whilst conforming to all password policies, or otherwise that the additional burden of dealing with cases of valid users being denied access to necessary resources is less than the burden of dealing with a successful password related security breach. It further assumes that the security surrounding password resetting at the helpdesk is equivalent to the security offered by the password mechanism itself. However, this may not be the case as: users will probably be authenticated at the helpdesk by a password, and users are more likely to keep their helpdesk password written down and available (and so available to attackers) than any of their other passwords because it is the only one they really depend on. The policy also assumes that prevention (of attacks) is better than detection and punishment (of

attackers) , and that helpdesks are the best place to decide the authenticity of users. The people being authenticated are often far removed from the people who work in helpdesks, whereas more local managers and colleagues are in a much better position to detect an identity theft in progress.

Access rights reviewing

Access rights reviewing protects against *ghosting,* where people who have left or moved within an organisation can freely roam its networks using their old accounts which have not yet been suspended (Hook, 2002). It is effected in two ways: manually by administrative personnel, and through automated techniques. The automated form suspends or deletes an account if it has not been used after a certain amount of time (3 months is recommended for high security by ISO 17799), forcing the account holder to seek authorisation to regain access to it.

The automated checks can reduce the frequency with which the costly manual check is required, or replace it entirely. If the automated check is relied upon, this assumes that, for example, 3 months of access is a good compromise between preventing attacks and incorrectly locking out current and authorised users. A more complex automated system tied into personnel department processes could drastically reduce these risks, but would also be more expensive to purchase and may have increased maintenance costs. It further assumes that the burden of proving an account is or is not still needed should be placed upon the users, as opposed to their managers, system administrators or the personnel department.

Password expiry

Password expiry/Password lifetime sets a time limit on the validity of a password. When this time has expired, the user must select a new password. This protects stakeholders against the undetected compromise of a user's password (FIPS, 1985), and the abuse of *ghost accounts* - accounts which give access to users who are no longer authorised to have access. This is usually because the users have left the organisation, but their accounts have not yet been deactivated. It is a particularly high *risk* when users are sacked, as they are likely to bare a grudge.

Early versions of the policy (e.g. FIPS, 1985) based the password lifetime on an equation relating the speed with which guesses could be made, the password's search space (see *password entropy* in section 3.2.5), and the level of risk that could be accepted for the password being broken during the password's lifetime.

More recent versions of this policy (e.g. ISO 17799 - BSI, 2001) do not advocate the calculation of password lifetime, but instead recommend baseline lifetimes according to the security required. Strong versions of the *Password expiry* policy require passwords to be changed every month or sometimes more frequently. Medium strength applications require three-month intervals, and the weakest versions require password changing at one-year intervals. The disadvantage of this policy is that it multiplies the number of password stakeholders are required to remember, and so adds to the administrative costs of password ownership.

Time-outs and screen locking

Time-outs and screen locks prevent unauthorised use of computing resources when authorised users are away from their computers, by activating some mechanism that renders the computer temporarily inoperable. In some circumstances, it may be suitable to rely on users activating the mechanism manually. In other circumstances policy can mandate automatic activation of this mechanism after periods of mouse and keyboard inactivity. Care must be taken when setting the trigger interval - too long and the attacker is given the opportunity he needs, whilst if it is too short then the mechanism will trigger whilst users are actually present and doing work (but not using the keyboard or mouse). As trigger intervals shorten password use increases. With this increase so too does the risk that users will find the policy more of a hinderance than a help, and so become disinclined behave securely (cf. Adams & Sasse, 1999).

Distribution

Password distribution policies improve security by reducing the opportunity of attackers to intercept passwords as new or replacement passwords are being distributed to users. Distribution policies can have profound effects on the costs of password problems, as the solution to most of these is a password reset, which requires password distribution. For example, the distribution policy that allows new passwords to be distributed over the phone will allow much speedier recovery from password problems than a policy that requires distribution by registered post-which may take days. Onerous distribution policies may dramatically impact users' compliance to password related policies, particularly if these increase the likelihood of a password reset (and distribution) being required.

Login feedback

This family of security policies make it more difficult for an attacker to gain useful information when faced with a login prompt, or when observing someone else using a login prompt. These policies trade reduced availability for increased confidentiality: The less information that is displayed during login the less information there is available to authorised users that may help them to prevent and recover from password errors. For example, fips 112 requires a non-printing keyboard for high security situations (see Table 2). This means that users will not get any visual feedback whilst typing the password, and so will not even be able to tell if they have typed enough characters. This policy is often used as a baseline safeguard, i.e. may be used where it is not beneficial compared to more tailored safeguards (Cho & Ciechanowicz, 2001).

3.2.9 Other attacks

Password security can be breached in other ways, which are summarised in Table 3. These methods for breaching password security have been put into four groups, most of which are methods for avoiding the password mechanism.

The first group of attacks relies on the password or information that the password is protecting being made directly available to the attacker in some way. The second group involves different kinds of observation of systems in use. The third group deploys software by the attacker that is specially designed to breach security. The third group of attacks exploits weaknesses in the design of computer software, which may be done manually or using malware. In some cases, an attack may fit into more than one group. Examination of these attacks is beyond the scope of the thesis. However, it should be noted that of the 26 attacks listed, only one is prevented by strong password policy (dictionary attack/password crackers); the three strikes policy is even tighter in its effect-online dictionary attack/password crackers. These two safeguards are ineffective against the other risks.

3.2.10 Summary

Recommendations about general password content (low security - 4-6 digits; medium security - 4-8 of upper and lower case letters and digits; high security - 6-8 of all the 95 printable characters; FIPS, 1985) are usually designed under the assumption of an off-line dictionary attack by a government agency on the output of a standard UNIX password mechanism. Given these assumptions, the user is allocated the task of choosing a password that will resist brute force cracking by several of the world's

Table 3 - Other means of breaching password security

Disclosure • Writing down • Sharing • Authentication logs • Social engineering • Bribery • Blackmail/coercion • Publicity material • Desk surfing • Theft	*Shoulder surfing / observation* • Unaided • Cameras • Keystroke logging • Inference • Man in the middle • Signals intelligence • Interception • Wire tap • War driving • Promiscuous mode ethernet card • Own a router • Tempest • Traffic analysis
Malware • Trojans • Viruses • Worms • Security scanners • War diallers • Privilege escalation aps • Password crackers/ dictionary-attacks/ guessing • Packet editors	*Exploits* • Buffer overflows • Windows messaging flaws • Backdoors • Core dump

fastest supercomputers for *several thousand millennia*. It would seem more effective and efficient to allocate this task to a computer.

However, of 26 types of attack against computer security listed in Table 3, these password recommendations would only protect against one (see section 3.2.9). Many of these 25 other attacks are far easier to mount, cheaper and more quickly successful than an off-line dictionary attack of the type assumed. These other attacks are therefore more likely to occur (cf. Mitnick & Simon, 2002; Schneier, 2000; Winkler, 1997).

On the whole, risks are managed with questionable assumptions and inappropriate technology (sections 3.2.1 and 3.1.8) to protect against attacks which are unlikely (cf. Mitnick & Simon, 2002; Schneier, 2000; Winkler, 1997). The result is that users are forced to chose unnecessarily complicated passwords (see section 3.2.5), have too many of them, and change them too often.

3.3 Human memory & skilled performance

Password systems require users to memorise passwords and recall them when logging on. Human memory is therefore the most important user characteristic on the human side of the authentication worksystem if the user is honestly trying. There is a huge body of research on human memory, but the most important issues related to passwords can be summarised as follows (Schacter, 2001):

- Recognition of a familiar item is easier than unaided recall,
- People cannot 'forget on demand' — items will linger in memory even then they are no longer needed,
- Memory decays over time — this means people may not recall an item, or not recall it 100% correctly,
- Frequently recalled items are easier to remember than infrequently used ones, and retrieval of very frequently recalled items becomes 'automatic',
- Items that are meaningful (such as words) are easier to recall than non-meaningful ones (sequences of letters and numbers that have no particular meaning),
- Distinct items can be associated with each other to facilitate recall — however, similar items compete against each other on recall.

The sections that follow will go over these in more detail. Firstly, the differing power of recognition, recall, and other routes to memory will be outlined (section 3.3.1). The next section (3.3.2) will outline the interference theory of forgetting. Following this will be an outline of practice effects (section 3.3.3). Section 3.3.4 will note that the strength of memory depends on what is being remembered. Section 3.3.5 brings together the work of the previous sections to outline the practical issue of human error, and shows what role human error has in computer security.

3.3.1 Recall and Forgetting

The accuracy of human memory varies widely with the circumstances of recall and the type of remembering used. This may be exploited by the developers of authentication by knowledge systems, who can choose the most appropriate type of remembering for their design. A partial list of the different types of recall includes:

- ***Unaided recall***
- ***Cued recall***
- ***Recognition***

- ***Fragment & stem completion***
- ***Matching***

Unaided recall is used in traditional password mechanisms. Users are asked to retrieve an item (password) from memory that was either shown to them in the past, or that they had chosen themselves. Unaided recall is one of the least accurate types of memory - its accuracy decays rapidly with time compared to the other forms.

Cued recall is used in *associative* and *cognitive password* mechanisms (section 4.2.2 & 4.2.3). Users are shown prompts which are associated with the password or passwords, that users are then required to recall. Cued recall can retain its accuracy better with the passage of time than unaided recall.

Recognition is used in authentication mechanisms such as Passfaces™ (RealUser, 2004). Users are shown a selection of items, and are required to pick the correct item or items to login. The accuracy of recognition surpasses cued recall with the passage of time.

Fragment completion has been used in the *personal entropy* mechanism (Ellison, Hall, Milbert, & Schneier, 2000). Fragment completion is similar to the game of hangman, where the password/phrase/sentence is displayed with parts missing throughout. Users would be required to complete the parts that were missing. *Stem completion* has also been used in the *personal entropy* mechanism, where the first part of a password is presented, and the user has to supply the ending. These two forms of remembering can show retention of information better than recognition.

Matching is not currently used in any authentication mechanisms. The user would be shown a series of two stimuli, and asked to remember the relationship between them.

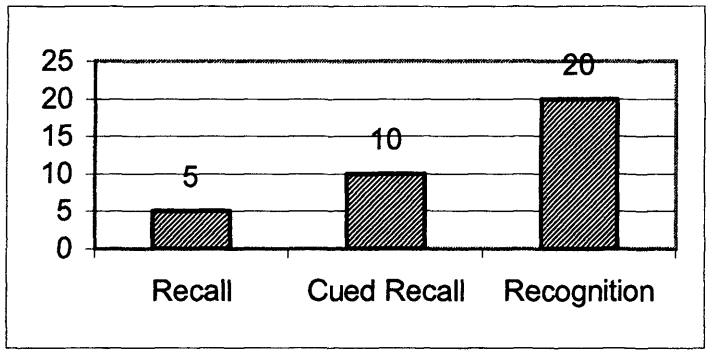


Figure 5 - Ratio of number of items correctly remembered using different measures of memory

The general effect of type of remembering on retention rates (how many target stimuli or how much of a stimulus is remembered after a particular interval) is displayed in

Figure 5 above (adapted from Parkin, 1993 p.74). Unaided recall returns the fewest correct responses, and recognition gives the most. The size and direction of these ratios is not fixed, but depends on the circumstances of the remembering, including the nature and number of the target and distractor stimuli. Authentication mechanisms therefore have to be implemented appropriately to the form of remembering employed to achieve optimum accuracy. For instance, unaided recall can be made more accurate than recognition if the targets and distractors are very similar (Baddeley, 1997); therefore in a recognition based mechanism targets and distractors should be made very different otherwise the benefits of recognition over recall will not be achieved. A psychological mechanism for this phenomenon will be discussed in the Human error section (3.3.5) below.

Unaided Recall

In his paradigm-creating work, Ebbinghaus (1885, cited in Parkin 1993) described a *forgetting curve* for lists of nonsense syllables. Because they were nonsense, what was revealed was a genuine property of the memory system, uncontaminated by previously held knowledge, frequency of words in language, or other extraneous variables. Ebbinghaus studied himself as a subject using the *savings method*, and produced the following curve - Figure 6 below (adapted from Parkin, 1993 p.4)

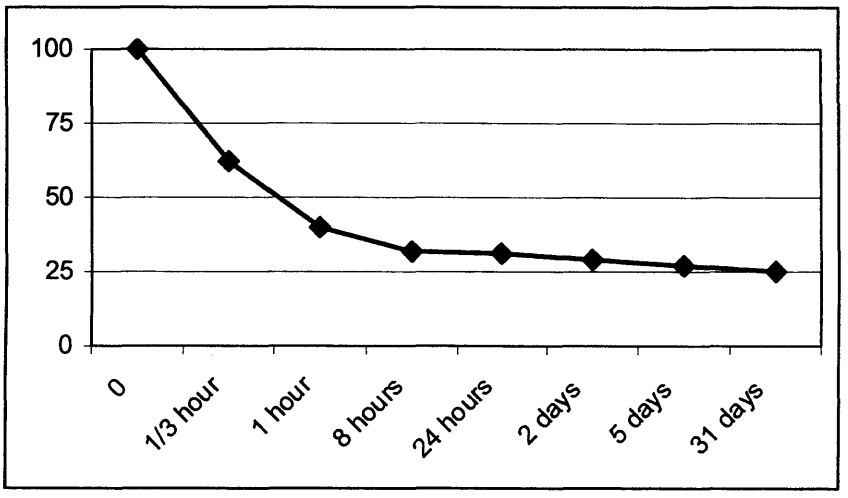


Figure 6 - Mean percentage savings made with nonsense syllable lists

The curve shows that memory for nonsense syllables decays quickly at first and tends towards an asymtote. After 8 hours, approximately only 40% savings are made - 60% of the effort of original learning must be expended to reach the same accuracy of recall.

Ebbinghaus was recalling lists of 13 items, while passwords are individual targets. His experiment demonstrates that memory for nonsense material quickly becomes less

accurate when not practiced. The target stimuli are reasonable approximations of passwords. If the user learns a random-character (i.e. strong) password and then does not practice it, this study suggests she will be liable to forget it. If she does practice, the study has no direct relevance.

Cued Recall

The amount of improvement in memorability offered by *cued recall* over unaided recall depends on the strength of the association between the cue and the response. Strong associations lead to better recall (Parkin, 1993).

Population stereotyped word association pairs would not make good *challenge-responses* (such as "leak", "beak" used in the strong condition of Parkin's (1981) experiment), because any attacker would be liable to guess them. However, similar effects have been demonstrated with semantic relatedness (Reason, 1990). This leaves the way open for user selected *challenge-response* items which are memorable because of some personal semantic relatedness, but hard for an attacker to guess without intimate knowledge of the victim. On the other hand, a spouse, family member or close friend might share all or some of that knowledge. This personal and difficult to guess knowledge has been termed *personal entropy* (Ellison et al., 2000).

Recognition

Recognition is the distinguishing of targets from distractors. Recognition accuracy depends on the nature and number of target and distractor stimuli. If there are many targets and few distractors, successful recognition may not be distinguished from plain guessing. This ratio would not be useful in a *challenge-response* system, because it would allow impostors to successfully log into a system by guessing. A high ratio of distractors to targets would reduce this risk. However, the accuracy of recognition steadily decreases as the ratio of distractors to targets increases (Baddeley, 1997). A balance must be struck between the risks of letting attackers in and denying access to valid users.

The accuracy of recognition also decreases with increasing similarity between target and distractor stimuli (Baddeley, 1997). This finding suggests that recognition-based authentication systems should have targets and distractors that are not similar, and so easily distinguishable by the authorised user.

Implicit versus Explicit

Human memory can be divided into two separate kinds according to the way in which it is tested (Parkin, 1993). So far, only explicit memory has been discussed.

Explicit memory can be defined as a task which requires the subject to recollect a previous learning event such as the presentation of a word list. ***Implicit memory,*** however, refers to any memory task in which a subject's memory for a learning event ***is tested without specific reference to that event.*** Implicit memory can be tested several different ways:

1. ***Electrodermal response,*** changes in skin conductance for a target stimulus relative to previously unseen distractor items.
2. ***Fragment completion***, a procedure where the subject must fill in the blanks in a word, similar to a partially completed hangman puzzle.
3. ***Stem completion***, the first few letters of a word are presented, and subjects asked to complete a word from it.
4. ***Picture completion priming***, completing a partially drawn picture to resemble a previously presented target?
5. ***Decisions about stimuli***, subjects are required to answer questions about stimuli, with something about their answers revealing whether or not they have remembered the target information.
6. ***Perceptual identification,*** subjects show increased accuracy in identifying previously seen target stimuli compared with new stimuli.

It has been repeatedly demonstrated that implicit memory is far more resilient than explicit memory (Parkin, 1993). For example, in the following study fragment completion is relatively unaffected by the passage of time, whereas recognition drops off substantially. Implicit memory tests might therefore form the basis of authentication-by-knowledge mechanisms that have the best retention of the target 'secret' over extended periods.

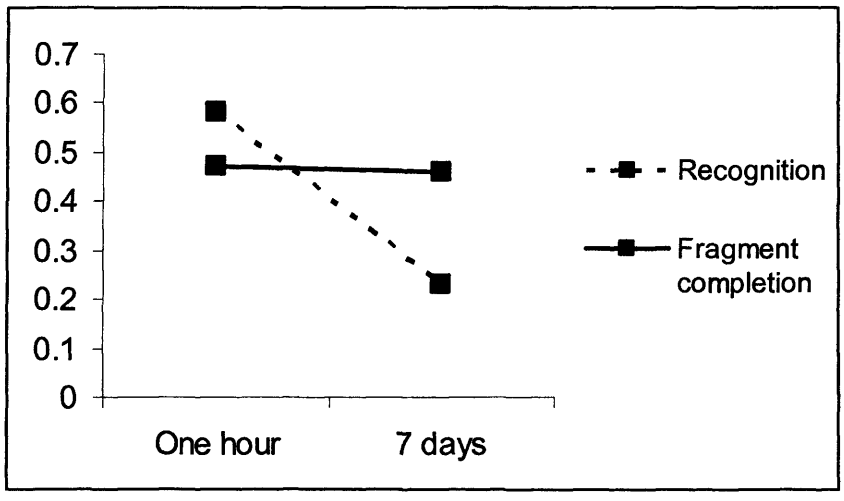


Figure 7 - Accuracy (frequency of correct response) of memory for words in a word list task. From Tulving et al., 1982, page 339

A related phenomenon is known as *implicit learning*: learning that occurs without the subject being able to explain how. *Artificial grammar learning* is one of two techniques that has been employed to investigate this type of learning. Subjects are asked to learn strings of letters generated by synthetic grammar which defines what letters are permissible and the sequences between them. Next they are told that the strings of letters are bound by rules, and are required to categorise new strings of letters as grammatical or ungrammatical. There are many demonstrations that subjects can learn to do tasks such as this even though they cannot explain the rules they are using (Berry & Dienes, 1993; Reber, 1989), and are still able to do than two years after the experiment (Allen & Reber, 1980).

Implicit learning exhibits the same durability as implicit memory, and offers the potential extra security benefit of not being easily divulged to a second party. However, implicit learning could only be used for authentication in unusual circumstances, as it would require extensive effort and time for the user to enrol (undergo implicit learning), and would be appropriate only for very high security or very high retention periods. It is likely that both these could better be served by biometric or token-based solutions. The only circumstance where implicit learning would be better would be where divulging cannot be tolerated.

3.3.2 Theories of Forgetting

Why does forgetting occur? There is much conflicting evidence (Baddeley, 1997). The rival hypotheses have been called *trace decay theory* and *interference theory*. Trace decay theory postulates that memories are eroded by the passage of time (Baddeley, 1997), whilst interference theory states that earlier memories become obscured by later ones.

Interference

There are two types of interference: retroactive and proactive interference (Baddeley, 1997). *Proactive interference* occurs when previously learned material interferes with the recall of later material (Baddeley, 1997). It will be discussed in detail in section 3.3.5 on human error

Retroactive interference is produced by later learning (Baddeley, 1997). Typically, the experimental group learns list 1, then list 2, and then is tested on list 1. Decreased performance is displayed in recalling list 1 compared to a control group that rested instead of learning list 2 (Baddeley, 1997).

Figure 8 below shows this effect on the recall of sentences (adapted from Slamenka, 1960). The number of interpolated lists is shown on the x-axis, and the mean percentage of words correctly recalled on the y-axis.

This experiment lacks ecological validity for the majority of password users, many of whom would not learn 4 or 8 passwords in a row. However, it does support the model proposed by Reason (1990), where similar targets interfere with each other, and the greater their number the greater the interference (more interpolated lists = worse memorability). It also shows that frequency of use increases a target's propensity to come to consciousness (more training trials = better memorability).

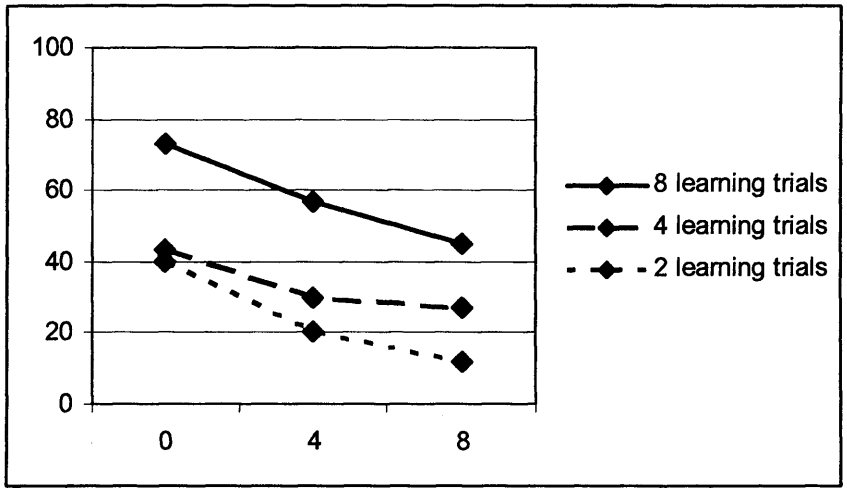


Figure 8 - Effect of retroactive interference on the retention of prose.

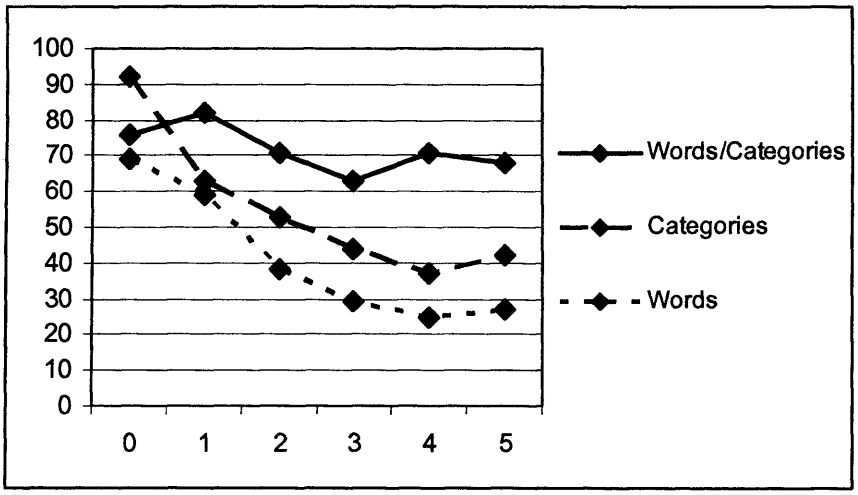


Figure 9 - The effect of retroactive interference on the recall of categorised word lists

Challenge-response systems use lists of target stimuli. There is therefore a risk that users of several different *challenge-response* systems would suffer reduced

memorability of their target pairs. This risk can be lessened. Figure 9 above (adapted from Tulving & Psotka, 1971) shows a retroactive interference experiment, where each list of words was based on a separate category. The number of interpolated lists between target list learning and testing are on x-axis, and percent recalled on y-axis. Participants showed the expected reduction in recall accuracy with increased numbers of interpolated lists when asked to recall either the categories or words of the lists. However, when cued with the categories, the participants were well able to recall the items in the lists.

This result suggests that *associative password* systems should use pools of target pairs that fall naturally under categories, and the systems should prompt users with the category when asking for a response. This effect has been replicated in studies of *challenge-response* authentication (Zviran & Haga, 1993).

3.3.3 Learning and Practice

While a single experience may lead to learning, the acquisition of new information usually requires practice (Baddeley, 1997). There are two general principles of practice (Baddeley, 1997):

- the *total time* hypothesis
- the *distribution of practice* principle.

The *total time hypothesis* states that the more time spent practising, the better the subsequent learning. The *distribution of practice hypothesis* is that it is generally better to spread out the practice over time, instead of doing it *en masse* (Baddeley, 1997). This second hypothesis holds true even for the inter-item repetition interval - the time taken between practising the same item. Combined with the retrieval practice effect - that successfully recalling something increases the likelihood of it being remembered beyond the simple effect of practice - these hypotheses show that password use tends to follow the ideal schedule of practice. Using passwords is likely to improve their memorability considerably. This means that regular users of passwords should experience far fewer recall problems than the participants in the experiments of conventional, associative and cognitive passwords (e.g. Bunnell, Podd, Henderson, Napier, & Kennedy-Moffat, 1997; Zviran & Haga, 1990, 1993), and that their results should be generalised with caution.

The total time hypothesis only holds true if the learner is attending to the stimulus; Baddeley, 1997). However, the learner does not have to actively intend to commit the item to memory, as long as it is processed sufficiently. The *levels of processing* theory (Craik & Lockhart, 1972 cited in Baddeley, 1997) describes a very robust phenomenon

where categorising the visual characteristics of target words leads to poorer retention than a deeper and more elaborate semantic processing of the words.

In password use, little or no learning will result from merely copying the passwords from paper into the screen. The passwords must be processed by the user at a deeper level, i.e. their meaning must be attended to. This is a problem for strong *passwords* (see section 3.2.5 for a definition) - they are meaningless. Inherently meaningless targets can be imbued with meaning (and *activation* - see section 3.3.5) through *mnemonic techniques* that associate targets with more intrinsicly memorable items; usually accompanied by deep processing of the new group of items. For example, creating a story from the characters in a strong password, and invoking all the senses in imaging it - hearing the sounds, seeing the colours and details, smelling the smells, feeling the textures and emotions, etc. Mnemonics are effortful, and require strong motivation which many subjects find difficult to sustain (Schacter, 2001). Here is where *challenge-response* systems have a theoretical advantage:they use word-associations (section 4.2.2) or autobiographical memory items (section 4.2.3) as targets, which already have *high activation* (section 3.3.5), are more meaningful, and lend themselves to processing at deep levels and learning.

3.3.4 Specialised memory systems

This section notes that there is evidence that different things are remembered by different, highly specialist cognitive systems (Cohen, 1996). These different memory systems have individual properties that distinguish them from the others. Some of these properties might be exploited to produce an authentication system with fewer or different memorability problems than passwords. This might allow side-stepping of the central dilemma of passwords: the mutual exclusiveness of memorability and crackability.

Authentication mechanisms have been created that rely on memory for pictures (Jermyn, Mayer, Monrose, Reiter, & Rubin, 1999) and faces (RealUser, 2004). The author defines these as out of the thesis scope.

3.3.5 Human Error

Reason (1990) unified the literatures of skill acquisition and use, memory, and human reliability in a treatment of human error. This treatment is widely respected and taught and has been successfully applied to accident prevention programs (Groeneweg, 2002). It offers a potential framework to explain and predict password system performance, and has thus been chosen to form one of the major theoretical underpinnings of this thesis.

In Reason's (1990) Generic Error Modelling System (GEMS), mental operations can happen under one of two conditions: attentional or schematic control modes.

The *attentional control mode* is related to consciousness and working memory. It is slow, limited, effortful and difficult to maintain for more than a short time. This control mode is used for setting future goals, selecting the means to achieve them, monitoring progress towards the goals and detecting and recovering from errors. When a user has to think what her password was, or try and reconstruct it, she is using this mode of control.

The schemata of *schematic control mode* are specialised processors (like software daemons) that are expert on some useful regularity of the world. The schematic control mode processes familiar information quickly, in parallel and without conscious effort. No limit has been found to the number of schemata that may be stored, or for the length of time of their retention. When you are in "automatic pilot", you are under schematic control.

Activation of schemata

Schemata are triggered when they reach a certain level of activation. The activation can be from two groups of sources (see Figure 10 below): general and specific activators (Reason, 1990).

Specific activators are usually required before any schemata can be put into action. The most important activator is the intention to do something. A plan is the description of intended action. Each of the described actions call up and activate a schema, which activates its subordinate schemata, etc. With repetition of the intended activity, the descriptions get *chunked* into higher-level schemata, and so the plan becomes a smaller list of descriptions of actions. As the number of repetitions gets larger, so does the process of abstraction and implementation of the intended action becomes increasingly automatic.

Most of the time this is advantageous. However, to change an established routine requires attentional intervention. If the attentional control mode is distracted or preoccupied at the critical moment, an absent minded slip occurs, and a user could type the wrong password.

While specific activators may push a schema to "critical mass", general activators provide a "background radiation" which lower the amount of specific activation needed for a schema to "ignite". In Reason's model, increased memorability would be described as giving a general activation to a password schema, which reduces the amount of specific activation required to trigger it.

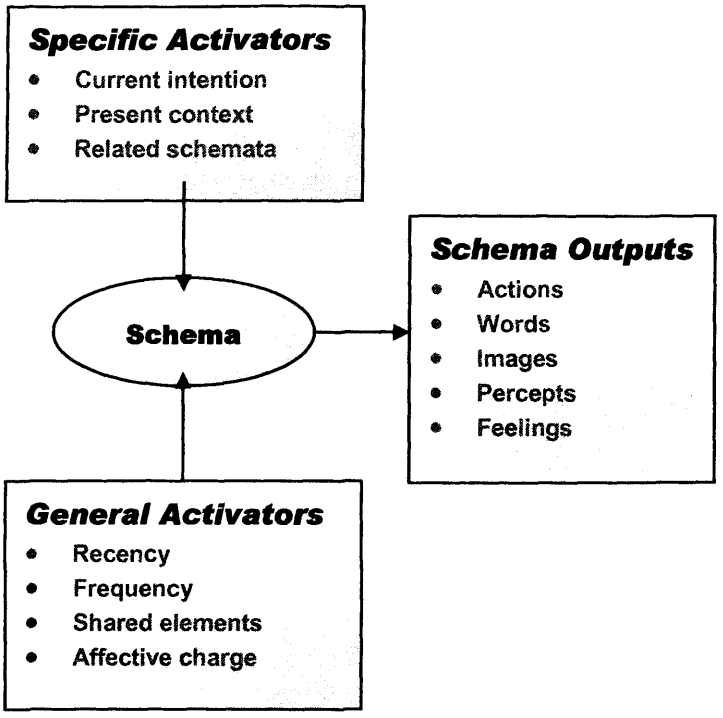


Figure 10 - How schemata are brought into play

Of the general activators, frequency of prior use is probably the most potent (Reason, 1990). Its effect is so strong, that only contextual cues may be required to trigger schemata, particularly in very familiar environments.

Error Forms

These are varieties of failure that exhibit themselves at all levels of problem solving and goal directed behaviour, and their ubiquity is evidence that they are universal processes in cognitive activities. *Similarity matching* and *frequency gambling* are automatic retrieval processes by which knowledge structures are located and their products delivered to consciousness (thoughts, words images, etc.) or to the outside world (action, speech or gesture). These two retrieval processes are what cause the error to happen. The actual sequence of events that lead to the error, and the error's effect are termed *error types* and will be discussed later. The effect of the two processes can be summarised as:

"When cognitive operations are underspecified, they tend to default to contextually appropriate, high-frequency responses." (Reason, 1990, p.91).

In the domain of passwords, the intent to use an infrequently used password (see section 6.3.1 for a definition) is likely to be subverted, and a high-frequency use password entered instead.

Error Types

Reason (1990) posits three levels of behaviour that may be used to distinguish the different types of error. *Skill-based* behaviour happens under schematic control - it is automatic and unconscious. Errors at this level are *slips* (unintended actions), or *lapses* (unsuccessfully completed intended actions). When this level of behaviour fails, the following two levels of control are applied until the failure is repaired: rule based control and knowledge based control.

Rule-based behaviour selects and applies previously stored rules to the data. This is an automatic, largely unconscious process that interacts with *skill-based* or *rule-based* activities in the following way: if (state), then (schema, motor program, rule), etc. *Knowledge-based* behaviour is by attentional control, and goes from first principles. Failure at either of these two levels results in *mistakes:* actions completed successfully or unsuccessfully where the intention is wrong. For example, having typed in the wrong password, it would be a mistake to put your foot through the monitor, as this would not achieve your goal of logging in.

Password use in general does not require much problem solving, so there is little scope for rule and knowledge based errors. The majority of errors will be skill-based. These error types will be outlined in the next section, whereas the reader is directed to Reason (1990) for descriptions of rule- and knowledge-based error types. However, two rule-based errors are plausible in password use: rigidity and redundancy.

Rigidity characterises the intrinsic "cognitive conservatism" of rule usage (Reason, 1990). There is a strong and remarkably stubborn tendency to use the same familiar but cumbersome solution when a more elegant one is available (cf. the Jars Test - Luchins & Luchins, 1950 cited in Reason, 1990). This would be revealed in repeated attempts to re-type the password information when another solution - perhaps using a different password - would mean success.

Redundancy refers to the nature of signs and problems. Some signs are more diagnostic than others, and will be attended to more than redundant information as the problem solver's skill and experience increases. This bias will favour previously useful signs, and tend to dismiss rare counter-signs. This might happen when the user makes a typo when entering her username. When trying to trouble-shoot the login failure, she might not attend to the username she typed in, and so might fail to diagnose the problem.

Skill-based errors

These can be grouped under two headings (Table 4): inattention and over-attention errors.

Table 4 - Skill-based errors

Inattention	*Over-attention*
Double capture slips	Omissions
Omissions following interruptions	Repetitions
Reduced intentionality	Reversals
Perceptual confusions	
Interference errors	

Inattention errors occur when attentional monitoring is omitted at a critical check on the progress of an action. This is particularly likely to happen if the intention is to deviate from a well-established action sequence - for example, when logging in with the newly changed password. It is predicted that interference between passwords will be a significant and prevalent problem.

Double-capture slips are one of the most common results of an omitted check. There are two kinds of capture involved. The attention is captured by some internal or external event, just when the attentional control mode is required to set the action along the intended path; from that point, the most activated schema captures control and leads on to the completion of an unintended activity.

Generally, the result is a strong habit intrusion. This would be the case after a forced change of a password, where the old password was used instead of the new one. Another subcategory is the branching error, where a common root action sequence leads to different outcomes, but a check is omitted at the vital choice point. This would be the case when two commonly used passwords have the same *account name*, for instance: intending to type your password A but using password B instead.

The necessary conditions for the occurrence of these errors are:

1. The performance of a well practised activity in familiar surroundings.
2. An intention to depart from custom.

3. A point where the activation of particular action schemata are very different.
4. Failure to make an attentional check

Omissions following interruptions are the same as double-capture slips, only the first capture is caused by an external event, not an internal one. A subclass of this error is the *program counter failure*. The steps take to deal with the external event are counted in as part of the action sequence. The action is restarted, but at a place further down the sequence than it should be. For instance, the user enters her username and goes to the password field; the phone rings, she deals with the phone call, then presses RETURN, having neglected to type in her password.

Reduced intentionality errors are a failure of prospective memory, where some delay occurs between the formulation of an intention to act and the committing of the act. The intention is not maintained by the necessary attentional checks, and a slip or lapse happens. Subclasses are: detached intentions (intending to log into one system but logging into another instead), environmental capture (going to a PC intending to log in but instead starting up *Minesweeper*), multiple side-steps (the user intends to log in, goes to the kitchen to make a cup of tea, then she needs to remember what she wanted to do). There are also the two lapse states, *"What am I doing here?"* and *"I know I should be doing something but I can't remember what"*.

Perceptual confusions arise when objects that are in the expected place, or look similar or perform a similar operation are taken to be the target object. An example would be mistaking the login prompt of one system for the one that you intended to log into.

Interference errors are blends or behavioural spoonerisms. Two highly activated schemata try to gain control of the user's actions. For instance, the user might type in the username of one account, and the password for another. (Though this type of error would be difficult to distinguish by observation from a *branching error*.) *Behavioural spoonerisms* are the result of transposing elements in the sequence of actions. For example, typing in the password instead of the username, and the username instead of the password. This could lead to your password being disclosed on screen, sent in plain text across the network, stored in a plain text log file.

Over-attention errors happen when attentional monitoring occurs at an inappropriate moment. These errors particularly occur in *test-wait-test-exit* type tasks, where a sequence of automatic tasks need be completed in order, but are punctuated by waiting. This might be case when logging into a machine by *telnet*. A mis-timed check can result in believing oneself to be earlier on or later in the sequence than is the case. Repetition or omission of elements may occur, for instance typing in the username

twice, or forgetting to type it in. The latter case would require user reports to distinguish it from an omission after interruption error.

Human Error and Computer Security

Recently computer security researchers have begun focusing on Human Error, producing statistics showing it to be a large component of problems in computer security: McCauley-Bell (1999) cites NIST (1992)where 65% of the economic loss attributed to information security breaches were caused by human error, whereas only 3% of the loss was attributed to malicious outsiders; Spruit & Looijen (1996) find that 41% of security incidents are caused by human error, whereas only 9% are due to wilful crime.

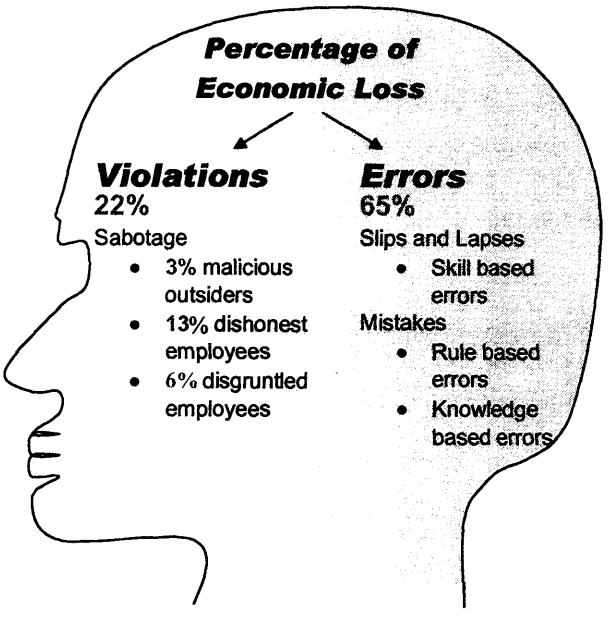


Figure 11 - Percentage of economic loss due to information security breach categories, adapted from NIST, 1992.

HCI provides a suitable basis for explaining human error during computer use, and identifying interventions that might prevent them. Indeed, *number of errors before task completion* is a primary measure employed in usability. Some security equipment has such severe usability problems that the most appropriate measure is not how many errors were made before the task was completed, but whether the task was completed at all (e.g. PGP 5.0 - Whitten & Tygar, 1999).

3.3.6 Summary

The accuracy of human memory varies with the method used to access it. This feature of human memory can be exploited by developers of authentication systems. *Cued recall* can be twice as accurate over the same period as unaided recall (Figure 5),

depending on the strength of the association between the cue and the response (Parkin, 1993). *Recognition* can be twice as accurate as cued recall, though the accuracy of recognition steadily decreases as the ratio of distractors to targets increases and with increasing similarity between target and distractor stimuli (Baddeley, 1997). Unaided and cued recall and recognition are all accesses to explicit memory.

It has been repeatedly demonstrated that tests of implicit memory are far less affected by time than measures of explicit memory (Parkin, 1993). Fragment completion, for example can retain accuracy when even recognition has faded (Figure 7), though the absolute level of accuracy is small. Tests of implicit memory may therefore not be appropriate for authentication mechanisms, though implicit learning may be appropriate if disclosure of authentication knowledge is intolerable.

Several principles have been proposed that can be used to enhance the learning of passwords, or other secrets used in authentication mechanisms:

1. The more time spent using a password, the better the subsequent ability to remember the password.
2. It is better to spread out the practice over time, instead of doing it *en masse*.
3. It is better to give deeper and more elaborate semantic processing to the password than to its visual or acoustic features.
4. Images are more memorable than words.

There are two theories as to why forgetting occurs. One is that memories are eroded by the passage of time. The other is that memories are interfered with by the presence of other memories. This can be in two ways: *proactive interference* occurs when previously learned material interferes with the recall of later material (Baddeley, 1997), while *retroactive interference* is the interference on learned material produced by later learning (Baddeley, 1997).

Reason's (1990) model of human error has been introduced, and used to predict that interference will be an important source of error in password use, with frequently used passwords intruding on infrequently used ones. Human error is a major issue for computer security: 41% of security incidents are caused by it (Spruit & Looijen, 1996). The discipline of HCI is put forward as a solution to human error in computer security.

Chapter 4

Performance of authentication mechanisms

This chapter consists of three main sections. Section 4.1 deals with the performance of traditional password mechanisms, and discusses different approaches to the measurement and meaning of performance. Section 4.2 describes the working and performance of text based alternative mechanisms for authentication. Section 4.3 brings together the literature review so far to redefine this PhD's research question.

4.1 Existing password mechanisms

4.1.1 Sociotechnical performance

Adams et al's (1997) study was the first instance of HCI research into the wider human factors of authentication. Its main finding was that the organisation investigated made choices in the implementation of its password mechanisms that encouraged their users to circumvent security. The finding led to a paradigm shift from the prevailing conception in password security: that tighter restrictions mean better security. Some aspects of this finding were modelled in the paper, and concrete operational recommendations were made for better practice. However, this did not supply a framework for designing or assessing the security of a system.

Adams et al.'s (1997) user research found four factors that influenced effective password usage:

1. Multiple passwords
2. Password content
3. Users perceptions of security
4. Information sensitivity

Multiple passwords - users were given passwords for different applications by the organisation and had to change them frequently because of password expiry mechanisms. The high number of passwords reduced memorability and thereby increased writing down and choosing poor passwords, including passwords with incremented suffixes.

Users did not how to create *strong passwords content*-for example choosing one's wife's Christian name because a complete stranger would not be able to guess it. The organisation had not educated them about what made a strong password, or how to create one.

Security perceptions - users' perceptions of security levels and potential threats was a key element in motivating their work practices, but because the organisation had not

supplied them with effective versions, users made their own incorrect risk models. For example, not publicising security breaches led to feeling that hacking was not a problem, and thus there was no threat.

Information sensitivity - users correctly categorised systems according to whether the information they contained were sensitive enough to be worthy or not of secure password practices. However, whilst using the right principle they often came to the wrong conclusion. In the absence of guidance from the organisation, users concluded that confidential information about individuals such as personal files and e-mail was sensitive, but that commercially sensitive information such as customer records and financial data were not.

Adams et al.'s study found other examples where choices made by the organisation resulted in situations where users subverted safeguards. This research provides important insights - into the causes of usability problems with password authentication, and highlighting the organisation's role in these problems. This research reported in this thesis carries forward these two themes.

4.1.2 Technical performance

Resistance to attack and memorability are two sides to the performance of password systems that have been most widely researched - the first is a measure of task quality, and the second is of stakeholder cost. Passwords selected by users tend to be weak (Petrie, 2002), and laboratory studies show worse memorability than naturalistic observations.

This section will first discuss the resistance of passwords to attack, and will go on to examine their memorability.

Real-world password strength

In real-world settings, the overwhelming majority of passwords chosen by users are weak (see section 3.2.5 above for definitions of cryptographic strength): 90% of 1,200 users reported choosing passwords that were dictionary words or names (Petrie, 2002), with 47% of the sample choosing their own name or nickname or the names of their partners, children or pets (the *Family* group of Petrie's taxonomy); 32% choosing the name of sports stars, cartoon characters, pop stars, favourite teams or film stars the *Fan* group of Petrie's taxonomy); and the *Fantasist* group (11%) who pick words like 'Sexy', 'Stud', 'Goddess' and 'Slapper'. These three kinds of passwords are among the weakest that can be chosen. Only 9% of this sample reported choosing strong passwords (the *Cryptic* group of Petrie's taxonomy). Even in environments where

users are asked to select strong passwords, large numbers of users do not comply: 32% of university student's passwords in one study were cracked with a short running dictionary attack (cf. Yan, Blackwell, Anderson, & Grant, 2000). Similar proportions are found in studies run in business environments (Belgers, 1993; Klein, 1990).

Password memorability

Studies concentrating on memorability of passwords give cause for optimism. Medium-strength passwords can come close in their memorability to weak passwords (e.g. see Zviran & Haga, 1990, 1993 in Table 5)- showing that improvements in resistance to attack can in some circumstances be without the cost of huge increases in helpdesk use. The password memorability results and external validity of several laboratory studies are discussed below.

Zviran and Haga (1993) gathered 103 graduate students for a comparative study of the memorability of secrets in 3 different authentication-by-knowledge systems. The average age of the student participants was 33, ranging between 25 and 42. The sex ratio of participants was 85% male to 15% female.

These participants were experienced computer users, with an average self-reported experience of 5 years. 80% of the sample reported previous use of PCs, while 56% of the sample reported experience of mainframes.

In this questionnaire study, different types of passwords were generated and then their recall rates measured after an interval of 3 months. The password types were:

- System generated - alphanumeric - strong
- System generated - pronounceable - weak to medium
- User generated - free choice - (weak)

The recall rate of strong passwords was **13%** (none of those successfully recalled had been reported from memory, but instead had been "remembered" by writing them down). In an earlier study using a similar participant sample (106 information management graduate students; mean age 32; age range 25 to 41; 76% male / 24% female; average 4 years computer use), the recall rate of strong passwords was **23%**, of which 34% had been reproduced from memory, and the rest written down (Zviran & Haga, 1990). This averages to about *18%* recall of system generated alphanumeric passwords after a 3-month interval. If this were a real system, it would be equivalent to a failure of availability for 82% of users.

Table 5 - Summary of password memorability studies, ordered by password type

Researchers	*Year*	*Password Type*	*Pwd chosen by*	*Study duration*	*Accuracy of Recall*	*N.*	*Policies*	*Study design*
Zviran & Haga	1993	Strong	CPU	3 months	13%	103	3 strikes	Enrol/gap/test
Zviran & Haga	1990	Strong	CPU	3 months	23%	106	3 strikes	Enrol/gap/test
Zviran & Haga	1993	Strong (sayable)	CPU	3 months	37%	103	3 strikes	Enrol/gap/test
Bunnell et al.	1997	Strong (sayable)	User	2 weeks	77%	86	3 strikes	Enrol/gap/test
Spector & Ginzberg	1994	Strong	User	1 year	0%	15	-	Enrol/gap/test
Spector & Ginzberg	1994	Strong	User	6 months	0%	15	-	Enrol/gap/test
Yan et al.	2000	Strong	User	3 months	99%	96	-	***Pwd in everyday use***
Spector & Ginzberg	1994	Strong	User	1 month	0%	15	-	Enrol/gap/test
Spector & Ginzberg	1994	Strong	User	2 weeks	33%	15	-	Enrol/gap/test
Yan et al.	2000	Pass-phrase	User	3 months	97%	97	-	***Pwd in everyday use***
Spector & Ginzberg	1994	Free choice	User	1 year	26%	15	-	Enrol/gap/test
Spector & Ginzberg	1994	Free choice	User	6 months	26%	15	-	Enrol/gap/test
Zviran & Haga	1993	Free choice	User	3 months	27%	103	3 strikes	Enrol/gap/test
Zviran & Haga	1990	Free choice	User	3 months	35%	106	3 strikes	Enrol/gap/test
Spector & Ginzberg	1994	Free choice	User	1 month	60%	15	-	Enrol/gap/test
Spector & Ginzberg	1994	Free choice	User	2 weeks	73%	15	-	Enrol/gap/test
Dhamija & Perrig	2000	Free choice	User	1 week	65%	20	-	Enrol/gap/test
Dhamija & Perrig	2000	Free choice	User	Minutes	95%	20	-	Enrol/gap/test

System-generated pronounceable alphanumeric passwords are designed to be a

compromise between password memorability and strength. It is assumed that by making a nonsense sequence of characters sound like a word, it will become more memorable. Making the sequence sound like a word cuts out those combinations that do not, thereby decreasing the search space and weakening the content of the password. However, the assumption of better recall is upheld: *37%* were recalled successfully (83% of which were reproduced from memory). The difference between alphanumeric and pronounceable passwords in the 1993 study was significant (chi squared = 7.898, p = 0.005). The results of this study are that pronounceable passwords have a retention rate more than twice that of random alphanumeric passwords (combining the strong password results of Zviran & Haga, 1990, 1993). It is equivalent to an availability failure for only 63% of hypothetical new users with a 3-month retention interval.

In a recent study of pronounceable passwords (Bunnell et al., 1997), the passwords consisted of two lower-case real English words (one with 3 letters, the other with 4), concatenated with a single digit (0 to 9). The retention interval was a two-week period between password generation and recall. The recall rates were better than those described above (77% recalled).

Self-generated free-choice passwords are expected to have the weakest content and be most memorable. These passwords were successfully recalled by 27.2% of the participants, with 42.9% reporting using their memory only and 7.1% reporting writing it down. 93% of recalled self-generated passwords contained only letters (Chi squared=7.324, p=0.022, significant) - more evidence that self-generated passwords contain "weak" content. In the earlier (1990) study, a retention rate of 35% was found, with 86 % reporting they were recalled from memory and 14% writing them down. Averaged, these figures are 31% recall, 64% of which from memory and 10% writing them down. These results are better than those obtained for random alphanumeric passwords and similar (if slightly worse) than those obtained for pronounceable alphanumerics. This suggests that pronounceable alphanumerics may be a sweet spot for password content.

The experimental situation above represents almost a worst-case scenario: generating a password and then not using it for 3 months. This situation also goes some way to representing the demands faced by users of several passwords which then compete for memory resources: participants in the study were asked to recall 3 passwords (and a pass-phrase, and tens of Challenge-Response pairs). The participants would still have to remember the passwords they use in the real world.

It cannot be decided from the published details how well Zviran and Haga's (1990, 1993) experimental situation mimics the situation of remembering an infrequently used

password, because it was not recorded how many passwords were owned by the participants, the similarity of their content, how frequently they were used, nor the effect of different contexts of recall. However, it is possible that the fit was good: the relatively large size of the participant sample (103) will have given a diversity of numbers of passwords owned and frequencies of use. This argument will gain more force as more participants are considered from similar experiments, yet to come.

Bunnell et al. (1997) performed an experiment similar to Zviran and Haga's in most respects (sample size 86, mean age 29.8, age range 19 to 53, enrolled on university course), bar the retention interval. Bunnell et al.'s participants were tested after only 2 weeks (a sixth of the time). The resulting recall rate was 77%, more than twice that of Zviran and Haga's combined studies. Bunnell et al. note this and the well-known decay of memory for alphanumeric material over time (Postman, 1985), though their recall accuracy is far larger than would be expected from the Ebbinghaus curve (Figure 6). They also note the larger numbers of their participants who wrote down their passwords (22% versus 10% for self-generated and 45% versus 17% for pronounceable. This is evidence of a powerfully beneficial effect of writing passwords down, and may be useful for cost/benefit analyses of disclosure policies.

Zviran and Haga's 3-month retention interval is unlikely to reflect the situation faced by people in the majority of their password use: the recall of *frequently used passwords* (see section 6.3.1 for a definition). This is very rarely studied. *A priori*, frequently used passwords are more likely to be confused or blended and forgotten- unless it is a frequently used password *that has just been changed* (forced by expiry policies for example). Dhamija and Perrig (2000) asked 20 participants to generate a password (with no constraints on the choice), then recall it only minutes afterwards. Only 95% of participants were able to recall the password. If this finding is extrapolated, it represents a 5% failure rate for password changes. This is a conservative generalisation, as in password changing there is the additional possibility of interference from the obsolete password (see section 3.3.5). This generalisation suggests that password changing might be a particularly sensitive period; and that a policy of password expiry could be a significant source of costs: lost productivity for *users*, and a large source of additional workload for secondary stakeholders (help-desk staff).

Methodological and substantive difficulties

With the exception of Yan et al. (2000), the password memorability research described above have been lab studies. The ecological validity of these lab studies is difficult to judge and the naturalistic observations are difficult to generalise.

Lab studies enrol the participant with a password, force an interval of known duration during which time the password is not used, then ask for the password to simulate login. It is difficult to judge the ecological validity of these designs as data describing real-world intervals between enrolment and first use are not published. Obtaining this data should become part of the research agenda. Moreover, it is difficult to model the effect of interference between passwords because the number, type and frequency of use of other passwords the user has is not known. Obtaining this should also form part of the research agenda. Lab studies so far have not attempted to study the effect of security policies on password memorability and behaviour - this is a major omission.

Naturalistic observation studies record real-world password use, and have high ecological validity. They specify their participants in less depth than lab studies, and have no descriptions of frequency of use and intervals following enrolment, nor do they specify the policy environment, importance of availability, onerousness of password distribution procedures, or other details that would aid comparisons. There is a particular practical need to support generalisations from student populations (who are easily available to researcher) to corporate populations (who are difficult to study).

4.2 Alternatives to traditional password mechanisms: text-based alternatives

Improving the encryption in passwords requires similar process in practice to installing an entirely new mechanism, and so will be discussed here with other alternatives to traditional passwords.

4.2.1 Better encryption for passwords

This allows pronounceable alphanumeric passwords to have the same strength as UNIX password with much more random content (see section 3.2.5). Its effect is to increase the security offered by any passwords given to or chosen by a user.

Though greatly improved mechanisms such as BCRYPT are freely available, they are incorporated in only the tiniest fraction of systems. Instead of allocating the task of encryption to technology, which is well placed to take the load, it is in effect allocated to users who are much less well suited to perform it.

4.2.2 Associative Passwords

Sidney Smith first proposed *Associative Password systems* in a 1987 paper with startling findings that suggested an end to the problem of forgetting passwords (Smith,

1987). Four participants were each asked to generate a list of 20 cue words and associated responses. They were not told that they would later be tested on them. Participants were, however, then tested, twice and unannounced. The results are shown in Table 6.

The rate of successful recall of cued responses far exceeds the successful recall rates of any of the passwords yet studied. After the initial 6-month interval (twice the interval used in the password studies reviewed in section 4.1.2) an average success rate of *94%* for *associative passwords* and only *37%* for the most memorable class of passwords. After another year of interval, the success rate for *associative passwords* averaged *86%* - still more than twice the best rate achieved with conventional passwords, and over an interval 4 times as long.

Table 6 - Percentage recall of "Responses" when shown the list of 20 "Challenges" (adapted from Smith, 1987)

Respondent	1	2	3	4	Average
	After 6 months				
Correct	100	100	90	85	94%
Wrong	0	0	5	5	6%
No response	0	0	5	10	
	After 18 months				
Correct	100	95	70	80	86%
Wrong	0	5	15	20	14%
No response	0	0	15	0	

However, Smith's (1987) results must be treated with caution. The sample sizes used were very small. The participants were not described - they could have been world memory champions, or have no other passwords to remember and so interfere with their performance. The memorability of *associative passwords* have been tested in two other lab-based questionnaire studies: Zviran and Haga (1993) and Pond et al. (2000).

Zviran and Haga (1993) directed their participants to choose both *challenge* and *response* words according to a theme, to form 20 CR pairs each. It is not known how many of the participants used a theme. The participants were tested after 3 months, and correctly recalled 69% of their *responses* upon being cued with the *challenges*.

Pond et al. (2000) supplied participants with *challenges* that had been chosen because they provoked a wide range of responses across individuals (Palermo & Jenkins, 1964), because it was predicted to reduce vulnerability to guessing as well as being

instructed to pick some with a theme. After 2 weeks, the participants were able to recall only 39% of their 20 *responses*, with a standard deviation of 15% and a range from 10% to 90%.

Participants were able to correctly recall 66% of their 20 word associations with a standard deviation of 20 % (Pond et al., 2000). A summary of findings about associative passwords' memorability is given in Table 7.

Table 7 - Memorability of associative passwords

Authors	Year	Recall interval	**Type of word assoc. selection**	**Recall rate**	No. of items in pool	No. of participants
Smith	1987	1 Year	**Themed**	***86%***	20	4
Smith	1987	6 months	**Themed**	***94%***	20	4
Zviran & Haga	1993	3 months	**Themed**	***69%***	20	103
Pond et al.	2000	2 weeks	**Themed**	***73%***	20	20
Pond et al.	2000	2 weeks	**Unthemed**	***65%***	20	18
Pond et al.	2000	2 weeks	**Given random cue words**	***61%***	20	19

It appears that the ability to generate both *challenge* and *response* is necessary for good memorability. In both Smith's (1987), Zviran and Haga's (1993) and Pond et al.'s (2000) studies, participants were instructed to pick CR pairs according to a theme. This appears to be associated with good memorability. Pond et al. state that the use of a theme in the selection of a pool of word-associations lead to a significantly better memory of CR pairs than with an un-themed selection - an average of 14.6 correct out of 20 (73%) with themes, compared to 12.1 out of 20 (61%) without (but they do not report inferential statistics).

In summary, *associative passwords* are more memorable than conventional passwords, despite participants having to remember more material overall. These results suggest that associative passwords would lead to fewer password *recall* problems than conventional password systems.

There are two general problems with the design of the studies described above - they do not test the task of concurrently using several different CR systems, and they do not predict the effects of the additional effort and time required to enrol with, authenticate with and manage the system.

Concurrent use of several different CR systems would be the task faced by computer users if associative passwords came to replace conventional passwords. How easy

would it be to remember 10 or so sets of CR pairs? What would the problems be? There is some evidence that if each set of CR pairs were generated according to a separate theme, then interference problems between CR pools would be low if the theme were used as a prompt (see the discussion of Figure 9 on page 65 above). The use of a theme in generating both *challenge* and *response* appears to be of great importance.

CR systems require a pool of CR pairs, which in many mechanisms the user will have to choose. Because there will be many times the number of CR pairs as the passwords they replace, it is plausible that users will have to expend many times the effort of choosing passwords in choosing the CR pairs. Authentication will take longer, as more information has to be transferred both ways through the mechanism than was required for passwords. The studies discussed above do not seek to predict the reaction of users to this increase in effort, or to identify appropriate circumstances for the use of these mechanisms - for example, users may be more willing to expend this effort on rare important occasions such as proving their identity to password helpdesks than on normal everyday logging in (cf. Just, 2003).

4.2.3 Cognitive Passwords

Bunnell et al. (1997) and Zviran & Haga (1990, 1993) studied the memorability of *cognitive passwords* (*challenge-response pairs* based on autobiographical data). All the studies used similar types of target items. Participants were supplied with a list of biographical questions and asked to answer them truthfully. The questions were either *fact based* (the validity of the answers could in principle be objectively checked) or *opinion based* (the validity of the answers could not be objectively decided).

Zviran and Haga's two studies used 14 opinion based questions and 6 fact based. The participants answered the same questions after an interval of 3 months. Bunnell et al.'s (1997) study used 20 opinion-based and 20 fact-based questions, and tested participants after an interval of only 2 weeks. The results are shown in Table 8.

The recall rates of both *fact-based* and *opinion-based cognitive passwords* are impressive. *Fact-based* passwords average a success rate of *89%* and *opinion-based* passwords average *77%*. These are much higher than the average of ***41%*** recorded across all types of conventional passwords in the same experiments.

Table 8 - Percentage successful recall of cognitive passwords

Type of Cognitive Item	Bunnell et al. (1997) n=86, 2 weeks	Zviran and Haga (1990) n=106, 3 months	Zviran and Haga (1993) n=103, 3 months	Average
Fact based	88%	84%	94%	89%
Opinion based	72%	70%	88%	77%

Cognitive passwords were not all successfully recalled to the same extent. In Bunnell et al.'s (1997) study the range for *fact-based* items was 58% to 100%, while *opinion-based* items ranged from 56% to 90%. Bunnell et al. (1997) point out that those questions should be chosen that have been demonstrated to have answers that are difficult to guess, as well as being highly memorable. A critical issue is "guessable by who?" - anyone who is familiar with the user's biography (such as a family member, friend or close working colleague) will have a natural advantage in guessing cognitive passwords compared to an attacker who is a stranger. By extension, cognitive password pairs should be difficult for an attacker to research without raising the alarm. Bunnell et al. (1997) used *significant others* to guess CR pairs, thus employing a difficult test for cognitive passwords to pass. Out of the forty items used by Bunnell et al. (1997), only five met their criteria for good memorability (70% or above) and resistance to guessing (20% or below). At that rate, it would be necessary to screen 800 potential cognitive items to gather a set of 100 from which all users would have to be authenticated. Though this may appear expensive, the result would be an authentication system that was approximately as resistant to guessing as conventional password systems (Bunnell et al., 1997). However, this new system would perform much better over long retention periods.

There is a related issue of an attacker harvesting CR pairs on one system to break into others. Because screening of cognitive password questions is expensive, there will be a tendency to use questions that have already been screened by other organisations (or not screen at all- which according Bunnell *et al.*, 1997 is dangerous). Once a cognitive password item is compromised, it is dangerous for an individual to re-use it. Cognititive passwords rely on the memorability of biographical data for their effectiveness, and biographical facts cannot be changed. If the user compensates for biography disclosure by giving an untrue answer to a cognitive password question, she is in effect using an associative password mechanism – these have lowered performance (see section 4.2.2). This performance may be further degraded through interference with the historically accurate answer (see sections 3.3.2 and 3.3.5).

4.2.4 Pass algorithms

Pass algorithms (Haskett, 1984) employ several rounds of questions and answers, instead of the single round that is used in traditional password mechanisms. In principle, it is similar to a zero-knowledge proof, with a human being taking the place of what would normally be a mechanism as the "prover". The user ("prover") has agreed a secret algorithm with the mechanism ("verifier"). When the prover attempts to login, she is issued with a challenge/question by the verifier. The prover takes this challenge and manipulates it using her secret algorithm, returning the answer to the verifier. If the answer is correct, some evidence is gained that the prover is who she says she is. However, there is always some possibility that the answer was guessed by chance. The process of challenge and response is repeated until the verifier is satisfied that the prover has answered enough questions correctly to establish her authenticity. The mechanism can be set to require greater or lesser amounts of proof/rounds of authentication according to the security required.

After first use, the memorability of the mechanism should be better than traditional passwords, as the secret information is employed in a repeated and elaborate mental processing of the challenge information supplied by the verifier. However, unless this process is carried out during enrolment, the user may forget the secret algorithm before using it. The mechanism's weakness compared to passwords is that it will take longer and more effort to authenticate, as several rounds of challenges and responses are necessary compared to passwords' single round. This may be more suitable for infrequent but critical tasks, such as authentication with a helpdesk for password recovery. Pass algorithms have not been empirically evaluated.

4.2.5 Pass Phrases

The most frequently used definition of *passphrases* is in the one adopted by this thesis - *extended length passwords that may include spaces* - though else where passphrases have been defined as *normal length passwords made out of the nth letter of every word in a phrase* (e.g. Yan et al., 2000). The memorability of pass phrases have not been researched in depth. Although the content of pass-phrases may vary as widely as that of passwords, only one type has been reported: self-generated phrases of 80 characters or less.

In their 1993 study (see section 4.1.2), Zviran and Haga tested the retention of newly self-generated pass-phrases over a 3 month interval. The recall rate was 21.4%, with 91% of them reported as recalled from memory without writing them down. This is slightly worse than the recall rates for self-generated passwords. It is possible that the

increased amount of information to be remembered caused this decrease - the average length of pass-phrases was 21 characters / 4.4 words. This is much longer than the 8 characters available for passwords. Another possibility is that fitting desired content into 8 characters results in deeper processing and therefore better memorability (see section 3.3.3) than is necessary for the unconstrained passphrase.

The ecological validity of this study is difficult to judge, as the proportion of authorisation attempts that occur at 3 month intervals is not known (see end of section 4.1.2). However, as with its password findings, it cannot reveal the memory problems associated with passphrases that are frequently used.

4.2.6 Pass Sentence

The pass sentence mechanism (Spector & Ginzberg, 1994) is a passphrase system (see section 4.2.5) using artificial intelligence techniques to support human memory such that the user must recall the *sense* of the pass sentence, *not its exact written form*. For example, the pass sentences "Daniel buys a book from John for 5 dollars", and "John sells a book to Daniel for 5 dollars", are equally valid. The two pass sentences have the same meaning, but are written differently. Moreover, if the user does not supply enough detail in the pass sentence, for example by forgetting some part of it, then the mechanism can prompt for further details.

All this is achieved while maintaining a huge search space-estimated as being 80 bits on average (allowing for the predictability of English in the mechanism's only empirical trial (Spector & Ginzberg, 1994). This dwarfs the search space of even the strongest UNIX password (52 bits), and exceeds the search space of commonly self-generated passwords (4 bits) by 23 orders of magnitude. In short, pass sentences should be difficult to guess, even by using automation.

The system offers the potential for far greater memorability than passwords overextended periods, but suffers the cost of forcing the user to enter many more characters during each authentication (an average of 40 characters in the trial, more than five times the length of a normal password). While the system may reduce the probability of memory problems leading to failed login, it greatly increases the potential for typos. However, the addition of some extra "intelligence" to recognise and correct for typographical errors could mitigate this problem, without greatly effecting the mechanism's security.

A prototype has undergone some empirical testing (Spector & Ginzberg, 1994), producing very positive results (Table 9). 15 participants generated a pass sentence each (and a PIN, and two passwords in the same study) and were asked to recall it

Table 9 - Memorability of pass sentences for 15 participants who could log in

Retention interval	Memorability: [number of] / [%]
Two weeks	15 / 100%
One-month	15 / 100%
Six months	14 / 93%
One-year	12 / 80%

after 2 weeks, then again three times at increasingly long intervals. No details were given of the participants, and their total number is small, so these impressive results should be treated with caution. However, if these results are to be believed, then pass sentences greatly improve upon passwords' memorability. However, length of login time may make them inappropriate for everyday use.

4.2.7 Rebus Passwords

Rebus passwords supplement system-generated pronounceable (but otherwise random) passwords with a rebus - a mode of expressing words and phrases by pictures of objects whose names resemble those words, or the syllables of which they are composed (King, 1991). For example, the randomly generated password *koucehur* is made from three randomly selected syllables: *kou-ce-hur*. To make the rebus, a phonetic search is undertaken to find English dictionary words that match the syllables in the password. In the example, the syllables are matched to the English words *cow*, *see*, and *her*, which displays their pictures when the user is enrolling with the system. If necessary, the user can display one of the rebus images as a cue to help him during normal use.

The system relies on several properties of human memory to make system-generated passwords more memorable and so improve the effectiveness of password authentication:

- memory is superior for pictures than for words
- cued recall is better than unaided recall, and
- the user is performing deeper processing of the password information than simple phonetic/acoustic processing by recognising the simple rebus images which are displayed above the password syllables (identification of the images

implies a semantic processing of their meaning), and deeper processing leads to better memory.

There is no published empirical evaluation of rebus passwords.

4.3 Re-defining the Password Recall Problem

The initial problem of the research reported in this thesis was "*how to make more memorable passwords*", because users forget them and this leads to unwantedly high use of helpdesks. Our theoretical framework (section 3.2.5) tells us that passwords are forgotten when they are rarely used, or when they are competing with more memorable passwords (i.e. that have been more recently or frequently used). Creating more memorable passwords will improve the first problem, but do nothing for the second- perhaps even make it worse. This chapter has reviewed research about different types of passwords, and found that there are small differences in laboratory recorded memorability between them over long intervals. However, authentication mechanisms that access different routes to memory offer twice the memorability over the same long intervals. A focus on password content will therefore lead to a relatively poor solution.

When examining the context of password use, security policies create interference between passwords by ensuring that they are regularly changed, and that there are many of them. The problem is exacerbated by policies limiting user's ability to prompt themselves by writing the password down. Furthermore, the initial question implies there is a choice of password content. Most security policies, however, mandate content of a particular kind, and will only become more restrictive as computers (and so brute force attacks) inexorably become more powerful. Focusing solely on password content cannot therefore give a lasting solution. A good solution requires that the scope of the problem be widened.

It has already been noted that systems are composed of three kinds of stakeholders (section 2.2.1) - not just the user. It is therefore concluded that the differing agendas of some stakeholders has resulted in the imposition of security policies and mechanisms that allocate security tasks to users (rather than to technology or other stakeholders in the work system). The research focus must therefore include all 3 groups of stakeholders:

- the *users* themselves (primary stakeholders) and
- their support staff and managers (secondary stakeholders) and
- their directors or share-holders (tertiary stakeholders).

The author falls back on HCI's traditional concerns - work system effectiveness, task quality and user costs. The original problem specification is reformulated as "how can password system effectiveness be improved". This PhD aims to produce substantive knowledge that can be used to reduce the costs of password authentication systems to all stakeholders, and to improve the quality with which password systems carry out the authentication task.

This research question has several dependencies. To achieve improved effectiveness the research will have to address:

- what causes password system performance to be good or bad

To answer this question one must be able to distinguish between good and bad performance. The research must therefore address:

- what is the performance of different password systems,

This requires that research is conducted about:

- how password system performance can be measured,

HCI research attempts to make generalised knowledge (see section 2.2.2). This thesis aims to make models abstracting across the particular combinations of password mechanisms and stakeholders about which data was collected. Eventually it is hoped that these will enable *general* conclusions about password authentication *as a class of system*. This thesis will also start model building that may later allow conclusions to be made about wider aspects of security in the corporation.

4.4 Summary

Traditional password mechanisms have substantial memorability problems. To improve matters mechanisms are modified so that they employ innately more memorable secrets, or support more robust routes to human memory than recall while ensuring "cryptographic strength". A hypothetical matrix of the relationship between memorability, the type of secret used in authentication, and the route to remembering supported by the mechanisms is presented in Figure 12.

Mechanisms that employ unaided recall such as passwords, pass-phrases tend to use one "all or nothing" round of authentication.

Mechanisms that use routes to remembering such as cued recall and recognition employ several rounds of authentication (except Rebus passwords which use only one). Instead of remembering one difficult to guess strong secret, these clases of mechanism ask users to remember and be authenticated on several weaker but easier

to remember secrets. By combining several weak secrets, the composite strength is high. Cognitive passwords (Zviran & Haga, 1990) involve a series of questions about the user's personal preferences and history. After a certain number of correct answers, the user is considered to have passed authentication. Associative passwords (Smith, 1987; Zviran & Haga, 1993) employ word pair or phrase associations in a similar manner while avoiding word association stereotypes (e.g. Dear-Sally , Spring-Holiday , Black-Settee). Ellison et al. (1999), refer to systems such as these that use the contents of episodic memory as employing *personal entropy.* The Pass sentence mechanism (Spector & Ginzberg, 1994) is unusual in that it uses one large difficult to guess secret. However, if the user does not get the secret completely right, the user is prompted with questions about it. When the user answers enough questions correctly, s/he is logged in. Weak empirical evidence (Spector & Ginzberg, 1994) suggests this mechanism shows promise.

The critical review of the literature has shown that focusing on password contents of traditional password mechanims cannot bring a good solution to the problem of password requests clogging up helpdesks. Whilst password content memorability is relevant, it is smaller part of a problem caused by the choices made by different stakeholders in the organisation. These choices have a great impact on the usability of authentication mechanisms, and on security more widely.

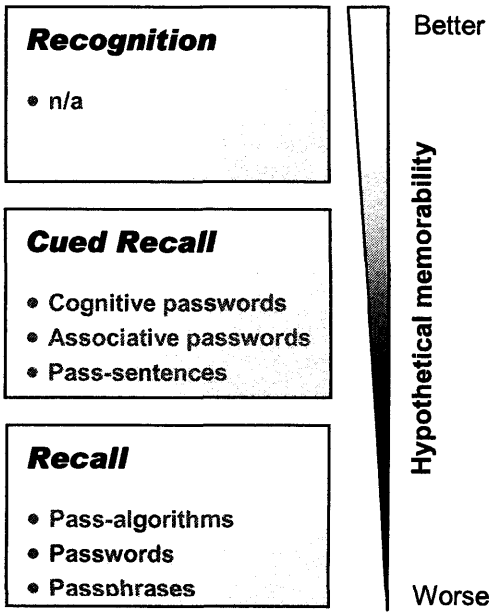


Figure 12 - Matrix of knowledge-based authentication mechanisms, categorised by Route to Memory.

In the light of this realisation, the goal of this PhD has been redefined as:

- How to reduce the costs of password authentication systems to stakeholders,

- How to improve the quality with which password systems carry out the authentication task. Both of these tasks involve:
- Identifying basic parameters of password authentication systems so that
    - Ineffectiveness can be diagnosed and
    - Interventions be designed

Finally, the PhD addresses the concerns of its parent discipline HCI to start on the path to generalising and validating its password related findings.

Chapter 5

Methods

5.1 Introduction

The previous chapter identified goals of this PhD as to collect basic parameters of password authentication systems in BT and UCL and to diagnose ineffectiveness. This chapter will outline two groups of HCI methods and select those appropriate for password research so that the process can begin of collecting the data to fulfil these goals. A discussion of the strengths and weaknesses of different techniques for collecting usability data and for diagnosing ineffectiveness in general, and the specific strengths and weaknesses for data collection about password systems, follows.

The purpose of evaluation is to measure the system's effectiveness, and if the system's balance between work quality and costs is not optimal, diagnose why it is not. The interaction between the human and technology parts of the system, and the properties of both human and computer parts of the system are studied.

There are two main categories of evaluation:

- ***Empirical***-where data is collected from real users, or artefacts
- ***Analytic***-where the user or technology is modelled

Each of these categories of evaluation will be discussed in separate sections below.

5.2 Empirical evaluation techniques

This section will review available data collection techniques with view to deciding which would be best for studying password system performance. The list of techniques in Table 10 is representative, though not complete.

5.2.1 Observation

Observation can vary according to the setting (in the field or laboratory), the extent of the observer's involvement with the task (ethnomethodology, participant observation, or observation from outside), the formality of the observation process (quick and dirty or highly controlled), the tools used to record the observations (from pen and paper to full audiovisual recordings) (Preece, Rogers, & Sharp, 2002), and the obtrusiveness of the techniques used (Dix, Finlay, Abowd, & Beale, 1998). Field observations allow the capturing of real world behaviours in the context of use, whilst laboratory observations allow easy comparisons and high levels of control (Dix et al., 1998). Most observation techniques require an observer to be present, so are expensive if the phenomenon being observed occur infrequently.

Table 10 - Evaluation techniques

Data collection techniques	Appropriate for password research	Strengths	Weaknesses
Observation	×	Collects data on what really happens.	Breaks security policies by its similarity to shoulder-surfing. Misses rare events.
Retrospective interview	✓	Collects data where observation is not possible	Factual data can be inaccurate due to workings of human memory
Retrospective verbal protocol	×	Fleshes out observation.	Breaks security policies by its similarity to shoulder-surfing.
Concurrent verbal protocol	×	Fleshes out observation.	Breaks security policies by its similarity to shoulder-surfing. Interferes with password task.
Questionnaire	✓	Collects data where observation is not possible	Factual data can be inaccurate due to workings of human memory
Focus group	✓	Collects data where observation is not possible	Factual data can be inaccurate due to workings of human memory
System logs	✓	Objective measure. Collects large amounts of data.	Little insight into cognitive processes - why events occurred. Limited by administrative domains.
Diaries	✓	Collects data where observation is not possible. More accurate than interviews for factual data.	Burdensome on diarist. Possible interference with behaviour being recorded.
Documentation	✓	Tells you how things should be done, ideally.	Does not tell you what activities really happen
Experimentation	✓	Strong control of variables.	Low ecological validity
Cognitive walkthrough/ Heuristic walkthrough/ KLM, GOMS	×	Can collect data without users. Can model task performance times.	Requires the preceding data collection activities. Designed for more complex interaction tasks.

Because there are no published studies of password frequency of use, no one can be sure how effective field-based direct observation of people using passwords would be.

It is predicted that some of the most problematic passwords will be some of the least frequently used (due to memory failures), perhaps on the order of once every several months. Some important password events are therefore unlikely to happen during any period of observation, and observations would be biased towards frequently occurring events. Moreover, there is only one observer available, and the amount of useful data that could be connected would be small compared to the resources spent collecting it. Furthermore, shadowing users would in many cases breach company security policies - i.e. watching people as they type in sensitive passwords. Observation of password users in the field is therefore not appropriate.

5.2.2 Interview

Interviews are a rich and efficient source of data, because the researcher can ask users about events that occurred when the researcher was not present. Interviews can therefore collect data that direct observation cannot. However, interviews suffer from inaccuracies-events may be remembered as better or worse than they really were, or as taking a longer or shorter amount of time (Preece et al., 2002). This can be a particular problem for regularly occurring events, since human memory stores such events as a general script from which memories are reconstructed, instead of retrieving verbatim what actually happened (Schacter, 2001). Memories of these events will tend to be regularised, with details missing that perhaps were important (Schacter, 2001). Users' may lack insight about an event being recalled- for example an interviewee may genuinely believe they experienced a problem with a password because the server was upgraded, when in fact the interviewee slipped and entered the wrong password. Interviewees may also attempt to give answers that make them appear more socially desirable (Oppenheim, 2000), or refuse to answer questions on sensitive topics. There are techniques to combat both these risks (Robson, 2001).

Interviews are more appropriate for password research than direct observation, because the researcher can question the participant about rarely used passwords or infrequent but important password problems. However, interviews are unlikely to be accurate in collecting data about password frequency of use or frequency and severity of problems, because of memory biases. Though it may not be very accurate, in many instances the user's memory of password related events will be the only source of information available to the researcher.

5.2.3 Retrospective verbal protocol / post test walkthrough

Interviews can be improved by presenting participants with recordings of what they were doing, to cue the participants' memory of events in a *post test walkthrough* (Dix et

al., 1998). However, such techniques have the same weaknesses as observation. They may miss rare but critical incidents and are expensive of resources if observations are made for long durations. Moreover, if the interval between the activity and the post test walk-through is long enough, the user may not be able to accurately remember her thought processes and the reasons for her actions (Dix et al., 1998).

This technique would be more difficult to use in password research than direct observation (see section 5.2.1), as it would have many of its disadvantages, with the added necessity of making a permanent record of the sensitive event - an augmented shoulder surfing attack; a greater breach of security policies than observation alone.

5.2.4 Concurrent verbal protocols - "Thinking aloud"

Concurrent verbal protocols suffer the same weaknesses as retrospective verbal protocols. In addition, there is the potential for interference between the task of using the password and producing the verbal protocol. Both thinking aloud and password tasks are essentially verbal, and so require the same mental resources (cf. Wickens, 1992) - the resulting competition for scarce resources is likely to degrade performance in both tasks.

5.2.5 Questionnaires

Questionnaires are very flexible, offering the opportunity of collecting both qualitative and quantitative data (Oppenheim, 2000). However, the data is *user reports*, and suffers some of the same biases of memory that interviews do (see section 5.2.2). Questionnaires are efficient, they are cheap to produce and distribute, and cheap to analyse if the responses are constrained (Dix et al., 1998). However, they lack some of the flexibility of interviews - the researcher is not able to pick up and explore topics as they appear serendipitously (Dix et al., 1998). Questionnaires do not allow researchers as much scope to establish rapport and put respondents at their ease as interviews do (Oppenheim, 2000), and so respondents may be less likely to answer sensitive questions. Questionnaires should undergo extensive pilot testing to ensure that respondents reliably understand the intended meaning of the questions, as the researcher will in many cases not be present to clear up misunderstandings (Oppenheim, 2000).

Questionnaires therefore can measure from the entirety of a users password experiences in a way that observation cannot, and do not break security policies. They are therefore appropriate for password research. However, extra steps may be necessary to collect useful data about passwords with questionnaires. An Internet-

based questionnaire survey was unable to collect information about password design in 50% of responses (Adams et al., 1997). Appeals to anonymity and authority can improve response rates, and so will be employed (Oppenheim, 2000).

5.2.6 Focus group

Flexibility is a particular strength of *focus groups*, where it is possible to quickly cover a wide range of views and home in on interesting topics as they are brought up. However, they also rely on retrospective user reports, and so suffer the memory problems of interviews, and are more suitable for collecting opinions than facts. The success of focus groups relies heavily on the skill of the facilitator (Robson, 2001). Effort may be required to prevent some group members from dominating the group agenda, and to ensure that every member of the group can express their true opinion when some group members are reserved than others, and in the face of opposing views (Robson, 2001). Moreover, focus group data are more difficult to analyse than interviews, because of the difficulty of transcribing overlapping speech and of obtaining clear recordings of everyone.

Focus groups are appropriate for use with password research because they can allow capture of data about events that are rare and occur when direct observation cannot be achieved because of security policy, etc.

5.2.7 System logs

Machine recorded system logs contain information recorded automatically by a work system. They are seen as objective - not subject to the biases of human memory. They are however subjected to the biases of the people who designed them (Bowers, 1992), and so record information of interest to the designers, rather than of interest to users or researchers. The researcher may have to make inferences because phenomena of interest have not been logged directly, and interpretations of the log can be unreliable and inaccurate.

They have the advantage that they are unobtrusive (Preece et al., 1994), are generally more accurate than records made manually, and can lead to large amounts of reliably coded data being collected which can be at least partially analysed automatically (Dix et al., 1998). The unobtrusiveness of system logs leads to an ethical and practical problem: do researchers inform users that they are being logged and risk them altering their behaviour (and thus confounding the observation), or not inform users they are being observed and fall foul of data protection laws.

System logs have several problems in password research. Passwords are generally not recorded in system logs for fear of security breaches. If they were recorded, it would be difficult to obtain these logs from their suspicious owners, who may also have a legal obligation to protect their confidentiality. It may even be difficult to identify the owners of the logs, or even the logs' existence. There is likely to be a large number of different logs for each individual participant, who may use passwords in many different systems which may be in different parts of the world by virtue of the Internet and the World Wide Web. It is therefore unlikely that researchers would be able to obtain a complete set of logs. However, helpdesks are likely to keep logs about password problems-our topic of interest.

5.2.8 Diaries

Diaries are especially useful for longitudinal studies and logging unusual or infrequent tasks and problems (Dix et al., 1998). Diaries are portable, and so can be carried round by participants, and used concurrently with the tasks being studied, or nearly concurrently. This can prevent the biases of human memory that can reduce the accuracy of factual interviews and questionnaires, whilst also offering some of the advantages of interviews and questionnaires. However, diaries only allow a coarse level of recording, and are burdensome to use (Breakwell, 1995) and inappropriate for frequent records (Dix et al., 1998). The participant may forget to use them after an event and so fill them in retrospectively, by which time the events have been forgotten that should have been recorded. This could be ameliorated with an alarm system that prompts the diarist to make entry. This requires extra equipment, which the diarist must also keep about their person, adding to their burden. However, the alarm may sound at an inconvenient moment when the diarist cannot make entry. This may be particularly unacceptable in a business context where it may interrupt meetings with colleagues or clients.

The diary may influence the behaviour that is being studied (Breakwell, 1995). Breakwell suggests that the diary's instructions should include warnings to participants to be aware of this problem and not to let the diary change their behaviour.

Diaries can capture password data across different administrative domains where system logs cannot, for example because a person may use computers controlled by different parts of the company. Furthermore, diaries can record details of password events from outside the company which may interact with the passwords used at work. Diaries can be used to record password events which cannot be directly observed because of security policy whilst avoiding the memory biases of interview. Diaries are therefore appropriate for password research.

5.2.9 Documentation

The easiest source of research data is manuals, instruction booklets, training materials, etc. (Dix et al., 1998). Equipment specific manuals may tell you about the functions of the device (Dix et al., 1998), and policy documents can tell you about how the devices are meant to be used (although this may not be how they are actually used - Dix et al., 1998). The researcher must be given access to it, which may not be possible if the documentation is commercially sensitive, or is seen as containing information that could be a security risk if it became known outside the organisation.

Documentation therefore is appropriate for password research. It can reveal important information about the construction and configurations of password mechanisms including the restrictions on password content enforced by mechanism. Accepting the evidence that some password content is more memorable than others that different password mechanisms enforce harsher restrictions on content than others, then password systems that restrict password content to that which is unmemorable will be systems that cause helpdesk use. It will therefore be possible to predict password system performance based on the restrictions that the system enforces on password content. However, because password related documentation describes safeguards, security departments may be unwilling to release them to outside researchers.

5.2.10 Experimentation

Experimentation simplifies complex phenomena, and controls conditions so that the effects of one or two important variables can be judged in isolation. Experimentation can be formative and summative-it can be used to investigate the causes of system performance, or to measure the performance of systems, for example to show if one is better than the other. However, the simplification of the research problem into something that can be modelled in an experiment may mean that results are no longer meaningful in the context of the original problem.

Moreover, the performance and contexts of password systems are currently not well understood, no one knows what all the relevant variables are, or what value to put on parameters such as how many passwords users have. It is therefore difficult to model them or control these variables and parameters in an experiment. Because these are not modelled in the experiments, the result of the experiment may not generalise well to the real-world. Despite this, experimentation is highly relevant for password system research, and has been one of the most frequently used techniques (see section 4.1.2).

5.3 Analytic evaluation techniques

When real users are not available, or resources (time, money, facilities, etc.) are scarce, it is possible to use analytic techniques. To the author's knowledge, none of these techniques have been used in previous password research.

There are good reasons for this. Logging in with a password mechanism is a simple and brief activity, as far as the user is concerned: It involves the recall and typing of two pieces of information (the username and password) followed by pressing the return key. Analytic evaluation methods such as *cognitive* or *heuristic walk-throughs* are designed for more complex interfaces (cf. Dix et al., 1998; Preece et al., 1994), and so are therefore not appropriate for use with passwords. *Keystroke level modelling* (KLM) and *GOMS* are formal methods for modelling a users physical and mental behaviour at an interface, and can be used to predict the time taken/effort required to perform particular tasks. These techniques are particularly useful for optimising the interfaces of heavily used equipment where the same task is repeated for an entire working day. These techniques are not good at modelling errors and require the basic parameters, variables and sequences of tasks as input (John & Kieras, 1996), which have not yet been established. Analytic evaluation techniques are therefore not useful at this stage in research about password system performance.

5.4 Description of studies

A summary of the studies and methods used is given in Table 11. More details are given in the following sections.

Table 11 - Summary of studies and methods

Study	Goal	Methods	Population	Comments
1	Begin to estimate the basic parameters of password use and performance in UCL. Pilot tests a combined method	Interview, diary, system logs	University students and staff	Designed for a very detailed view of individual's password use on all systems.
2	Start to estimate the basic parameters of password use in a BT sample. Start to diagnose performance	Questionnaire	Corporate R&D	Requires greatly reduced effort for participants than study 1, and so more suitable for use with BT population. Reliance on human memory makes this relatively inaccurate.
3	Supplement study 2 with data from a different sample of users in BT	Questionnaire	Corporate general	Supplements data from study 2, sampling different parts of the BT population.
4	Estimate the basic parameters of password use and performance in UCL with more accurate instruments. Continue to diagnose performance.	Password mechanism system logs	University students	Very accurate view of performance: observes both normal use and occurrence of problems but on only one system. Permission could not be obtained to instrument corporate password mechanisms
5	Expanded data collection for study 4 with greatly expanded numbers of observations.	Password mechanism system logs	University students	As above
6	Observe password system performance for a BT sample using objective measure. Diagnose performance	Helpdesk system logs	Corporate - general	Performance measured across many systems. Incomplete view of performance: observes problems but not normal use.
7	Validate sub-set of interventions with a sample of users.	Focus groups	Corporate R&D	Interventions chosen as being easily to implement, and so more likely to be implemented.
8	Validate sub-set of interventions with a different sample of users.	Focus groups	Corporate operations	Supplementing study 8 with data from users in a very different part of the organisation.

5.4.1 Study 1 (Combined methods)

Study 1 pilot tests a methodology and estimates some of the basic parameters of password system performance in a real world setting (using the pilot methodology). These parameters include:

1. Number of passwords owned
2. Content of passwords (proportion containing numbers and symbols) / password strength
3. Proportion of passwords system generated and self generated
4. Password length (number of characters)
5. Proportion of passwords entered on the user's behalf
6. Ease of recall of passwords (automaticity)
7. Proportion of user ids that are self generated or system generated
8. Frequency of use of passwords
9. Frequency of problems with passwords
10. Type of problem with passwords

The methodology involves an interview, followed by diary use and system logs. The interview is necessary to build an individual diary for each participant, which will reduce the effort required to fill it. The primary reasons for using a diary were that it could collect data across administrative domains, and that it could collect data about normal use as well as problems (see section 5.2.8). System logs are also taken so that the objectiveness and suitability for password research can be judged.

5.4.2 Study 2 (Questionnaire survey I)

Studies 2 and 3 are to collect information about the basic parameters of password use and performance, collecting this information from a corporate population of stakeholders. The method from study 1 was dropped because it was found to be more effortful and time-consuming than originally hoped, and was therefore considered to be inappropriate for a corporate population. Two questionnaires were therefore designed that took approximately 10 minutes to complete (the study 2 questionnaire can be found in the appendix, the study 3 questionnaire has one question removed but is otherwise identical). Both questionnaires had four types of questions:

- Questions about the respondents' password behaviour on the mechanism for which the respondents last required a password reset

- Questions relating the respondents' password behaviour on this mechanism to the respondents' behaviour with other mechanisms
- Questions about the respondents' general use of passwords
- Demographic questions (Qs 23-26)

The questions were derived as follows (Table 12):

Table 12 - Derivation of questions in Studies 2 & 3

Function of question	Derivation of question	Questions
To replicate findings relating perceived sensitivity, risk and value of resources to password behaviour	Adams et al	3, 4, 5
To replicate findings about automaticity vs conscious recall of passwords	Adams et al	3, 4, 5, 7, 16
Investigate importance of the number of passwords owned by users as a factor	Adams et al	15
To investigate the hypothesised importance of *frequency of password* ***changing*** as a factor influencing password system performance	Reason's model of human error	8, 10
Investigate the effect of password content re-use, which is part of *frequency of password use*.	Reason's model	12, 13, 14
To investigate the hypothesised importance of *frequency of password* ***use*** as a factor influencing password system performance	Reason's model	9, 11
To explore different facets of password content as a factor influencing password system performance	Literature of password memorability	17, 18, 19, 20
Explore automated password entry as a factor.	Allocation of function concept from HCI	6
Open ended questions about the users' methods for managing and using passwords	Open ended questions to capture factors not yet hypothesised	21
Users' insights into the cause of their password reset		2, 22

Study 2 participants were selected geographically: questionnaires were distributed by hand to three buildings at the research park belonging to BT where the author was based. Study 2 is named as being about *passwords in general.*

The questionnaires did not ask respondents to estimate the frequency of their password problems, as Study 1 showed them to be a relatively rare event which

respondents could reasonably be expected to have difficulty remembering accurately. Studies 2 and 3 therefore could not measure password system performance, though were able inform us about the parameters of password use and to contribute to diagnosis.

5.4.3 Study 3 (Questionnaire survey II)

Study 3 was designed to supplement Study 2 (see 5.4.2) by collecting data from a different sample of users in a corporate population. Although it collected information about password use in general, it also collected information about the use of Personal Identification Numbers (PINs), an important class of passwords.

The study 3 questionnaire was identical to the study 2 questionnaire (see section 5.4.2) with its first question removed because its answer was assumed to be known (see 7.2.2): it was distributed by a 3rd party attaching it to the end of a form necessary to get a voicemail system's PIN reset. Study 3 will therefore be considered as a survey of PIN use and performance.

However, how representative of PIN performance this authentication mechanism is cannot be known, as the prevalence, performance context of use of PIN based authentication has not yet been studied. The author claims that the feature of PIN authentication that distinguishes it the most from passwords in general is that the content is numeric rather than alphabetic. All PINs are by definition numeric. It will be assumed therefore that the PIN mechanism surveyed in study 3 is representative of all PINs.

5.4.4 Study 4 (University coursework system logs I)

The goal of the study was to improve our general understanding of password system performance, and to collect data about password use and content to aid in the diagnosis of performance. These goals were achieved by instrumenting a real-world authentication mechanism to record the timing, success or failure, and content of login attempts, and to allow us to compare these attempts to the passwords that should have been used. This allowed unique psychological insights that could not be accurately available by any other means.

The study was carried out on a sample of students at UCL, where the researcher was able to negotiate permission. Unfortunately, similar data was not available from the corporate environment, as permission was not given to instrument an authentication mechanism in BT.

Theoretically system logs are a rich source of data about password use and password system performance, as best practice dictates that they are kept (see 3.2.7). Study 1 found that the standard system logs of conventional UNIX workstations were inappropriate for password performance research. This study made use of an opportunity to collect system log data that was richer and more appropriate to the research objectives than had been available through standard UNIX logs.

5.4.5 Study 5 (University coursework system logs II)

Study 5 was designed to supplement the password performance and diagnostic data of Study 4 by improving the sample size by a factor of 10, so as to enable statistical inference tests about the performance of different types of passwords. This was achieved by repeating a simplified version of Study 4 over a number of years with more classes of students, and pooling the data.

Study 4 had captured login and error data for both passwords and pass faces, in a repeated measures design (see Brostoff & Sasse, 2000). Study 5 removed the Passfaces™ component of Study 4, as the IT infrastructure available to participants was found to significantly disadvantage Passfaces™ use (Brostoff & Sasse, 2000) and could not be changed.

5.4.6 Study 6 (Corporate helpdesk logs)

Study 6 was designed to gain objective password system performance data from a corporate sample, and to aid in diagnosing the performance observed. This supplements Studies 2 and 3, which collected subjective data from a corporate population, and studies 4 and 5, which collected objective data from students.

The study gives an incomplete picture of password system performance, as it examines records of helpdesk use rather than password use. These records are a direct measurement of important password system costs, but reveal little about their causes. However, the data can falsify some hypotheses about password performance (see *the repeat offender hypothesis*-section 8.3.3).

The results of the study will be optimistic, tending to overestimate password system performance. This is due to a systematic bias in the data: the helpdesk does not support all of BT's Information Systems, and individual users are likely to use a spread of supported and unsupported systems. This means that password resets of unsupported systems will not be recorded in the helpdesk's logs, and so the logs will underestimate the actual number of passwords reset for each user.

There is a further difficulty with this method. The helpdesk has no part in the provision of user accounts on BT's Information Systems, and so does not have access to information about the number of accounts on each of these systems. Without supplementary information, it is not possible to calculate the number of resets per system per user, which is an important performance measure. This supplementary information was available for some of the systems through the helpdesk staff's contacts, but its accuracy could not be validated.

5.4.7 Study 7 & 8 (Focus groups I & II)

Studies 7 and 8 were designed to begin validation of interventions described in Chapter 9 with BT users - as evidence suggested BT had more password problems than UCL (section 8.4.4). The effects of interventions would likely be greater at BT and so more easily detected. BT was the better place to test interventions. However, BT did not give permission for any of the interventions to be tried experimentally. BT also had a new policy that reduced the priority given to research, and so greatly reduced the availability of users for participation in research. Moreover, it was necessary to share research participants with another research project to achieve levels of efficiency acceptable to BT.

Focus groups were chosen, as they could be used to collect data from BT users about interventions without implementing them (a BT requirement), and would allow participants to be more easily shared by researchers (another BT requirement).

A set of interventions that relied on users changing their behaviour (rather than that relied on infrastructure changes) were chosen for two reasons:

1. The interventions required users to change their behaviour without putting in place mechanisms to enforce it. This would mean that users' attitudes to and perceptions of the interventions would be particularly important, which focus groups are especially good at exploring.
2. It was believed that these interventions had a greater chance of being implemented, as they required little investment in infrastructure and so were cheaper to implement at a time of economic hardship for BT.

Study 7 was conducted at a large research park at BT, where participants were generally higher corporate grades with little access to confidential customer records. Study 8 was conducted at a local office where many jobs require access to confidential customer records, and participants were generally of lower position in the hierarchy.

5.5 Summary

This chapter describes standard HCI evaluation techniques, describing them and their general strengths and weaknesses, and then discussing their strengths and weaknesses in relation to password research. The applicability of these methods to password research are summarised in Table 10. It was concluded that empirical data collection techniques were more useful at the current stage of research than analytic techniques. The purpose and methods of eight studies were outlined and summarised in Table 11.

Chapter 6

Study 1: Diaries & system logs pilot study

6.1 Introduction

This chapter describes a pilot study of a diary- and system log-based methodology for collecting basic data about password systems and their performance. Results are presented and discussed, both substantive results about the basic parameters and performance of certain password systems, and methodological results about the suitability of diaries as instruments for collecting data about passwords.

A pilot study was performed to assess diaries as the main data collection instrument for password performance data. Diaries show great promise as instruments for collecting this data (section 5.2.8). By using participants from UCL, it was possible to collect system log data, which could be used to assess the concurrent validity of the diaries.

It is understood that using the diary may influence the behaviour that is being studied (Breakwell, 1995). Breakwell suggests that the diary's instructions should include warnings to participants to be aware of this problem and not to let the diary change their behaviour.

The diaries in this pilot study did not contain such warnings, as the researcher had not been aware of the issue at the design stage of this study. However, it *was* hypothesised that participants would try to reduce the burden of completing a diary using 3 techniques (Table 13 below):

Table 13 - Hypothesised participant behaviour to reduce the burden of diary filling

Hypothesis 1	Not recording events in the diary
Hypothesis 2	Using passwords less frequently.
Hypothesis 3	Retrospective entry making in batches, instead of individual entries concurrent with the events they describe.

6.2 Method

6.2.1 Participants

Of the 6 participants, 4 were within the Computer Science Department (3 PhD students, one system administrator) where the author had access to the system logs.

The other 2 were undergraduates from the School of Library and Information Science, UCL, studying Information Management.

6.2.2 Procedure

The participants were given a 40 minute semi-structured interview about their passwords (the interview schedule is in Appendix 2). Interviews were recorded onto audio cassette if the participants gave their permission (5 of the 6 did).

The results of this interview (section 6.3) were used to personalise password diaries, which the participants were instructed to carry with them for one to two weeks, noting down each use of a password with its time and date shortly after it had occurred. If a problem occurred, participants were asked to record the nature of the problem. At the end of the period participants were asked to a debriefing interview, their diaries were examined and system logs obtained from the Computer Science Department's Systems Group.

6.2.3 Apparatus

The password diaries were constructed from paper, and were made to A6 size so that they might be easily carried in a pocket or purse, and so facilitate rapid recording of password uses to lessen the likelihood that password events would be forgotten before they could be noted down. To achieve such small size and maintain legibility, it was decided to use a 3 character code (*item id*) to identify each password that participants used, entering this code into a table with the date and time, and any problems. Each table contained a memory aid for the passwords and their *item ids*.

Diaries are effortful to fill in (Breakwell, 1995). Employing a multiple-choice/checkbox format in the diary would have reduced the effort required for record-keeping. However, this would have greatly increased the size of the diary, as much more space is required on the page to display the relevant information. In the trade-off between ease of data entry and portability, it was decided that portability was more important. Even so, the diary was designed to reduce effort; most entries would require only the writing down of a time, and a three character *item id*.

Example pages from a diary are shown in Appendix 3. Each diary consisted of the following:

1. A front page with a serial code identifying the diary's owner and the number of the diary, which maintains the privacy and anonymity of participants in case of loss
2. Two pages of instructions about the use of the diary

3. A section where *item ids* could be generated for any new or expired and replaced passwords participants acquired during the course of the study, with instructions on how to do so,
4. The password use record tables themselves, including customised memory aids. There are ten slots per record table, and 22 tables, one each per double page spread.
5. A section where participants could record the details of how and why they shared passwords with other individuals during the course of the study
6. A section in which participants could make their own notes,
7. A back page, which gave instructions about returning the diaries if they had been lost and found.

Participants from the Computer Science department used either SUN or SGI workstations. Participant's consent was given for the release of the Computer Science Department (CS) systems logs that recorded their password use, covering a period from 2 weeks before password diary use to 2 weeks after. System logs were captured by using the UNIX command *grep* to select relevant items from the UNIX system log *wtmpx* contained on the participants' workstations.

6.3 Results

6.3.1 Substantive

Summary statistics for each participant are shown in Table 14 and Table 15. The average number of passwords was 10.5, the minimum 6 and maximum 14. The frequency of use (logins per week) fell into 3 bands: low (7,11); medium (29,34); high (91,94). Of the 352 diary recorded uses of passwords and PINs, 11 errors were reported by 3 participants, which is approximately a 3% error rate. Typographical errors were in the majority (7) giving an typo rate of 2%, with *other errors* making up the remainder (1% error rate). No errors were reported for the 9 recorded PIN uses.

Table 14 - Summary by participant of password frequency of use

Participant	No of Passwords.	Logins/week	Total logins	Min/pwd/week	Max/pwd/week	Mean
C	6	7.0	7	0	7.0	1
B	8	10.5	21	0	8.0	3
D	11	22.4	32	0	11.9	3
F	14	33.5	67	0	27.0	5
G	10	91.0	91	0	34.0	10
E	14	93.8	134	0	42.7	10

Table 15 - Descriptive statistics of details of participants' passwords

	Percentage	Mean	Median	Mode	St. Dev.
Self generated	50	-	-	-	-
Contains symbols	15	-	-	-	-
Contains numbers	79	-	-	-	-
Length (characters)	-	7.2	6	4	10.5
How often changed per year	-	.9	0	0	1.4
Change forced	18	-	-	-	-
Automated entry of password by mechanism	5	-	-	-	-
Passwords that came to mind automatically	43	-	-	-	-
Self generated user ID	15	-	-	-	-

6.3.2 Methodological

It was possible to discover substantially detailed information about interviewees passwords (Table 15). Participants were able to give details about all of their passwords during a 40 minute interview. However, detailing each and every password was tedious, and some details subsequently were discovered to be incorrect.

Table 16 - Descriptive Statistics for Participant D's System Logs

Condition	Mean Logins/Day	Standard Deviation	No. of Days Observed
D Pre-diary	0.3	0.5	10
D Diary	0.7	0.6	16
D Post-diary	0.5	0.6	4

Table 17 - Descriptive Statistics for Participant E's System Logs

Condition	Mean Logins/Day	Standard Deviation	No. of Days Observed
E Pre-diary	0.4	0.5	14
E Diary	0.7	1.2	11
E Post-diary	0.6	0.6	14

Table 18 - Descriptive Statistics for Participant F's System Logs

Condition	Mean Logins/Day	Standard Deviation	No. of Days Observed
F Pre-diary	1	0.4	14
F Diary	1	0.0	14
F Post-diary	1.8	1.4	14

There would appear to be 2 different patterns in the 3 participants: increase in the number of logins during the period of diary use (participant D), and increase in the number of logins after diary use (participants E and F). However, it would not be appropriate to test for these differences using inferential statistics due to the small number of data points.

The hypothesis that diary use leads to reduced password use (Hypothesis 2) is not supported by the data, as only cursory examination of Table 16 to

Table 18 reveals. Although conclusions must be tentative due to the small sample size, it appears that diary use did not significantly reduce the usage of passwords.

Diary Accuracy

41 (12%) of the reported uses were of passwords where system logs were available. System logs were used as a baseline to check the accuracy of the diaries. The daily totals of logins recorded in the diaries were taken away from system log daily totals. Summary statistics are in Table 19.

Table 19 - Summary statistics of System logs daily totals minus diaries daily totals for password use

No. of observations	Mean	Median	Mode	Min	Max	Std. dev.
41	-1	-1	-2	-11	8	3.4

A distribution of these differences was prepared (Figure 13). The distribution is approximately normal, and shows that participants tended to over-estimate the numbers of passwords they had used. On one day, one participant over-estimated (compared to system logs) by 11 uses of a password. Over-estimations in diaries by between 3 and 6 uses a day were frequent.

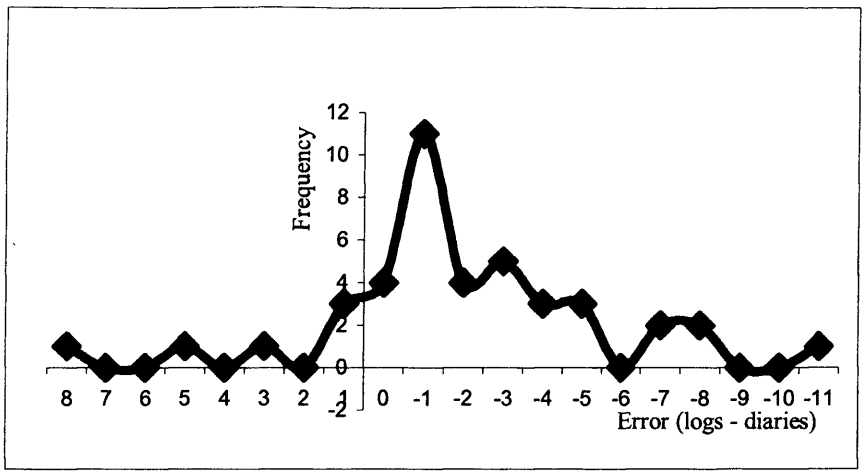


Figure 13 - Frequency distribution of diary errors compared to system logs, in recording number of password uses

There was a strong positive correlation between the numbers of uses in the diaries and in the system logs (Pearson's r = .749, df=39, p<.0001, significant). However, a correlation of this size gives an R^2 of 0.56, which means that 44% of the variance in the data is unexplained.

6.4 Discussion

6.4.1 Diary Accuracy

The password diary appears to be promising as a method for measuring the numbers of uses of passwords. It shows good correlation with an 'objective' measure of password usage - Computer Science Department System logs. . This is a methodological contribution, as diary methodologies have not been used in this field before.

The diaries show a small systematic bias in favour of over-estimation, and large random bias. The systematic bias was in an unexpected direction. Hypothesis 1 was that participants would act to reduce the burden imposed upon them by not recording some of their password uses. Several of the participants reported this during debriefing but it could be not be detected in the diary or log data.

Hypothesis 2 - that participants would use passwords less often, and so avoid making diary entries was not detectable in the data, though two of the participants reported they had behaved this way on more than one occasion.

Hypothesis 3 - that participants retrospectively filled in their diaries making entries in batches was partially supported by the data. The analysis was ambiguous but

suggestive, and agreed with the participant reports. Batch completion of the diaries may explain the difficulties encountered in matching some diary entries to records in the system logs. Perhaps participants could not accurately recall the time, and so entries made would reflect this problem. To measure the former problem and counter the latter, it would be necessary to allocate the task of recording the time away from the participant to a more accurate source. The most elegant solution would be a diary that time-stamps each entry.

Another method would be to instruct participants that diary entries may not be made later than the same day as the event. However, there is evidence that participants have already ignored instructions about the timing of entries, and no evidence that further instruction will be effective.

The most efficient but potentially least effective solution is to go without diaries, and rely on the system logs for data. This would drastically curtail the number of passwords about which data could be collected (nearly 90% of password uses recorded in this study would not have been captured using this strategy).

6.4.2 System log critique

Diary studies are known to have high drop out rates (Breakwell, 1995). All the participants who expressed an opinion about using the diary were negative, reporting that it was onerous, and the length of time they used the diary was the most they would consider using it for. It appears more likely than not that their avoidance behaviours were significant, but were not picked up in the analyses that searched for them. It is necessary to try and objectively measure participants' reported "burden lifting behaviours" for there to be trust in the results of future password diary studies.

Assuming that the method of comparison was good, the only remaining source of error is the system logs data.

Several problems were encountered when collecting and analysing the system logs.

At the time of the study, the Computer Science Department (CS) officially supported: SunOS, Solaris, Irix, Windows 95, Windows 98, Windows NT, NTrigue (emulated Windows NT), and FreeBSD. These systems record system logs in different ways: storing different information, using different nomenclature and locations. The participants in CS used a mixture of Irix, SunOS and Solaris. When asked for "system logs" of the participants, the systems group initially forgot about the Irix users.

```
<username>  ttyq1                Fri Feb  6 10:21 - 15:54  (05:33)
<username>  ttyq0                Fri Feb  6 10:21 - 15:54  (05:33)
<username>  console              Fri Feb  6 10:20 - 15:54  (05:33)
<username>  ttyq1                Thu Feb  5 10:22 - 16:45  (06:22)
sonic,<username>,pts/24,henry,@CS,19980220 1638,19980220 1830,0:01:52,
henry,<username>,ttyq1,,@LOCAL,19980220 1637,19980220 1830,0:01:52,
henry,<username>,ttyq0,,@LOCAL,19980220 1637,19980220 1830,0:01:52,
henry,<username>,console,,@LOCAL,19980220 1637,19980220 1830,0:01:52,
sonic,<username>,pts/5,henry,@CS,19980219 1909,19980219 2012,0:01:03,
sonic,<username>,pts/15,henry,@CS,19980219 1028,19980219 1641,0:06:12,
henry,<username>,ttyq2,,@LOCAL,19980219 1131,19980219 1131,0:00:00,
```

Figure 14 - A small sample of wtmpx system log data. henry and sonic are workstation names. Participants' usernames have been replaced with <username>

Successful password use

A small sample of the raw log data is shown in Figure 14. The *wtmpx* logs did not record instances of passwords being typed by the participants, recording instead instances of successful connections being made between machines by particular protocols, and terminal windows being opened. On further investigation was found that there is no direct mapping between connection or terminal establishment and a user entering password. Some protocols may or may not require the user to enter passwords, and one action by a user may result in several terminals being established and logged. Password use must be inferred from the system logs and what is known of the user's habits and preferred methods for connecting between machines. This is in effect informed guessing. Though it is informed, it is still guessing and so prone to error.

The process of inferring involves removing the distracting redundant entries from the log (laborious, painstaking work) so that useful log entries remain and can be interpreted. Useful entries can be taken out by mistake. This source of error could be removed or standardised by automating the log 'cleaning' process, but this would not remove the ambiguity of some of the log entries.

Unsuccessful password use

Unsuccessful use of passwords is logged in a different way. In standard UNIX, 5 failed attempts at using a password must be made consecutively before it will be logged. This has been reduced to 2 in CS. Failed attempts at password use do not have to be inferred, however they are most often not recorded. The system does not classify

failed attempts, severely reducing the utility of the logs. For instance, entering an often used but inappropriate password would be recorded in the same way as a typographical error.

There are two approaches to collecting this error type data. The first approach is altering the password authentication system.

The source files of UNIX "login" applications are sometimes in the possession of systems administrators. It is possible that the login application could be altered to log each failed attempt, or even to categorise the failures. The second approach is to alter the diary.

Impact of password error

The purpose of the diary is to gauge the size and scope of the problem of password memorability. So far, the impact of the errors has not been considered, and is not recorded in the current design. It should be recorded in future designs. It is suggested that a list of options be supplied to participants, reducing the data entry burden upon them. The categories in the list should be impacts of password error, indicate differing levels of inconvenience, and be applicable for users in many different situations. A suitable class of measures would be "generic error correction behaviours". For example, re-entering a password would be less serious impact than contacting the system administrators.

For effective logging of password use it is necessary to consider the mixture of applications running as well as the mixture of operating systems, as some may require password use. A software audit is recommended. Otherwise it is possible to miss out software that uses passwords, as happened in this study with XLOCK-a password protected screensaver that the author did not know about at the time of data collection. There is another reason that audits should be included and system administrators recruited as far in advance of data collection as possible-not all password using applications will keep logs (for example XLOCK), but may be easily adaptable to do so, and without significant risks. Most applications in UNIX can be modified to generate a message for *syslog*, UNIX's general logging facility (Garfinkel & Spafford, 1996).

An additional methodological problem was discovered-study participants may treat separate authentication mechanisms as the same if they use the same password content on them - one participant did not regard XLOCK as different to his network login - despite its use for a different purpose and its consequences for data logging and analysis. Given the advanced computing knowledge of the participant (final year Computer science PhD) this may be a problem for system administrators, and may be

more acute of stakeholders who have less technical computing backgrounds. This will affect user reports data collection (diaries, interviews, questionnaires, focus groups, etc.) as well as logging in studies. This problem may be tackled by recording the context or purpose of password use, as well as its happening.

The problems with the password diaries were not limited to the diarists' behaviours. The nature, design and media of the diary led to inefficient methods for construction - which is appropriate in small studies, but would not scale well. Each diary has to be individualised with its owner's passwords and printed pages processed to make a booklet by hand. Precious data can be lost if the diarists' handwriting cannot be understood. Ideally the diary would configure itself to suit each diarist, do away with handwriting and when completed automatically or semi-automatically transfer the data to computer, with little intervention from researcher or participant.

These goals, and others mentioned throughout the discussion could be met if the diary was an electronic device (PDA, etc.). They are more expensive, weigh more and are more vulnerable than paper based diaries. However, the additional functionality and attractiveness of the PDA may even encourage individuals to participate.

6.5 Substantive summary & conclusions

Real world data has been collected that begins to answer Research Question A - *What is the performance of password systems in actual use*?

All participants in Study 1 used a password at least once per day, with a total of 361 uses of passwords recorded. This is roughly equivalent to 2 uses per day, 6 uses per day, and 18 uses per day (for a 5 day week). Participants were categorised into 3 groups according to their password usage:

- Low useage, 7 to 11 uses/week;
- Medium useage, 23 to 34 uses/week, and
- High useage, 91 to 94 uses/week

The range of uses of individual passwords was from 0 to 43 per week, or 0 to 6 per day. 11 errors in password use were recorded, giving an overall login error rate of 3%. Two thirds of the errors were typos. This suggests that most login problems are simple to recover from at UCL, and that login problems at UCL are rare in absolute terms.

Data was collected that puts parameters on the password system performance observed. This data will be useful in designing highly controlled laboratory stdies, and for making comparisons with performance of password systems in other real-world

contexts (such as BT) - which may later help in the diagnosis of password system performance (Research Question C). The average number of passwords that participants reported owning was 10.5, the minimum 6 and maximum 14. Their average length was 7.2 characters, and 80 % contained numbers, 15% contained symbols. These participants chose relatively strong passwords.

Half of passwords owned had been chosen by the participant, the other half chosen by the mechanism for them. Participants reported that 18% of their passwords had enforced password expiry, and that their passwords were changed on average once per year.

6.6 Methodological summary & conclusions

Progress was made towards answering Research Question B - How can the performance of password systems be measured? Password diaries and system logs were shown to collect useful data. Their performance is summarised below, with recommendations for their improvement.

6.6.1 Password Diary

Overall there was a good correspondence between diary reports and system logs, with a small systematic bias. Diaries over-reported password use by an average of 1 use per person per day in the study, but there was some difficulty matching times recorded in diary to times recorded in *wtmpx* logs (see system log summary). However, the data did not show participants altering their behaviour to avoid making diary entries. Diaries therefore provisionally show good concurrent and face validity as a technique for password research.

Amendments to diaries were suggested to make them fitter for password research. A number of these came together in the recommendation for an electronic password diary, using a personal digital assistant or palm-top computer. This would allow studies to scale up more easily (through efficiencies gained in diary configuration, transfer of data, etc.), and to control for or measure retrospective diary entries (with a daily deadline and associated statistics for "missing value" imputation; or time/date stamp for entries).

In addition to the data already captured, it was proposed that future studies should capture error correction behaviours, the consequences of errors and error types, and if possible what the passwords are used for. It was also suggested that diary studies should be preceded by software and operating system audits with systems

administrators to identify password mechanisms (as diarists may treat separate password systems as the same).

6.6.2 System Logs

There are challenges in using UNIX authentication system logs (such as *wtmpx*) for analysing password system performance. They record a successful password use as several terminals opening, not as a password being typed. Logs are therefore ambiguous and full of distractors and must be interpreted according to what is know about the habits of the user. It's time consuming cleaning log data and so more difficult to be consistent in interpretations. There is therefore error proneness in processing logs by hand. Automated log analysis/log cleaning would be more reliable.

Moreover, 2 failed attempts are required before a record is made (in the study's environment, 5 in standard UNIX configuration), and errors are not categorised into types. UNIX system logs therefore do not have enough content to accurately record password system performance, or be particularly useful in diagnosis. To make optimum use of system logs in password research it may therefore be necessary to implement new logging software.

In designing this logging software it should be noted that there is much variation in logging systems. Different systems record same events in different ways, logs are not stored centrally (though some may be configured to), and users can identify completely separate password system as being the same one. The researcher inadvertently missed out some password mechanisms from a system log study. It is important therefore to perform an audit, and involve system administrators at an early stage.

6.7 Contributions

6.7.1 Substantive

A start has been made on identifying the basic parameters of password system performance. Real world data has been collected that begins to illustrate:

- Number of passwords owned by users at a large acadmic organisation
- Frequency of their use, and
- Frequency of problems occurring

6.7.2 Methodological

Diary based password research has been attempted for the first time. This will improve researchers' appreciation of the diary's strengths and weaknesses as a password research tool. Workarounds have been suggested for some of its shortcommings. The strengths and weaknesses of system logs for password research have also begun to be outlined, and improvements to logging systems and studies identified.

Chapter 7

Studies 2 & 3: Survey Data

7.1 Introduction

This chapter presents the first set of data collected from BT. The previous chapter began to address the measurement of basic parameters of real-world password systems' performance, by collecting empirical data from 6 UCL based users about their: number of passwords (mean=10.5, range=6-14 passwords), frequency of password use (between 2 and 18 uses a day), frequency of password problems (mean error rate of 3%), and password construction (mean password length = 7.2 characters; 80% contained numbers, 15% contained symbols). However, it would not be feasible to repeat this in BT as the data collection was time consuming for participants, and moreover did not capture the consequences/recovery behaviours required to cope with the problems that were recorded.

To reach users in BT it was decided to use a questionnaire survey methodology instead: participants could give useful data at a time and place of their choice, and in a matter of ten minutes rather than the 40 minute interview and subsequent diary use employed in the previous chapter.

7.2 Methods

7.2.1 Study 2

152 questionnaires were distributed by hand in Study 2 to three buildings selected pseudo-randomly at BT's research park, asking respondents to give details about the last password that they were forced to have reset at a helpdesk, the circumstances surrounding it, and some details of their other passwords and approach to password management (see section 5.4.2 for more details and Appendix 4 for the questionnaire itself). Within each building, half the floors were selected for questionnaire distribution, and approximately every fourth occupied desk approached and the occupant asked if they would participate in the survey. These questionnaires will be referred to as the ***passwords-in-general*** survey.

7.2.2 Study 3

A version of the questionnaire was designed for Study 3 which was identical to the Study 2 questionnaire (see section 5.4.2) except that its first question had been removed as its answer was already known due to the distribution technique used: the last password reset for these respondents was for a voicemail system used widely within the organisation.

This voicemail system's authentication mechanism had the following properties:

- It's passwords could contain only numbers
- They must also be exactly 6 digits (i.e. a 6 digit PIN),
- It's PINs expired after only a month,
- The last 12 versions of the PIN could not be reused.

Study 3 will therefore be called the ***6-digit PIN*** survey. This PIN mechanism is a severe example of a password system - the security policies applied to it increase the user's cognitive workload (see all policies in section 3.2.8; *strong password content* policy is being applied here from a psychology of memory viewpoint because the user is forced to choose a number instead of a more memorable word; furthermore, constraints that are put on the number to prevent it being easy to guess are likely to make it more difficult to recall).

The questionnaire was attached to the fax-back form necessary to get the voicemail's PIN reset. Study 3 respondents are therefore selected by having recently suffered a PIN problem (they had to request a voicemail PIN reset to receive the questionnaire).

7.2.3 Response classification

Responses were classified into 4 categories, relating to the cause of users last password reset (Table 20). The answers to two questions in the surveys were examined for each classification (see Appendix 4 for full questionnaire):

1. Please describe below why it needed to be reset?
2. What caused the password management system you described to break down for the BT work related password you last requested to have reset?

The author acted as sole judge. If both answers suggested the same classification, the questionnaire was given that category. If only one of the answers clearly suggested a classification but the other answer neither confirmed or contradicted it, the questionnaire was also given that category. However, if the two answers contradicted each other, or neither answer clearly suggested a classification the questionnaire was given a *can't classify* category.

Users reports were treated as truthful and accurate. Questionnaires were anonymous and used neutral language to reduce responses aimed at improving social desirability (Oppenheim, 2000). It was assumed respondents had insight into the cause of their reset.

Table 20 - Categories for causes of password resets

Category	Definition	Example response
Forgetting	The respondent could not recall their password.	"Very low usage - forgot password" n.b. the response indicates the password couldn't be retrieved from memory, but no other memory related phenomena are described.
Slips, Lapses, Mistakes	A slip, lapse or mistake related to password content was implicated in the problem (see section 3.3.5),	"user error - I was using the password for a similar system which had a different password" n.b. this response describes a slip or a mistake rather than an incident of forgetting.
Technical or Organisational	Causes were implicated, unconnected to a failure of human memory for the target password content.	"because it was in use on both an NT and Win95 machine concurrently. This gives some form of conflict and therefore lock out" n.b. this indicates that technical incompatibilities between different parts of the authentication infrastructure caused a problem; no aspects of human memory are implicated
Can't Classify	Not enough information was available to classify the response into one of the other categories.	"not used for a long time" n.b. this response could describe either *forgetting* or an incident of password expiry, which is a *technical or organisational* cause of password resetting. No other information was available to decide between these two categories.

7.2.4 Statistical tests

Chi squared tests were used for inferential statistics. Unless otherwise stated, the observed values were counts of problems in each group, rather than proportions of problems within each group. Values of Chi squared were calculated by manual construction of formulae in MS Excel 2002 to perform the calculations described by Rosenthal and Rosnow (1991). Significance values were calculated from these using Excel's CHIDIST function.

7.3 Results

152 *passwords in general* questionnaires were distributed in Study 2. 96 questionnaires were returned, giving a response rate of approximately 63%. The respondents were 13% higher management, 59% lower management and professional grades, and the remaining 28% technical and administrative staff.

137 responses were received for the *6 digit PIN* questionnaire (Study 3)- the researcher has not been able to obtain information about the exact number of questionnaires distributed. However, approximately 60 PIN resets were requested per

week (based on data from Jan and Feb 1999) and the questionnaire was being distributed for approximately 18 working weeks. A 12% response rate is therefore inferred. Respondents were 22% higher management, 32% lower management and professional grades, and the remaining 46% technical and administrative staff.

7.3.1 Number of passwords

Respondents' numbers of passwords are displayed in Table 21. Study 3 respondents had fewer passwords on average than Study 2 respondents, with 11 passwords versus 16. However, the range of passwords owned was higher in Study 3, even though this group was more tightly focused around the mean.

Table 21 - Number of systems owned that use passwords. *number of passwords owned, not just six digit PINs.

Source	Mean	N	Minimum	Maximum	Std. Deviation
Study 2 - passwords in general	***16.2***	86	3	52	9.7
Study 3 - 6 digit PIN	***11.3****	121	2	65	7.9

7.3.2 Types of problem

Respondents were asked to describe the causes of the problem that last required them to have a password reset. Table 22 shows summary statistics. Forgetting the PIN or password was the biggest source of error - accounting for almost half the total responses and 60% of those that could be classified. Problems that could not be attributed to memory accounted for 25% of all responses and 30% of responses that could be classified. 10% of responses that could be classified could be attributed to slips, lapses, and mistakes.

From this point on, results will be given separately for the Study 3's 6 digit PIN system, and for passwords in general from Study 2.

Table 22 - Overall sources of problems for questionnaire respondents

Error	Number	% of total responses	% of responses that could be classified
Can't classify	40	17.2	-
Technical or organisational	59	25.3	30.6
Forgetting	114	48.9	59.1
Slips, lapses, & mistakes	20	8.6	10.4
Total	233	100.0	100.0

Table 23 - Technical and organisational problems, Study 3

Source	Count	Percentage
Expiry	13	41%
Unknown server problem	8	25%
Comms failure	1	3%
Expiry+strikeout	1	3%
Inherited equipment	7	22%
Covering for absence	1	3%
Striked out due to tampering	1	3%

Table 24 - Technical and organisational problems, Study 2

Source	Count	Percentage
Operating system conflicts	1	4%
Expiry	9	35%
Expiry+autologin scripts	3	12%
Autologin across incompatible NT domains	2	8%
Unknown server problem	2	8%
Server upgrade	2	8%
Software installation	1	4%
Caps lock	2	8%
New system	1	4%
SSO bug	1	4%
Login interrupted by Comms failure	1	4%
Inherited equipment	1	4%

7.3.3 Frequency of use

The questionnaires asked for the uses per month of the password for which respondents last required a reset. The passwords-in-general responses are summarised in Table 25, and displayed in Figure 15.

Table 25 - Summary statistics of frequency of use of passwords/month in Study 2

N	Mean	Median	Mode	Min	Max	St.dev.	33.3° %ile	66.6° %ile
85	30.3	4	30	.08	300	53.2	1	30

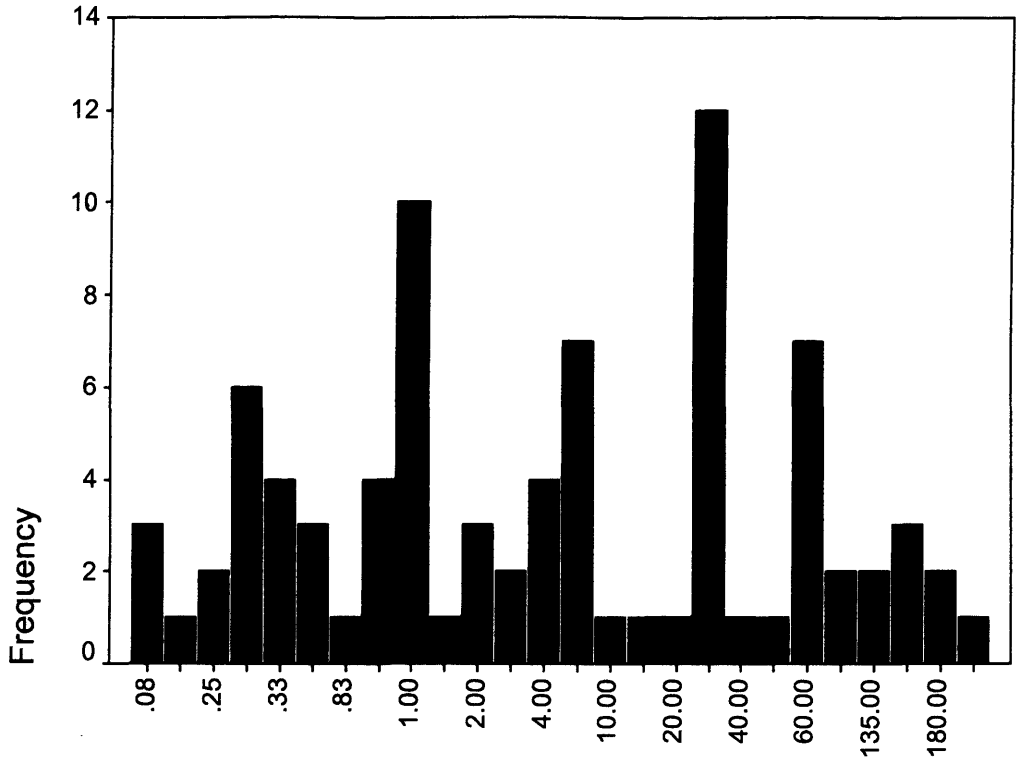


Figure 15 - Frequency of use of passwords/month, Study 2

The responses divided naturally at the 33rd and 66th percentiles, making three groups, which were:

Light = <1 per month

Medium = 1 to 29 times/month

Heavy = 30 + times/month

Proportions were calculated of how many respondents belonged to each frequency of use group, for both questionnaires. The proportions for the 6 digit PIN users (Study 3) are shown in Figure 16, the proportions for users who responded about passwords-in-general (Study 2) are shown in Figure 17. The proportions are very different. Study 3 respondents were in the majority heavy users of their PIN. However, Study 2 respondents tended to be light users of the password they most recently had a problem with.

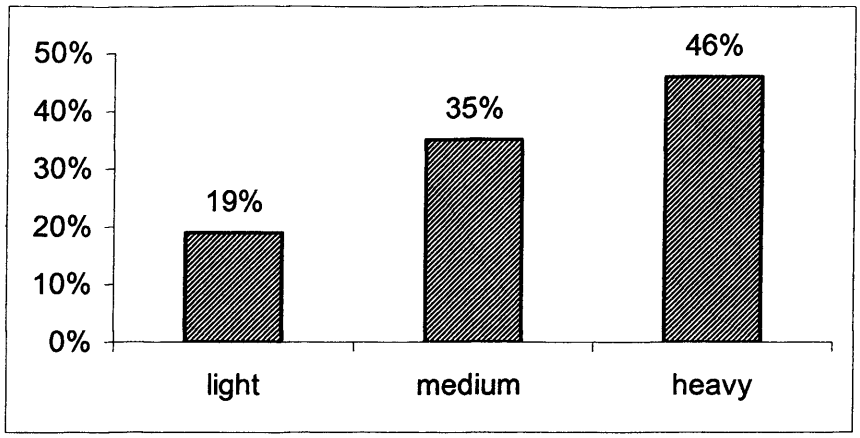


Figure 16 - Percentage of Study 3 responses about 6 digit PINs by frequency of use. n=127

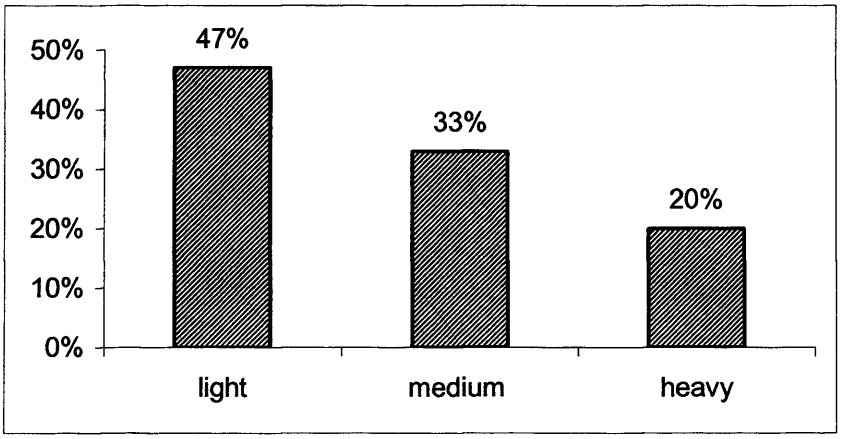


Figure 17 - Percentage of Study2 responses about passwords-in-general by frequency of use. n=85

Frequency of use responses were combined with types of problem responses, to show which users of which systems suffered which kind of problems. Table 26 shows the data for 6 digit PINs (Study 3), and Table 27 shows the data for passwords-in-general (Study 2).

These data were refined further, creating a problem profile in Figure 18 for PINs (Study 3), and Figure 19 for passwords-in-general (Study 3). Each profile shows three clusters, one cluster for each frequency of use: light, medium, and heavy. Each cluster contains three bars which add up to 100% of respondents in that category, with the bars representing how many respondents suffered problems of a particular kind.

Table 26 - Study 3 sources of error, according to frequency of 6 digit PIN use

6-digit PIN, with use that is:	Technical or Organisational	Forgetting	Slips, Lapses, Mistakes	Row total
Light (<1 per month)	6%	12%	1%	19%
Medium (1 to 29 times/month)	7%	26%	3%	36%
Heavy (30 + times/month)	11%	26%	8%	45%
Column total	24%	64%	12%	100%

n=108 missing cases/unclassifiable =30 (22%)

Table 27 - Passwords-in-general sources of error according to frequency of password use, from Study 2.

Passwords in general, with use that is:	Technical or Organisational	Forgetting	Slips, Lapses, Mistakes	Row total
Light (<1 per month)	5%	29%	0%	33%
Medium (1 to 29 times/month)	11%	21%	6%	38%
Heavy (30 + times/month)	18%	6%	5%	29%
Column total	33%	56%	11%	100%

n=66 missing cases/unclassifiable =24 (27%)

The two profiles appear to be quite different. 57% of respondents who were heavy users of a 6 digit PIN had forgotten it. However, only 20 % of users who responded about passwords-in-general which they used heavily had forgotten them. The proportions of forgetten 6 digit PINs and passwords-in-general were significantly different (Chi^2= 7.2, df=1, p=0.0075).

There are other differences. Forgetting is an important problem for light users of both 6 digit PINs and passwords in general. The proportions of each group reporting forgetting is 65% and 86%, though there is no statistical difference (Chi^2= 2.6, df=1, p=0.10). There is a significant difference in the problem profiles of medium users of 6 digit PINs and medium users of passwords-in-general (Chi^2= 6.2, df=1, p=0.045). Heavy users of passwords-in-general suffer proportionately about twice as many problems that are due to technical or organisational causes than do heavy users of 6 digit PINs (Chi^2= 9.0, df=1, p=0.003, significant). Finally, heavy users of passwords-in-general have highly significantly different problem profiles than light users of passwords-in-general (Chi^2= 35.8, df=1, p<0.001).

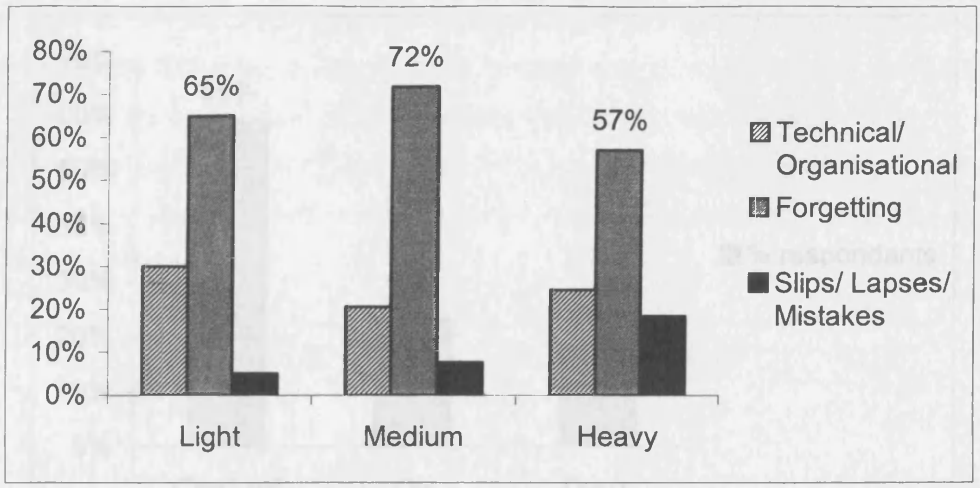


Figure 18 - Problem profile for 6 digit pins, by frequency of use, n=108

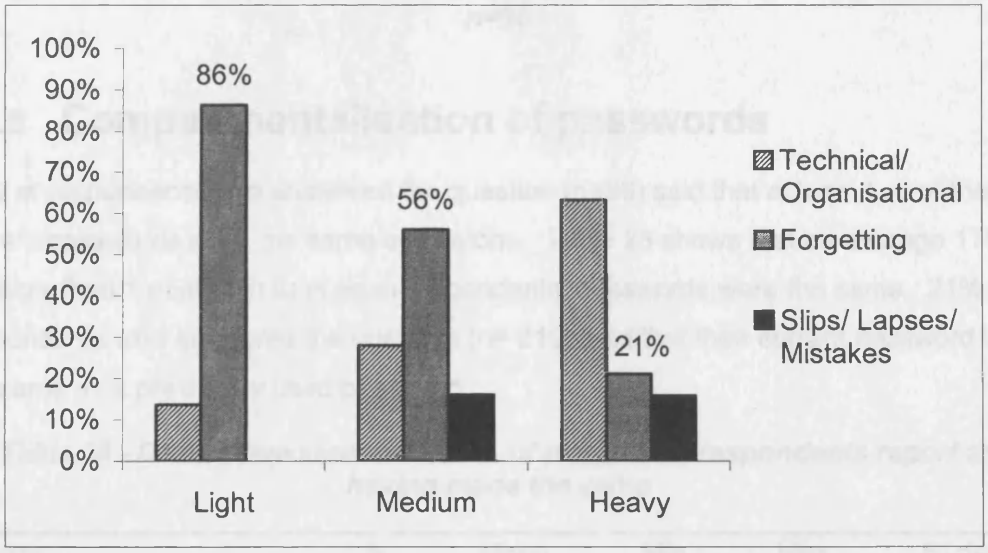


Figure 19 - Problem profile for passwords in general, by frequency of use , Study 2, n=66

7.3.4 Append digit

Respondents were asked if they appended a digit to the password, as a strategy for coping with historical compartmentalisation of passwords. This is not an appropriate question for Study 3 users, since their password is wholly numeric. However, respondents to the passwords-in-general questionnaire (Study 3) had alphanumeric passwords, where this question makes sense, and their responses are shown in Figure 20. Where respondents answered the question, and their answer could be clearly interpreted, the majority of respondents (56%) did append a digit to their password.

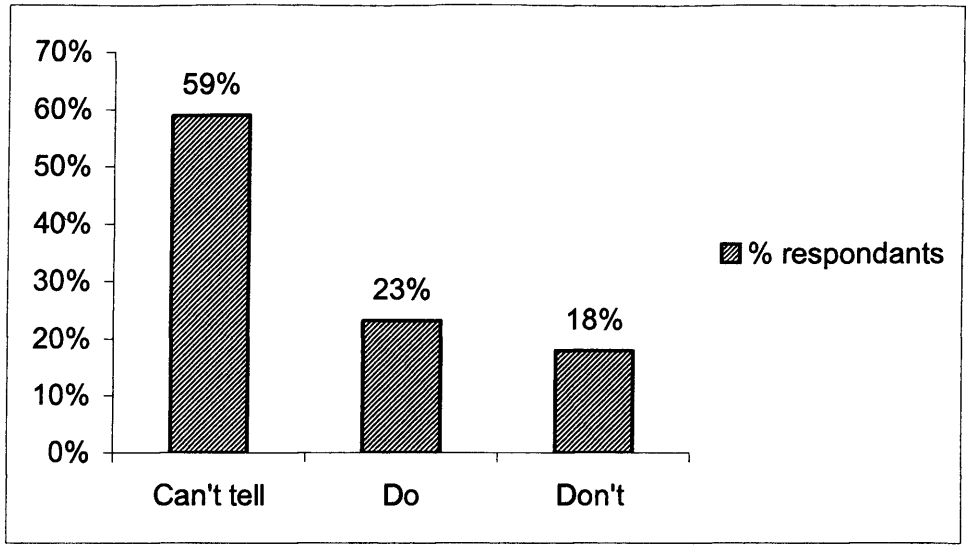


Figure 20 - % of respondents (Study 3) appending a digit to their passwords, n=95

7.3.5 Compartmentalisation of passwords

59% of respondents who answered the question (n=85) said that at least two of their current passwords were the same as this one. Table 28 shows that an average 17% (a bit more than 1 person in 6) of each respondents' passwords were the same. 21% of respondents who answered the question (n= 219) said that their current password was the same as a previously used password.

Table 28 - Descriptive statistics for % of passwords respondents report as having made the same

Source	N	Mean	Min	Max	St.dev.
Study 3 group	16	12.8%	0	50%	18.0
Study 2 group	58	18.3%	0	100%	24.3
All samples	74	17.1%	0	100%	23.1

7.3.6 Password changing and expiry

Another frequently employed password policy is password expiry, where users are forced to change their passwords at set intervals. Respondents were asked whether their last serious password problem occurred just after changing a password or a password had expired. 29 of the 233 respondents suffered problems just after changing, 17 % (a bit more than 1 person in 6). For each of the three frequency-of-use groups, a ratio was calculated for the number of respondents who suffered their last serious password problem just after changing to the number of respondents who suffered their serious problem at some other time (Figure 21). There was a statistical trend that these ratios were different (Chi^2= 5.5, df=2, p=0.06). Lightly used passwords

had a 1 in 20 chance of suffering a problem just after being changed, heavily used passwords had 3 chances in 20, and with medium use passwords had a chance of 5 in 20. The ratio for lightly used passwords was statistically different to the other frequencies of use combined (Chi^2= 4.4, df=1, p=0.04). Passwords that had medium use also had a statistically different ratio to the other frequencies of use combined (Chi^2= 3.8, df=1, p=0.05).

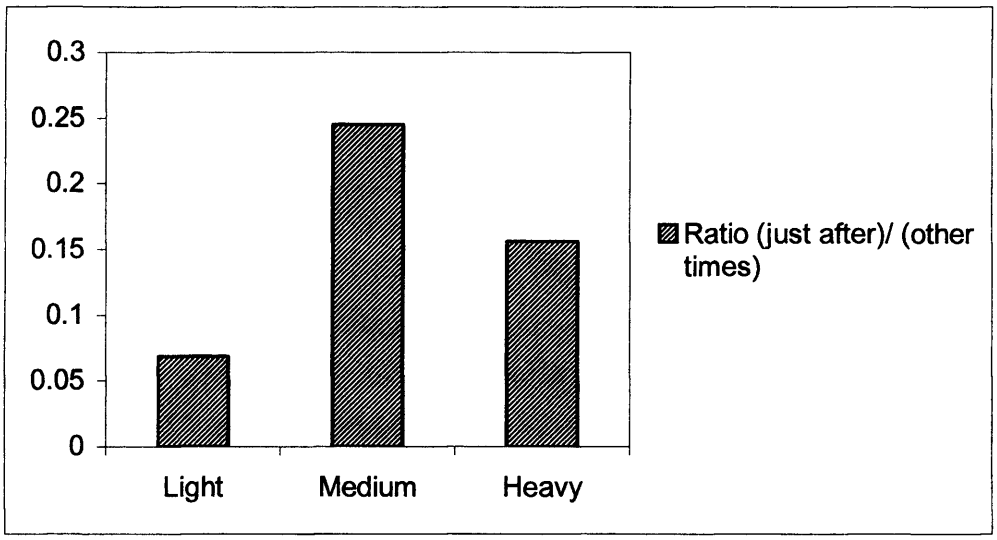


Figure 21 - Ratio of problems that occured just after changing a password to problems occuring at other times, for light medium and heavily used passwords, n=213

Table 29 appears to show that with increasing frequency of changing passwords comes increasing likelihood that there will be a problem just after the password has changed. However, it should be noted that the results for the first column come from a very small number of data points, and so cannot be relied upon statistically.

Table 29 - Relationship between frequency of password changing and occurrence of problems just after password changing (data from studies 2 and 3)

Frequency of password changing	Every 3 months	Every month
% of problems occurring just after changing	10%	14%
N	10	112

Table 30 shows how the proportion of different problems experienced by both groups combined can change with frequency of password changing. There is a statistical trend in the table (Chi^2= 6.3, df=4, p=0.17). As the frequency of changing a password increases, the proportion of slips, lapses, and mistakes increases too. The proportion of technical/ organisational problems remains fairly level, and the amount of forgetting problems decreases. The trend does not reach statistical significance. However, the

numbers of responses is very small, and approximately 800 responses would be required per frequency of changing group for an 80% chance of detecting effects of these hypothetical sizes (w= 0.05, 0.11 and 0.16) (cf. Rosenthal & Rosnow, 1991).

Table 30 - Relationship between frequency of password changing and types of problems leading to password resets

Frequency of password changing	Technical or Organisational	Forgetting	Slips, Lapses, Mistakes	N
Never	33%	67%	0%	18
Every three months	24%	67%	10%	21
Every month	29%	55%	16%	96

Finally, the percentages of users appending a digit to the end of the password was calculated for respondents who changed their password less often than once every 3 months (low change frequency), and for those who changed them every three months or more frequently (high change frequency). 18% of the low change frequency group appended a digit to their password, while 64 % of the high change frequency group did so (n=22 for both groups, passwords-in-general questionnaire). This difference was highly significant (Chi^2= 9.6, df=1, p=0.002).

7.3.7 Writing down passwords

BT's password policies mandated that passwords should not be written down. The questionnaires asked respondents if they wrote down passwords, and if so qualitatively how much they did so. Figure 22 shows the proportion of different problems suffered by respondents who claim not to write down any of their passwords versus respondents who claim to write down some or all of their passwords. The writing down of passwords is associated with a 9% drop (from 67 % to 58 %) in the proportion of forgetting problems compared to not writing down passwords. This drop is not significantly different (Chi^2= 1.2, df=1, p=0.26). However, a hypothetical effect of this size (w=0.09) would have only a 15% chance of being detected with this number of respondents, and would require 800 respondents in each group to give it an 80% chance of being detected using Chi squared (cf. Rosenthal & Rosnow, 1991).

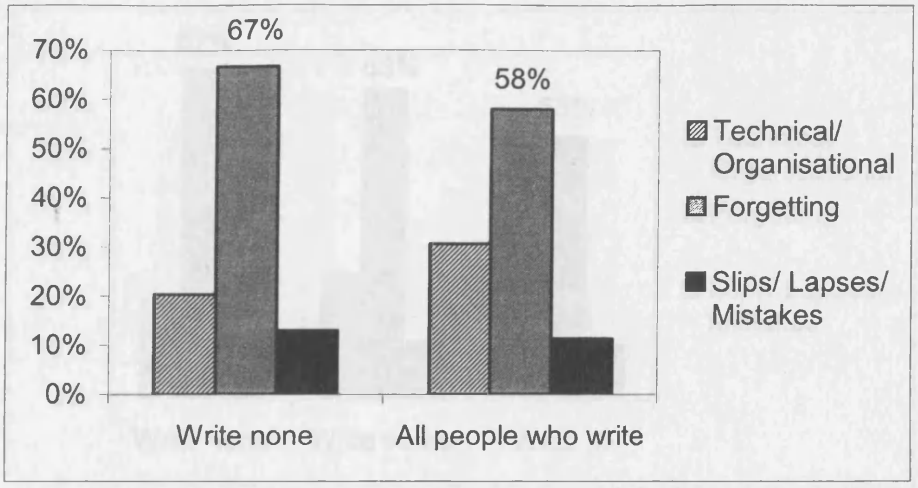


Figure 22 - Problem profiles for respondents who wrote down passwords, and those who did not (Studies 2 and 3 combined) (n=157, or 67% of responses)

These profiles were refined further, splitting the *writing* group into those who wrote down only some of their passwords, and those who wrote down all of their passwords (Figure 23). From this chart you can see that the proportion of slips, lapses, and mistakes remains almost constant. However, with increasing numbers of passwords written down, the numbers of passwords forgotten appears to drop (from 67% to 53%) and the number of technical/organisational problems appears to increase by about the same amount. These changes are not significantly different (Chi^2= 2.1, df=2, p=0.35 and Chi^2= 3.3, df=2, p=0.19 respectively). However, hypothetical effects of these sizes (w=0.11 and 0.14 respectively) would have only a 15% chance of being detected with this number of respondents, and would require 800 respondents in each group to give them an 80% chance of being detected with Chi squared (cf. Rosenthal & Rosnow, 1991).

The charts appear to show that users who write down all their passwords tend to forget fewer of them than users who do not write them down. Statistical falsification of this will require 10 to 20 times the current sample size.

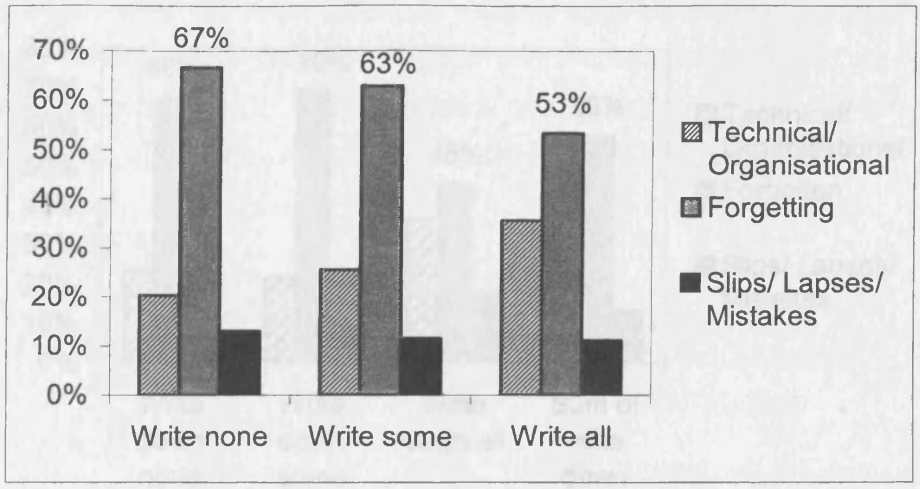


Figure 23 - Problem profiles for respondents in 3 groups by propensity to write down passwords (Studies 2 and 3 combined), n=157 - 67% of responses

Figure 24 (below) shows the proportion of respondents in each writing category. The proportions do not add up to 100 percent, because some respondents did not answer the question. These responses show that a large proportion of respondents write down all their passwords.

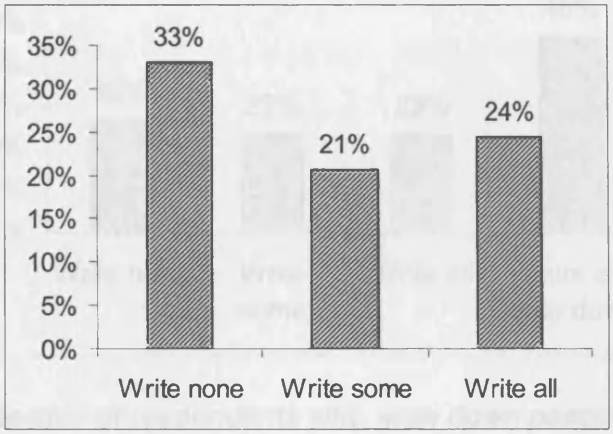


Figure 24 - Proportion of respondents who write down passwords (Studies 2 and 3 combined), n=233

The analysis is now refined further, looking in turn at the responses from users about passwords-in-general in Study 2 (Figure 25), and those respondents using 6 digit PINs in Study 3 (Figure 27). Looking first at passwords in general, there was a trend for users who reported that they write down their passwords to have fewer incidences (a third less, 45% vs 68%) of forgetting problems (Chi^2= 3.1, df=1, p<0.07), though the proportion of respondents forgetting passwords was still high.

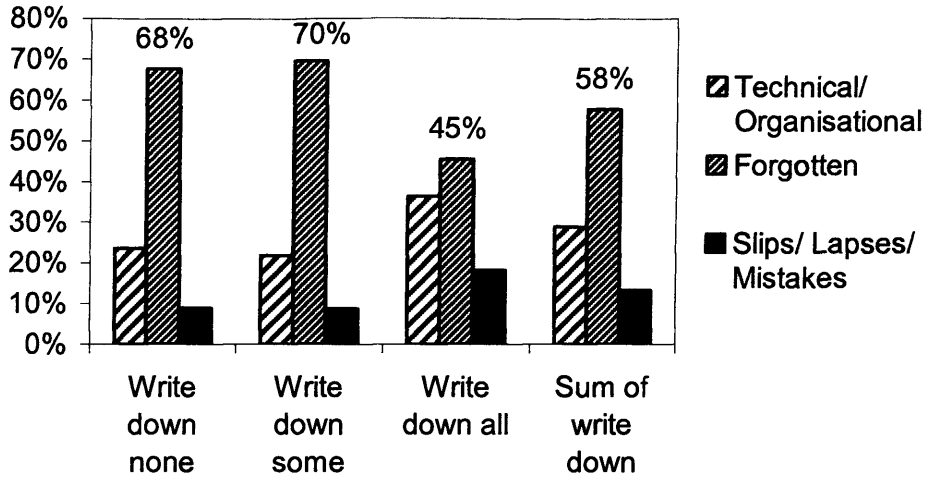


Figure 25 - Problem profiles for respondents in 3 groups according to how they write down passwords (for passwords in general), n=79 (57% of Study 2 responses)

Figure 26 shows that the numbers of respondents in each writing category was about the same, with about half of these respondents writing down at least some passwords.

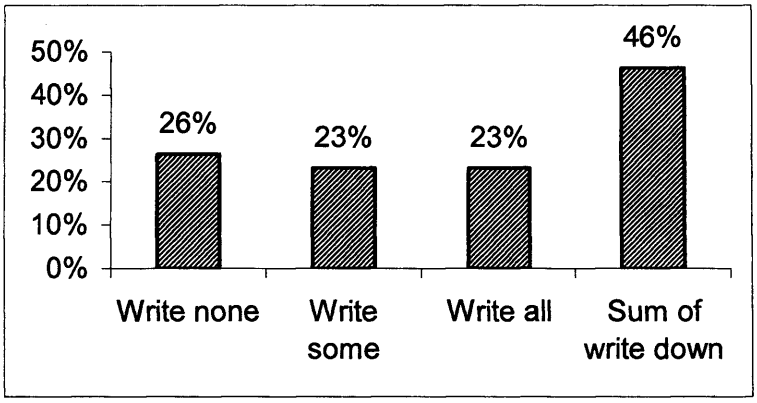


Figure 26 - Proportion of respondents who write down passwords-in-general (Study 2 data only), n=95

Figure 27 shows problem profiles versus writing down group for users who responded to the Study 3 questionnaire. A pattern similar to Figure 25 is shown, whereby writing down passwords is associated with a reduction in forgetting-74% of Study 3 users who claim not to write down any passwords had last experienced a forgetting problem, versus an average of 60% of Study 3 users who wrote down at least some of their passwords. This test suffered the sample size problem of previous figures - the test had a 15% chance of detecting a real effect with a hypothetical size indicated by the data (w=0.10) - 800 responses per group would be needed to have a test of the recommended power (cf. Rosenthal & Rosnow, 1991), approximately 20 times the

existing sample size. However, there was a strong trend for the relative proportion of Technical/ Organisational problems to increase from respondents who did not write down passwords to all respondents who wrote down passwords (Chi^2= 3.4, df=1, p<0.06). This supports a hypothesis of reduced proportions of forgetting.

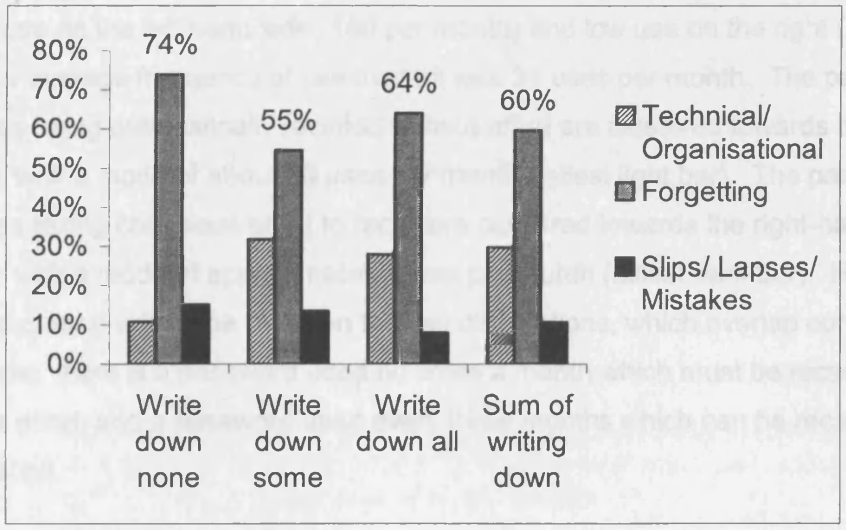


Figure 27 - Problem profiles for 6 digit PINs according writing down behaviour (Study 3 data only), n= 93 (66% of Study 3 responses)

Figure 28 shows the proportion of respondents in Study 3 who reported each kind of writing down, showing a similar pattern to Study 2. The proportion of respondents claiming to write down at least some of their passwords is about half (44%), with slightly less than this (35%) claiming not to write down any passwords at all. As before, about a quarter of respondents claimed they wrote down all of their passwords.

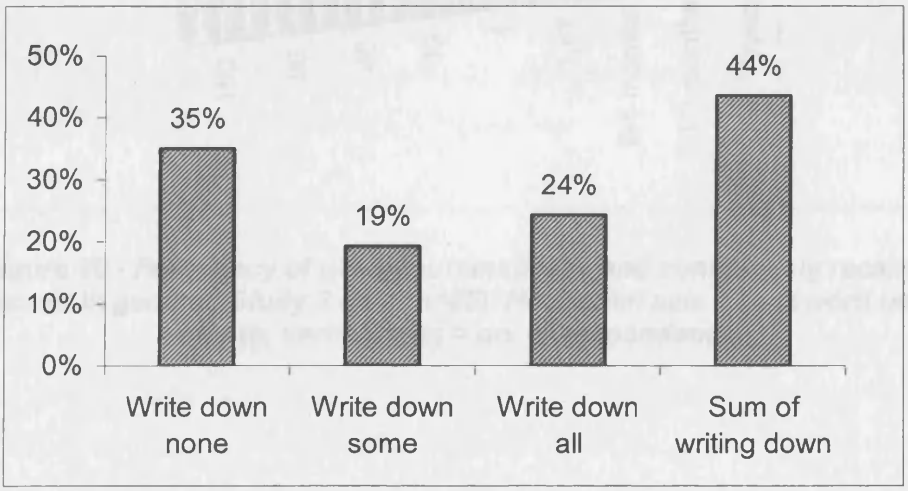


Figure 28 - Proportion of respondents who write down passwords-in-general (for Study 2 data only), n=140

7.3.8 Automaticity

Respondents were asked about the frequency of use, and the automaticity with which they are able to recall the password that they most recently asked to be reset. Figure 29 displays this data from Study 2. The x-axis of the figure shows frequency of use, with high use on the left-hand side (180 per month) and low use on the right (1 per year). The average frequency of use over all was 31 uses per month. The passwords reported as being automatically recalled without effort are clustered towards the left of the figure, with a mode of about 30 uses per month (tallest light bar). The passwords reported as taking conscious effort to recall are clustered towards the right-hand side of the figure, with a mode of approximately 1 use per month (tallest dark bar). However, there is no clear dividing line between the two distributions, which overlap considerably. For example, there is a password used 60 times a month which must be recalled with conscious effort, and a password used every three months which can be recalled with no effort at all.

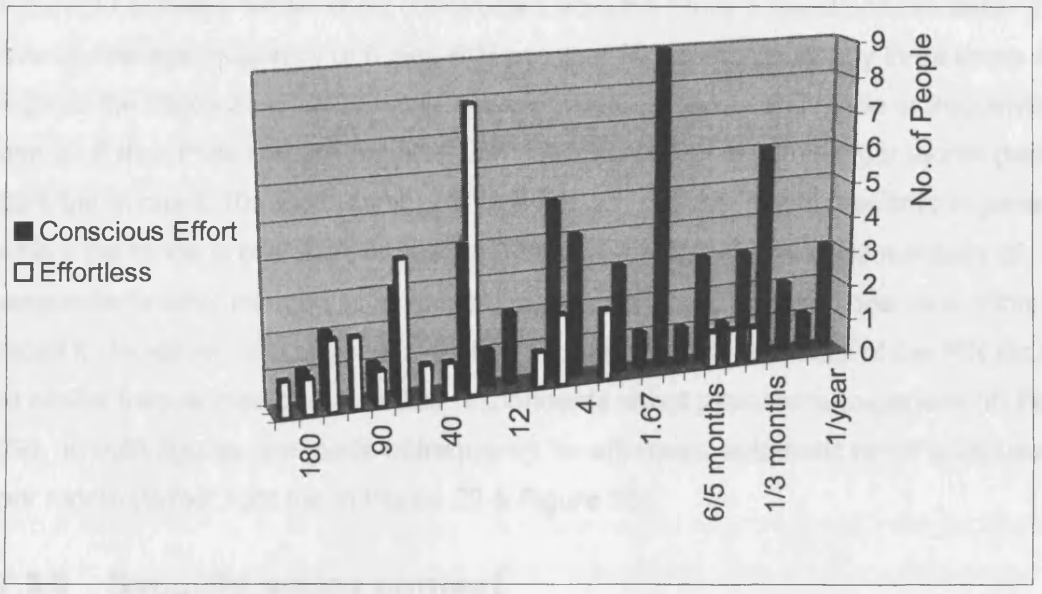


Figure 29 - Frequency of use of automatically and consciously recalled passwords in general, Study 2 data (n=85). Horizontal axis = password uses per month, vertical axis = no. of respondents.

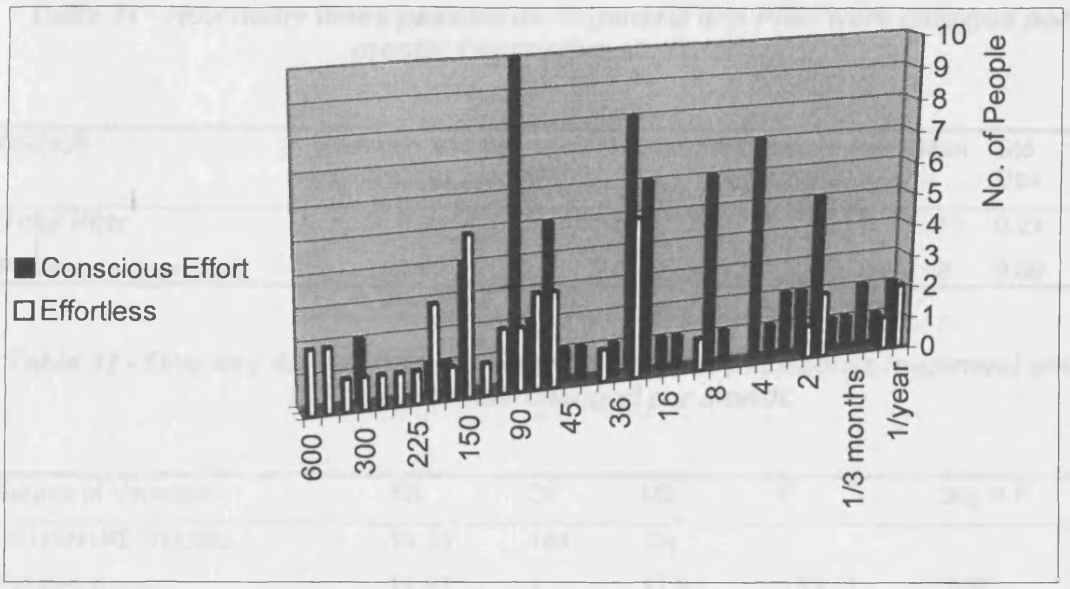


Figure 30 - Frequency of use of automatically and consciously recalled 6 digit PINs, Study 3 data (n=117). Horizontal axis = password uses per month, vertical axis = no. of respondents.

Figure 30 shows a similar chart constructed from the Study 3 questionnaire data. The overall average frequency of 6 digit PIN use was 80 per month, nearly three times as high as the Study 2 responses about passwords in general. The mode of frequency of use for 6 digit PINs that are recalled with conscious effort is 90 uses per month (tallest dark bar in chart, 10 respondents). This is in stark contrast with passwords in general, where the mode is one use per month. The 6 digit PIN also has large numbers of respondents who use it 60 or 30 times a month, who must expend conscious effort to recall it. However, respondents reporting effortless/ automatic recall of the PIN did so at similar frequencies of use as had respondents about passwords-in-general (in Figure 29). In both figures, the mode of frequency for effortless/ automatic recall is 30 uses per month (tallest light bar in Figure 29 & Figure 30).

7.3.9 Security policy context

Table 31 & Table 32 show that study 3 respondents reported that they had to change their 6 digit PINs (0.9 times/month) more frequently than study 2 respondents reported they had to change their passwords (0.4 times/month). This difference was significant ($F_{1,184}$=133.11, p<.0001,R^2=.42, observed power = 1).

Table 31 - How many times passwords-in-general and PINs were changed per month; Descriptive statistics

GROUP	Mean no. of times changed per month	N.	95% CI for Mean	Min	Max	Std Dev
6 digit PINs	0.93	109	(.89, .97)	0	1	0.21
Passwords in general	0.42	77	(.33, .51)	0	2	0.39

Table 32 - One way ANOVA table for how many times passwords-in-general and 6 digit PINs were changed per month.

Source of Variation	SS	DF	MS	F	Sig of F
WITHIN+RESIDUAL	16.33	184	.09		
Password type	11.81	1	11.81	133.11	.000
(Model)	11.81	1	11.81	133.11	.000
(Total)	28.14	185	.15		

The frequency of mandatory password changing increases with the strength of other security policies (see Table 2, section 3.2.7). The 6 digit PIN system in study 2 therefore had harsher security policies associated with it than the general password mechanisms examined in study 3. During study 2 it emerged that the 6 digit PIN system had inflexible security configuration - it allowed either harsh policies (6 digit PIN, etc., see 7.2.2) or nothing.

7.4 Discussion

7.4.1 Many passwords

Users in BT have a lot of passwords. Two questionnaire surveys were conducted by the author. In the survey which had pseudo random distribution of questionnaires (Study 2), the **average** number of passwords was 16. Published figures are rare and reported with little statistical detail, but this appears to be more than previous studies - 172 office workers surveyed at Liverpool street station had an average of 4 passwords (Wagner, 2004); 2,500 European IT administrators, executive management and security professionals reported an average of 4.3 passwords each. Other surveys present password ownership in bands - but these also suggest an average of lower than 16: 40% of users in the highest band were reported to have between 6 and 10 passwords (Protocom, 2003); whereas only 23% of SafeNet's 3,050 survey respondents had 8 passwords or more; 70% of 500 computer users surveyed at

Victoria Station had less than 10 passwords (Wade, 2002); and only 20% of SearchSecurity.com poll respondents had 15 passwords or more (Hurley, 2003).

7.4.2 Problems not related to memory

A second striking feature of the surveys is that a large proportion of problems that result in a password needing to be reset have nothing to do with the user's memory. They are predominantly caused by passwords expiring whilst the user is away (Table 23, Table 24), automated login systems triggering password policies such as three strikes, server problems, inheriting equipment without the password to access it, and a number of other less frequent problems. In some cases, the user could have taken action that would have prevented the problem from occurring-such as changing their password before they went away on business or on holiday so that it would not expire while they were away. However, this is more an issue of policy, organisation, or technology than of human memory. Such issues account for approximately 30 percent of password problems. Previous published discussions of password problems imply that they are the users' fault (Bunnell et al., 1997; Pond et al., 2000; Yan et al., 2000; Zviran & Haga, 1990, 1993), but this is not true. These figures agree substantially with an internal survey at BT which this author has since gained access to: 14% of password resets are caused by access control problems, 5% security management problems, 4% systems/integrity of data problems (MacKay, 1999) - 23% overall due to technical or organisational issues.

7.4.3 Password resets due to forgetting

While users are not the single point of failure for password mechanisms, they do account for the vast majority of instances where a password must be reset. Approximately 70% of password problems recorded in our surveys were reported to have resulted immediately from a failure of the users' memory. In this respect, conventional wisdom is accurate-users do forget a lot of passwords. However, users do not exist in a vacuum. They exist as part of a security architecture which allocates them certain functions and workloads, not all of which may be compatible; a particular stance has been taken by the organisation in risk management, balancing some rules against others. If an organisation finds the high rate of password forgetting to be unacceptable, then it might consider redesigning its security architecture and the trade-offs it made in risk management.

7.4.4 Password expiry has security costs

A good example of this is the policy of password expiry-where users are forced to change their passwords at regular intervals. This has a security benefit that attackers who have unauthorised access to a password are locked out of the system after a while, unless they have installed malware/backdoors. However, this benefit must be weighed against certain security demerits: weaker passwords selected, higher helpdesk use, and greater disclosure of passwords. These will be discussed below.

The appending of a digit to a password is a sign of users attempting to circumvent historical compartmentalisation of passwords, which are associated with strong password contents policies. For example, while the word "password" would not pass a strong password policy, the addition of a number to its end could transform it into something that does -it goes from being a dictionary word (weak) to a mixture of letters and numbers (stronger). 18% of our respondents who changed their passwords infrequently appended digit, versus 64 % who changed their passwords frequently. This is a statistically clear and negative relation between changing passwords and password content. Therefore, the enforcing of password expiry makes password content weaker.

There appears to be a particular risk for users associated with password changing-17% (1 in 6) of our respondents' problems occurred just after they had changed their password (see section 7.3.6), and it seems that the more frequent the changes, the greater the number of problems (Table 29). The percentages in Table 29 suggest that password expiry is causing the majority of these problems. If password problems were randomly distributed, then only 2% of respondents (with a three monthly change) would expect to have a problem on the same day they changed a password[1], instead of the actual figure which is 10%. For passwords changed every month, the expected figure would be 5%, and the observed figure is 14% (1 in 7). This supports users reports that 22 of their 233 resets (9%) were due to password expiry when they were away (section 7.3.2).

There is a trend for the types of problems experienced by users to be different when password expiry is in force: instead of merely forgetting their passwords, they confuse them more often (Table 30). This increases the risk of unintentional password disclosure.

[1] the random chance of having a problem on the day you change a password is: 1 chance in (5 working days per week x 4 weeks per month x 3 months), or 1 in 60, or 1.7%

7.4.5 Writing down is common and helpful, and so should be supported

Another striking feature of our results is how widespread is the writing down of passwords - approximately 45 percent of our respondents admitted to it. It seems odd that so many users would make the effort to do this unless there were real benefits in doing so. Our results seem to point to the expected benefit - a trend for the reduction in the numbers of passwords forgotten. For example, the numbers of respondents whose last serious password problem was forgetting: 68 % vs 45 % of our passwords-in-general respondents claiming not to write down any passwords vs writing down all of them (Figure 25). This is reducing the rate of forgetting by a third. Anything that is so common and so useful would seem appropriate to support, taken under the umbrella of the organisation and made safer. However, it should be noted that writing down passwords is not a panacea. It would still leave 45% of problems being due to forgetting. How could this be? It is the author's hypothesis that it is due to a combination of intermittent writing down and lack of the record's availability at the time it is needed. Writing down is effortful, and so it is unlikely that users will be scrupulous with it. Particularly when the immediate cost of not making the record is so low. Record-keeping is also less likely unless it is properly supported with the materials and/or apparatus necessary to complete it. If the writing down is ad hoc, then it may be difficult to gain access to the record when it is needed. Moreover, it may be needed when the user is away from their normal place of work-unless there is organisational support to help this user to get to their password record, then they will not be able to do so.

7.4.6 Frequency of use protects against forgetting

We have seen that writing down passwords is associated with reduced forgetting. The evidence shows what is expected, that frequent use of a password is also associated with reduced forgetting. Figure 19 showed that forgetting accounted for 86% of problems in lightly used passwords, but only 21% of problems in heavily used passwords. This result was highly statistically significant. These findings are consistent with a hypothesis that increased password frequency of use and writing down protect against forgetting, and that frequency of use is a far greater protection against forgetting than writing down.

7.4.7 Security policies destroy protection given by frequency of use

However, the results also suggest that high frequency of use can only go so far. The effect is moderated by the burden of security policies placed upon it. This is evident when data from Study 3 is examined. This 6 digit PIN system has the lion's share of security policies-extremely restricted password content (a minimum of six characters, and numbers only, i.e. close to meaningless), monthly password expiry, and a 12 password history. Even when this is heavily used- more than once a day-57% of its problems are forgetting (Figure 18). This is more than 2 1/2 times the rate of less restricted password systems (Figure 19), and highly statistically significant. In fact, Figure 30 shows how difficult it is to remember passwords of this type-respondents had to expend conscious effort to remember them even when they were used three times a day (90 uses per month). More average passwords had become so effortless respondents classed their recall as automatic at a third of this frequency of use (Figure 29).

7.5 Summary & Conclusions

The average number of passwords owned by respondents was 16, which is substantially more than other published figures. This could be because:

1. BT users have a larger password burden than users in other organisations, or
2. More passwords were being reported due to differences in methodology (such as question phrasing).
3. Both the above.

These hypotheses should be explored - the hypothesis 1 is theoretically relevant to Research Question C (what are the causes of good or bad password system performance?), but the size of its effect has not yet been studied; and hypothesis 2 is an important component of Research Question B (how can the performance of password systems be measured?) that has also not been investigated.

Further data was presented towards answering Research Question C. About 70% of password problems were related to human memory. A high level of memory related problems is expected from the literature of human error (see section 3.3.5).

However, this leaves approximately 30% of problems reported by users as resulting in a password reset as being due to technical or organisational problems and so not to do with failure of the user's memory. This is a surprising result in the literature of

password system performance. Password expiry whilst the user was away was the most frequent category (approx. 45%) of technical or organisational problems reported, and a large proportion (approx. 14%) of all problems reported. Evidence was also presented that password expiry policies lead to weak passwords, increase the numbers of problems requiring helpdesk support.

Progress was made toward answering Research Question D - *What interventions can be made to improve the performance of password systems*? Data suggested that using a password frequently appears to protect against forgetting it. This is consistent with the literature of human memory. The protective effect of frequent use of passwords is moderated by the severity of password policies. Severely restricted passwords can require little effort to recall if used three times a day, but are not as memorable if used once a day. However, even when used 3 times a day, passwords created under severely restrictive policies were still forgotten by some of the users surveyed.

The writing down of passwords is widespread (admitted to by approx. 45% of respondents) and appears to prevent memory problems leading to password resets. It was concluded that it should be better supported so it can be appropriately defended.

7.6 Contributions

7.6.1 Substantive

The impact of security policies (in particular password expiry) on password system performance has been empirically demonstrated. The data from this chapter has allowed the beginnings of a diagnosis of password problems. Data has therefore been presented that can be used as a basis for more rational and informed risk management in authentication.

Chapter 8

Studies 4, 5 & 6: System logs revisited

8.1 Introduction

This chapter returns to system logs as a source of data for password research, bringing together three studies that use them more successfully than was possible in Study 1. Studies 4 and 5 build upon the methodological conclusions of Study 1 (see section 6.6.2), implementing and employing an improved password use logging mechanism on a computer system at UCL. Study 6 opportunistically uses a large data set made available by BT's central password helpdesk.

The work done by password systems is letting through authorised users (and denying access to unauthorised users). When an authorised user needs access to a computer for part of her job but is denied it by a password system, a *login failure* has occurred, a breach of availability has happened and the password system is performing less than perfectly. Studies 4 and 5 examine logs of login success and failures - direct measures of password system performance. After repeated attempts the user may have logged in successfully, or may be experiencing problems that leave her unable to go further. Study 6 examines logs of helpdesk use that occur when users are unable to progress beyond login failures, and so are prevented from doing their job.

The goals, datasets, and analyses performed in each study are summarised in Table 33. Studies 5 and 6 contributed data from one specific password system at UCL and three at BT towards answering our Research Question A (*what is the performance of password systems in actual use?*), in a way which allows comparisons between different systems. Previously Study 1 had presented password system performance data that had focused on the user rather than systems, and so did not allow such comparisons.

Studies 4, 5 and 6 contribute data towards answering Research Question C (*what are the causes of good or bad performance?*). This data allows hypotheses about the causes of password system performance to be tested that could not have been in previous chapters. Two of these hypotheses are implicit in the literature of password memorability (the *long interval* and *character set size* hypotheses). One hypothesis was supplied by BT (the *repeat offender* hypothesis), and the remaining (*frequent changing*) hypothesis was a result of research in Chapter 7 (see section7.4.4).

Study 5 starts to investigate one answer to Research Question D (*what interventions can be made to improve the performance of password systems?*). Altering the number of strikes allowed is explored as an intervention in thought experiments based upon data collected during the study.

Table 33 - Summary of purpose and data of Studies 4,5 and 6

Study	Goal	Data	Participants	Analyses performed
4	Estimate password system performance in UCL with enhanced instruments. Continue to diagnose performance.	Password mechanism system logs	33 UCL students	1. Types & frequencies of password errors 2. Validity of long interval hypothesis 3. Validity of frequent changing hypothesis
5	Expanded data collection for Study 4 with better precision through greatly increased numbers of observations. Estimate the basic parameters of password system performance, continue to diagnose performance. Explore a possible intervention.	Password mechanism system logs	386 UCL students	4. Frequencies of errors 5. Password content 6. Validity of character set size hypothesis 7. Number of strikes observed
6	Observe password system performance for a BT sample using objective measures. Measure & diagnose performance.	Helpdesk system logs	47,145 BT users	8. Validity of repeat offender hypothesis 9. Identification of problem password systems

8.2 Methods

Data were collected from two sources: the Teaching And Coursework Online (TACO) system at UCL Computer Science (a web-based coursework system), and the logging system of a large central password helpdesk at BT called *Newpass*. Studies 4 & 5 were conducted using the TACO system (see section 5.4.4). Study 6 used the *Newpass* system (see section 5.4.6).

8.2.1 Study 4 (UCL coursework system logs I)

Participants

Thirty-four 1st year undergraduate students (Information Management majors taking a 1 term course in 1999 in Systems Analysis at UCL's Computer Science Department) participated in the trial. 33 students logged in frequently enough to be included in the study.

Context of Experiment

As part of their Systems Analysis course, the students had to complete 6 courseworks on-line over a period of 10 weeks; the coursework was authored and managed through the TACO system (Sasse, Harris, Ismail, & Monthienvichienchai, 1998). To interact with TACO, students used computer terminals at University or from home.

The system is designed to give scores and feedback, so frequent practice tends to result in better grades. Students could practice on the question sets as often as they liked before submitting an assessed version. Users of TACO are required to go through authentication before being allowed to interact with courseworks. Authentication usually consists of entering a username and system-generated password (both of which had been previously supplied to the user through a secure channel). In addition TACO was fitted with both a facility for participants to select their own passwords, and a facility to select and use Passfaces™. Passfaces™ is an alternative authentication mechanism, which uses recognition rather than recall.

Participants in Study 4 used both passwords and Passfaces™, swapping halfway through the ten-week period. Study 4's password results are presented in this chapter, and its Passfaces™ results are discussed elsewhere (e.g. Brostoff & Sasse, 2000).

Apparatus

The password mechanism of TACO is executed at the server side, and appears instantaneous to users. User IDs and an enrolment password are distributed through a secure channel, such as being handed out on slips of paper in the first lecture. The enrolment password consists of 6 randomly selected lowercase letters, appended with two random digits.

The password enrolment procedure gives guidance about the selection of strong password content, and requires users to submit a password of their choosing twice, and their enrolment password. TACO does not enforce any constraints on users' self-generated passwords. If the enrolment password is correct, and both submissions of the chosen password match, then enrolment is complete.

TACO was further changed to offer participants reminders of their passwords. On their request, participants were emailed a copy of their password, or sent the address of a web page where they could view their Passfaces™.

TACO log files were enhanced so that the failure or success of login attempts could be determined, all requests for reminders were recorded, and all log entries were timestamped. TACO stores users' passwords in plain text. The enhanced logging system took advantage of this to display the correct password next to what was typed

in by a user during a failed login attempt, so that the two could be easily compared and the nature of the failure inferred. The logs did not introduce greater vulnerabilities to the TACO system, as logs were stored in the same database that stores users' passwords.

TACO log files were cleaned and analysed in MS Excel 2000 and 2002, and also analysed with SPSS 10.

Procedure

Participants used the web-based coursework system as normal to complete course-works. A repeated-measures design was used, with each student using both passwords and Passfaces™. Halfway through the term the authentication mechanisms used by students were changed over (those using passwords were now using Passfaces™ and vice versa). The participants were pseudo-randomly assigned to Group 1 (starting with passwords, then using Passfaces™) or Group 2 (starting with Passfaces™, then using passwords). This maximised power for the test of difference between passwords and Passfaces™.

Participants were required to re-enrol after the changeover, and *new* enrolment passwords were distributed by email to all participants and on paper slip by request. The academic term ended for Christmas, and participants were asked to log in during their first week back - approximately 1 month after last using the system.

8.2.2 Study 5 (UCL coursework system logs II)

Participants

386 undergraduates students participated over one academic term each (10 weeks), though the data were collected from students over the course of several years. The students were enrolled on one of the courses B160, 2B14, C363 given at UCL CS, and who used the system between the dates of 20th Oct 1999 and 18th Nov 2002. Study 5 includes the data from Study 4. No participants could be in more than one class at a time, although they may have moved from one course to another course in subsequent years. Individuals are given separate accounts for each course.

Context of Experiment

Though focusing exclusively on passwords, Study 5 expanded on Study 4 by capturing data for several years from several other courses. Students in Study 5 used passwords exclusively, whilst students in Study 4 also used Passfaces™ and were

forced to change authentication mechanisms halfway through the term. In all other respects, participants use of the TACO system was the same.

Apparatus

The same apparatus was used as in Study 4.

Procedure

Participants were given enrolment passwords (on paper slips or via email) at the start of term, to authenticate them for their subsequent selection of passwords. Participants used TACO as normal to complete course-works over the course of the term, and their use of TACO passwords was logged.

8.2.3 Study 6 (BT helpdesk logs)

Apparatus

Data were collected using BT's *Password Control* helpdesk's normal in-house logging system (*Newpass*) as part of everyday business. The data was analysed in MS Access 2000 and SPSS 10.

Participants

Password Control covers 650 of the organisation's several thousand computer systems. At the time of data collection, 120,000 of the organisation's 150,000 employees (80 percent) were registered for services with *Password Control*. *Password Control*'s callers are therefore representative of users in the organisation. It should be noted however that most of BTs password mechanisms are not served by *Password Control*.

Users contributed data if they called *Password Control* for help during six months (May, June, July, August, October, and November) in the year 1999. During this time, 47,145 people called the helpdesk, requiring 138,687 password resets.

Procedure

Callers to *Password Control* have information about them logged such as their name, workgroup code, time and date of the call, and the systems that required a password reset, among other things.

8.3 Results

8.3.1 Study 4 Results

Error types

By comparing the failed attempt with the participant's correct password, problem types could be inferred. Table 34 shows the relative frequencies of password problems encountered during Study 4. The most frequent problem was entering an expired TACO password in place of the current one. The next most frequent problem was substituting a password-like sequence for the correct password - the sequences were classed as passwords because they did not have the appearance of random key-presses - though these classifications were not verified with participants.

Table 34 - Password problems encountered by participants in Study 4

Problem type	Proportion	Description of problem
Use Expired Pwd	37%	Using the old TACO password instead of the currently valid one.
Substitution	15%	Using some password like sequence of characters instead of the currently valid password.
Enter	9%	Pressing Enter instead of typing a password - no password was entered in the login attempt
Omission	6%	Omitting a necessary character from an otherwise correct password
Addition	5%	Having an unnecessary extra character in an otherwise correct password
Partial Recall	5%	Recalling part of a password, but not the rest of it
Distribution	4%	A problem occurred during the initial distribution of the account details
System Error	3%	A malfunction in the password mechanism
Replacement	2%	Having incorrect characters in an otherwise correct password
Blend	1%	Mixing parts of passwords together
Capitalisation	1%	Using the wrong capitalisation in all or part of a password
Userid	1%	Entering the username instead of the password

Long interval hypothesis

The *long interval hypothesis* is that people forget passwords because they have not used them for long periods. It is an implicit explanation for the usability problems of passwords in password memorability research, and is consistent with the literature of human memory (see section 3.3.2). It has been demonstrated in laboratories using highly artificial experimental tasks (Zviran and Haga, 1990; Zviran and Haga, 1993; Bunnell et al., 1997). This is the first time that it has been observed using system logs in a population of real users during their normal work.

Some of the data from Study 4 was used to assess our preliminary finding from Section 7.4.3 that infrequently used passwords are forgotten. The descriptive statistics are shown in Table 35.

Table 35 - Descriptive statistics for login success and interval since last successful login

Group	N	Mean interval since last use, in minutes		95% C.I. for mean	Std. Dev.
Successful logins	1349	1341	(22 hours)	1152 to 1530	3538
Failed logins	103	2631	(44 hours)	1745 to 3517	4534

The interval between a failed login and the last successful login was approximately twice as long as the average interval between successful logins - long intervals between uses of a password lead to problems. This effect was highly significant ($F_{1,1448}$=14.74, p<.001, observed power = .969) though relatively small (eta^2 =.01), and supported the hypothesis and previous observations.

Frequent changing hypothesis

Many users in large organisations are forced to change their passwords regularly. There appeared to be a trend in our questionnaire data (Sections 7.3.6 and 7.4.4) for users to experience problems with their passwords, just after they had changed them.

An attempt was made to validate the effect of password changing using system log data from Study 4. The elapsed time since the password had last been changed was calculated for each successful and failed login. Descriptive statistics are shown in Table 36.

Problem logins occurred on average at less than half the elapsed time since the password had last been changed compared to successful logins. This result was

highly significant ($F_{1,1174}$= 43.75, p < .001, eta^2 = .036, observed power = 1.0). Our hypothesis and observation that changing or resetting of passwords can cause password problems was supported. This is evidence that password expiry causes people to have problems with their passwords.

Table 36 - Descriptive statistics - login success and elapsed time since password was changed

Group	N	Mean time since pwd was changed, in minutes		95% C.I. for mean	Std.Dev.
Successful logins	1018	15123	(10 days)	14321 to 15925	13037
Failed logins	160	6022	(4 days)	4455 to 7590	10039

8.3.2 *Study 5 Results*

Error frequencies

Study 5's experimental apparatus captured successful and failed login attempts, and their descriptive statistics are shown in Table 37. Out of 386 participants, 87 (23%) required password reminders over the period measured, which was effectively 10 weeks. The average failure rate for passwords in Study 5 was one login failing per 10 attempts (10%). Approximately 7% of these failed login's were converted into requests for password reminders. Therefore approximately 0.7% of Study 5 login attempts resulted in a password reminder request (an approximation to a helpdesk call).

Table 37 - Descriptive statistics of login success and failure in Study 5

	No. Successful logins	No. of failed logins	Login failure rate	No. of Reminders requested
Total	12044	1261	N/A	87
Average	31	3.3	0.1	0.2
Min	0.0	0.0	0.0	0.0
Max	339	71	1.0	9.0
St.dev	32	7.6	0.2	0.9
No. of participants	386	386	-	386

Password content

Although the apparatus in Study 5 requires the use of a password, it does not enforce any other password policies. Users can therefore elect to use their enrolment password as their active password on the system. However, 229 of the participants in Study 5 chose a password that was different than their enrolment password - 59% of the 386 participants.

The passwords selected by participants were analysed using Petrie's (2002) taxonomy (see section 4.1.2). The proportions in the different categories are shown in Table 38. 90% of passwords were strong (*Cryptic* in Petrie's taxonomy). Examining only participants who chose not to use their enrolment password (Table 39), 84% of their password choices were also strong. The proportions in each category were significantly different from Petrie's (2002) (Chi^2= 656, df=3, p<0.001).

Table 38 - Content of TACO passwords (including self generated and system generated passwords that users had chosen to keep)

Strength	Weak			Strong	
Type	*Family*	*Fan*	*Fantasist*	*Cryptic*	Total
Number	16	14	6	350	386
%	4%	4%	2%	90%	100%

Table 39 - Content of self generated TACO passwords

Strength	Weak			Strong	
Type	*Family*	*Fan*	*Fantasist*	*Cryptic*	Total
Number	16	14	6	193	229
%	7%	6%	3%	84%	100%

Table 40 - Descriptive statistics about self generated TACO password length

Measure	Value
Average Length	7.8 characters
Min. Length	4 characters
Max. Length	13 characters
St.dev. of Lengths	1.5 characters
N	229

Table 40 shows some descriptive statistics about the length of self-selected TACO passwords. TACO passwords were on average quite long, at nearly eight characters. Table 41 shows the distribution of self generated password lengths

Table 41 - Distribution of password lengths, for self generated TACO passwords only

Password length (characters)	4	5	6	7	8	9	10	11	12	13	Total
Number of participants	2	5	39	33	106	18	12	7	6	1	229
%	0.7	1.8	13.9	11.8	37.9	6.4	4.3	2.5	2.1	0.4	100

Character set size hypothesis

One of the fundamental hypotheses about password usability is that more complex passwords are less usable. Password usability is commonly operationalised as memorability (see section 4.1.2). Here usability is being operationalised as login error rates - an index of breakdowns in memorability (though are also other reasons for login errors - see section 8.3.1). The passwords chosen in Study 5 were analysed to see how many character sets they contained of:

- Uppercase letters
- Lower case letters
- Digits
- Symbols

Descriptive statistics for the login failure rates were calculated for each group (Table 42). The mean error rates for passwords with between 1 and 3 character sets were approximately the same, with the error rates for passwords with four character sets being nearly double the error rates for passwords with one or two character sets.

Table 42 -Descriptive statistics about the number of character sets in Study 5 passwords, and their respective error rates

No. of charsets	Average of error rates	Std. Deviation	Number	%
1	0.09	.16	67	17%
2	0.09	.16	277	72%
3	0.11	.21	26	7%
4	0.17	.21	15	4%

This data was entered into a one-way independent groups ANOVA, with four conditions-one for each number of character sets (Table 43). No significant difference

was found ($F_{3,383}$=1.2, p=.3 not sig., eta^2=.009, power = .326). Power analysis showed that this test would only have stood a one in three chance of detecting an real effect of this hypothesised size.

Given the low power of the omnibus test, it was decided to use a more focused test, adding together the data from passwords with between 1 and 3 character sets, to compare these against the error rates from passwords with four character sets. The data was entered into a one-way ANOVA, the results of which are shown in Table 43. No significant difference was found, although there appeared to be a trend ($F_{1,385}$=3.27, p=.07 not sig, eta^2=.008, power= .438).

Table 43 - ANOVA table testing for the effect of number of character sets on login failure rates in Study 5

Source	Sum of Squares	df	Mean Square	F	Sig.	Eta^2	Observed Power
Character sets	0.1	3	0.03	1.2	.303	.009	.326
Error	10.3	383	0.03				
Total	14.1	387					

Table 44 - ANOVA table for number of character sets per Study 5 password and associated error rates, comparing passwords with between 1 and 3 character sets to passwords with 4 character sets.

Source		Sum of Squares	df	Mean Square	F	Sig.	Eta^2	Observed Power
Character Sets (1,2,3 vs. 4)	Hypothesis	0.08	1	0.08	3.27	.07	.008	.438
	Error	10.31	385	0.03				

How many strikes?

One of the more favoured methods of configuring password mechanisms is the three strikes policy (see section 3.2.8), which usually locks a users' account after three failed attempts at logging in. This section investigates the potential impact of setting the number of strikes to 3, rather than another number. Figure 31 shows the distribution numbers of login failures-with the light bars representing the participants who required password reminders, and the dark bars representing the participants who did not. Bars are total numbers of login failures for each participant over the whole study - not failures per session. Figure 31 shows that participants who required password reminders suffered proportionately greater numbers of failed login attempts.

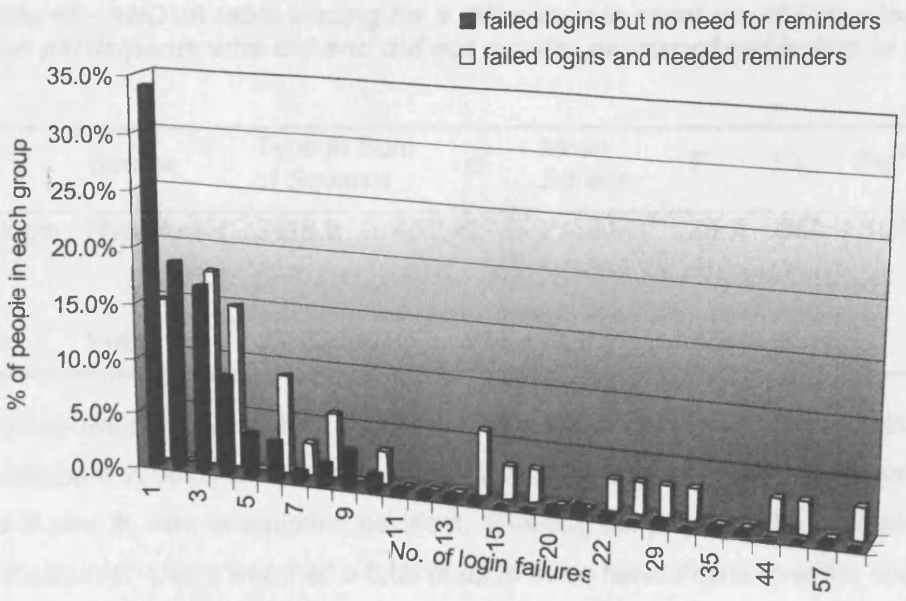


Figure 31 - Distribution of numbers of login failures, for users who did not require password reminders, and users who did. Study 5.

Table 45 shows descriptive statistics about the relative numbers of failed login attempts for the two groups. Participants who required password reminders experienced on average three times as many failed login attempts as participants who did not. Table 46 shows that this difference is highly significant ($F_{1,234}$=28.7, p<.001, eta^2 = .109 , power = 1).

Table 45 - Descriptive statistics of failed login attempts, for people who did and did not require password reminders in Study 5

Group	Mean failed login attempts	Median	Min.	Max.	Std. Dev.	No. of people
Failed logins but no need for reminders	4.1	2	1	57	6.5	202
Failed logins and needed reminders	12.7	6	1	71	16.3	34
Total	5.4	3	1	71	9.1	236

Participants who required password reminders experienced an average of 6.9 failed login attempts per reminder request, with a standard deviation of 7.5, and an average number of 2 reminders.

Table 46 - ANOVA table testing for a difference in numbers of failed logins between participants who did and did not require password reminders in Study5.

	Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Eta^2	Observed Power
Failed login group	Hypothesis	2118.9	1	2118.9	28.7	.000	.109	1.000
	Error	17292.3	234	73.9				

Figure 32 shows cumulative data about the total number of failed login attempts of each participant in Study 5. The orange line shows participant data. The smooth line is a line of best fit, with its equation beside it. Reading the graph will be illustrated with three data points. Users who had a total of up to three failed logins over the course of the study made up about 33% of participants. About 65% of participants had 10 failed logins or less, and 90% of participants had 30 login failures or less during the study. If you assume that users had all their failed logins in one session then you can start to reason about how many strikes would impede the login attempts of how many users. The figure is attempting to answer the question, "How many strikes do users want before giving up and asking for a reminder?". Under this assumption, the graph shows that approximately 90 % of Study 5's participants would give up trying to login before triggering a 30 strikes policy. Approximately 80 percent would give up before hitting a 20 strikes policy, about 65 percent before a ten strikes policy, and about a third would give up before hitting a three strikes policy.

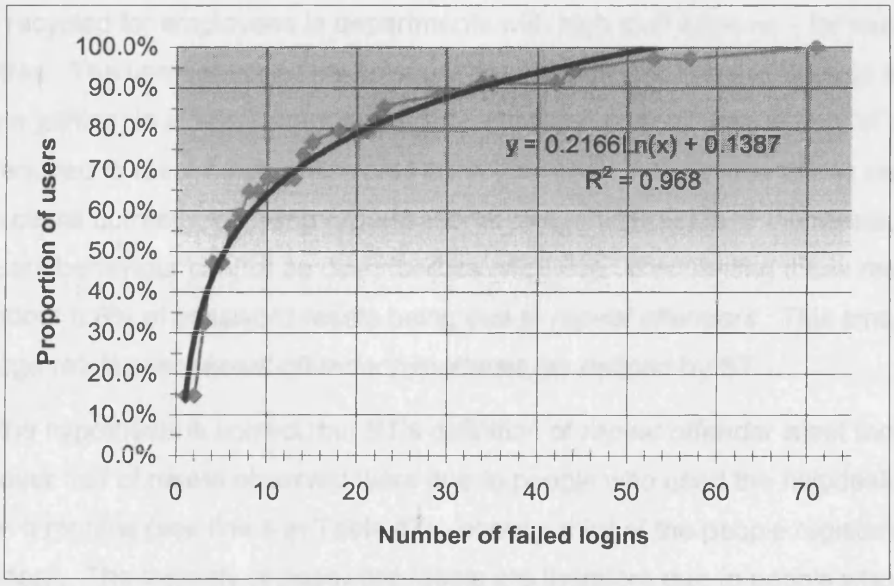


Figure 32 - Graph of the number of strikes allowed, and the proportion of participants accommodated, data from Study 5

8.3.3 *Study 6 Results*

Repeat offender hypothesis

BT's *repeat offender hypothesis* is that the password related load on helpdesk's is due to a small minority of people who do not take due care with their passwords, and who consequently must contact the helpdesk repeatedly. This hypothesis has large implications for both measuring password system performance and diagnosing it: it suggests that password system performance is acceptable, and that any problems observed are due to misuse of password mechanisms by a small minority. The hypothesis is already followed up by BT, who identify and refer *repeat offenders* to their line managers, who may initiate disciplinary procedures. *Password Control's* definition of *repeat offender* is someone who has 6 passwords or more reset a month

The data presented in this section comes from Study 6 - from a central password helpdesk at BT, which has approximately 80 percent of BT's workforce on its books. The data shows that 90,000 different people a year use this helpdesk. This is approximately 60% of BT, and 75% of those registered with the helpdesk. This suggests that the majority of people who use information technology experience difficulty with passwords, and is evidence that the hypothesis is wrong.

Repeat offenders (using *Password Control*'s definition) account for only 8.3% of password resets (Table 47). Approximately 30% of passwords reset by *repeat offenders* are associated with *Self Administered User IDs* (Prince, 1999) - i.e. user-ids that are recycled for employees in departments with high staff turnover - for example call centres. The user-ids for all the tempory or new staff (i.e. that are likely to leave soon after joining) in a department are registered with a permanent member of staff who is required to reset these passwords each time somebody leaves and is replaced. In these cases correctly following organisational procedures leads to the resets, so these users behaviour cannot be described as *offending*. Discounting these resets leaves about 5.8% of password resets being due to *repeat offenders*. This small percentage refutes the *repeat offender hypothesis,* as defined by BT.

What if the hypothesis is correct, but BT's definition of *repeat offender* is set too high? Slightly over half of resets observed were due to people who used the helpdesk 4 times or less in 6 months (see line 4 in Table 47) - about a third of the people registered with the helpdesk. The majority of password resets are therefore due to people who have 2/3 or less of a password reset a month on average. This evidence refutes the *repeat offender hypothesis* in its wider definition.

Table 47 - Summary data for 6 months of Newpass logs. * approximate values

Resets per month	No of resets in 6 months	No of people	% of users registered at helpdesk	Cumulative No. of People	Cumulative % of users registered at helpdesk with this number of resets or less	No. Of Resets	Cumulative number of resets	% of total resets caused by people with this number of
	1	19695	16.41%	19695	16.41%	19695	19695	85.8%
	2	10585	8.82%	30280	25.23%	21170	40865	70.5%
	3	6970	5.81%	37250	31.04%	20910	61775	55.5%
	4	3474	2.90%	40724	33.94%	13896	75671	45.4%
	5	2047	1.71%	42771	35.64%	10235	85906	38.1%
1	6	1490	1.24%	44261	36.88%	8940	94846	31.6%
	7	806	0.67%	45067	37.56%	5642	100488	27.5%
	8	538	0.45%	45605	38.00%	4304	104792	24.4%
	9	418	0.35%	46023	38.35%	3762	108554	21.7%
	10	267	0.22%	46290	38.58%	2670	111224	19.8%
	11	185	0.15%	46475	38.73%	2035	113259	18.3%
2	12	149	0.12%	46624	38.85%	1788	115047	17.1%
~	~	~	~	~	~	~	~	~
3	18	47	0.04%	47018	39.18%	846	120939	12.8%
4	24	19	0.02%	47146	39.29%	456	123641	10.8%
5	30	10	0.01%	47214	39.35%	300	125490	9.5%
6	36	4	0.00%	47267	39.39%	144	127244	8.3%
7	42	7	0.01%	47303	39.42%	294	128667	7.2%
8	48	5	0.00%	47325	39.44%	240	129664	6.5%
9	54	2	0.00%	47339	39.45%	108	130390	6.0%
10	60	7	0.01%	47360	39.47%	420	131598	5.1%
15*	91	1	0.00%	47384	39.49%	91	133410	3.8%
20*	121	1	0.00%	47396	39.50%	121	134689	2.9%
35*	212	1	0.00%	47406	39.51%	212	136133	1.8%
50*	296	1	0.00%	47413	39.51%	296	137905	0.6%
75*	448	1	0.00%	47415	39.51%	448	138687	0.0%

Identification of problem password systems

Figure 33 shows average password reset rates per user for nine major information technology systems in BT, based on telephone survey data from April 1999. It shows that of the first three systems, there is a steady increase from System 1 to System 3 which has the most resets per user.

This chart also shows that System 7 has a relatively high rate of password resets. System 7 has a particularly elaborate password authentication mechanism employing a

Secure-ID token, that requires a random number that changes every few seconds to be copied from the token and appended to a PIN.

Table 48 is based on reset log data from a six-month period in 1999, with the same number of users per system information as in Figure 33. Table 48 shows the opposite pattern - there is now a decrease from System 1-which has the most resets per user- to System 3, which is now shown as having the least.

It is estimated that under similar operating conditions (continuous use throughout the year instead of only ten weeks, and with uniform distribution of resets through time), TACO would have 0.7 resets per user per year.

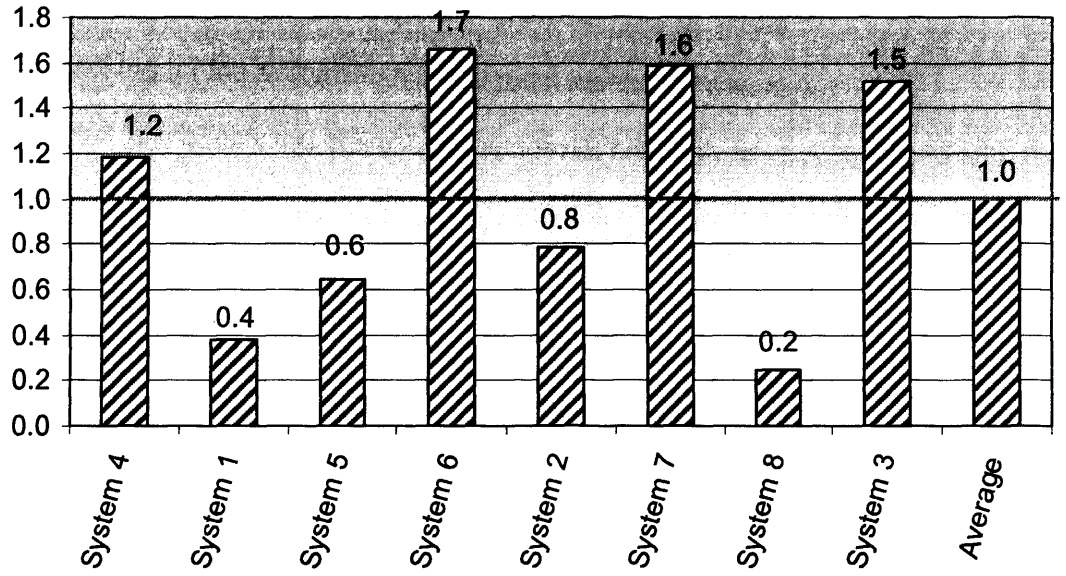


Figure 33 - Ratio of relative password reset rates per user of different IT systems compared with to System 9- data from Password Control phone survey, April 1999.

Table 48 - Number of password resets for each IT system administered by Password Control

System Name	No. of Resets per ½ year	% of total resets	No. of users[2]	Rate of resets per user per year
System 1	31,825	22.9	17,490	3.6
System 2	11,703	8.4	20,000	1.2
System 3	6,642	4.8	15,000	0.9

[2] Data from April 1999 (Mackay, 1999a)

8.4 Discussion

8.4.1 Identifying a problem password system can be difficult

It emerged from the data that it would be possible to identify problem password systems by observing performance of a number of systems so that diagnostic efforts could be sure to encompass those systems that perform badly. However, identifying a problem password system in a large corporation is not as straightforward as it first appeared. There are at least 2 ways it might be tackled:

1. looking for the system with the highest number of resets
2. looking for the system with the highest number of resets per user

Approach 1 is probably the easiest, as helpdesks are likely to keep logs of problems dealt with, which include which systems problems originated with. In this case, you can see in Table 48 that System 1 generates the highest volume of password resets out of the three. In one sense, this is a problem system because it produces the most work for the helpdesk. However, how does it compare to other systems? It may be for example that it simply has more users than other systems.

To assess this, it is necessary to get both numbers of resets per system, and number of users per system, which allows you to generate a chart such as Figure 33 (shows the April 1999 data for some of BT's major computer systems), and which shows that system 1 generates a very low ***rate*** of helpdesk requests compared to other systems and that System 7 has a relatively high rate.

After identifying System 7 as having relatively severe problems, it might be examined to find out their cause. System 7's password authentication mechanism is configured to be *unusually complex* (see section 8.3.3) with more steps performed by the user during login than most password systems, giving more room for human error. This is further evidence to support our finding in Chapter 7 (section 7.4.7) that restrictive password policies reduce the effectiveness of password mechanisms (by increasing login problems and helpdesk use).

The principle used in making the Figure is sound, and could be used to target problem systems to be given priority for intervention. However, in practice it can be difficult to get the data. For example, while reset rates are stored at the helpdesk, numbers of users data may be kept in other parts of the organisation, which may have an incompatible political agenda, or may simply be difficult to track down and ask. The

inquiry may be left with outdated or otherwise inadequate information, as occurred in the pursuit of this PhD.

8.4.2 Three strikes is too little, ten strikes is better

A common policy in high security environment is that users should be allowed no more than three strikes at login. Enlightened security specialists have recommended up to 50 strikes (e.g. Viega & McGraw, 2001), though such a high number may not be acceptable to traditionally conservative security personnel. Although the number of strikes may vary from system to system, it does not appear that this number has ever been selected from analysis of empirical data. A first attempt has been made here. As will be discussed in the next section, the sample of UCL users on which this attempt is based faces different circumstances than our BT population of users - so direct extrapolations to them are inappropriate. Moreover, some precision has been lost by examining totals and averages of failed login attempts and resets, rather than counting for each reset how many attempts preceded it. However, our attempt remains an improvement over the existing state-of-the-art, which is based solely on speculation.

There are two issues:

1. Reducing use of a helpdesk by giving users enough chances to successfully enter their password
2. Accommodating users by allowing them to give up and not forcing them to call a helpdesk before they are ready to

The first issue is to do with balancing the risks of increased vulnerability to attackers and loss of availability to authorised users. The second issue is about the respect with which users hold safeguards.

For the first issue we must look at the distribution of failed logins among users who did not later require password reminders (dark bars in Figure 31) - these are the users who could be unnecessary forced to use a helpdesk. The existing norm of three strikes is predicted to accommodate only a third of these users. Ten strikes would be enough to accommodate people who could recover from login failure. There are 202 of them in total. 107 of them suffered two failed logins or less, and so would have passed a three strikes rule (28% of all participants). However, a further 80 of them suffered between 3 and 9 login failures. These users would have triggered a three strikes policy and probably be required to contact the helpdesk had the policy been in force. Instead of just 87 helpdesk requests (section 8.3.2) there might have been 80 extra, an increase of nearly 92%. With ten strikes instead of three, these extra helpdesk requests could be avoided. Moving from three strikes to ten strikes should reduce password related

helpdesk calls. However, the amount of reduction should be predicted with a more accurate method, and may still be inaccurate if users adapt their behaviour to the new policy.

To address issue 2, we must look at the distribution of users who experienced login failures that did result in helpdesk requests (light bars in Figure 31). 11 of these (32%) would have passed a three strikes policy. Extending the number of strikes to ten would double the number of people accommodated (68%), leaving only a third prematurely forced to call a helpdesk. An extra 12 stakeholders accommodated out of 236 may not seem many, but security depends on the weakest link - alienating 5% of the user group cannot be sensible.

By increasing the number of strikes, the chance is increased that an attacker may successfully guess the password and break into your system. Moving from three strikes to ten approximately triples this risk. However, if an organisation enforces strong password content policies, then the actual risk will still be very small. Moreover, Viega and McGraw (2001) suggest another strike counter operating in conjunction with the first-recording the total number of strikes rather than the number of strikes in a session. After a suitably small total number of strikes is reached, such as 50 (see Figure 32), then additional security procedures are started. This would help to reduce any negative impact of moving from three strikes to ten.

8.4.3 Unusually strong passwords

A striking feature of our results is how security-conscious the participants in Study 4 and 5 (all from UCL) were in their choice of passwords. 84% of the sample chose strong passwords, in stark contrast to Petrie's (2002) recent survey, where only 9% were as virtuous. Moreover, the passwords they chose were of recommended length (8 characters). This is highly unusual behaviour for users given completely free rein in which passwords they chose. It raises 3 important questions:

1. How can other users be encouraged to choose strong passwords ?
2. Would it be a good thing to do this, or would it be better to try other interventions?
3. To what extent can results from an academic environment be generalised to corporate populations?

The first question is difficult to answer, because it is not known why the participants behaved like this. The solution to this problem is a Holy Grail of computer administrators.

It is the author's opinion that on balance it would be better for an organisation to improve password mechanisms (for example by using BCRYPT, see section 4.2.1) than to improve stakeholders' behaviour. By improving the encryption of passwords, an organisation would dramatically improve the effective quality of all its passwords- raising the bar on dictionary attacks without increasing costs for users or secondary stakeholders, and more than likely decrease them.

However, as has been seen in section 3.1.4, many techniques for breaking security concentrate on the human rather than the password mechanism itself. Attempting to replicate the good behaviour of Study 4 and 5's participants in corporate users would likely harden them against these other attacks to a far greater extent. Moreover, this approach may not greatly increase stakeholder costs. Though there is much evidence from laboratory studies to show that complex passwords are more difficult to remember than simple ones, the few available real-world studies (such as Yan et al., 2000 in section 4.1.2, and this chapter' s studies 4 and 5) show relatively little difference. For example, no statistically significant difference was found in error rates between passwords of differing numbers of character-sets. However, all of these real-world studies were carried out on computer science undergraduates at prestigious universities, rather than users in corporations- this brings up our third question - how well can the results be generalised? This has not yet been studied, and remains an important topic.

8.4.4 Password system performance and password policies

We are beginning to get an idea of password system performance in different environments, and with different configurations and policies enforced. It was possible to calculate rate of password helpdesk use per user per year for three BT information systems, and one at UCL. The UCL system had the lowest stakeholder costs-with 0.7 password related helpdesk calls per person per year, compared to 0.9,1.2, and 3.6. The UCL system had no password policies in force, and also had the smallest stakeholder costs. Other evidence was presented in Ch.7 (particularly section 7.4.7) that common password policies increase stakeholder costs. By comparing the stakeholder costs across systems whose configurations and policies are known, better understanding of the impact of these policies on system performance will be gained. If this happens, we will be better able to diagnose password system ineffectiveness, and have better guarantees that interventions will work. In that spirit, a performance matrix for TACO is presented in Table 49.

Table 49 -Observed password performance matrix for TACO. No password system policies in force.

Goal - let through users	*Real user*	*Attacker*
Denied access	Incorrect rejection rate 10%	Correct rejection rate ?
Let through	Correct acceptance rate 90%	Incorrect acceptance rate ?

8.4.5 Password problems are everyone's problem

Study 6 took place inside a corporate environment (see section 8.3.7). It was calculated that 60% of BT's users would require password reminders over the course of the year, and about three-quarters of users phone the helpdesk once a month or less, and that only 5.8% of password resets were due to repeat offenders (using BT's definition). This refutes the repeat offender hypothesis. Password resets were found to be a problem for all, not just the few.

However, the data collected at UCL showed only 23% of users requiring password reminders. This would appear to contradict the previous findings, as it is much less than BT's 60%. However, the participants of Study 5 used their system for only 10 weeks of the year, while those of Study 6 were observed for six months of their all year round computer use. The participants of Study 6 had more time to experience problems. Only if Study 5's results were projected to a year's worth of system use could a proper comparison be made. However, the relationship between time and cumulative number of helpdesk users is unlikely to be linear. More research is needed to construct a method for projecting annual numbers of password helpdesk users from short duration studies.

8.4.6 Password changing is a vulnerable time, so password expiry increases problems

Study 4 (section 8.4.1) shows us more evidence that password changing is a sensitive time that is predisposed to password problems. This is more evidence that password expiry has negative consequences on password system performance. To fully understand its impact, it would be necessary to investigate this experimentally, manipulating the intervals between password expiry and recording the result.

8.4.7 Passwords interfere with other passwords

Another striking result was the large extent to which passwords interfere with each other-37% of the errors found in Study 4 were the use of expired passwords, and another 15% were the use of other password like sequences. This points to several fairly obvious conclusions:

- The more passwords, the greater the trouble with them
- The more frequent password expiry is, the greater the number of problems
- (and together with section 7.3.2) password confusion is the next biggest memory problem with passwords after forgetting

This makes password confusion the second or third largest source of password problems. Password confusion is therefore a significant problem, and it has to date not been addressed by research on password system performance.

This issue shows a difficulty in password content or memory based interventions-if you make passwords more memorable, you make them more prone to confusion because the distractor passwords come more easily to memory (section 3.3.5). There are three main approaches to solving this dilemma. One could either ignore the problem of password confusion as it is a lesser problem, one could engage in a difficult balancing act to make passwords just memorable enough, or one could try and reduce the number of passwords. In practice, it seems likely that the first approach will be taken. It also seems unlikely that the real number of passwords will be decreased, given that security departments are conservative and that corporations have so much to defend. It seems more likely that corporations would pursue a technical solution, reducing the effective number of passwords by allocating their use to a machine/or middleware on the user's behalf. Such technology is called a Single Sign-On (SSO). Such technologies tackle both main problems-password numbers are reduced so they are less likely to be confused with each other, and what passwords are left are used more frequently so they are less likely to be forgotten. SSOs can also protect against *ghost* accounts: by centralising the control of authentication, it makes it easier to remove an individual's permissions when they have left the organisation. SSO can therefore take up some of the responsibilities given to password expiry mechanisms and so lessen the need for them. However, the solutions can be expensive to implement, both because of the complexity and amount of integration required, and also because they must be better defended than individual password systems-as more systems fall under an SSO it becomes more fundamental to the running of the organisation, and so a more devastating and therefore likely target.

8.4.8 Long interval hypothesis supported

More empirical evidence has been found (section 8.3.3) supporting the long interval hypothesis-that passwords are forgotten if they are infrequently used. Though this is not a startling result, it is important to check fundamental assumptions. Moreover, studies of these kinds may eventually help to predict helpdesk usage for IT systems which are on the drawing board-by knowing the approximate frequency of their use, it may be possible to estimate their future authentication problems.

8.5 Summary

This chapter collected data toward answering Research Question A - *What is the performance of password systems*? Study 5 found that users of a UCL password system experienced substantial login failure rates (overall 1 in 10 attempts failing).

Whilst user report data has identified similar problems (Adams and Sasse 1999), the extent of failure had not been quantified in the security or HCI literature to date. This study therefore represents an increment forward in the evaluation of user-authentication mechanisms, and computer safeguards more widely.

Progress was made towards answering *Research Question C - What causes this performance to be good or bad*? Data was collected that shed light on the role of three strikes policies on password system performance, the validity of the repeat offender hypothesis, and the effect of password expiry policies & long intervals between password uses on performance.

By examining the number of login problems experienced by people who did and did not have to request a password reminder, we found that three strikes would be likely to cause substantially more reminders than ten strikes. This finding led to progress in answering Research Question D - *What interventions can be made to improve password system performance*? It was suggested that moving from three strikes to ten would lead to substantially reduced helpdesk use, with a very small change in the amount of protection offered against attackers, so little in fact that it would be negligible.

The strain imposed on BT password helpdesks was not due to a small minority of repeat offenders, but in fact was due to huge numbers of normal users. It was calculated that 60% of BT's users would require support for their passwords from helpdesks in a typical year, and that only 5.8% of password resets were due to repeat offenders.

Further evidence was found that changing a password is a particularly sensitive activity that can cause problems. Furthermore, password confusion was found to be the second or third most frequent password related problem, and that entering obsolete passwords instead of the currently valid one was the largest source of such errors. It was suggested that these were further evidence for the deleterious effect of password expiry policies. It was further suggested that single sign-on technologies could in some senses replace password expiry mechanisms.

Evidence was presented that long intervals between uses can lead to problems with passwords. This examination was justified as supporting our fundamental theoretical assumptions.

It was discovered that the participants in studies 4 and 5 were atypical because they chose particularly strong passwords. This and other aspects of the UCL environment (such as its lower rate of password problems) give us caution in extrapolating the results from UCL to BT. This caution should be shared by researchers in interpreting their studies results.

8.6 Contributions

8.6.1 Methodological

A novel paradigm for capturing password usability data (an enhanced password system logging mechanism) was introduced. This allowed diagnostic information to be collected that was not previously available.

8.6.2 Substantive

An empirical model of failed log in attempts was presented that can be used to make better decisions about the number of strikes allowed by password policies. The data on which the model was based suggested that ten strikes would lead to less helpdesk use than three strikes, with little reduction in the security offered.

Differences between university and corporate user populations were documented that could prevent meaningful extrapolations of results from one population to the other. For example: undergraduates in study 4 had substantially stronger passwords than the general population. This can form the basis of further research.

A hypothesis about the causes of password system performance was refuted. Evidence was presented that password helpdesk use is by the majority of users, not a small hard-core of *repeat offenders*. This finding may be used in risk management or further research.

The documenting of the use of password system support services (0.7 password reminders per user per year) for a system with known password policies. This could be used to compare to other systems with different but known password policies, and so study the effects of different password policies on password system effectiveness.

Password confusion was identified as a major source of password errors and requests for help. This source of error has been previously neglected in the literature, and is evidence that password expiry policies increase helpdesk use. This finding can feed into further research and eventually into risk analyses.

Ecologically valid evidence was presented that supports a fundamental assumption about password system performance-long intervals mean forgotten passwords.

Chapter 9

1st pass abstraction and interventions

9.1 Introduction

This chapter will present a 1st pass of abstraction of the research from the previous 4 chapters to diagnose the problems in password system performance. It will start by introducing a model that summarises the causes of password problems identified. A preparatory section introduces a taxonomy of password costs. Finally, different interventions to reduce password costs will be introduced and related to the model.

9.2 Diagnosis of Password System ineffectiveness

This section introduces a model that is abstracted from the results so far-the diaries and interviews (section 6.3.1), questionnaire responses (section 7.3), and system log data (section 8.3). The model summarises the causes of password system performance, and so forms a diagnosis of it. However, the model evolved as a result of validation activities described in Chapter 10. Rather than present both old and new models in detail, only the new model is given, and the reader is referred to Chapter 11 to view it. However, an outline of the revised model is given below.

The model consists of eight stages, describing different sets of factors that impact password system performance. Each stage feeds into the one below it, but stages lower in the model may feed up to higher stages:

1. Mechanism stage is at the top of the model, describing perceived and real properties of password mechanisms
2. Security administration stage, describing the administrative contribution to password systems through allocation of function between users and mechanisms, security policies, disciplinary procedures and publicity surrounding breaches
3. Environment stage , describing the social, psychological and business context surrounding a user's participation in a password system: the users perceptions of resource sensitivity and associated risks, the availability of sources of support, the prevailing working practices, and the timing of critical business tasks to which the user must conform.
4. Design stage, which describes the user's strategies for designing and using passwords.
5. Input stage, describes the properties of passwords and prompts, and the task environment in which they are used

6. Activator stage describes psychological phenomena that occur in planning and execution of plans to use passwords - taken from the literature of human error (see section 3.3.5).
7. Schemata stage describes the actual cognitive processing of plans to use passwords
8. Output stage describes the resulting successful or unsuccessful use of passwords.

Notes on the derivation of interventions from the model are given in section 9.4.

9.3 Costs of password problems: taxonomy and estimates

Section 2.2.2 defined a worksystem's performance as being divisible into quality of the work achieved, and the costs accrued in achieving that quality. This section will examine the various stakeholder costs that may occur in password systems. They have been split into three categories outlined in Figure 34, and given in more depth in Table 50.

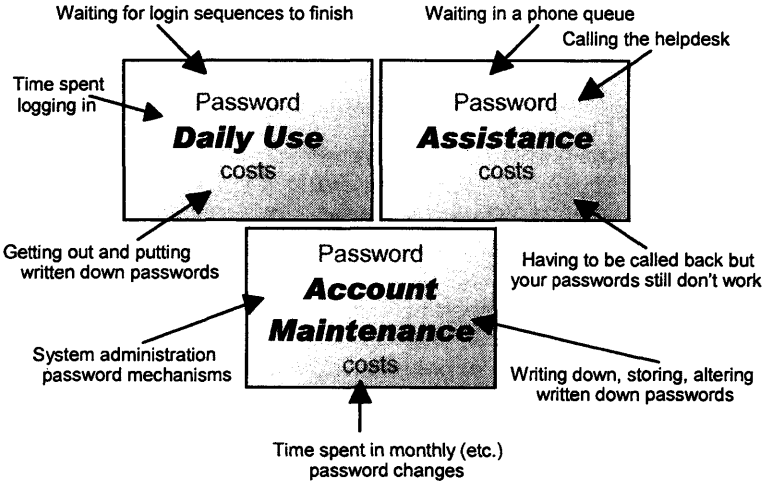


Figure 34 - Stakeholder costs of password systems.

Table 50 - Taxonomy of Password system costs

	Password *Daily Use* Costs	Password *Account Maintenance* Costs	Password *Assistence* Costs
Users	*A* 1. Getting out and putting back written down passwords & prompts 2. Time preparing the workstation to connect to remote resources 3. Time spent typing in passwords 4. Waiting for response from resources 5. Re-establishing timed out connections 6. Time spent opening screen locks	*B* 1. Locating and initiating password change function 2. Getting out and putting back written down passwords & prompts 3. Entering exisiting passwords 4. Designing new passwords 5. Confirming new passwords 6. Redesigning and re-entering passwords if designs don't meet mechanisms' requirements. 7. Ammending written down passwords and prompts	*C* 1. Time spent a. Locating and initiating contact with password helpdesk / support systems b. Waiting in queue c. Authenticating with helpdesk d. Describing problem e. Receiving new account details f. Storing new account details / Ammending written down passwords and prompts 2. Reputation damage 3. Deadline pressure
Secondary stakeholders (Systems administrators, line managers, helpdesk analysts, etc.)	*D* 1. Included in Cell A	*E* 1. Providing new users with accounts 2. Performing audits 3. Removing obsolete accounts 4. Patching / updating the system 5. Contributing to purchasing decision making 6. Installing new systems 7. Decommisionning old systems	*F* 1. Authenticate users 2. Listen to problem description 3. Initiate resetting of passwords 4. Transmit new passwords to users in secure fashion 5. Ensure details of transaction are properly stored
Tertiary stakeholders (Executives, board members, shareholders, etc.)	*G* 1. Sum of Cell A for all users	*H* 1. Sum of Cells B & E across all employees	*I* 1. Sum of Cells C & F across all employees

9.3.1 *Costs to the user*

Password *daily use* costs, is the time spent in normal use of passwords-logging in, preparing for logging in, etc. This is a cost for users and tertiary stakeholders-users are delayed from doing their work, and tertiary stakeholders therefore receive less productivity from users. This cost is rarely mentioned, but may become significant if users are required to use many different computer systems in the course of their work. While this may intuitively seem trivial, the steps required to authenticate are more numerous than recall alone (see section 3.1), and are repeated frequently. Bellcore (cited in Brentano & Wiseth, 1996) estimate that a user with four applications will spend 5.5 days per year simply logging in. If 240 working days per year are used, then this cost averages out at about *half an hour per user per working day*.

How realistic is this estimate? This figure seems high-at least from this researcher's own personal experience. However, the author has not yet found other literature corroborating or falsifying this amount, or been able to find and verify the original research on which this amount was based. Never the less, *daily use* is likely to be a substantial cost for users with an average of 16 passwords (section 7.3.1), and one which it seems reasonable to expect to be not less than a tenth of the reported value (i.e. not less than 3 mins per BT user per working day).

Users must also regularly perform *account maintenance* tasks such as choosing new passwords, amending their password records, etc. This may become significant if they use many password systems that force frequent changing. Studies within BT (Leung, 2000) have found changing passwords to take *15-30 minutes per user per month* (giving a midpoint of 22.5 minutes). If the maintenance tasks are scheduled, then this is the only cost. However, users may be compelled by password expiry mechanisms to perform account maintenance at a critical point in a production task, interrupting it, and potentially causing other costs: missed deadlines, reduced appearance of professionalism with customers, etc.

As we have seen, the password system can break down for a number of reasons (section 8.3.1 and section 7.3.2). When this occurs there are password *assistance* costs. For BT users, this is primarily in time spent contacting the helpdesk to be given a new working password. The average time spent on the phone to Password Control (including being held in the queue) is 9 minutes (Brennan, 2000), with average of 0.8 calls to the helpdesk per BT user per year, or once every fourteen months [3] (though there is a very large spread, as has been shown in section 8.3.3).

[3] 33,000 resets per month on average, equivalent to 8,000 calls per month, times 12 months per year, divided by 120,000 users

Password assistance may be required at any time, is not planned, and so may interrupt production tasks. This is a relatively small interruption, but may occur at critical points during production, such as when dealing with customers. Moreover, some users and tertiary stakeholders may be in circumstances where any interruption is intolerable - for example in call centres, where every second must be accounted for to the extent that even toilet breaks require permission.

9.3.2 *Costs to the organisation*

The relative cost to BT of these different sources of password related expense is displayed in Figure 35. It is hypothesised that these costs will be in a similar ratio and magnitude for most organisations of a similar size. Relevant data were not available for UCL, so a comparison was not possible.

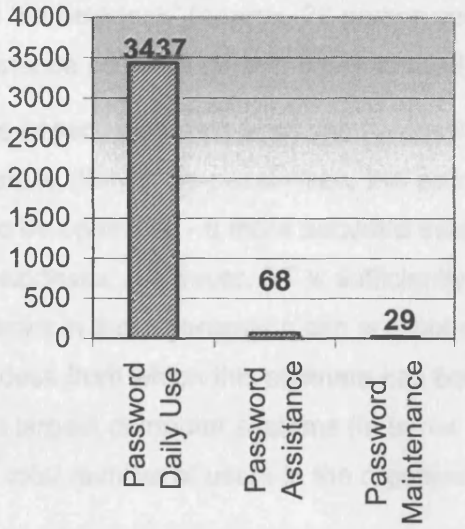


Figure 35 - BT's annual password costs, in person-years.

When users spend time interacting as part of a password system, this unproductive time can be thought of as a cost to the organisation and tertiary stakeholders. In an organisation on the scale of BT, password *account maintenance* costs *is equivalent to 29.25 person years* annually (for an average of 22.5 mins per user, assuming an 8 hour working day, 240 working days/year, for 150,000 employees). The work of secondary stakeholders who administer the password systems (adding and deleting accounts) is generally replicated dozens of times over in the organisation, because the authentication mechanisms that are installed are not designed to interoperate. However, it is assumed that the cost of replicated administration by secondary stakeholders is relatively small compared to the cost of users' account maintenance, and so this part of password maintenance costs has not been included in the estimate.

The Bellcore estimate of password *daily use* costs (cited in Brentano & Wiseth, 1996)is far more significant when considered at the scale of BT. 5.5 days per year per user *is equivalent to 3,437 person years annually*, across the whole of BT - an enormous cost for tertiary stakeholders.

From the perspective of an organisation, Password *assistance* costs are those that are associated with helping users when the password system breaks down, including all the costs associated with running a helpdesk, and time spent by the users themselves in troubleshooting, or liasing with secondary stakeholders. Approximately half of computer helpdesk calls are password related (Carden, 1999; Murrer, 1999). This organisational cost is particularly mentioned in arguments promoting the use of biometrics (e.g. Murrer, 1999). BT's central password helpdesk currently costs 40 person-years annually to run, in support staff's salaries alone. At 9 minutes per call, with an average monthly call volume of 30,000, including the costs of primary users' time spent interacting with the helpdesk[4] (approx. 28 person years annually) would make BT's password assistance costs 68 person-years annually.

However, these figures are based upon data from one helpdesk. Whereas the password daily use estimate is likely to be pessimistic, this estimate of password *assistance* costs is likely to be optimistic - a more accurate estimate would include the costs of other password helpdesks. However, BT is sufficiently large and complex that even identifying the helpdesks in the organisation can present a significant challenge for researchers. The helpdesk from which this estimate has been made is BT's largest helpdesk. It supports BT's largest computer systems (in terms of numbers of users), with a large fraction of the total number of users in the organisation registered with it.

9.4 Potential Interventions

A number of potential interventions to improve password system effectiveness (though not an exhaustive list) are presented in Table 51.

The interventions are derived from the model. Some of the derivations are simple. For example model component 6 of the Security Administration stage (see section 11.2 for model) contains a variable *number of strikes* - where a low number of (three) strikes increases login problems and so reduces password system performance. There is a simple step from this to the intervention (#7 in Table 52) *more than three strikes.*

However, many of the interventions do not have simple one to one derivations with model components because the model components are interrelated. For example, the

[4] (9min per call /60min per hour/8 hours per day/240 days per year) times (30,000 calls per month times 12 months/year)

Table 51 - Potential interventions to reduce password costs

Relevent Model Stage		Intervention	Model component
Mechanism			
	1.	Better encryption of passwords	1
	2.	Standard password policy requirements in purchasing	3
Security Administration			
	3.	Password synchronisation SSO	5
	4.	Automated login SSO	5
	5.	Automated login SSO with centralised administration	5
	6.	Less frequent password changing	6
	7.	More than three strikes	6
	8.	Declassification of systems	6
	9.	Standardised password restrictions	6
	10.	Support for Writing down passwords	6
	11.	Punishing users bad password behaviour	7
	12.	Rewarding of users good password behaviour	7
	13.	Highlighting the occurrence of incidents	8
Environment			
	14.	Education about Attack methods	10
	15.	Education about Risks	10
	16.	Job description changes	10
	17.	Electronic assistant	11
	18.	Common password format generator	11
	19.	Improve helpdesk turnaround time	11
	20.	Cultural change	12
	21.	Intrusion detection systems	13
	22.	Link policy enforcement to personnel dept. (provisioning)	13
Design			
	23.	Mnemonic training	14
	24.	Selection of hints training	14
	25.	Password selection training	15
	26.	Encourage Manual password synchronisation	15
Input			
	27.	Sending reminders of password expiry	20
	28.	7-day multiples for password expiry intervals	20
Activator			
	29.	Increase number of repetitions to verify password	22

intervention *better encryption of passwords* crosses model components 1 (variables: speed and cryptographic strength) and 2 (variable: resistance to dictionary attack) (see section 11.2 for model, and section 4.2.1 for the background of this intervention). Moreover, some interventions entail dependencies between model components-for example *more than three strikes* is a policy intervention (component 6) that is dependent on mechanisms supporting the desired number of strikes (component 3).

In general there will be a single component that is particularly associated with an intervention, and this is noted in Table 52, with the intervention's predicted primary effect on performance. Each intervention could potentially affect all three sources of password cost as well as task quality - even if these effects are not expected. The

author does not attempt to analytically derive estimated effect sizes. Collecting observational data from intervention trials would inform the estimation of expected primary effects and unexpected secondary effects.

Table 52 - Description of interventions and their predicted primary effects.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
[Mechanism stage]			
1. Better encryption for passwords	1	Use slower encryption so that dictionary attacks become orders of magnitude harder.	Improves the strength of password content
2. Standard password policy requirements in purchasing	3	Only purchase authentication mechanisms that will allow a standard set of preferred password policies to be configured and enforced.	Assisted manual password synchronisation, reducing the complexity of the task and thereby reducing the frequency with which it breaks down and requires helpdesk use.
[Security Administration stage]			
3. Password synchronisation SSO	5	Extra software is added behind-the-scenes to synchronise the users passwords. The user is still required to login as frequently as before, only using the same password whereas previously she would have used many. This intervention is less costly to implement and maintain than automated login SSOs (Yasin, 2002). One password synchronisation solution provider claims a 50 to 70 percent reduction in password resets (Yasin, 2002). Purchase costs from $15 per user (Bort, 2002).	The user is able to employ the same passwords across a wider range of systems, thus effectively increasing the frequency with which these passwords are used, and therefore lessening the probability that they will be forgotten, therefore lessening use of helpdesks.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
4. Automated login SSO	5	Extra software is added so that the user only has to login once to use all her computer applications, whereas previously she would have logged in many times. There are different methods of achieving this, which have different costs and benefits (Brentano & Wiseth, 1996). The scripting approach is fairly cheap to implement but has higher administration costs than standard passwords (Brentano & Wiseth, 1996). The middleware approach is expensive to implement but has no additional administration costs compared to standard passwords (Brentano & Wiseth, 1996). Purchase costs around $80 per user (Bort, 2002).	Some logging in tasks are allocated to the computer rather than the user, thereby reducing overall the time spent in logging in.
5. Automated login SSO with centralised administration	5	Extra software is added so that the user only has to login once whereas previously a separate login would be required for each computer application. In addition, administration of access rights for all applications is centralised, making administration much easier (Carden, 1999). This is the most expensive form of SSO to implement, and the cheapest to maintain (Brentano & Wiseth, 1996). Can cost about $1 million to implement (Bort, 2002).	Some logging in tasks are allocated to the computer rather than the user, thereby reducing overall the time spent in logging in.
6. Less frequent password changing	6	Password changing increases the likelihood of password related slips and forgetting. Reducing the frequency of password changing lowers the probability of these problems occurring, and so low as the probability of helpdesk use.	Study 5 showed that password changing is a particularly dangerous time for password problems. Reducing the frequency of passwords changing therefore will reduce password helpdesk use.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
7. More than three strikes	6	Gives users more chances to remember their passwords when logging in, and so predicted to increase availability of computer applications when there is difficulty in remembering of passwords, and so reduce helpdesk use.	Giving people more chances to enter their passwords means that they are more likely to successfully enter it if they are having problems remembering it. This means that they are less likely to get locked out of their accounts, and therefore less likely to require password helpdesk assistance to unlock their accounts.
8. Declassification of systems	6	Reducing the number of password authentication mechanisms reduces the opportunity for password problems. Some computer applications may not require passwords or other user authentication mechanisms, but may be protected in other ways.	Declassifying some systems means that password authentication will not be necessary on them. This means that there will be less daily use of passwords.
9. Standardised password restrictions	6	Allows users to manually synchronise their passwords so that they are able to use one password for their applications, rather than being forced to select many different passwords because of differing requirements.	This intervention means that users will be able to use the same passwords across multiple systems, thereby effectively using what passwords they have more frequently and so making them more memorable and less confusable, and so reducing the probability of problems, and so reducing helpdesk use.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
10. Support for Writing down passwords	6	Mandating that users write down all of their passwords, and giving them the infrastructure required to store their password records securely. Allows users to consult their password record rather than call a helpdesk for a reset.	Writing down passwords means that you have a quicker source of assistance than calling a helpdesk , which means that time and effort are saved if you forget your password.
11. Punishing users bad password behaviour	7	Punishing users for frequent helpdesk use and cryptographically weak password selection may motivate them to improve their practices.	This intervention is designed to motivate people to use password helpdesks less.
12. Rewarding of users good password behaviour	7	Rewarding users for exemplary password use and lack of helpdesk use may motivate them to put more effort into selecting good passwords and managing them effectively.	This intervention is designed to motivate people to use password helpdesk's less.
13. Highlighting the occurrence of incidents	8	Increases users perceptions of risk and vulnerability, and so predicted to motivate users to behave in a more security conscious fashion, including expending more effort in the selection and management of passwords.	Increasing the strength of password content., improving password system task quality.
[Environment stage]			
14. Education about Attack methods	10	Users are taught how passwords are compromised by crackers, which helps them to understand how to make a strong password.	Improve password content strength.
15. Education about Risks	10	Enables users to make informed judgements about the relative risks of computer use, so they can better judge their own vulnerability and risk, and behave in an appropriately security conscious fashion.	Improve password content strength.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
16. Job description changes	10	By adding security duties to job descriptions users are better motivated to behave in a security conscious manner, including choosing and managing passwords.	If people believe that it is their role to pursue security behaviours such as managing their own passwords, and they are less likely to attempts to get others to do these tasks-such as the password helpdesk – Studies 7 and 8. This means that helpdesk's will be used less frequently.
17. Electronic assistant	11	An electronic version of writing down (intervention 10) - allows users to keep a record of their passwords, which can be consulted instead of helpdesk use. May offer the advantage over paper based methods of easier remote access.	Having a record of the passwords to refer to when it cannot be remembered reduces the need for going to a helpdesk.
18. Common password format generator	11	Will give you password format that will allow you to manually password synchronise with a minimum number of passwords.	Simplifies the manual password management task, thereby reducing opportunity for error and so reducing helpdesk use.
19. Improve helpdesk turnaround time	11	Makes helpdesk use quicker, so less time is spent by users and helpdesk workers in resetting passwords. One instance of self-service password resets has been recorded leading to a 60% reduction in helpdesk calls (Yasin, 2002), whilst another gave a 10-13% reduction (Bort, 2002). It is not recorded how long users spent interacting with the software per reset.	Improved helpdesks means less time and effort spent by users when they require password assistance, and potential reduced costs in supplying assistence (through automated, self service systems).

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
20. Cultural change	12	A concerted programme to put security as one of the core values of the organisation, so that people are highly motivated to choose strong passwords and make efforts to use and manage them appropriately.	Increasing the strength of password content. Reducing the risk of disclosure.
21. Intrusion detection system	13	The intrusion detection system monitors logins and system used to highlight unusual patterns. This shifts the emphasis to detection and reaction to attacks rather than using password mechanisms to prevent them-enabling password policies to be relaxed on systems that are less frequently used.	There is reduced need to change the password each time it is used, and so less password maintenance
22. Link policy enforcement to personnel dept. (provisioning)	13	Tying the password policy enforcement system into the personnel department will allow expiry times to be extended if the user is absent.	Reduced helpdesk use as less passwords will expire and need to be reset while users are absent.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
[Desgin stage]			
23. Mnemonic training	14	Gives users techniques to memorise passwords, so that they are less likely to forget them. Techniques need to be applied every time a password is changed, and can seem effortful.	The use of mnemonic's will make passwords more memorable, so there will be fewer cases of them being forgotten, and so fewer instances of helpdesk use. It may also increase the incidence of password confusion, as obsolete passwords would harder to forget, and so more likely to interfere with the new passwords. This would tend to increase helpdesk use. The ballence of effects is likely to depend on each user's relative frequencies of long durations between password uses and password expiries.
24. Selection of hints training	14	Guidance about the creation of hints that can be used to jog one's own memory if one has forgotten a password. Users may therefore be able to avoid calling a helpdesk.	Use of good password hints may enable stakeholders to assist themselves more speedily then a helpdesk, and therefore save some assistance costs.
25. Password selection training	15	Gives people guidance on the creation of strong and memorable password content.	More memorable passwords mean less forgetting, which means less assistance required.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
26. Encourage Manual password synchronisation	15	Users make as many of their passwords the same as possible. When an information system forces the user to change her password, the user voluntarily changes all her other passwords to synchronise them.	Because passwords have been synchronised, there are in effect fewer of them to remember. This should lead to less memory load on the user, and therefore fewer chances for slips, lapses, or forgettings.
[Input stage]			
27. Sending reminders of password expiry	20	Prompts users to change their passwords at a time of their own choice, when they can better pay attention to what they're doing and so be more likely to learn their new password effectively and update their records of it, and so make them less likely to forget it and require helpdesk use.	Sending reminders in advance that passwords will have to be changed means that users will be able to pick a time when to make the changes that will interfere least with their work, and so when they are less likely to suffer divided attention. This means they will be able to pay attention to what they're doing, and so less likely to forget the new password this means they are less likely to require a helpdesk.
28. 7-day multiples for password expiry intervals	20	Allows users to more easily plan their password changing activities, as it leads to easier manipulation of calendars or diaries. Better planning of password changing activities means less opportunity for slips lapses, and mistakes with passwords, or their being forgotten.	This more easily allows people to note in the diaries when password changes are due. This means that password changing is less likely to be a surprise, and therefore that people will be able to do it better, and so are less likely to need helpdesk support.

Intervention Name [& Model Stage]	Inspired by Model Component	Intervention Description	Predicted primary effect
[Activator stage]			
29. Increase number of repetitions to verify password	22	Gives users more practice in correctly recalling a new password, and so leads to better learning of the password, and so accurate recall over longer durations, and so less chance of forgetting their passwords and consequent helpdesk use.	Increasing the number of repetitions to verify passwords would make passwords more memorable , which means they would be less often forgotten , and so users would have less need for helpdesk's.

9.5 Summary & Conclusions

A model was introduced that summarised the causes of password system performance observed in this PhD's data collection (see Chapter 11) - and so has answered Research Question C - *What are the causes of good or bad performance*? This model tied the PhD's results to the literature of human error (section 3.3.5) and HCI research of password system performance (section 4.1.1).

Three categories of password system cost were presented - *Daily use*, *assistance*, and *maintenance* costs. These costs were estimated for BT as 3437, 134 and 29 person years annually (relevant data were not available for UCL), adding to this thesis' answer to Research Question A - *What is the performance of password systems in actual use*? It was hypothesised that similarly sized large organisations would have similar ratios of password costs.

The model of password performance causes was used to inspire the design of 29 interventions - forming an answer to Research Question D - *What interventions can be made to improve the performance of password systems*? These were outlined with their predicted main mode of action and primary effect. It was noted that interventions may not have a simple mapping to components of the model.

9.6 Contributions

9.6.1 Substantive

A taxonomy of password costs was presented, with an estimate of their magnitudes in a large organisation. These may inform further password system research and practice.

Interventions designed to improve performance were presented that were based on a model of the causes of password system performance in large organisations. These may also further password system research and practice.

Chapter 10

Studies 7 & 8: Validation of interventions

10.1 Introduction

It is impractical within the time-frame and scope of a Ph.D. to implement and validate more than a few of the interventions identified in the previous chapter (section 9.4). It was decided that validation would be most effective with BT users, but this entailed restrictions in the choice of research methods, which guided the choice of interventions to be validated (see section 5.4.7). The interventions chosen were a subset focussing on 'redesigning' users and avoiding changes to infrastructure or policies. All but three (interventions: 23 - mnemonic training, 14 - education about attack methods and 15 - education about risks) of this subset were chosen. The selected interventions were predicted to improve performance by reducing password costs. The three rejected interventions were thought to be unsustainable (intervention 23 - mnemonic training), or increase costs (14 - education about attack methods and 15 - education about risks) by motivating users to behave more securely but not more efficiently. The included interventions comprised:

1. ***Manual password synchronisation*** (intervention 26 in Ch 9)
2. ***Password selection*** (intervention 25 in Ch 9)
3. ***Selection of hints*** (intervention 24 in Ch 9)
4. ***Writing down passwords*** (intervention 10 in Ch 9)

It was decided to combine the interventions into one package, to be called a Password Manager (PM, see Appendix 6).

This validation activity attempts to address two questions:

1. Relevance - If adopted, would the interventions address users' problems in a positive way?
2. Acceptability - Would users be motivated to adopt or subvert the interventions?

Both of these questions need to be answered in the affirmative for the interventions to be effective: if the interventions do not positively address users' problems, then there will be little impact on password system performance so the interventions will be pointless. Moverover, interventions change the password system, and if users do not accept the new security system they would attempt to subvert it (Adams & Sasse, 1999) and so password system performance would be lowered through reduced security and/or additional costs instead of improved.

10.2 Description of PM

The Password Manager was designed to make the password task easier for users so that they would suffer fewer problems (reduced *assistance* costs), and give users training about password related security issues to enable them to be more security conscious without significantly increasing costs. An example Password Manager is shown in Appendix 6. The features of the PM are described according to the intervention they implement:

1. The first feature of the Password Manager (PM) is support for ***Manual password synchronisation.*** It directs users to split the computer systems they use into groups according to how often the password needs to be changed. This combines a support tool with training - computer systems that require frequent password changing usually protect more sensitive data and resources - the systems require stronger security, and therefore stronger passwords. The tool prompts users to group systems together in this way, and give stronger passwords to systems with requirements for stronger security.

   Users are instructed where possible to give one password to each group of systems-and to synchronise the passwords within each group by changing all the passwords of a group at the same time. This is designed to simplify the password task, by reducing the effective number of passwords that have to be remembered. Synchronising the passwords on different systems is a technique already used by many users. However, the PM version improves security because it reduces the threat of password harvesting by reducing the use of sensitive passwords on insecure systems, for example: using one's customer records system password on an external web site.

   The PM supports this by asking users to write down the names of the systems in each group on separate pages. When the time comes to update the passwords in each group, the users will be reminded of each group's contents so that they do not miss a system out. Moreover, users are instructed to record which systems they have updated, so they do not lose track of which systems they have updated and which they have not. This should increase the rate of successful synchronisation, and reduce helpdesk use and assistance costs.

   By grouping systems together and changing them at the same time, the user is less likely to forget her password for two reasons: by using it more frequently at the beginning (when changing all the passwords in a group) she will have more practice, which leads to better memory; and by having the same password on

several systems, this password will by definition be used more frequently, again leading to more practice and so greater memorability.

The PM has the additional feature that it prompts users to login to systems that would otherwise suspend or delete their accounts due to their inactivity. This is intimately associated with password expiry policies, which are a common cause of password resets (see section 7.3.2). Again, users are prompted to record which systems they have logged into, and which they have not. The PM therefore trains people about the importance of logging into systems that one might not otherwise do, and acts as a tool that will help them to keep track of which systems are outstanding, and which systems they have dealt with. This should reduce helpdesk use due to expired accounts, and so reduce *assistance* costs, though would increase *maintenance* costs.

The PM also supports users in changing passwords on systems that have automated password synchronisation. The automation on the systems often works in a complex web of propagation of password changes. If the user changes the password of the wrong system some of the systems will be synchronised to it, and others will not be. This will cause difficulties that will result in a call to the password helpdesk to have all the passwords reset. The complex password propagation relationships between the systems are simplified in the PM so that they are linear- users are forewarned about the possible difficulties of the automated synchronisation, and shown a safe route through it.

2. The second feature of PM is ***Password selection:*** it tells users how to select passwords of different strengths that are designed to be memorable, thereby reducing *assistance* costs (passwords are designed to reduce helpdesk use due to forgetting). Users are given algorithms for generating passwords of different strengths that they can learn (a training aspect), or merely refer to when needed (a tool aspect).

3. In feature three, ***Selection of hints***, users are given instructions on how to compose hints for their passwords, so that the hints are helpful to users should they forget their passwords without being clues to help attackers make guesses. This is designed to improve security, by lowering the risk of disclosure, for example cutting down the number of passwords that need to be written down on a sticky note that has been stuck to the screen. The PM therefore has training aspect, instructing the user about secure behaviours, and has a tool aspect, a reference to aid the users' memory of passwords.

4. The fourth feature of the PM is a limited form of ***Writing down passwords***. Users are instructed to write down passwords that are for systems requiring low security. This should reduce *assistance* costs, as users will be able to refer to their passwords, and so be less likely to forget them and require helpdesk use. *Maintence* costs should be relatively unaffected, as low security systems are associated with reduced or absent password expiry policies, which would therefore require few updates to the written down passwords. The limitation to low security systems only was introduced to lower the risk of *disclosure*, because it was not believed the PM would be kept securely (i.e. not in a locked drawer).

Finally, should the worst happen and a password reset is required, the PM facilitates this by containing the password reset team's contact details, potentially cutting down the time to establish a connection to the helpdesk, and so reducing *assistance* costs.

10.3 Method

Five focus groups were held at BT. The focus groups were shared between the author and another researcher because access to users in BT was only possible at BT's request that researchers doubled up. In both cases the other researcher was another PhD student researching human factors and security.

10.3.1 Study 7 (Focus groups I to 3)

Participants

Three focus groups were held in a research park belonging to BT. The groups contained 4, 5 and 5 participants, in addition to the author and co-researcher. Participants were recruited by the author's industrial supervisor, who had advertised inside BT with e-mail circulars. The participants consisted of:

- Security researchers
- System administrators
- Human factors specialists
- Development technologists/engineers
- Contractors
- Receptionists
- Postgraduate students on placement

Apparatus

The password manager was combined with a document testing a fear appeal (created by the co-researcher) that had been designed to encourage users to behave in a more security conscious way, offering the password manager as a guide to illustrate best practice and tool to help achieve it.

Each discussion was recorded onto standard D90 cassette, using an omni-directional boundary microphone plugged into a standard cassette recorder. Each recording was subsequently transcribed.

Procedure

Half of each focus group's duration was spent discussing the password manager, and the rest spent discussing topics related to the other student's research. Focus groups lasted approximately one hour, with half an hour devoted to the password manager per session.

Participants were welcomed to each group, asked permission that the session be recorded to audiotape, offered refreshments, and all focus group members introduced themselves to each other, and the discussion started. Participants were first asked to read the fear appeal, and asked questions about this by the other researcher. In the second half of the focus group, the password manager was outlined to the participants (illustrated with a sheet taken from it), who then gave their reactions to it, and spoke more generally about their use of passwords.

10.3.2 Study 8 (Focus groups 4 and 5)

Participants

Two further focus groups were held at BT, this time in one of its operational arms - where the organisation carries out some of its core profit-making activities. Participants were recruited internally by a contact of the researchers inside the organisation, with the promise of drinks bought for them after work.

The participants consisted of two groups of 5, containing a mixture of people who often came into contact with sensitive customer data:

- Administrative personnel
- Technical support engineers, and
- Their line managers

Procedure

Less time was available, and so focus groups were restricted to 40 minutes split between the two researchers. As before, participants were welcomed to each group, asked permission that the session be recorded to audiotape, and discussion started.

The topics of these groups were slightly different than in the previous series. The author led the first half of the groups in a discussion of how many passwords/systems the participants used, how they managed them, and the ways and consequences of their management techniques failing, covering all of the techniques in the PM. The other researcher then led the rest of the group in a discussion of their attitudes to security, the understanding of it, and their feelings of vulnerability to attack.

Apparatus

As before, the discussion with recorded onto a 90 cassette with an omnidirectional boundary mic and standard cassette recorder, and the recordings were then transcribed.

10.4 Results

10.4.1 Study 7

Relevance of interventions

Participants described having experienced problems that the password manager was designed to solve. These included:

1. The proliferation of passwords/how to remember them all (7 users / 50%)
2. Losing track of one's place in the process of updating passwords (4 users / 29%)
3. Problems with password expiry being more frequent than the need to use systems (3 users / 21%)
4. Failing to sign into systems that consequently suspended their accounts because of lack of activity. (2 users / 14%)
5. Forgetting infrequently using passwords (2 users / 14%)

Some of the participants used solutions to these problems that are similar to those promoted in the password manager. For example:

6. Giving different systems the same password, to reduce the overall number to be remembered (10 users / 71%)
7. Grouping systems according to the security that they need (2 users / 14%)
8. Making reminders instead of writing down the password (1 user / 7%)
9. Changing all passwords at the same time to keep them synchronised (1 user / 7%)

Acceptability of interventions

Participants made negative comments when asked about the password manager implementation that they had been presented with, indicating they found it to be unnacceptible. Criticisms were various, including:

10. The wrong problem is being addressed-changes should be made to the systems, not the people who have to use them (8 users / 57%)
11. A call to the helpdesk would be preferable to the effort involved in using the PM (2 users / 14%)
12. Techniques used in the password manager may clash with personally preferred techniques already used, for example: how to choose passwords (5 users / 36%)
13. It's hard to tell how often each password needs to be reset, and so which password should go in which PM group (1 user / 7%)
14. It is clumsy and effortful to store securely, so it will end up being stored not-securely therefore making security worse (2 users / 14%)
15. It would make its users vulnerable to version control problems and / or denial of service if they work in more than one location, or are travelling. (3 users / 21%)
16. It would only be suitable for people who work in a very organised kind of way, which is no one round here! (1 user / 7%)

17. It would take too long to learn how to use (and be used too infrequently to remember how-implying that it need be re-learned each time), too long to set up and too long to use (3 users / 21%)
18. They did not believe it could offer a way of creating memorable and secure passwords and password reminders (4 users / 29%)
19. It's just not very good (5 users / 36%)
20. They would not use it unless they were sure they would be caught and punished (4 users / 29%)

10.4.2 Study 8

Relevance of interventions

As in Study 7, participants in Study 8 described experiencing problems that the password manager was designed to solve; evidence that the interventions were relevant and would make a positive impact if adopted. Study 8 participants mentioned some of the password problems given in Study 7:

1. Losing track of one's place in the process of updating passwords (4 users / 40%)
2. Failing to sign into systems that consequently suspended their accounts because of lack of activity. (2 users / 20%)
3. The proliferation of passwords/how to remember them all (1 user / 10%)

Study 8 participants mentioned in addition:

4. Difficulty in giving the same password to different applications/systems (implicit in Study 7) (4 users / 40%)

Study 8 participants used fewer techniques from the PM already than those in Study 7:

- Losing track of one's place in the process of updating passwords (4 users / 40%)
- Giving different systems the same password, to reduce the overall number to be remembered (4 users / 40%)
- Changing all passwords at the same time to keep them synchronised (3 users / 30%)

Acceptability of interventions

Fewer negative points were raised, consisting only of:

- The problem of secure storage and (1 user / 10%)
- The problem of access for mobile workers (1 user / 10%).

10.5 Discussion

The data in previous chapters were analysed to diagnose the performance of password authentication systems. These results were used to generate potential interventions (section 9.4). A subset of interventions focusing on user behaviours were brought together in a paper-based tool, which was discussed in five focus groups of BT users (permission was not given to deploy the tool experimentally). If users experienced problems that the interventions were designed to tackle, and users found the interventions acceptable, then some validation of the interventions had been achieved. This discussion will therefore focus on the relevance of the proposed interventions, and their acceptability to users.

Relevance of interventions

It was found that users in both focus groups had password problems the PM was designed to combat:

- The proliferation of passwords/how to remember them all (8 users / 3%)
- Losing track of one's place in the process of updating passwords (8 users /33%)
- Failing to sign into systems that consequently suspended their accounts because of lack of activity. (4 users /17%)
- Forgetting infrequently used passwords (2 users /8%)

The PM was therefore relevant in that it addressed problems experienced by a substantial number of participants. Some of the participants used solutions to these problems that were similar to those promoted in the password manager:

- Giving different systems the same password, to reduce the overall number to be remembered (14 users / 58% of sample)

- Changing all passwords at the same time to keep them synchronised (4 users / 17%)
- Grouping systems according to the security that they need (2 users / 8%)
- Making reminders instead of writing down the password (1 user / 4%)

This showed that the *Manual password synchronisation* and *Selection of hints* interventions in the PM in particular were relevant in that users already deployed them. However, Study 8 users raised specific issues militating against their personal use of a paper based PM that was used as a password repository: their mobility - working at different locations and the PM's consequently reduced availability.

Acceptability of interventions

The data collected from Study 7 focus groups show that the PM interventions on their own would not prove acceptible, with a large proportion (57%) of this powerful group of users (manager grades, researchers and developers) predicting that they would merely pay lip service to the interventions unless they were accompanied by technical or administrative or cultural changes. These participants described having experienced problems that the PM would solve if used, and in fact already employed some of the techniques suggested. However, a third of these participants reported that they would be unwilling to use the PM unless they were sure to be caught and displined for avoiding it, showing that they found it unacceptable, and would risk punishment in their attempts to subvert it.

There is a potential feature of the password manager that could defeat this unwillingness by means of usefulness. This feature was raised in Study 8. Though it was not initially conceived as being an explicit part of the PM functions, it is very much in its spirit of training users how to pick memorable and secure passwords, and with applying the same password where possible within each of the groups of systems. These interventionscould be extended to promote algorithms that produce universal passwords that would be acceptable on all the systems in BT; or at least as widely accepted as possible within each password expiry group. Even if the resulting password was complex, the overall reduction in complexity of the password management task would more than compensate. By offering such a useful feature, previously unwilling users may be seduced into using, or at least looking at a password manager. However, permission was not given to test this. Moreover, user based password synchronisation is still a lot of effort for users. The organisation is likely to receive far greater benefits by supporting users with machine based password

synchronisation or single sign-on technologies than with a PM. Indeed, by highlighting the ease with which passwords can be manually synchronised, the organisation also reveals the repetitive nature of the procedure and so its ripeness for automation. By forcing users to perform simple yet repetitive and time consuming password maintenance, greater discontent might be brewed with the security architecture. There is therefore a paradoxical risk that simplifying the password management task (with a half-way measure) could foment greater dissatisfaction with and less respect for security among users.

The Study 8 participants (who were from a more operational and lower status part of the organisation) were generally more concerned with security than Study 7 participants, for their own and the organisation's sake. They too reported a similar mix of problems that the PM would help with. They seemed more open to techniques that could help them.

It should be noted that some participants in both series already employed some of the techniques promoted in the password manager - particularly grouping systems together and giving the same password (70% in Study 7, 40% in Study 8), and that many of these participants still found password management difficult, and had what they reported as extensive contact with helpdesks as a result. Employing the password manager in its entirety rather than only a small selection of its recommendations could lead to far fewer forgettings and confusions. However, despite best efforts to make it simple and quick, learning and using the password manager is likely to require a significant investment in time and effort. In weighing the benefits and costs of using it, users may come to a conclusion that it is better for them to avoid using it and simply call the helpdesk instead - this is a rational decision on their part, as the password costs are shared out with other parts of BT, not just borne by them personally.

In this context, encouraging users to take more responsibility for password management was seen by users in Study 7 as a cynical ploy by the technical support department to reduce its costs at the expense of everyone else. Thus users lose yet more respect for security in the organisation; see less point in conforming to it, and so the "weakest link" is weakened (cf. Adams & Sasse, 1999). Study 7 users' attitude of "someone else's problem" represents a difference in culture and perception of risk between the two study sites. Extra interventions could encourage Study 7 users to act in a way that would improve security and reduce helpdesk use: cultural change programs, publicising *impacts*, contractual obligations to perform security management functions and punishment if they don't (to make users perceive higher personal risk). However, these will come at the expense of allocating security tasks to stakeholders who could be carrying out production tasks instead. Overall, a strategy of having users

shoulder the security burden rather than technical and organisational infrastructure could cost BT more to achieve the same security.

10.6 Summary & Conclusions

A package of interventions designed to improve password system performance was created containing information and paper-based tools to support users in: the selection of passwords, selection of password hints, manual password synchronisation, and the writing down of passwords. This package was reviewed by BT users from two different parts of the organisation in a series of focus groups, to increase the level of confidence that these interventions addressed appropriate problems and would be acceptable. It was found that these interventions, and by extension user based interventions have a role to play, but will not be very effective on their own because they suffer from lack of buy in from powerful groups of stakeholders.

User based interventions *might* be made more acceptable to users if they included algorithms for generating passwords that would be universally acceptable across systems with different password constraints (ie, official support for user based password synchronisation), thereby dramatically reducing the number of passwords users' had to manage. This is at best a tentative conclusion - such an intervention might also reduce security's standing among users.

However, a paper-based password management tool approach would not suit mobile users. Moreover, machine based password synchronisation would lead to greater cost savings to the users themselves than only manual password synchronisation. User based interventions would therefore be much more acceptable to users (and so apopted by them rather than subverted) if combined with technical and organisational changes.

10.7 Contributions

10.7.1 Substantive

The existence of different agendas of different groups of stakeholders in the organisation has been identified. This has important implications for researchers and practitioners designing interventions to improve password system performance. For example, it reveals the necessity of supporting user based interventions with technical and organisational interventions, and identifies a risk in attempting user based interventions on their own. Studies 7 and 8 therefore partially validate the requirement for a holistic approach to security. By so doing the necessity is established for models

such as have resulted from this research and presented in chapters 11 and 12, that encompass human memory as well as technical and organisational aspects of password system performance.

Chapter 11

2nd pass abstraction: password security and human error on a small-scale

11.1 Introduction

This chapter abstracts the findings from the previous chapters into a model (Figure 36). The model is described in both prose and diagram form. Some benefits of the model are presented.

11.2 Description of model

This small scale framework focuses on how factors combine inside the user to produce password system performance. This framework is composed of several different stages: mechanism, security administration, environment, design, input, activator, schema and output stages. The first five stages build upon this PhD's findings and Adams et al's (1997) model of password usability, and the final three stages integrate Reason's (1990) conception of human error.

11.2.1 Mechanism stage

The first stage addresses the perceived qualities of password mechanisms. In some cases, the perceived qualities match the actual qualities of password mechanisms - the strengths and weaknesses of password encryption algorithms (model component 1) and password files (model component 2) are well understood. These qualities are given in the model as they are known to be for UNIX password mechanisms. This is the model from which password content recommendations are implicitly generated, and is therefore the underlying model of password mechanisms that are used in risk management. This is important because the UNIX password mechanism is considered poor by modern standards, and so its use requires users to choose password content that makes up for its inadequacies (section 4.2.1).

In contrast, password mechanism performance (model component 4) is a quality that is either assumed to be adequate, or not adequate (but not researched). For example, the *incorrect rejection* rate (see 8.4.4) of password mechanisms has never been studied. Similarly, there does not seem to be published work about the *incorrect acceptance* rate associated with password systems in real world studies. Moreover, the costs associated with users' password daily use, maintenance, and assistance is not discussed within the password performance literature.

Different password mechanisms may have built into them different configuration options (model component 3). These may limit the policy choices available - for example a choice between no policies and 6 digit PINs forced upon the administrators of the voice mail system in Study 3 (section 7.3.9).

These data, perceived and real values and configurations options feed into risk management, which is undertaken in the next stage: *security administration.*

11.2.2 Security administration stage

The basis of security is the management of risk-the assessment of the likelihood and severity of impacts, their subsequent prioritisation, and the allocation of resources to protect against them. However, security administrators are not in possession of all the information they require: for example the costs of password mechanisms to individual users, and in aggregate to the organisation (section 9.3) are not fully appreciated, and it is still not certain what protection they offer.

Security administrators set policy (model component 6) and allocate security functions (model component 5) on the basis that the allocations have an acceptable cost, but these administrators do not know what the costs are because the costs had not started to be measured until this PhD. Moreover, instead of adapting the password mechanisms to protect against dictionary attack -an intervention which is know to have high costs: security administrators allocate this function to users (where the costs are not measured).

There are other difficulties in risk management relating to passwords. Password mechanisms are vulnerable to many different attacks, but few can be subjected to persuasive numerical analysis. Dictionary attacks are one of the threats where it is possible to (seemingly) easily calculate the effect of a defensive measure. Because defending against dictionary attack is easy (and is believed to have negligible costs), it becomes a focus and priority of the security architecture. Contrast this with the social engineering vulnerability: it cannot easily be predicted how much protection is offered by n-minutes of training. This is a general problem in the allocation of resources to production and security (see Brostoff & Sasse, 2001).

The next factor is disciplinary procedures (model component 7). These appeared to promote secure behaviour in Study 9 focus group participants (section 10.4.2, 10.5), who feared for their jobs if some security breach occurred with one of their computer accounts. However, these were relatively politically weak employees. Study 8 participants were generally several grades higher in the organisation, and were far less fearful of disciplinary procedures. Disciplinary procedures feed into the perceived risks part of the environment stage. Disciplinary procedures are in tension with reputation management, because formal and punishing responses to security breaches necessarily admit their existence.

The final factor of the security administration stage is reputation management (model component 8). If breaches or even foiled attacks become known, it is predicted that users will behave more securely (Adams & Sasse, 1999; Adams et al., 1997, and see perceived risks in the environment stage). However, there is currently a tendency to cover up security incidents rather than publicise them. There are two main reasons for this:

1. it could become negative publicity and harm the organisation's reputation, and so negatively effect sales or production; and
2. it can spread knowledge of vulnerabilities that can then be exploited by greater numbers of attackers, and so can increase the organisation's level of risk.

Because the feedback from security is delayed and unrewarding (i.e. nothing happens), and the feedback from production is so immediate and reinforcing (see Brostoff & Sasse, 2001), it is difficult to justify an intervention that can so clearly harm production. Until it can be shown that publicising security breaches protects more business than it spoils, then it is unlikely that breaches will be publicised. However, this cannot be shown until breaches *are* publicised.

11.2.3 Environment stage

The *environment stage* represents the social, work, and perceptual context in which password design and use occurs.

Perceived resource sensitivity (model component 9) relates to users' perceptions of the sensitivity of the information or other resources being protected by the security system. Security procedures protecting information perceived of as sensitive were complied with (for example by designing strong passwords) and security procedures protecting unimportant information were subverted (Adams, 1996; Adams & Sasse, 1999). This finding applies to information only, though attackers may wish to access computing facilities rather than the information associated with them. The finding is therefore extended in the model to include more general resources, in the assumption that its psychological validity remains intact.

Perceived risks (model component 10) are a function of the visibility of attacks against the system, and sanctions enacted by the organisation against the user. It has previously been found that users who believed that the system (and the information it protected) were vulnerable make more effort to implement strong password design (Adams, 1996; Adams & Sasse, 1999). However, if the threat against the system was perceived to be small, strong password design was believed to be unnecessary and laborious, and so was not adopted. The perceived risks were larger if system breaches

were public knowledge and if physical security was perceived to be low. In contrast, the existence of highly visible and tight security (and little publicising of security breaches) gave the perception of safety. In addition, some users feared punishment from the organisation if a breach should occur, and took steps to protect themselves by complying with security policies - this was observed in Study 8 focus groups (sections 10.4.2 and 10.5).

The next factor is *support* (model component 11). This consists of three facets, which together reduce the negative consequences of experiencing a password problem, either by reducing the likelihood of a problem (tools) or by maintaining productivity after a problem has occurred (colleagues, help desks). They mediate the motivation for and so the effort put into managing and designing passwords. For example, Study 7 focus group members felt it would be a better use of their time to call the helpdesk than systematically manage their passwords (section 10.4.1, and 10.5). In other cases, it might not matter if you forget your own password if a colleague offers his to carry out your work with. Finally, it is less necessary to pay attention to your passwords if you have them recorded somewhere in a password safe, diary, e.g. a prompt.

Working practices (model component 12) corresponds to the compatibility between security systems and the amount of information sharing between colleagues. Adams et al. (1997) found that users who commonly had to share information as part of their work subverted security systems that enforced separate passwords. The security systems were seen as obstacles to task completion. Users subverted incompatible security systems by altering the design of their passwords so as to make them more usable (less costly) with the consequence that they were weaker as passwords. However, if security systems were compatible with work practices then users would conform to the principles of strong password design.

Business cycles (model component 13) reflects longer-term temporal components of work which may interact with password use. For example, some systems may only be accessed at the end of the financial year. Furthermore regular patterns of system use may be interrupted; for example through vacation or business trips, or other kinds of leave. All these examples could combine with password expiry policies to create helpdesk use. However, even the interruptions have predictable temporal features. None of these regularities appear to have been taken into account in the design of password systems.

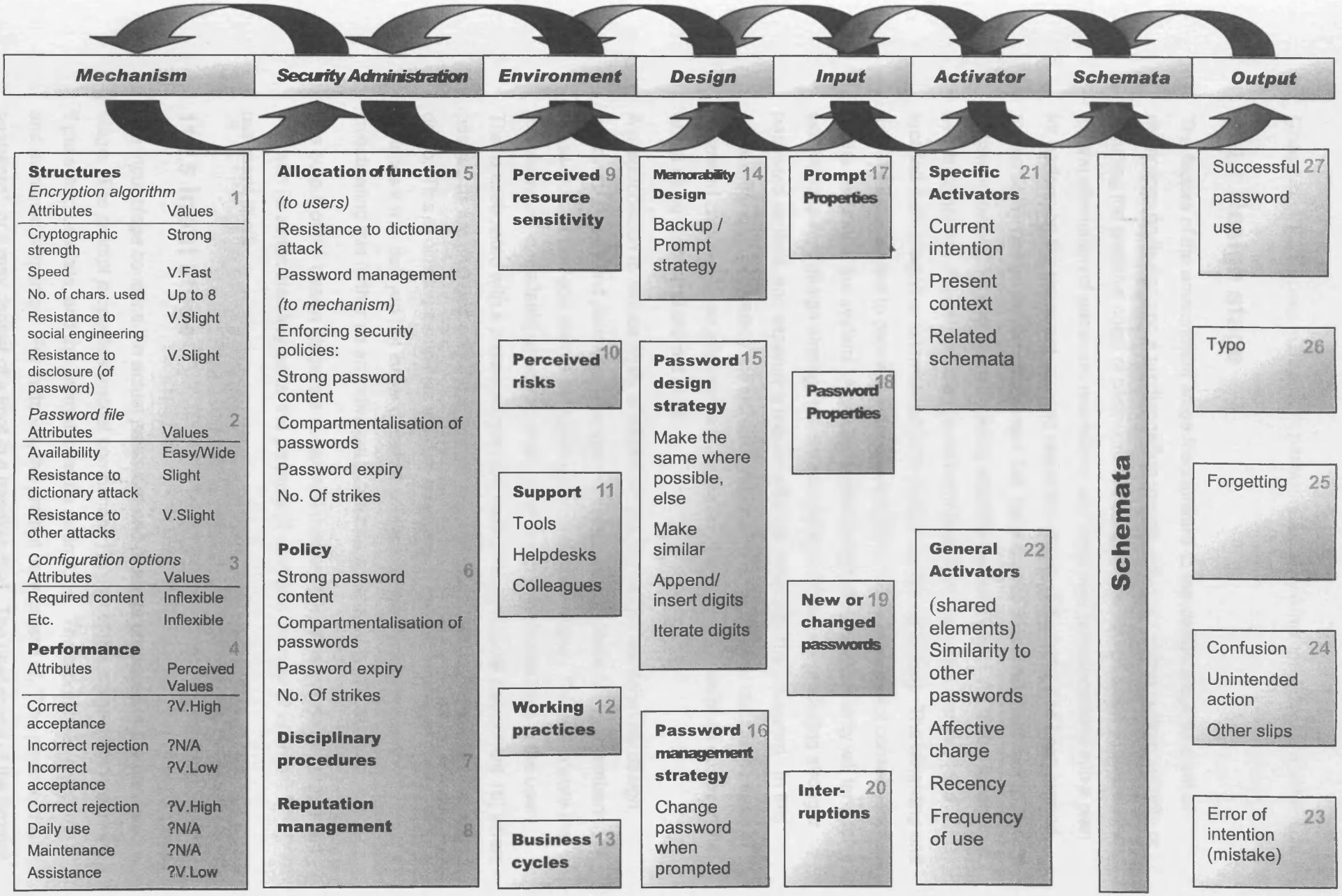


Figure 36-Small Scale model of the corporate password problem

11.2.4 Design stage

The factors of the *environment stage* feed forward to the *design stage* as a set of restrictions on design and a predisposition toward either upholding system security or reducing the personal costs of password ownership. The *design stage* conceptualises the transformation of password restrictions and behaviour predispositions into a plan for performing the password use and ownership tasks. This factor has been termed *memorability design* (model component 14), because the most important task in password ownership and use is knowing what the password is at the time it is required by the computer. Passwords may be stored internally in the users memory (and recalled during login) or externalised (and copied out during logins). The weighting and detail of these routes to password storage and retrieval will have direct consequences on the security of the system. A predisposition to uphold system security will tend to weight *password design strategy* (model component 15) toward producing stronger password content, and expending mental effort to remember the password. In this context the *prompt strategy* will increase the users mental effort to reduce the ease of password cracking by an attacker, e.g. by making the physical prompt an obscure one; or not using a physical prompt at all.

A predisposition to reduce costs of password use will weight *memorability design*, *password design* and *password management strategies* towards easy remembering of passwords, or perhaps even a complete reliance upon prompts. This will ensure that the password is available when required, but at much less mental cost to the user. This is associated with a *password management strategy* (model component 16) where passwords are changed only if the system requires them to be, and requires the user to do so. This avoids the monolithic effort of changing all the passwords at once, but replaces it with delayed but larger negative consequences. It may be worth investigating this further as an individual difference in delayed gratification.

The output of the *design stage* is a completed *memorability design* (model component 14), with its associated key parts, the prompts (if any) and the new or changed password itself.

11.2.5 Input stage

The *input stage* contains the actual password and prompts produced in the previous stage. The *prompt properties* (model component 17) part of this stage is only relevant if password use fails, and the prompt must be referred to. The prompt has structural and semantic properties. For instance, the prompt may consist of, "The password is: password", or it may consist of a knot in a handkerchief. The usefulness of the prompt

rests in the context of its use, and its activation of related schemata (increasing the activation of the password itself), which combine to form the residual risk it will not work. Many users will set this residual risk to zero by making the prompt identical to the password itself.

Password properties (model component 18) refers to the number of characters/complexity of the password, the meaning of the password if it has one, its personal significance, etc. One of the most important properties is whether the password is *new or changed* (model component 19).

All these properties are processed in the next stage, along with *interruptions* (model component 20). These have been identified as a significant source of human error (see 3.3.5). There are many compelling and frequent sources of interruption in large organisations-telephones, colleagues, e-mail alerts, etc. They can occur at any point during password construction or use.

11.2.6 Activator stage

This stage draws out some of the properties outlined in previous stage that have been previously identified as of psychological interest in human performance. It is where the findings from this thesis tie into an existing psychological framework -schema based control as illustrated in section 3.3.5. The properties in the previous stage are converted into *activation*, which is used in the *schema stage*. Each password schema gains activation through *general activators* (model component 22) and *specific activators* (model component 21). For example the newness of a password identified in the previous stages contributes activation through the general activators *recency* and *frequency of use*; the password's personal significance is converted into activation through the general activator *affective charge*, and so on. Specific activators have less to do with the password properties, and more to do the context and intention of the user.

11.2.7 Schema stage

Passwords are represented as schemata in the user's mind. The password schemata compete to gain enough activation to be triggered as behaviour at the keyboard. The password schemata gain activation by the stimulation of *general* and *specific activators* in the previous stage. It is possible for the wrong password schema to be triggered. The result of the competition to be triggered is dealt with in the next stage.

11.2.8 Output stage

Depending on which schema was triggered in the previous stage, a different outcome may result. The correct password may be used correctly, or there may be several different kinds of errors committed (see section 3.3.5).

11.3 Benefits

This model explains novel findings about password system performance using existing psychological theories. Password expiry (see section 7.3.2) and password confusion (see section 8.3.1) have been found to be large causes of password problems, and more significant than the password literature suggests. Their magnitude in this PhD is explained by the model as the result of competition between rival schemata. The model also explains why strong passwords can be difficult to remember in lab studies but comparatively memorable in real world studies (see section 4.1.2) - *activation* gained through higher *frequency* and *recency of use* of the passwords in the real world studies make them easier to recall.

The model ties together attributes of the password mechanism itself, with the administrative response to these attributes, to the users' adaptations to the administrative response, and explains the resulting performance. The model can be used to assess the relevance of proposed interventions by linking their modes of action to the known causes of password system performance.

11.4 Summary & Conclusions

A model is presented that summarises the previous findings of this thesis about the causes of password system performance in BT and UCL, and by so doing bring us closer to answering Research Question C. The model builds upon the models of Adams, Sasse & Lunt (1997), Adams and Sasse (1999), and Reason (1990).

The model shows that the properties of password mechanisms lead to a security administration response to compensate for these properties by applying password security policies, and allocation of security tasks to users rather than weak password mechanisms. This results in users making particular design choices about their passwords and password backup strategies. These all conspire with the schema based control of human performance to produce password problems.

11.5 Contributions

11.5.1 Substantive

A new model is presented that ties problems with password system performance to existing psychological explanations and explains novel findings from previous chapters. This model can be used to as a starting point for further research that can be used to create generalised theories of password system performance. The model may also be used as a starting place for reasoning about potential interventions (see section 9.4).

Chapter 12

Conclusions

The purpose of this chapter is to present and reflect upon the conclusions of the research conducted in this thesis, and to propose future research topics and directions. To date, little empirical work has been carried out in the area of the usability of password authentication. This thesis has undertaken exploratory research in both the field and the lab, and has presented a body of empirical evidence that represents a contribution to our understanding of the performance of password authentication in large organisations. This chapter will start with a restatement of the research problem. There will then be a section summarising the paths that organisations choosing an authentication mechanism can take as a result of this PhD. This will be followed by an overview of the contributions made in the thesis in pursuing its four research questions:

Research Question A)	Substantive contributions presenting quantitative measures of password system performance, and the effect of different security policies upon it.
Research Question B)	Methodological contributions made during the research that enabled the measurements above.
Research Question C)	Substantive contributions of models of password system performance and organisational security system performance.
Research Question D)	A substantive contribution of the proposing of interventions for improving password system performance based on a new theoretical appreciation, with the start of validation for some of these interventions.

Then there will be a section critically examining the research contained in the thesis. The chapter will close with proposals for further work.

12.1 The problem restated

The research on which this thesis is based was sponsored to address the pressing business problem of unacceptable numbers of password related requests at internal-facing computer helpdesks in BT. BT originally specified the research problem as users choosing unmemorable password content, and forgetting it. As research progressed, it became clear that the problem of password helpdesk use was deeper, and that treating it as an issue of memorable password content would result in a poor solution. Different questions needed to be asked.

The research became "improving password system effectiveness" in large organisations such as BT and UCL, and was based on the Long and Dowell applied science conception of Human Computer Interaction (e.g. Dowell & Long, 1998 see

section 2.2.1). Restated in the terms of this conception, the PhD aims to generate knowledge to help:

1. *Reduce the costs* of password authentication systems to stakeholders, and to *improve the quality* with which password systems carry out the authentication task, and to
2. *Start model building* that may eventually enable the findings of this research to be *generalised and validated,* so that they may be re-used for other classes and scales of work system in the domain of information security.

These aims were argued to require answers to the following research questions:

a) What is the performance of password systems?
b) How can password system performance be measured?
c) What causes password system performance to be good or bad?, and ultimately
d) What interventions could improve the performance?

12.2 How to choose an authentication mechanism

Organisations wishing to implement authentication mechanisms can take two paths as a result of this thesis

1. Focus on the device - a classic HCI perspective, or
2. Take the wider view embodied in the new *Elevation* model of Chapter 12.

For the first path, the organisation should predict the pattern of device use. High-frequency use makes the traditional password mechanism appropriate. Strong password content can be used, particularly with extended enrolment that increases the amount of practice with and depth of thought about the password. Mnemonic techniques may be possible. With low-frequency use, enhanced password mechanisms will give better availability while maintaining confidentiality and integrity. Alternative paradigms such as tokens or biometrics could be considered.

For the second path, the organisation should consider how authentication fits into the culture and working practices that are local to the new site of authentication. Tying in authentication to existing organisational infrastructures, such as personnel department should also be considered. The existing authentication load on the intended users, the burden imposed by password recovery and distribution systems, and the interaction of existing password mechanisms with the incumbent mechanism must also be considered. Additional support may be provided to users, or security functions allocated to infrastructure rather than users. Motivational techniques could also be considered. By taking a wider view, it is possible to avoid or ameliorate additional risks

that are brought by installing a new mechanism. There is a better chance that overall system security can be improved, rather than local improvements that produce impacts in other parts of the system.

12.3 Contributions of the thesis

12.3.1 Research Question C: Allocation of function

One of the primary contributions of this thesis is the documenting of harm done to security by the seemingly overzealous application of standard and respected computer security techniques. Evidence has been collected that the following increase the use of helpdesks:

1. Password expiry (in Chapter 7 it was the single most frequently reported technical and organisational cause of password resets in the data; and greater proportions of problems occurred just after password expiry as frequency of expiry increases; in Chapter 8, problem logins were on average closer in time to password changing than successful logins).
2. A bar on writing down passwords (in Chapter 7 evidence was presented that it increases forgetting problems).
3. Minimum password strength requirements (in Chapter 7 evidence was presented that this policy requires greater frequency of password use for passwords to become automatically recalled; in Chapter 8 a trend was detected for greater numbers of resets to be associated with added password complexity).
4. Compartmentalisation of passwords between systems (i.e. different password required on different systems; in Chapter 8 using the wrong password was found to be the second biggest login problem in the data) and historical compartmentalisation within systems (i.e. different passwords are required after each expiry; in Chapter 8 - expired passwords being used in place of current passwords was the largest single source of login problems recorded).
5. Three strikes policies (in Chapter 8 data it was found that there were a large proportion of users requiring more than three attempts to login when there are no three strikes policies).

While it may be privately acknowledged that these risks exist - they have not before been demonstrated emprically. This demonstration also forms part of the answer to one of our research problems - what is causing poor password system performance. Complex security functions in the organisation were allocated to users rather than to

technology or infrastructure (particularly in the case of creating strong passwords to resist dictionary attack). Security technologies and techniques were applied where they were inappropriate - passwords and associated password security policies enforced on systems that are infrequently used. This is a novel position in a discourse about password performance that tends to blame the user for not knowing how to remember or choose passwords, or for not trying hard enough.

12.3.2 Research Question A: Password system performance

On the route to the contribution above, it was necessary to measure and document the performance of password systems:

1. 11 errors in password use were recorded by a total of 3 (out of 6) participants, a 3% login error rate (Chapter 6), with 7 of 11 of these errors being typos. No helpdesk uses were recorded.
2. 37% of *login failures* observed (in Study 4) were due to entering the most recently expired password, with another 15% entering a different password than the one required - making approximately 50% of failed logins due to confusion between passwords.
3. Approximately 30% of login failures observed (in Study 4) were due to other slips, lapses, and mistakes by the user.
4. Approximately 60% of causes of password *helpdesk use* (in Studies 2 and 3) are forgetting, followed by approximately 30% caused by technical or organisational problems, with about 10% caused by users' slips, lapses, and mistakes (combined n=174, Chapter 7).
5. Types of errors are different according to how frequently the password is used. For heavily used passwords (>30 times/month) the proportion of helpdesk use caused by forgetting drops to about 20%, while technical/organisational problems jump to 70%. For lightly used passwords (< 1 time a month), technical/organisational problems account for only 10% of helpdesk uses, with the remainder caused by forgetting (Chapter 7, Figure 17)
6. Three separate password authentication systems at BT had password helpdesk use rates of 3.6, 1.2, 0.9 resets per user per year (Chapter 8). Assuming similar patterns of use the UCL-CS based password system in Study 5 would have 0.7 resets per user per year (Chapter 8).

Password system performance data of this kind has not previously been available to researchers and practictioners, and so forms a substantive contribution. These findings can be used to assist practitioners in planning of authentication systems and support services such as help desks, and can aid researchers in the planning of studies - they will have a better idea of what password system performance to expect (findings 1, 6 above), under what conditions (finding 5 above), and the relative proportions of different kinds of problems (findings 2, 3, 4). Moreover, these findings act as a starting point from which to make comparisons. If practitioners or researchers measure password systems as having substantially different performance then this indicates important operational and research issues that should be given attention.

12.3.3 Research Question B: Measuring password system performance

In the course of collecting the data which led to the contributions above, different methodologies were investigated. These investigations resulted in knowledge about benefits and drawbacks of these methodologies in this field of research, which together form a methodological contribution.

Measurement is the cornerstone of research, without it there is no data to analyse. The methodological findings of this thesis will enable researchers to collect data about passwords more easily, and to collect data of a better quality that allows deeper insights.

System logs

Chapter 6 found that standard UNIX password system logs are not appropriate for tracking the password use of individual users. These logs give ambiguous data that allow little insight into the psychological causes of password problems, and so are not appropriate for password system research (section 6.4.2).

If it is not possible to alter the logging system, Chapter 8 shows that by shifting focus to a different kind of system log it is possible to gain valuable insights at a wider scale. It is possible to estimate password system performance using *helpdesk logs*, and to identify problem systems without having to alter security policy, or change the way authentication mechanisms operate or record login events - which may breach security policy. This is useful in environments where the researcher has little control.

If however it is possible to make changes then authentication mechanism logs become much more useful. Chapter 4 suggests the measurement of intervals between enrolment and first login generates data that can help assess the ecological validity of laboratory studies, which is very difficult to do at present.

Study 5 (Chapter 8) shows that recording every login success and failure produces data that allows researchers to start reasoning about the setting of password policies such as *three strikes*. Chapter 8 also illustrates an authentication logging system specifically constructed to support password research, which allowed direct inspection of password error types and their real world frequencies through comparing actual passwords with the contents of login attempts. This allowed a more ecologically valid and informative analysis of login problems than had previously been available from the prevailing methodology - lab studies of password memorability. The utility of this methodology was shown by finding an unexpected result - that password confusion is a very frequent cause of password problems (section 8.3.1). This result could have profound implications for research about password system performance - the literature has focussed on password forgetting, with an effort towards making more memorable passwords. However, password confusion must be tackled by reducing the power of distractor passwords, by reducing their number (fewer to be distracted by) and/or reducing their memorablity (each one is less distracting) - a strategy opposite to the one currently being pursued by researchers and practitioners. The novel system logging method initiated in this thesis may therefore result in changing the direction of research in password system performance.

The logging system above collects password content data which causes a *vulnerability* - the data can be used by attackers to impersonate authorised users. It is likely that there will be few circumstances when such logging systems are deployed. Chapter 8 has demonstrated logs that recorded less sensitive information such as password enrolment, use intervals, login failures and helpdesk uses would still improve our understanding of password system performance and its parameters, and aid in the design of and extrapolation from lab studies.

Diaries

This PhD formed the first trial for diaries in password research.

- Overall there was a good correspondence between diary reports and system logs
- Diarists over-reported password use by an average of 2 uses per person per day in the study
- The data did not show participants altering their behaviour to avoid making diary entries

Some difficulties were experienced, and recommendations were made for their study, amelioration and avoidance (section 6.6.1).

These findings about diaries will enable researchers to have more confidence in the accuracy of results in password diary studies. This is important because password diaries offer a uniquely accurate view across the many different password systems a user is part of that is not practicable using other techniques.

With the recommendations for enhanced diaries researchers will be able to plan and implement studies which can collect better quality data with a larger sample of users. This will allow greater confidence in the accuracy of results (for example by checking timestamps against diary entries), and (by using the installed base of PDAs in large commercial organisations) will make it more easy to achieve sample sizes that enable conclusions to be made with known levels of certainty about the populations from which the samples were drawn.

12.3.4 Research Question D: Interventions for improved password system performance

It was possible to propose interventions for improving password system performance based on a new theoretical appreciation of it. Validation was begun for a subset of these interventions. This subset was selected because of its focus on "redesigning" users. The interventions were combined into a package that consisted of training embedded in a paper-based tool to support the following:

- Password selection for strength and memorability of content
- Selection of password hints
- Manual password synchronisation
- Writing down passwords

The tool was presented to some of its potential users at BT in two series of focus groups. Despite describing many of the problems that the tool was designed to solve, and in several cases employing some of the techniques proposed in the tool, the relatively powerful users from the research arm of the organisation made many negative comments about it. Though some of these were practical critiques, the chief criticisms were that the tool was the wrong approach-changes should be made to the authentication systems not people who have to use them. These users predicted that they would prefer to use the helpdesk than make the effort of using the tool, and they would avoid the tool even under significant risk of punishment. The less powerful users from the operational branch of the company made far fewer criticisms, which could be addressed by moving the tool from paper to electronic media. It would appear then that if used the tool would work, but that the tool would not be used by a significant

proportion of its intended user population, who would object to it on a political or cultural level.

12.4 Critical review of own work

This section will present some criticisms of the research presented in this thesis.

12.4.1 Other Human Factors based models of Security

There are four models of human factors in security that have not been considered during this research and which may be viewed as competitors or a serious omission:

1. Spruit (1998),
2. McCauley-Bell et al. (2000),
3. McCauley-Bell (2002), and
4. GAO-98-68 *Information Security Management* (GAO, 1998).

None of these models were published when the current research was started, all of them emerged while it was being conducted.

12.4.2 Social and cultural factors

In Chapter 10 it was found that relevant and otherwise promising interventions would not improve password system performance due to management, cultural and social obstacles (section 10.5). Social and cultural issues have not otherwise been studied in this thesis. Section 5.1.1 identified this as an aspect of the HCI discipline's necessarily limited worldview - industrial politics and sociology does not come under the domain knowledge of HCI, and so it is not treated in the model presented in Chapter 11. However, it is predicted that some of the interventions proposed in this thesis will avoid cultural/ social objections similar to those observed in users in Study 8, as the interventions require changes to infrastructure - exactly the changes proposed by the objecting users. While the wider context of interventions has not been considered throughout this work, at least some of the interventions proposed in this thesis remain potentially effective from the user's point of view. However, there are secondary and tertiary stakeholders, and their political and cultural stand will alter the effectiveness of any intervention. Managerial, social and cultural factors must be considered in future research and attempts to improve system in practitice. A model with a wider scope than Chapter 11's is needed.

12.4.3 Unexploited indices of usability and performance

This thesis has focused on performance metrics which are various kinds of *errors* in the use of passwords, and to a much lesser extent has assessed the *time taken to complete tasks*. However, the discipline of HCI recognises many other performance metrics including learnability, cognitive workload, physiological stress (galvanic skin resistance, blood volume pulse, etc.), users' satisfaction, and fun. By focusing on a limited set of metrics, it is possible to arrive at a view of the problem which is less full than could be achieved with the wider set. For example, analysis of the merits of rival authentication technologies might lead to quite different conclusions if user experience type metrics were considered. If it were found that some authentication mechanisms lead people to want to use them for the sheer pleasure of it then there could be significant implications for solving the password problem. However, most of the implications and contributions of this thesis would still stand: installing a more intrinsically pleasurable authentication technology would likely be as costly as fixing the existing password mechanisms due to the extensive changes required to infrastructure.

12.4.4 Psychology of risk and motivation

While much of this thesis is both directly and indirectly to do with the management of risk, there has been no direct treatment in this thesis of the individual's psychology of risk. A full account of the human factors of computer security cannot avoid this topic, which is a component of the wider question of what motivates users to behave securely (Weirich & Sasse, 2001). Given ideally usable security systems, users must still have the motivation to use them (Weirich & Sasse, 2001), and if they do not use them there will be no security. The previous statement shows how intimately related usability and motivation are to security. However, people cannot be motivated to use ideally usable security systems that do not exist. While not directly addressing the issue of motivation, this thesis has nevertheless pointed out its importance. In Chapter 10 interventions were assessed and found to offer the promise of improved password system performance. However, at the same time it was demonstrated that these interventions would not succeed because a significant group of stakeholders would not be motivated to use them in part because they felt less personally at risk. Further research may focus on manipulating stakeholders perceptions of risk to provide interventions to motivate them to behave more securely; though at the probable expense of an increase in stakeholder costs.

12.4.5 Legal aspects, national and international standards

This thesis has not considered the legal aspects of password authentication, nor the role that national and international standards and certification play in security architecture. It may be the case that companies are bound to enforce password policies that this thesis has found to be counter-productive because laws or standards require them to. However, this would still not invalidate the contributions made by this thesis, which could be used to inform and reform counter-productive laws, guidance and certification.

12.4.6 Ethical and legal issues

Participants in Studies 4 and 5 were unaware that their coursework system account details or their system use trails would be subject to examination for the purposes of this research. Their consent was expressly not sought, as it was predicted that participants might alter their behaviour, particularly in the choice of passwords. This was seen as a substantial risk, because the validity of these studies hinged on measurements being taken of real users performing real work in real circumstances.

There are three related issues about these participants' knowledge and consent. The first is that participants passwords were available to this researcher, without participants being warned that this would be the case. Not knowing that the passwords were available to this researcher, the participants may have selected their password to be the same as used on other systems, thus unwittingly revealing information that the researcher could use for identity theft. The second issue is that participants' consent was not sought for the examination of their passwords for research purposes. The third issue is that participants' consent was not sought for examination of their system use trails for research purposes.

UCL's Information Systems conditions of use policy states that all student's data may be inspected for security purposes, and that systems and their contents may be examined to see if they are secure (UCL, 2004). It is common practice for systems administrators to examine passwords in very similar ways to those used here-for example in password strength audits. The researcher had already been acting as the system administrator for the coursework system, and so was already trusted with the participants' account details, and could be reasonably expected by participants to perform password strength audits. Moreover, the researcher was routinely asked by the participants to remind them of their forgotten passwords (rather than asking to be supplied with new ones), implying a belief that the researcher already knew the passwords. This addresses the first issue.

Users of the coursework system may have expected that sensitive information like their passwords would be used only for the purposes of administering the system - for example supplying users with forgotten account details, or performing security audits. These audits are not designed to benefit the user, though they may do inadvertently by preventing their account from being the point at which security is breached. Rather, these audits are designed to benefit the organisation by improving its knowledge of its users behaviour, and enabling it to improve its password systems' performance. Similarly, this research is designed to improve knowledge of users' behaviours to improve password system performance, though the inadvertent benefit to users whose account data has been examined is likely to be less. In both cases the sensitive information is used for very similar purposes. Participants had already agreed to uses of their data very similar to the additional uses to which it was put. This partially addresses the second and third issues.

The researcher now knows that UCL's data protection policies required that informed consent should have been sought for research use of the participants' data. This policy was not knowingly breached. Future research must seek this consent from participants. The standard technique for obtaining consent is to ask users to agree to the conditions of use / privacy policy during registration. This would not be feasible on a coursework system where use of the system is compulsory. An opt-in facility for research should be provided instead, with an explanation of the purpose and use of data collection.

12.4.7 Establishing causality

Throughout this thesis assertions have been made about the effect of security policies, but the evidence presented for these assertions has not met the traditional level of proof required by HCI and its parent disciplines: demonstration through the controlled manipulations of a laboratory experiment. The evidence presented has been descriptive and correlational, and furthermore has in some cases been based on user reports rather than direct observation, and in some cases reports of what users might/would do rather than what they have done. Despite falling short of science's gold standard of proof, it is the author's claim that this evidence is persuasive and useful. It represents a significant contribution to the amount and detail of evidence previously available. It identifies issues of importance in the real-world that could be investigated under more controlled and laboratory conditions, whereas starting with a programme of laboratory studies could not guarantee the same level of real-world relevance. Being applied and field based research it by definition cannot meet the exacting standards of pure and laboratory research. It was not possible to manipulate conditions in the field:

in many cases this would have required amending security policies, either to make them more lax-which BT's security departments would be unwilling to do-or by making them harsher, and thereby reducing productivity which would not be ethical and would not be acceptable to line managers.

12.4.8 Sampling issues

Many of the contributions of this thesis were based on survey data collected at BT. It was unfortunately not possible to follow standard sampling procedures in the collection of this data, as it was not possible to get the relevant information about the composition and whereabouts of BT of users. Consequently, it is not possible to confidently assess the generality of the findings from the survey respondents to the rest of the organisation. However, sociodemographic information was collected in the surveys, which shows that 43% of the respondents were management grade, and a further 18% were senior management. The author presumes that even if the sample was not representative, this is a sample of users that is powerful within the organisation, and whose password problems the organisation would want to address. They are therefore a valid and useful sample to study.

12.4.9 Analysis of residual risks

This thesis does not address whether risk of loss of availability due to the organisation's password practices outweighs the changed risks to confidentiality and integrity of relaxing these practices. It could not, because this would involve deep risk analysis, which would not be practical in a PhD and would be too commercially sensitive to be released to the researcher had it been done by the organisation.

This thesis also does not address the cost of interventions, or the effect of interventions on password costs. It would be difficult to make reasonable generalisations. The most accurate costings would involve trialling the interventions, which would not be feasible in a PhD. Less accurate costings would involve the recruitment of specialists within BT to make estimates-however the author has not been able to access them. Specialists from outside the organisation may not give as accurate estimates, but would be easier to contact. In the author's opinion however, determining the size of intervention effects must rely on further empirical work rather than specialists' estimations.

12.5 Further directions

12.5.1 Password design for recall with competition

This thesis has shown the large extent to which confusions and interference between passwords can effect password system performance. However, studies of password design have focused almost exclusively on their memorability in isolation from other passwords. Their resistance to confusion has been completely ignored. The next stage in research about password design should address this, as it is implicated in two of the causes of poor password system performance: the proliferation of passwords, and their forced expiry. This will likely require experimental designs where participants recall several passwords concurrently instead of just one, or are forced to learn and forget passwords in quick succession. By pursuing this route it may be possible to generate guidelines for the optimum number of passwords to be used at any one time, and algorithms for their selection.

12.5.2 Security and system administrators

Further research is required with systems administrators and security personnel about their selection of security technologies and their configurations. In particular, the roles of national and international standards should be explored, and attitudes and reasoning about the deployment of relatively new technologies such as *bcrypt* passwords. Are there any issues which prevent otherwise beneficial technologies being employed? How do systems administrators decide that a technology has become mature or safe enough to use? Anecdotal evidence suggests that otherwise desirable changes are not adopted in case the changes are used to target scapegoating activities at the systems administrators.

12.5.3 The costs & effects of interventions

Further work is needed to make it possible to predict costs of interventions, both in terms of purchase/implementation costs, and also to the three sources of password costs. This will require more validation activities to be carried out on the interventions proposed in as real circumstances as possible. Laboratory conditions may have to suffice because of the difficulty of performing these interventions in a company without high-level management support. Further work should include evaluating the effects of these interventions on the categories described by the two new models.

12.5.4 Sampling and surveys

If possible, work should be done to improve the surveys conducted in this thesis. The surveys could be rerun using better sampling. This will require much more aid from the hostBT than was available during the time of this PhD. Information regarding the numbers, types, and locations of different kinds of stakeholders within the hostBT will have to be made available to researchers. Moreover, researchers will have to be given enough access to these stakeholders to conduct the survey, and will then require suitable inducements to answer the survey questions-all of which may be difficult to achieve for stakeholders at the extremes of corporate class.

All of the participants in the diary studies reported in this thesis described the burdensome nature of making entries for every use of a password. The use of advanced sampling techniques in conjunction with electronic password diaries implemented on personal digital assistants (PDAs) could greatly reduce this burden, by recording a much smaller sample of each individual's password events and building a bigger picture using statistical techniques. Techniques such as these would benefit from the probably large installed base of PDAs in organisations that are large enough to face particularly difficult password problems.

Feeding into the enhanced surveys should be studies about the performance of different types of question for uncovering the number of passwords owned and used by stakeholders. In particular, the role and effectiveness of prompts about the different systems that passwords might be attached to.

12.5.5 A wider focus

This thesis has been explicitly placed in an Ergonomics/HCI viewpoint - focusing on issues such as allocation of function, time and errors, and purposefully ignoring cultural and social factors that have been the preserve of the Information Systems discipline viewpoint (section 4.1). Despite this, the work reported in this thesis has inexorably lead to the conclusion that motivation, managerial, social and cultural factors must be addressed (Chapter 10). Research is already beginning on the first of these issues (e.g. Weirich & Sasse, 2001), and work on the latter issues is ongoing in the Information Systems community (reviewed in Dhillon & Backhouse, 2001). However, there may be difficulty bridging the theoretical chasm between the disciplines so that the sociological work can be operationalised in ways that Computer Scientists and Ergonomists can understand. Attempts should be made in this area, and a preliminary attempt is made here.

This section presents a model describing security and human error on a large-scale, showing at a high level what an organisation needs to do to be secure. It is a view that extends to parts and concerns of the organisation beyond the security system. By taking this wide view the model allows reasoning about the causes of password problems, such as the findings described in Chapters 9 and 11, and security problems more generally.

The first part of this section describes the model in overview, and then in detail. The benefits of the model are then briefly discussed. A start on validation of the model is then made, by peer review. A revised version of the model is then presented based upon the comments received during the peer review procedure.

Overview of model

Instead of modelling an individual information system, the model describes security from the perspective of a (commercial) organisation (though the model should apply equally to UCL as to BT). For security breaches to occur, *active failures* must combine with *latent failures* (Figure 37), which are built-in weaknesses in the organisation that predispose it towards *active failures* and disaster. *Latent failures* can be found at all levels of the organisation, which are modelled as (Figure 37): *decision makers* (strategy makers, the board/ executives), *line management* (departments that enact particular parts of strategy), *preconditions* (prerequisites lying between *line management* and *productive activities*), *productive activities* (actions by users), and *defences* (to protect the organisation from attack).

Violations-the intentional breaking of security policies are categorised as being part of the group of *not-secure acts*. The other *not-secure acts* are due to human error and comprise: *slips* and *lapses* (unintended actions), and *mistakes* (intended actions with unintended consequences). *Not-secure acts* are also known as *active failures*-the easily observable actions made by users that tend to compromise security.

The most benefit is to be gained by tackling *latent* rather than *active failures*. In the same way that it is better to drain the swamp instead of killing mosquitoes one by one, it is better to rid the organisation of *latent failures*, which are the breeding grounds for *active failures* (Reason, 1990).

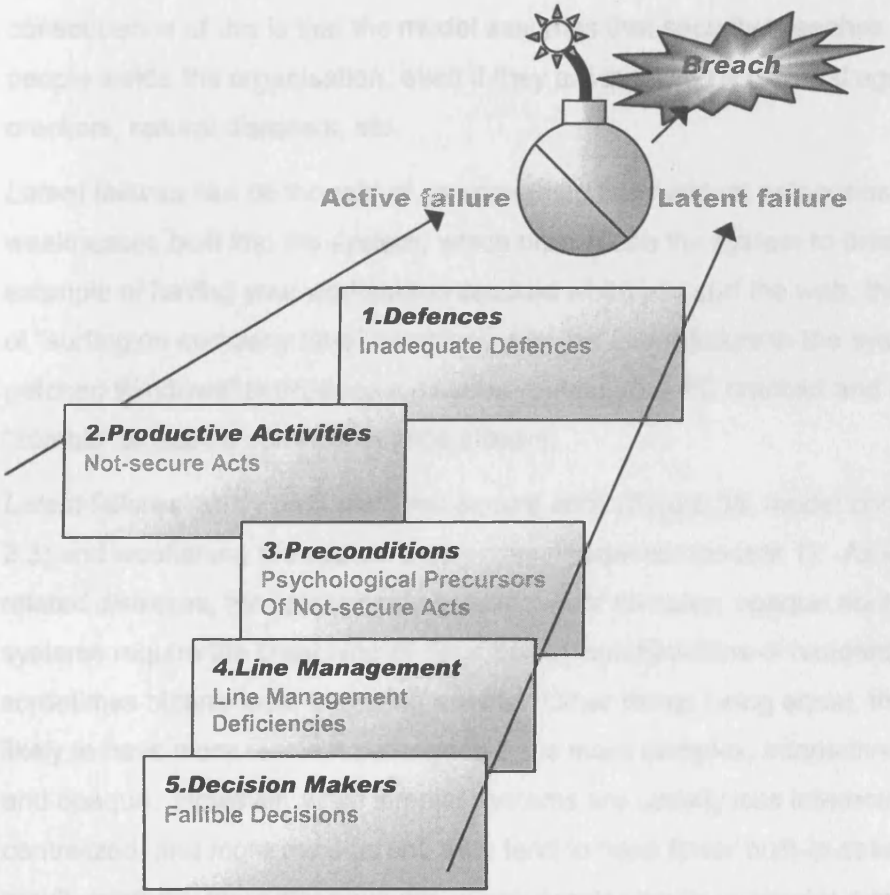


Figure 37 - Overview of large scale model

Detail of model

Reason's GEMS (see section 3.3.5), combined with *Basic Elements of Production,* is a comprehensive model for ensuring safety in organisations. The research reported in this thesis demonstrates that the model can be applied to reasoning about security design for organisations. It is a model that is informed by a detailed understanding of both individual and organisational characteristics that direct user behaviour. The model (Figure 38) is described in two sections: firstly, failures at the system (organisational) level - which are the more serious of the two kinds of failures, and secondly, failures at the individual (user) level.

Latent failures, and Basic Elements of Production

For a security breach (disaster, or accident) to occur, *not-secure acts* must combine with *latent failures* and or unusual environmental conditions (see Figure 37). On their own, *not-secure acts* are necessary, but not sufficient to cause system disasters. A

consequence of this is that the model assumes that security breaches are due to people inside the organisation, even if they are initiated by external agencies such as crackers, natural disasters, etc.

Latent failures can be thought of as something like resident pathogens. They are weaknesses built into the system, which predispose the system to disasters. Using an example of having your workstation cracked while you surf the web, the *not-secure act* of "surfing on company time" combines with the *latent failure* in the system of "un-patched Windows" to produce a disaster-getting your PC cracked and used as a "zombie" to launch denial of service attacks.

Latent failures act by promoting *not-secure acts* (Figure 38, model components 2.1 to 2.3) and weakening the system's *defences* (model component 1). As with pathogen-related diseases, the catastrophic breakdown of complex, opaque socio-technical systems require the breaching of *defences* by combinations of resident pathogens and sometimes bizarre local triggering events. Other things being equal, the system is likely to have more resident pathogens if it is more complex, interactive, tightly coupled, and opaque. However, while simpler systems are usually less interactive, less centralized, and more transparent, they tend to have fewer built-in *defences.* As a result, relatively few pathogens can wreak greater havoc in simpler systems than in more advanced ones.

Having the concept of a *latent failure* as something that predisposes a *system* to security breaches necessitates some definition of one. A system or organisation is described in the model in the following way (Figure 38):

Decision-makers (model component 5) direct the organisation at a strategic level (CEOs, VPs, etc.), and

Line management (model component 4) implement the strategies of decision makers. This implementation creates the

Preconditions (model components: 3.1 usable and secure equipment of the right kind, 3.2 a skilled and motivated workforce, 3.3 appropriate attitudes and motivators, 3.7 appropriate work schedules, 3.5 appropriate system administration and maintenance programs, 3.6 environmental conditions, 3.4 codes of practice and policies, etc.) for

Productive activities (model component 2) which are the activities the organisation and its users carry out to attain its payoff, e.g. telephone service provision, degree level education.

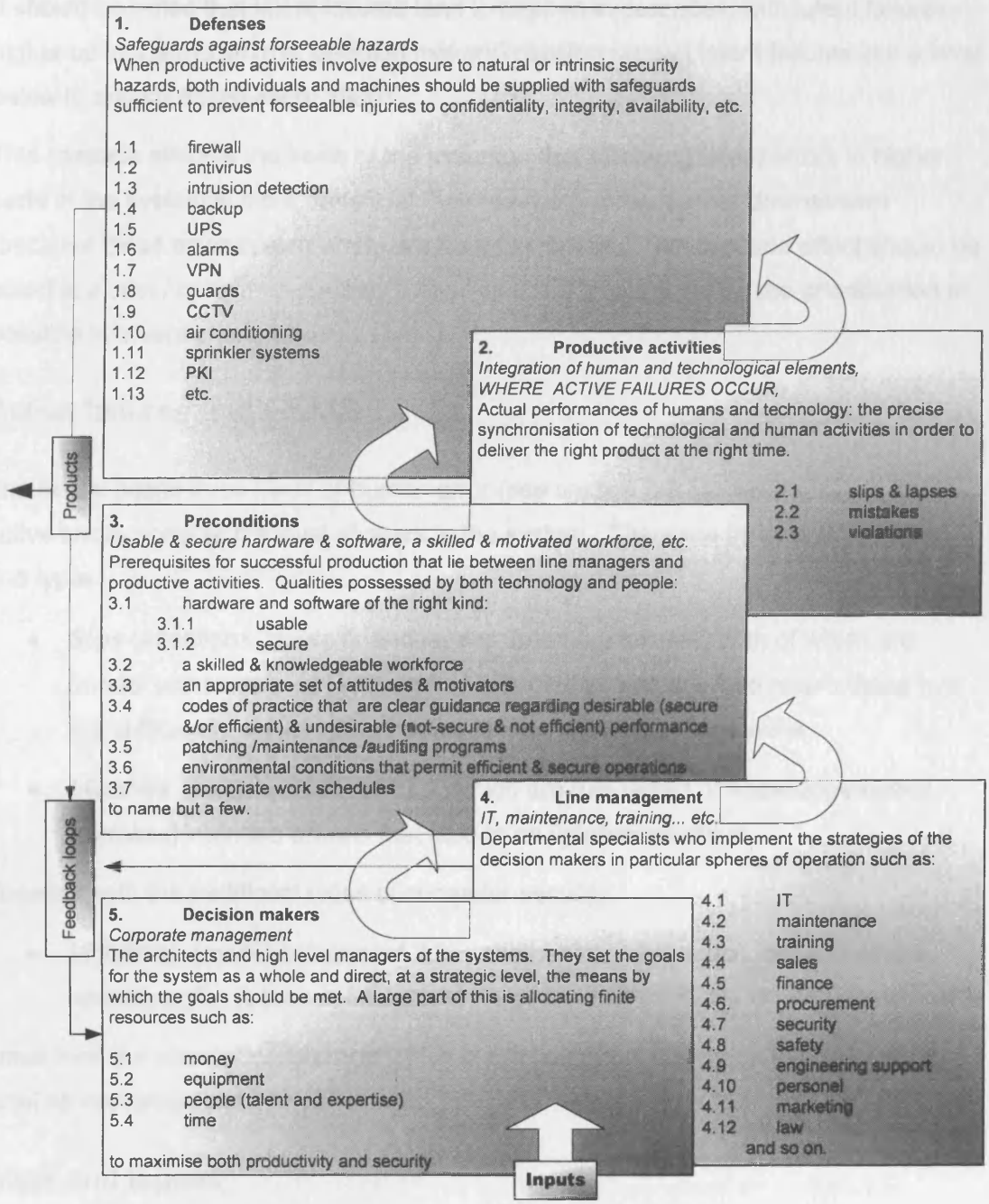


Figure 38 - Detailed view of large scale model

Defences (model component 1) protect the organisation, such as: 1.5 uninterruptible power supplies, 1.1 firewalls, 1.7 virtual private networking, 1.4 data backups, 1.11 sprinkler systems, etc.

The causes of a breach can be traced to failures at all levels listed in this model. To indicate this, the model's levels have been relabeled appropriately (Figure 37).

It should be noted that latent failures tend to happen in cascades, with latent failures higher up in the model (e.g. decision makers) causing several latent failures in the level below it, and so on (Reason, 1990).

This cascade effect is the basis of the assertion that removing latent errors in higher parts of the system is more beneficial than removing errors further downstream (because these downstream errors are then prevented). This cascade effect should be noted in a cost / benefit analysis as it has important implications for the prioritisation of possible interventions.

Active failures and GEMS

The model posits three kinds of human error (see section 3.3.5). *Active failure*, or active errors occur at the level of users in the system. There are three active failure and types:

- *Slips* (attentional failures), and *lapses* (memory failures) both of which are (model component 2.1) unintended actions that lead to a bad result, these two are sufficiently similar to be treated as one category. The third is
- *Mistakes* (model component 2.2, which are rule-based or knowledge-based *mistakes*) intended actions that lead to an unintended result.

Together with the traditional focus of computer security,

- *Violations* (model component 2.3) - another intended action, but one where security policy is broken intentionally,

these form the class of unsafe acts which in the domain of information security we re-label as *not-secure acts*.

Slips and lapses

These active failures are described in section 3.3.5 and the Appendix 1. The key property of these *not-secure acts* is that they are unintended. The user attempts a secure act, but it is executed incorrectly. For example, entering the most recently expired password when meaning to enter the new password that replaced it (e.g. one explanation for the prevalence of expired passwords – see Table 34 in section 8.3.1).

Mistakes

This stage of model contains another class of *active failure* known as *mistakes*. A *mistake* is an intended action which has unintentionally bad consequences. The user

is able to successfully execute the action that he planned to, but his act was not-secure because his plan was faulty in some way. For example, entering the expired password believing it to be the new password (e.g. another explanation for the prevalence of expired passwords in Table 34 in section 8.3.1). In this section two ways will be considered in which actions are planned/problems are solved, and the different kinds of errors that these planning mechanisms produce.

The first of the two attentional control mode methods of planning actions (see section 3.3.5) is by using heuristics. Actions are planned by using a series of *if then* rules, such as: *if* sending confidential e-mail *then* use PGP. These rules can be wrong in two ways: they can be good rules which are applied in the wrong situation (the *if* part is wrong), or they can be bad rules (the *then* part is wrong). Appendix 1 lists some of the ways that good rules can be misapplied, and that rules can be considered to be bad.

The second of the two attentional control mode methods of planning actions is by problem solving from first principles. These *knowledge-based errors* are described in more detail in Appendix 1.

Violations

Although *violations* have traditionally been the focus of computer security, there is evidence that they cause less damage than is caused by human errors (see end of section 3.3.5). A short taxonomy is presented in Appendix 1.

Although *violations* are intentional breaches of security policy, they are not always malicious. *Routine violations* are accepted as normal behaviour by people inside the organisation. They can occur when users believe the official policy to be inefficient, and where a shortcut exists. The intention is not to harm the organisation, but rather to support its goals by doing work more efficiently. An example might be leaving a workstation without locking the screen - many users do it because they trust their colleagues and believe it is unnecessary, not because they want hackers to have access to sensitive information. In fact, they may believe that it is necessary not to leave it locked as there may be social repercussions for being security conscious - colleagues may think that you distrust them or that you are otherwise being disrespectful (Weirich & Sasse, 2001).

Exceptional violations occur under unusual circumstances, where an action is perceived to be for the greater good of the organisation. The solution to an emergency might be to pursue a course of action which is against security policy. For example, when much needed information is in a soft document that has been quarantined by antivirus software so that it cannot be accessed, an operator disables the anti-virus

software and opens the document to save the day. A special case of *exceptional violation* is when a user is tricked into unknowingly harming the organisation through *social engineering*.

All intentionally malign security breaches by users are classed as *sabotage,* whether or not the user was tricked into them by *social engineering*. Included in this category are breaches where harm is not intended per se, but the advantage of the breach accrues to the individual at the organisation's expense. An example of this might be the salami attack, where the individual aims to make himself rich by skimming off the fractions of pence from interest calculations, rather than aiming to impoverish the bank (which does not even notice the attack).

Preconditions

Stage 3 of the model lists some of the preconditions for successful productive activity, such as codes of practice giving clear guidance about behaviours that are desirable and undesirable (model component 3.4). If these preconditions are not met, then latent failures have been introduced into the organisation and productive activity is likely to result in active failures and perhaps disaster.

Examples of this from previous chapters include:

- *Disclosure policies* (section 3.2.8) /not informing users that it is all right to write down passwords given suitable precautions. This predisposes users to try and remember their passwords, which causes them to forget them more frequently. When this happens, productive activity must stop until the password is reinstated by the helpdesk.
- *Password expiry* and *compartmentalisation of passwords* (section 3.2.8) both make the password management task more difficult, because it is allocated to users who are poorly equipped to deal with it rather than infrastructure. By spending more effort in managing their passwords, users are less able to perform productive activities. Moreover, the complex password management task is more likely to go wrong, requiring helpdesk support before productive activities can continue.

It is easy to construct hypothetical examples using this stage of the model. If users are not knowledgeable, then they may not know that threats exist or what to do about them, which predisposes the system towards security breaches and so is a latent failure. If users are overworked, then they are more likely to suffer slips and lapses or make mistakes, etc.

Line Management

Line management (model component 4) interpret the strategy set down by decision makers. They are responsible for ensuring that the preconditions exist for productive activity. Deficiencies or bad decisions made by these specialists result in the lack of a necessary precondition. If line management do not liaise appropriately, then there can be conflicting priorities and interests, which predispose the system to disaster. For example, a line manager in purchasing may keep costs of a new IT system down by not purchasing its optional single-sign-on module, despite the resulting additional password resets being charged to another line manager's account. Line management in turn depends upon decision makers.

Decision makers

If decision makers (model component 5) do not allocate appropriate resources to line management, then line management will not be able to put in place the necessary preconditions for productive activity. For example, funding may not be available for the deployment of single sign on technologies. Management also set the goals of the organisation, which may produce latent failures. For example, the goal of cutting costs may mean that line managers are encouraged to complete projects under-budget, which may cause a line manager not to purchase the optional single-sign-on module (see previous section).

The higher up in the model the latent failure occurs, the more dangerous it is because its consequences are wider reaching. Single bad decisions by decision makers can cascade to several latent failures lower down the organisation.

Benefits

The new model has several benefits, which are listed below. The first set of benefits relate to the modelling of the context of users in security systems:

1. It guides security practitioners and researchers to consider the wider context around security technologies - it reminds us that there is more to security than software and mathematics; there are people too, and their interactions with the above and each other. This helps practitioners and researchers expand their focus to consider a wider set of critical factors, and so makes their solutions more comprehensive. This gives greater chance of preventing security breaches than by merely blaming users for not complying.

2. Non-technical and social aspects such as organisational procedures and training are an essential part of the model, not an ancillary to it (cf. Dobson, 1993). This ensures that solutions based on the model integrate well into the enterprise, and help to build a "human firewall".
3. The model gives the context for lower-level models such as BLP (e.g. Bell & LaPadula, 1973). This helps the practitioner engineer the context, so it supports rather than subverts interventions or designs based on lower-level models.

In particular, the model focuses explicitly on organisations:

4. It situates the user in the context of the organisation, rather than treating the user as a single unity devoid of context (Dobson, 1993). This enables the practitioner/ researcher to better cope with the security problems of large distributed organisations.
5. The model contains an enterprise description, which LaPadula (1993) has identified as a desirable property. The model helps researchers and practitioners to identify relevant components of the enterprise/organisation which he or she might otherwise be unable to identify. It therefore better helps researchers and practitioners to understand and investigate breaches / security incidents
6. The model points to the importance of line management, which has been left out in rival approaches such as Spruit (1998) (see section 12.4.1). This gives the researcher/practitioner an extra route through which to intervene, thus giving greater flexibility.

The model has benefits related to its flexibility in application:

7. The model avoids premature formalisation (cf. Dobson, 1993). This can allow the model to be rapidly tailored to suit the organisation's individual needs.
8. The model can form the basis of tools to help incident investigators and security auditors in ways that purely technical models cannot. It can therefore help the practitioners to intervene in a wider variety of situations than purely technical models can.

And finally:

9. The model points to the area of security that is weakest, and where therefore the largest gains can be made. This gives the practitioner a way to make more powerful, efficient and effective interventions than those aimed at other areas.

Validation Method

The new model is too large to be validated within the confines of a PhD by conventional means such as experiments. Instead, a start on validation was made using peer review.

The new model was outlined in Brostoff (2002), together with examples of its application, and sent to two experts for review. The experts were Mary Ellen Zurko and Dieter Gollmann.

Mary Ellen Zurko, currently Lotus Next Generation Security Architect in IBM Software Group, was principle author of a paper that helped to start the recent movement for usable security (e.g. Zurko & Simon, 1996). She has been a prominent member of the ACM's New Security Paradigms Workshop for many years, serving on the Program committee and Steering committee, both as member and chair. Much of her extensive career in software engineering has been on computer supported collaborative work (CSCW). For the previous 5 years she has been developing and researching security for Lotus Notes, CSCW software designed for corporate wide use. This has given her insight into the security issues facing organisations like BT, and the integration and performance of software in such a diverse environment. For the 15 years before this she has been developing user interfaces for and performing user-tests of security software.

Dieter Gollmann is one of the editors-in-chief of the International Journal of Information Security, is serving on the program committees of the major European conferences on computer security (ESORICS), and cryptography (EUROCRYPT), has authored a computer security text book (Gollmann, 1999), and is a visiting Professor at Royal Holloway, University of London while working at Microsoft Research, Cambridge UK. His current interests are "best described as foundations of security".

Validation Results

The results are presented in two parts, overview and then detail. Within each part, the benefits of the model identified by expert reviewers are given first, and then their criticisms.

Overview

The expert reviewers identified a number of benefits of the new large scale model, which are summarised in Table 53. The quotes that give rise to these summaries are given in the next section (Detailed results - Benefits). The author agrees with these benefits of the large scale model, and they will not be discussed further.

The expert reviewers made a number of criticisms of the new large scale model, which are summarised in Table 54. The quotes that give rise to these criticisms are given in

Table 53 - Summary of the large scale model's benefits identified by expert reviewers.

Benefit type	Summary	Quote#
Scope	1. Gives an comprehensive view of the system	1, 2
Usefulness	2. Helps to find powerful & comprehensive interventions	3, 4
Integration	3. Human factors are well integrated with more traditional security concerns	5,6,7,8
Validity	4. Human error components relating to violations are well done	9
Novelty	5. It allows interesting tools and products	10
Relevance	6. It is complementary to other models	11

Table 54 - Summary of expert reviewers' criticisms of the large scale model.

Criticism type	Summary	Criticism#
Usefulness	The model is not useful because particular critical aspects of security are not treated in it.	1,2
Practicality	Various issues prevent the model being deployed in practice.	3,4,5,6,7,8,9
Validity	The model does not point to the area of security that can be improved the most, as is claimed	10
Relevance	The model contains irrelevant details	11
Clarity	The "Products" aspect of the model is not clear	12

the section *Detailed results - Criticisms.* The author does not agree with the majority of these criticisms. Arguments against these criticisms will also be given in *Detailed results - Criticisms.*

Detailed results - Benefits

This section will first give quotes where the expert reviewers' have identified benefits of the large scale model. These benefits are not contested by the author, and so are not discussed further.

Benefit 1 Gives a comprehensive view of the system

1. The Macro strengths of the model are that it encourages a full systems view of security and serious consideration of the implications of decisions at all levels.
2. It gives a good overall view

Benefit 2 Helps to find powerful and comprehensive interventions

3. it's good to provide a model that shows where the most impactful fixes would come from
4. On attacking latent vs. active failures (end of 2.1): Generally in security, it's good to pay attention to all aspects. You both want a system that is set up to be secure, and attention paid to making the human component secure as well (with both usability and consideration of morale issues). It's true you can't get rid of attempted external active failures, but you want to do what you can to minimize both latent and active failures. It's a belt and suspenders kind of approach.

Benefit 3 Human factors are well integrated with more traditional security concerns

5. It is nice to have a model that gives a place to slot usability failures.
6. I think it's quite useful to integrate human error into such a security model, and you do so quite well.
7. It does quite well in fleshing out and emphasizing usability, social, and organisational aspects,
8. I think the biggest win of this model is the emphasis it gives to social and usability issues. It can be cited when trying to ensure that they are properly considered.

Benefit 4 Human error components relating to *violations* are well done

9. The description of routine violations is good.

Benefit 5 It allows interesting tools and products

10. [about latent failure profile, based on proactive security review suggested in Expert Pack] it would be interesting to see such a bar chart generated.

Benefit 6 It is complementary to other models

11. ...the proposed model is complementary to existing security models.

Detailed results - Criticisms

A summary of each of the expert reviewers' criticisms of the large scale model is given below, preceded by a reference number matched in Table 54. Underneath each summary will be the quote(s) from which the summary is made, followed by discussion of the criticism's merit.

Criticism 1 *The model cannot cope with active adversaries.*

> 12. I think there's less attention to the implications of having an active adversary (the aspect that makes security different from most other disciplines).

This is a particularly serious charge. The active adversary notion is applied to human beings who are attacking the organisation (rather than the malware they are employing), who use intelligence, knowledge and skill to switch attacks when their current attack is not working. A significant proportion of attackers may be said to be active adversaries, therefore were this criticism true it would mean the model is useful in relatively few circumstances. This criticism shows that the model may not have been understood, which also severly limits it's utility.

The model does help against active adversaries, in three ways:

1. It posits that users actively subvert security systems in order to advance the organisation's agenda (e.g. they make *routine* and *exceptional violations* – see p.232). The model proposes that this is due to *latent failures* in the organisaiton, and that removing these will lessen the risk and impact of these violations.
2. It explicitly promotes knowledgeable and well motivated employees (in the *preconditions* layer of the model). This has been proposed as one of the primary safeguards against active adversaries (e.g. Schneier, 2000).
3. The most part of the model is about detecting vulnerabilities in the system, and plugging them - the security technique *prevention* (see section 2.1.2 on page 23). Adversaries of any kind must exploit vulnerabilities to breach security. If there are no vulnerabilities, then the attacker does not have anything to exploit and so cannot be successful. If the vulnerabilities exist but are too expensive or risky to attack, the security bar has been raised, and the adversary's options have been reduced. The model therefore protects against any kind of adversary. The only conditions

under which the model could not assist against active adversaries is if *prevention* does not work.

In addition, the model specifies that there be procedures and mechanisms in place that promote security, for example the Defences and the Preconditions levels of the model. *Detection* and *response,* the two other main techniques of security, are not explicitly specified in the current version of the model they can be easily accommodated, and should be in future revisions.

Criticism 2 *The model misses a necessary component of security - compliance testing.*

4. One aspect that seems less well covered by your model is checking for compliance. This includes tiger and penetration teams, running tools like SATAN, running audits of physical security compliance, etc. I suppose not considering these would be a line management deficiency. Perhaps is a precondition/motivator - 3.3

While the model does not explicitly use the words "compliance testing", it does imply that this is necessary. Layer 3 of the model, preconditions, asks that there be appropriate auditing programmes (stage 3.5 of the model). Compliance testing is a type of auditing programme. Since compliance testing is an important part of computer security and its implicit treatment in the model was not effective, it needs to be made explicit in the model.

Criticism 3 *The model's recommendations to proceduralise work are not practicable.*

5. workable procedures and relevant training are not possible (to the extent you suggest) for all jobs. And it's not just the higher in the organisation that works against proceduralization. It's also the size of the organisation. In small orgs, everyone is expected to show greater initiative and diversity of skills."

The model was used to assert that to avoid knowledge based errors leading to security breaches, work should be proceduralised *as much as possible* so that processing of work occurs at the rule based or skill based level, thus avoiding the knowledge based level and its potential errors. Where it is not practical to proceduralise work (as in the reviewer's example of a small company), then this strategy will not work. However, there are likely to be many situations where it *is* possible and practicable to proceduralise work, or parts of work, particularly in large organisations such as BT & UCL at which the current research is aimed. Moreover, the greater contribution of the

model is to identify the existence of errors, rather than their solutions. Procuduralisation is only one strategy to combat knowledge based errors, and others may be possible.

Criticism 4 *The model cannot help people who are constrained to intervene at a lower level than management..*

6. doesn't help someone "Figure out what they can fix, based on what they have responsibility for. While the greatest improvements may come from organisational fixes, they may not be within the scope of the decision maker. You do allude to that issue a couple of time (for instance, at the end of 2.2.1), but I'm not sure how it would be integrated into a concrete use of the model."

This is partially correct. The model points out that it is better to intervene at the level of management if it is possible. However, the model contains levels below management at which interventions can be made. The model can point to aspects of safeguards (for example) that may be improved on a technical level to better to prevent and respond to active failures, for example increasing the number of times passwords are confirmed during password changing, to prevent forgetting or confusion with the expired password. However, most security related interventions require at least management oversight.

Criticism 5 *The model would be hard to operationalise for novices.*

7. It's probably pretty hard for a non-security expert to translate from the abstractions of the model into concrete possibilities. It may be that you're just targeting intensive use by experts at this time, but that's not clear 8. There are so many issues, it's hard for the non-expert to know what to emphasize.

The model is not intended for novices to operationalise. It is intended that the model be operationalised by experts, who would create tools from the model that could be used by novices. A checklist is derived from the model and given in Brostoff (2002) as an example.

Criticism 6 *The model has not been used on real data, so we don't yet know if its possible to get the real world data that the model requires..*

9. I think it would be very useful to the model to use it on a real case; not just docu-dramas based on snippets of real cases. I know how hard that can be, but I think it could offer useful insight into what kind of information is available, and what kind isn't, in reality. If the information isn't available for some aspects of the model, then what would that imply?

Organisations are often unwilling to release details of their security breaches to outsiders because of the risk of damage to their reputations-therefore it is difficult to get real world data with which to test the model.

However, internal security departments necessarily have access to confidential and sensitive information. Moreover, in some circumstances there is protection in law to encourage sharing of sensitive information about vulnerabilities (Poulsen, 2004). The model is aleady used as the basis for regular safety audits in a process of continuous improvement (Groeneweg, 2002). This requires both management and employee commitment to the idea that investigating safety is good for the company even though safety breaches bring bad publicity. Safety and security have similar properties, and so a similar auditing process for security should be possible.

Criticism 7 *If assumptions of trusted hardware and software are necessary, then the model's useless.*

10. if this model can't be usefully used without trustworthy hw and sw, then it can't be usefully used, and it must be purely theoretical

This criticism rests on how terms are defined. "Trusted" is often used as a technical term in computer science, with a meaning similar to "logically or mathematically verifiable". The model requires that the system uses hardware and software "of the right kind", which is defined as hardware and software which is "usable" and "secure". The model assumes that there is hardware and software *of the right kind* available, which is a realistic assumption to make. However, the model does not make the assumption that *trusted* hardware and software is available. This is a far less realistic assumption, as the construction of trusted hardware and software is very difficult and expensive and its use is constrained, putting it beyond the reach of many organisations.

The explanation of the model did however use the term "trusted", although it was meant to mean "of the right kind". To avoid further confusion, future versions will employ less confusing terminology.

Criticism 8 *It is difficult to see how this model would be used in practice.*

11. There are a lot of issues with figuring out how to slot this into a corporations natural processes, which I think I've touched on.

It is proposed that the model is used as the basis for tools which would be "used in practice" rather than the model itself. Examples of these tools, such as a checklist for use in incident investigation have been given in Brostoff (2002). Moreover, tools and processes based on Reason's original (1990) model have been used in the petrochemicals industry (Groeneweg, 2002).

Criticism 9 *Needs a catchy name.*

12. need a catchy name

This is justified.

Criticism 10 *The model does not point to the area of security that can be improved the most, as is claimed.*

13. I'm still not convinced this points to the areas of security where the largest gains can (in practice) be made. None of your examples hone in on that aspect; they seem to work hard instead to cover lots/all areas

It would be difficult to test this assertion empirically. However, there is evidence that the human component of computer security leads to the most security breaches (Spruit & Looijen, 1996), and to the majority of the cost of security breaches (NIST, 1992). Taking this view, the assertion that humans are the weakest link and so the area that has the most ground to make up is correct.

Criticism 11 *The model contains irrelevant details: about the environment.*

14. Around page 46, it looks more like safety and less like security. 15. What environmental conditions could make a security breach more likely? Ones that make the system higher risk, like a high profile site or organisation, political, social, or cultural issues, well funded competition, etc

Human error is a large part of computer security (section 3.3.5). If the environment can cause human error, as this PhD's results demonstrate, then those aspects of the environment which can cause human error definitely are relevant to computer security.

Expert One described aspects of the environment which would tend to make the organisation more likely to be attacked as being relevant. The model also describes aspects of the environment which make the organisation more vulnerable to attacks, therefore they are relevant.

Criticism 12 *The "Products" aspect of the model is not clear.*

> 16. Looking back to table 4, I can't figure out the relationship between 1. Defences and the Products box.

This criticism probably refers to the *products* box in Figure 38, and the input and output lines connected to it. Product is used in its abstract sense in the model, meaning the output of work performed by the organisation. This may not have clearly emerged from the model's presentation.

As well as this, the model assumes that there are feedback loops from all levels of the model leading back to senior management. For convenience, these have been shown on the diagram in Figure 38 as a single loop, perhaps suggesting that feedback from the *defences* has something directly to do with *products*. This is not the case, and could be made more clear in future versions of the model.

Revised model

Stages 1 and 3 of the new model were revised in the light of criticisms made by the expert reviewers. This revised model is presented below in Figure 39.

The description of stage 1 was revised to counter the misapprehension that the model cannot help in situations where there is an active adversary. The new description makes more plain the principles that the defences are designed to uphold. These principles are the general principles of computer security, which are effective against active adversaries.

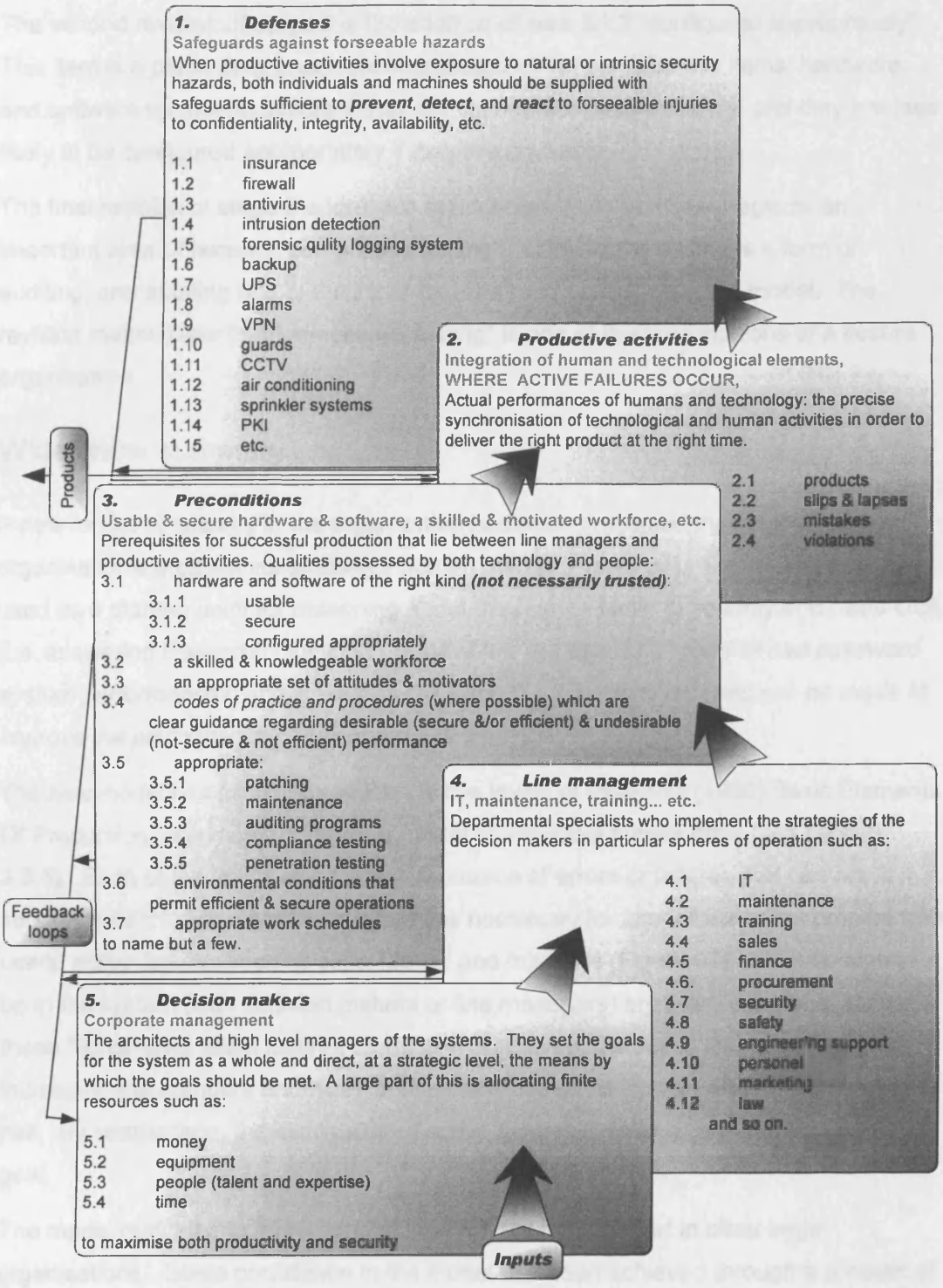


Figure 39 - The revised "Elevation" model

Section 3 of the model has had several amendments. The first of these addresses the criticism that the model is not useful if it relies upon trusted computer systems. The first revision emphasises that what is required is hardware and software *of the right kind,* rather than having properties and behaviour that are known and verifiable.

The second revision of stage 3 is the addition of item 3.1.3 "configured appropriately". This item is a particularly important intersection of the previous two items: hardware and software will not be secure if they are configured inappropriately, and they are less likely to be configured appropriately if they are unusable.

The final revision of stage 3 addresses the criticism that the model neglects an important area of security "compliance testing". Compliance testing is a form of auditing, and auditing one of the items that is already specified in the model. The revision makes clear that "compliance testing" is one of the preconditions of a secure organisation.

Wider view summary

A new model of security in the organisation has been presented that addresses the organisation's greatest vulnerability - human beings (Figure 39). This model may be used as a starting point for reasoning about the human factor in security in BT and UCL (i.e. answering Research Question C - *What are the causes of good or bad password system performance*?, and Research Question D - *What interventions can be made to improve the performance of password systems*?).

The new model has as its foundation the five levels of Reason's (1990) Basic Elements Of Production, which also contains a model of individual human error (see section 3.3.5). Each of the levels can serve as a source of errors or failures that can result in a security breach. For a breach to occur it is necessary for *latent failures* to combine with users' *active failures* such as slips, lapses and mistakes (Figure 37). Failures higher up in the system (with decision makers or line managers) are more important, because these higher-level *latent failures* produce a cascade of failures in levels below them, increasing the numbers and types of vulnerabilities in the system and so increasing its risk. By comparison, the elimination of *active failures* is seen as a much less fruitful goal.

The model may lead to more general models that can be used in other large organisations. Some confidence in the model has been achieved through a process of expert review. Many of the criticisms were due to misapprehensions of the model and the model was revised to address the criticisms. However, this review has shown that the model has difficulties in being understood, particularly in the key area of security against active adversaries. Further work is needed to address these difficulties.

12.5.6 Construction of investigation, auditing and design tools

The new models presented in this and the previous chapter may be used as the basis of auditing tools that can be part of an ongoing process of examining and improving security in the organisation. These may also form the basis of tools for incident investigation, which will help to prevent repetitions of the incident and may improve security more generally. The new models were based on Reason's models of human error. Tools already exist in the industrial safety sector that are based on these models: the Tripod series (Reason, 1997). These tools could be investigated for their applicability to security problems, and be adapted if necessary. The Tripod series are quite complex tools. It may be possible to use our new models to construct simpler and quicker tools such as checklists. This could also be investigated. Finally, the new models should be investigated as a basis for security design tools and methodologies.

References

Adams, A. (1996). *Reviewing human factors in password security systems.* Unpublished M.Sc., University College London, London.

Adams, A., & Sasse, M. A. (1999). Users are not the enemy: Why users compromise security mechanisms and how to take remedial measures. *Communications of the ACM, 42*(12), 40-46.

Adams, A., Sasse, M. A., & Lunt, P. (1997). *Making passwords secure and usable.* Paper presented at the HCI '97 - People and Computers XII, Bristol.

Allen, R., & Reber, A. (1980). Very long-term memory for tacit knowledge. *Cognition, 8,* 175-185.

Baddeley, A. (1997). *Human Memory: Theory and Practice* (Revised ed.). Hove, UK: Psychology Press.

Belgers, W. (1993). *UNIX password security* [html]. www.het.brown.edu/guide/UNIX-password-security.txt.

Bell, D., & LaPadula, L. (1973). *Secure Computer Systems: Mathematical Foundations and Model* (M74-244). Bedford, MA: MITRE Corp.

Berry, D. C., & Dienes, Z. (1993). *Implicit Learning:Theoretical and Empirical Issues.*: Psychology Press.

Bort, J. (2002, 21st October). Identity management begins with the humble password. *Network World.*

Bowers, J. (1992). The politics of formalism. In M. Lea (Ed.), *Contexts of computer mediated communication.*: Harvester/Wheatsheaf.

Breakwell, G. M. (1995). Diary Methods. In G. M. Breakwell (Ed.), *Research Methods in Psychology.*

Brennan, C. (2000). Password factors.: Unpublished.

Brentano, J., & Wiseth, K. (1996). *Enterprise-wide Security: Authentication and single Sign-on* (position paper). San Francisco: Network Applications Consortium.

Briney, A. (2001). Industry Survey 2001. *Information security,* 34-47.

Brostoff, S. (2002). *Expert pack* (Research Note RN/02/6). London: Department of Computer Science, University College London.

Brostoff, S., & Sasse, M. A. (2000, 5th - 8th September). *Are Passfaces more usable than passwords? A field trial investigation.* Paper presented at the HCI2000: People and Computers XIV - Usability or Else, Sunderland, UK.

Brostoff, S., & Sasse, M. A. (2001, September). *Safe and sound: a socio-technical approach to security.* Paper presented at the New Security Paradigms Workshop 2001, Cloudcroft, New Mexico.

Brostoff, S., & Sasse, M. A. (2003, April 5). *"Ten strikes and you're out": Increasing the number of login attempts can improve password usability.* Paper presented at the CHI 2003 Workshop on Human-Computer Interaction and Security Systems, Ft. Lauderdale.

BSI. (1996). *Information technology — Guidelines for the management of IT Security — Part 1: Concepts and models for IT Security* (BS ISO/IEC TR 13335-1: 1996). London: BSI.

BSI. (1998). *Ergonomic requirements for office work with visual display terminals (VDTs). Guidance on usability* (BS EN ISO 9241-11:1998). London: BSI.

BSI. (2001). *BS ISO/IEC 17799:2000 Information technology Code of practice for information security management.* London: BSI.

Bunnell, J., Podd, J., Henderson, R., Napier, R., & Kennedy-Moffat, J. (1997). Cognitive, associative and conventional passwords: Recall and guessing rates. *Computers and Security, 16*(7), 629-641.

Carden, P. (1999, March 22). The New Face of Single Sign On. *Network Computing.*

Cho, S., & Ciechanowicz, Z. (2001). *Selection of systems for detailed risk analysis in combined risk analysis approach.* Paper presented at the IFIP WG 9.6/11.7 Working Conference (Security & Control of IT in Society-II), Bratislava, Slovakia.

Cohen, G. (1996). *Memory In The Real World* (second ed.). Hove: Psychology Press.

Craik, F. I. M., & Lockhart, R. S. (1972). Levels of processing: A framework for memory research. *Journal of Verbal Learning and Verbal Behavior, 11*(6), 671-684.

Deane, F., Barrelle, K., Henderson, R., & Mahar, D. (1995). Perceived acceptability of biometric security systems. *Computers and Security, 14*(3), 225-231.

Dhamija, R., & Perrig, A. (2000, August 14-17). *Deja Vu:A User Study Using Images for Authentication.* Paper presented at the 9th USENIX Security Symposium, Denver, CO.

Dhillon, G., & Backhouse, J. (2001). Current directions in IS security research: towards socio-organisational perspectives. *Information Systems Journal, 11*, 127-153.

Dix, A., Finlay, J., Abowd, G., & Beale, R. (1998). *Human Computer Interaction* (2nd ed.). Hemel Hempstead: Prentice Hall Europe.

Dowell, J., & Long, J. (1998). Conception of the cognitive engineering design problem. *Ergonomics, 41*(2), 126-139.

Ebbinghaus, H. (1885). *Uber das Gedachnis* (H. Ruyer & C. E. Bussenius, Trans.). Leipzig: Dunker.

Ellison, C., Hall, C., Milbert, R., & Schneier, B. (2000). Protecting secret keys with personal entropy. *Future Generation Computer Systems, 16*, 311-318.

FIPS. (1985). *PASSWORD USAGE* (FIPS PUB 112): U.S. DEPARTMENT OF COMMERCE/ National Bureau of Standards.

GAO. (1998). *Executive Guide: Information Security Management: Learning From Leading Organisations* (GAO/AIMD-98-68). Washington, DC: United States General Accounting Office, Accounting and Information Management Division.

Garfinkel, S. (2000). *Database Nation: The Death of Privacy in the 21st Century.* Sebastopol, CA: O'Reilly.

Garfinkel, S., & Spafford, G. (1996). *Practical Unix and Internet Security* (second ed.). Cambridge, UK: O'Reilly & Associates.

Gollmann, d. (1999). *computer security.* Chichester: John Wiley and sons.

Groeneweg, J. (2002). *Controlling the Controllable, preventing business upsets* (Fifth ed.).

Haskett, J. A. (1984). Pass-algorithms: a user validation scheme based on knowledge of secret algorithms. *Communications of the ACM, 27*(8), 777-781.

Hook, B. (2002). Protecting your organisation against ghost workers. *TechRepublic.*

Hurley, E. (2003, 27 May). Survey: Most workers must remember six passwords or more. *SearchSecurity.com.*

Intel. (2003, 17th June). *Moore's Law*, [webpage]. Intel. Available: http://www.intel.com/research/silicon/mooreslaw.htm [2003, 14th July].

Jermyn, I., Mayer, A., Monrose, F., Reiter, M. K., & Rubin, A. D. (1999, August). *The Design and Analysis of Graphical Passwords.* Paper presented at the Proceedings of the 8th USENIX Security Symposium.

John, B. E., & Kieras, D. E. (1996). Using GOMS for user interface design and evaluation: Which technique? *ACM Transactions on Computer-Human Interaction, 3*, 287-319.

Just, M. (2003, April 6). *Designing Secure Yet Usable Credential Recovery Systems With Challenge Questions.* Paper presented at the Workshop on Human-

Computer Interaction and Security Systems, CHI 2003, Fort Lauderdale, Florida.

Kedem, G., & Ishihara, Y. (1999, August 23-26). *Brute Force Attack on UNIX Passwords with SIMD Computer.* Paper presented at the 8th USENIX Security Symposium, Washington, DC.

Kim, H.-J. (1995). Biometrics, is it a viable proposition for identity authentication and access control. *Computers and Security, 14*(3), 205-214.

King, M. M. (1991). *Rebus Passwords.* Paper presented at the Seventh annual computer security applications conference, San Antonio, Texas.

Klein, D. (1990). *Foiling the cracker: A survey of and improvements to password security.* Paper presented at the USENIX security workshop.

Leong, P., & Tham, C. (1991, January 21-25). *UNIX Password Encryption Considered Insecure.* Paper presented at the usenix 91, Dallas, TX.

Luchins, A. S., & Luchins, E. H. (1950). New experimental attempts at preventing mechanization in problem solving. *Journal of General Psychology, 42*, 279-297.

MacKay, J. (1999). Password Control Reason-for-calling survey. In s. brostoff (Ed.). london: Unpublished.

Macleod, M., Bowden, R., Bevan, N., & I., C. (1997). The MUSiC Performance Measurement Method. *Behaviour and Information Technology, 16*(4), 279-293.

Manber, U. (1996). A simple scheme to make passwords based on one-way functions much harder to crack. *Computers and Security, 15*(2), 171-176.

McCauley-Bell, P. (1999). *Predictive Modeling to Evaluate Human Impact on Internet Security.* Paper presented at the HFES99, Houston, TX.

McCauley-Bell, P. (2002). *A Holistic Paradigm for Evaluating the Role of Humans in the Security of Networked Information Systems.* Sandia.

McCauley-Bell, P., Carstens, D. S., & Malone, L. (2000). *Development of a model for determining the impact of password authentication practices on information security.* Paper presented at the Human Factors and Ergonomics Society Annual Meeting 2000, San Diego, CA.

Microsoft. (1998, 21 April 1998). *PDP-11/70*, [webpage]. Microsoft Information Research Services. Available: http://research.microsoft.com/~gbell/Digital/timeline/1975-2.htm [2003, 14th July].

Mitnick, K. D., & Simon, W. L. (2002). *The Art of Deception: Controlling the Human Element of Security.*: John Wiley & Sons Inc.

Murrer, E. (1999). Fingerprint Authentication. *Secure Computing*(March), 26-30.

NIST. (1992). *1991 Annual Report of the National Computer System Security and Privacy Advisory Board.*: National Institute of Standards and Technology.

Oechslin, P. (2003, August 17-21). *Making a Faster Cryptanalytic Time-Memory Trade-Off.* Paper presented at the Crypto'03, Santa Barbara, California.

Oppenheim, A. N. (2000). *Questionnaire Design, Interviewing and Attitude Measurement.*: Continuum International Publishing Group - Academic and Professional.

Palermo, D., & Jenkins, J. (1964). *Word association norms: Grade school through college*. Minneapolis: University of Minnesota Press.

Parkin, A. J. (1981). Determinants of cued recall. *Psychological Research, 1*(4), 291-300.

Parkin, A. J. (1993). *Memory: phenomena, experiment and theory*. Oxford, UK: Blackwell.

Petrie, H. (2002, Friday July 13th 2002). *Password Clues*, [web page]. CentralNic. Available: http://www.centralnic.com/page.php?cid=77 [2002, 12 September].

Pond, R., Podd, J., Bunnell, J. K., & Henderson, R. (2000). Word association computer passwords: The effect of formulation techniques on recall and guessing rates. *Computers and Security, 19*(7), 645-656.

Postman, L. (1985). Human learning and memory. In G. A. Kimble & K. Schlesinger (Eds.), *Topics in the history of psychology*. Hillsdale, NJ: Erlbaum.

Poulsen, K. (2004, 20 February). US info-sharing plan draws fire. *SecurityFocus.*

Power, R. (2002). 2002 CSI/FBI Computer Crime and Security Servey. *Computer security issues and trends, VIII*(1).

Preece, J., Rogers, Y., & Sharp, H. (2002). *Interaction Design: Beyond human-computer interaction.*: John Wiley and Sons, Inc.

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., & Carey, T. (1994). *Human-Computer Interaction*. Harlow, England: Addison-Wesley.

Protocom. (2003). *Global Password Usage Survey.*: Protocom Development Systems.

Provos, N., & Mazieres, D. (1999). *A future adaptable password scheme.* Paper presented at the USENIX annual technical conference.

RealUser. (2004)., [web page]. Available: http://www.realuser.com [2004, 5th August].

Reason, J. (1990). *Human Error*. Cambridge, UK: Cambridge University Press.

Reason, J. (1997). *Managing the risks of organisational accidents*. Aldershot, UK: Ashgate publishing Ltd.

Reber, A. S. (1989). Implicit learning and tacit knowledge. *Journal of Experimental Psychology: General, 118*, 219-235.

Robson, C. (2001). *Real World Research: A Resource for Social Scientists and Practitioner-Researchers.*: Blackwell.

Rosenthal, R., & Rosnow, R. (1991). *The Essentials of Behavioural Research* (second ed.). Singapore: McGraw Hill Book Co.

Sasse, A., Brostoff, S., & Weirich, D. (2001). Transforming the 'weakest link' — a human-computer interaction approach to usable and effective security. *BT technology journal, 19*(3), 122-131.

Sasse, M. A., Brostoff, S., & Weirich, D. (2002). Transforming the 'weakest link' - a human-computer interaction approach to usable and effective security. In R. Temple & J. Regnault (Eds.), *Internet and wireless security* (pp. 243-258). London: IEE.

Sasse, M. A., Harris, C., Ismail, I., & Monthienvichienchai, P. (1998). Support for Authoring and Managing Web-based Coursework: The TACO Project. In R. Hazemi & S. Hailes & S. Wilbur (Eds.), *The Digital University: Reinventing the Academy* (pp. 155-175): Springer-Verlag.

Schacter, D. L. (2001). *The seven sins of memory*. boston: Houghton Mifflin Company.

Schneier, B. (2000). *Secrets and Lies.*: John Wiley & Sons.

Schneier, B. (2002). *CRYPTO-GRAM: Fun with Fingerprint Readers*, [mailing list]. Counterpane.com. Available: http://www.counterpane.com/crypto-gram.html [2002, May 15th].

Semjanov, P. (2002). *Databases and spreadsheets password crackers*, [web page]. Available: http://www.password-crackers.com/crack2.html#winnt [2003, 7th July].

Shaffer, G. (2001). *NT's Poor Password Encryption*, [web page]. GeodSoft Website Consulting. Available: http://geodsoft.com/howto/password/nt_password_hashes.htm [2001, 27th May].

Slamenka, N. J. (1960). Retroactive inhibition of connected discourse as a function of practice level. *Journal of Experimental Psychology, 59*, 104-108.

Smith, S. L. (1987). Authenticating users by word association. *Computers and Security, 6*, 464-470.

Spector, Y., & Ginzberg, J. (1994). Pass sentence - a new approach to computer code. *Computers and Security, 13*(2), 145-160.

Spruit, M. E. M. (1998). *Competing Against Human Failing.* Paper presented at the 15th IFIP world computer congress, Vienna.

Spruit, M. E. M., & Looijen, M. (1996). IT security in Dutch practice. *Computers And Security, 15*(2), 157-170.

Svigals, J. (1994). Smartcards - A Security Assessment. *Computers & Security, 13*(2), 107-114.

Thalheim, L., Krissler, J., & Ziegler, P.-M. (2002, May. 22, 2002). *Body Check: Biometric Access Protection Devices and their Programs Put to the Test*, [web page]. c't. Available: http://www.heise.de/ct/english/02/11/114/ [2003, 4th July].

Tulving, E., & Psotka, A. (1971). Retroactive inhibition in free recall: Inaccessibility of information in the memory store. *Journal of Experimental Psychology, 87*(1), 1-8.

UCL. (2004) *Supporting Policy 2* (UCL Computing Regulations). London: UCL.

Viega, J., & McGraw, G. (2001). *Building Secure Software: How to Avoid Security Problems the Right Way.*: Addison Wesley.

Wade, N. (2002). *NTA Password Survey*. Rochester: NTA Monitor.

Wagner, M. (2004, April 19). The Password Is: Chocolate. *Information Week.*

Weirich, D., & Sasse, M. A. (2001). *Pretty good persuasion: a first step towards effective password security in the real world.* Paper presented at the new security paradigms workshop, cloud croft,NM.

Whitten, A., & Tygar, J. D. (1999, August). *Why Johnny can't encrypt: a usability evaluation of PGP 5.0.* Paper presented at the 9th USENIX security symposiom, Washington.

Wickens, C. (1992). *Engineering Psychology and Human Performance* (second ed.). NY: Harper Collins.

Winkler, I. (1997). *Corporate Espionage. What it is, why its happening in your company, what you must do about it.* Rocklin, CA: Prima Publishing.

References

Yan, J., Blackwell, A., Anderson, R., & Grant, A. (2000). *The Memorability and Security of Passwords -- Some Empirical Results.* (Technical Report 500). Cambridge, UK: Computer Lab, Cambridge University.

Yasin, R. (2002, April). Password Pain Relief. *Information Security.*

Zurko, M. E., & Simon, R. T. (1996, 17-20 September). *User Centered Security.* Paper presented at the New Security Paradigms Workshop, Lake Arrowhead,CA.

Zviran, M., & Haga, W. J. (1990). Cognitive passwords: the key to easy access control. *Computers and Security, 9*(8), 723-736.

Zviran, M., & Haga, W. J. (1993). A comparison of password techniques for multilevel authentication mechanisms. *The Computer Journal, 36*(3), 227-237.

Appendices

Appendix 1 Error types

Table 55 - Active failure components - Slips and Lapses

Slips & Lapses	*[skill-based errors]*
Errors in automated control of human action	
Manual error (typo, etc.)	Error due to bad physical execution of an intended and good plan: - *typos while entering passwords, over or under-shooting with the mouse, etc.*
Omission	Omission of an essential step in a sequence - *omitting to click Options and select the "Encrypt message contents and attachments" check box before sending a mail*
Repetition	Repetition of a step in a sequence - *clicking the encrypt button twice, so that a sensitive message is encrypted, then unencrypted again, before it is sent*
Interference errors/ Blends	The blending of two separate actions together, steps in each sequence crossover into the other sequence - *putting your password in the subject line of an e-mail, and typing "Hi how is it going" as your password*
Mis-ordering	Execution of all the steps in a sequence, but in the wrong order - *typing the password where the user-id should go then entering the user-id as the password*
Reversals	All the steps in a sequence are executed, in reverse order - *typing the password where the user-id should go then entering the user-id as the password*
Branching errors / Strong habit intrusions	Starting with one action sequence, and ending with another - *meaning to enter your web-password , but typing your email password instead*
Perceptual confusions	Objects that are in the expected place, or look similar or perform a similar operation are taken to be the target object, and actions meant for the target object are misapplied to them - *typing your email password into something you have mistaken for your email programme*
Reduced intentionality	Forgetting what you wanted to do - *meaning to digitally sign a contract to ensure its integrity before sending it, but then forgetting to*

Table 56 - Active failure components - Mistakes #1 (Rule based)

Mistakes #1	*[rule -based errors]*
Errors of heuristic control of human action, if... then...	
Misapplications of good rules	
First exceptions	An user is likely to misapply a good rule the first time that he meets a situation that is an exception to that rule. - *if you only read an e-mail, then you cannot be infected by it*
Informational overload	Only a limited number of diagnostic signs are adequately processed by the user, who then incorrectly determines which *if* applies - *Only looking at the first part of a double extension to a file attachment.jpg.bat*
Rule strength	Frequently used rules are stronger than infrequently used rules - *Typing your user name and password into XLOCK, only it requires a carriage return and your password*
General rules	By their nature, general rules are more generally applicable than specific rules, so general rules may be tried first. *See Rule strength example above.*
Redundancy	The user may focus on cues in the situation which are not diagnostic, particularly if they have been diagnostic in the past.. *Looking at the letters .jpg in the name of an e-mail attachment, but not looking at the letters .exe*
Rigidity	A tendency to repeat the same tried and tested solution when another one would be more appropriate. *Writing sensitive documents in French to keep them confidential, instead of encrypting them*
Application of bad rules	
Encoding deficiencies	Misunderstanding how things work. - *Deleted files have been safely destroyed.*
Action deficiencies	
Wrong rules	Will not achieve its purpose - Typing the same user name and password combination again, when it has not worked the first time
Inelegant rules	Achieves its purpose of the expense of causing other problems - *Managing your passwords by getting the helpdesk to reset them all for you*
Inadvisable rules	Achieve their purpose but many lead to an accident with continuous use *Remembering your passwords by writing them on a sticky stuck to your screen*

Table 57 - Active failure components - Mistakes #2 (Knowledge based)

Mistakes #2	***[knowledge-based errors]***
Problem solving from first principles, used to control human action if faced with a novel situation that can not be accommodated by skill or rule, or policies, etc	
Selectivity and Out of sight, out of mind	⇒ The problem space is inefficiently sampled by the user's conscious processing resources
Workspace limitations	The portion of the problem filling the user's conscious processing resources is small compared to the entire problem
Confirmation bias	Rapidly favouring one explanation then becoming loath to part with it
Overconfidence	Ignoring contradictory evidence
Biased reviewing	The "check-off illusion", confirming comprehensiveness by counting the factors considered instead of exploring their adequacy of consideration
Illusory correlation	Poor detection of co-variation
Halo effects	An aversion to discrepant orderings
Problems with causality	Underestimating the irregularities of the future and planning for fewer contingencies than will occur
Problems with complexity (subdivided below)	
Problems with delayed feed-back	When there is delayed feedback, users lose synchrony with the current situation and lag behind actual events
Insufficient consideration of processes in time	Being more interested in the way things are now than in considering how they have developed previously
Difficulties with exponential developments	Having no intuitive feeling for exponential growth, underestimating the rate of change
Thinking in causal series not causal nets	Being sensitive to the main effects upon the immediate goal, but being unaware of their side-effects on the remainder of the system
Thematic vagabonding	Flitting from issue to issue quickly, treating each one superficially
Encysting	Topics are lingered over and small details attended to lovingly while other more important issues are disregarded.

Table 58 - Active failure components - Violations

Violations	
Intended breaches of security policy.	
Routine violation	⇒ A Violation that is accepted as normal behaviour
Exceptional violation	A Violation that is necessary due to unusual circumstances, and is for the good of the organisation
Sabotage	A Violation that harms the organisation intentionally

Appendix 2 Password Diary Interview Schedule

Participant I.D. _____

Sex: Age: Occupation:

Do you use computers?: Industry sector:

How many people work under you?:

Interview schedule:

1. What is your opinion about password systems, what's wrong and how could they be improved?

2. What problems do you have when using them, and when do they occur?

<Go to separate sheets. Attach them to this document when completed>

	PASSWORD
	TYPE OF PWD/PIN
	Password chosen by: Self / Other ?
	Shared?
	Number
	Characters
	Symbols
	How long?
	Same UserID as pwds?
	Same as PWDs?
	How often changed (per year)?
	Change forced?
	Automated PWD entry by system?
	When you do it, is it Automatic/Consciou s
	Did you choose your UserID?
	Restrictions on PWD
	Exact size
	Min Size
	Max Size
	Num Only
	Num Inc
	Letters Inc
	Symb Inc
	No Names
	No WORDS
	No Last
	No Similar to Last
	No Previous

														PASSWORD
														TYPE OF PWD/PIN
														Password chosen by: Self / Other ?
														Shared?
														Number
														Characters
														Symbols
														How long?
														Same UserID as pwds?
														Same as PWDs?
														How often changed (per year)?
														Change forced?
														Automated PWD entry by system?
														When you do it, is it Automatic/Consciou s
														Did you choose your UserID?
														Restrictions on PWD
														Exact size
														Min Size
														Max Size
														Num Only
														Num Inc
														Letters Inc
														Symb Inc
														No Names
														No WORDS
														No Last
														No Similar to Last
														No Previous

9. *continued.* What is the purpose of these constraints?

How did you choose your passwords?There are many strategies that could be used to design a password. Please look at the following password design strategies, and annotate those you have used with the I.D. of the password you designed with them. <hand flash cards to participant>

11. There are some other ways of choosing passwords (in the table below), what do you like and dislike about these?

+	***A word (including names of people, places, and things).***	-
	A word that has been unaltered	
	Based on the name of someone you know	
	Based on the name of a famous person or group	
	Based on the name of a food	
	Based on the name of a car	
	Based on the name of something you own	
	Based on the name of a thing you want	
+	***A phrase***	-
	Based on a phrase (joke, speech, conversation, story, quote)	
	Based on something of personal importance to you	

	The first letter of each word of the phrase	
	Parts of a phrase joined together.	
+	***A number***	-
	Based on a number	
	Based on a birthday	
	Based on a number-plate	
+	***Based on something in the past***	-
	Based on a way you felt	
	Based on something that happened to you or someone else	
	Based on a famous event	
	Based on an experience (a smell, a sound, a feeling, a sight, or a taste)	
+	***Based on something in the future***	-
	Something that will happen	
	Something you or someone else will have to do	

	Something that you hope will happen	
+	***Based on a place***	-
	Based on where something is	
	Based on a route	
	Based on a position or orientation	
	Based on a scene	
+	***Based on some security algorithm***	-
	Swapping digits for letters	
	Adding symbols	
	Doing it backwards	
	In another language	
	using the shift key	
	mis-spelling	
	mixing things together	

	Altered in some other way	
+	***Related to another of your passwords***	-
	Based on a sequence that you can remember	
	the same password but containing a different number each time: tom1, tom2, tom3..	
	Based on a different sequence (please describe)	

12. How have you shared your passwords with other people, and why did you do it?

Is there anything else you'd like to say?

Appendix 3 Password Diary

Example front page of diary, showing participant and diary serial code

A0001

The first double page spread in the diary, showing instructions on the diary's use.

Instructions

Please fill in this diary for seven days, beginning the day after you received it. After seven days, start filling in the next diary. Carry on filling in a new diary every seven days until you have no more diaries to fill in. If you fill up a diary in less than 7 days, continue in the next diary. When they are all complete, post them back to me in the envelope I have provided.

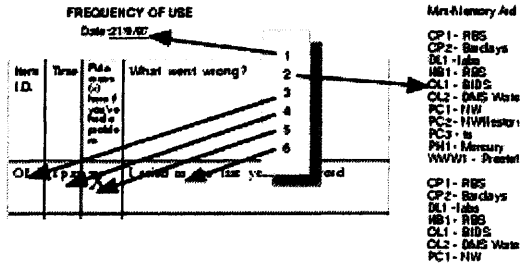
KEEP THE DIARY WITH YOU, AND FILL IT IN EVERY TIME YOU USE A PASSWORD. Even if you tried to use a password but couldn't for some reason, you should fill in the diary. At the latest, fill it in at the end of the day (if you leave it too long, you may have forgotten exactly what you did!).

Check the Memory Aid section to see that your passwords are all there and have been entered correctly. Please amend any incorrect entries, and add any that are missing.

Instructions Continued

Using your diary

Every time you use one of your passwords, try to use one, or want to use one but can't, make an entry in the diary. To make an entry: **1** Write the date; **2** Find the "Item I.D." in the list for the password you've just used; **3** Write the Item I.D.; **4** Write down the time the password was used; **5** Write a cross if you had a problem when you tried to use the password; **6** Write down what went wrong.



The example above shows that on the 21st September 1997 you used your BIDS password ("OL1") at 1pm, and it didn't work because you tried to use last years password.

The next double page spread in the diary, showing the memory aid section. Each diary had to such double page spreads. The next double page spread in the diary, showing the memory aid section. Each diary had two such double page spreads.

MEMORY AID

Check & amend existing passwords, add new passwords

How to make new entries

Use the list of situations below (and their letter codes in brackets). Use the next available number for each password category: for instance, your first Cash Point password should be given the I.D. 'CP1', and your second should be called 'CP2', etc.

Next, write down a short description of item in the Item Reminder column. This should remind you what the item was if you forget. If you are replacing an old password, in the last column write down the I.D. of the old password.

Alarm (AL)
Cash point (CP)
Door lock (DL)
Home banking (HB)
On line information system (OL)
PC Workstation (PC)
Phone (PH)
Pad locks / clamps (PL)
World Wide Web (WWW)
Other (other)
Sales (S)

Item I.D.	Item Reminder	Item replaces Item I.D:

An example password use table. Specimen item ids are shown down the right-hand side of the table, which would be personalised for each participant. The bulk of each diary would be made of these tables.

FREQUENCY OF USE

Date:_____

Item I.D.	Time	Put a cross (x) here if you've had a problem	What went wrong?

Mini-Memory Aid

CP1 - RBS
CP2 - Barclays
DL1 - labs
HB1 - RBS
OL1 - BIDS
OL2 - DMS Watson
PC1 - NW
PC2 - NWfilestore
PC3 - ts
PH1 - Mercury
WWW1 - Prestel

The inside back cover of the diary, showing one of the two password sharing tables, and the notes page.

SHARING

Date:_____

Item I.D	Why did you share it?	How did you share it?

Notes:

The back cover of the diary, showing instructions for the diary's return.

This booklet is part of an experiment carried out in University College London.

If you find this, please return it to Sacha Brostoff, in the Computer Science Department, University College London, Gower Street, London, WC1E 6BT.

0171 419 3462
s.brostoff@cs.ucl.ac.uk

Appendix 4 Study 2 Questionnaire

Password Questionnaire

Security Futures and Advanced Communications Engineering are collaborating in an effort to make computer security more usable and appropriate to the needs of BT people. We have commissioned a researcher at University College London, Sacha Brostoff, to study password based security. He has particularly been instructed to investigate password memorability, and the problem of forgetting passwords. To understand the scope and nature of the problem, Sacha will be conducting a survey, which he is pilot testing with this document. The survey takes approximately **10 minutes to complete**. If you have any comments about the survey questionnaire, please do not hesitate to make them where appropriate, in the space provided at the end or by email. Your help is greatly appreciated.

THIS SURVEY IS ANONYMOUS

Your answers will not be linked to you as an individual or used in anyway other than to support our analysis. If however you are willing to take part in follow up research then please give your name and a contact number or e-mail *on the tear-off slip at the back of the questionnaire*, or by e-mail (to s.brostoff@cs.ucl.ac.uk) and Sacha may contact you at a later date.

The questionnaire can be returned in the envelope provided, or by following the instructions at the end of the questionnaire.

Thank you for taking time to help us with our work.

Charles Brennan,
Senior Professional, ACE

We'd like to find out about the possible causes of passwords needing to be reset. The better we are able to understand what causes this need, the more probable it is that something can be done to reduce their burden.

1. Think of the BT work-related password you have last asked to be reset. What is the name of the system which you used the password for? Please write its name in the space below.
*

Password Questionnaire

2. Please describe in the space below why it needed to be reset. If you think you know why this happened, please give details below. If you run out of space, please use additional sheets.
*

We want to know if perceptions of security are associated with passwords needing to be reset. Please answer the following questions about the ***BT work-related password that you most recently requested to be reset.***

3. In your opinion, how sensitive is the information system for which this password was reset. Consider the information stored in the system, the functions performed by it and what these could be used for by somebody who knew your password? Please circle the most appropriate number.

Extremely Un-sensitive — Extremely sensitive

*1 *2 *3 *4 *5 *6 *7

4. In your opinion, how bad would it be if somebody broke into this system?

Wouldn't matter — Extremely Bad

*1 *2 *3 *4 *5 *6 *7

5. How likely is it that somebody would try to break into this system?

Extremely Unlikely — Extremely likely

*1 *2 *3 *4 *5 *6 *7

We want to know whether it is particular properties of passwords that cause the need for passwords to be reset. Please answer the following questions about the ***BT work-related password that you most recently requested to be reset.***

6. Normally when using this password, is it you that enters it into the system ? For example, some e-mail programs store and enter your password for you, or your secretary may operate the system on your behalf. Please circle most appropriate answer:

*I enter the password *The password is entered for me

7. Normally, when you enter this password into the system (or try to), do you have to *consciously* try and remember the password, or does it come to you *without thinking*? Please circle most appropriate answer:

*Consciously *Without thinking

8. How often do you change this password? Please write your answer in the spaces provided.

*I change this password _____ times per _____.

s.brostoff@cs.ucl.ac.uk [2]

Password Questionnaire

9. How often do you *use* this password? Think about the period leading up to this password needing to be reset. Please answer in the spaces provided below:
*I use the password _____ times per _____.

10. How long have you used this password? Please answer in the space provided below:
*_____

11. In what pattern is this password used? Please tick the most appropriate answer.
*❑ About a constant amount, which doesn't change much
*❑ Periods of intense use followed by periods of little use
*❑ A generally increasing amount
*❑ A generally decreasing amount

12. Is the last BT work-related password you asked to be reset the same as any passwords you currently own? Please circle the most appropriate answer.
*yes *no

13. If the answer to the last question was yes, please write below how many of your other current passwords are the same as this password?
*_____ are the same as this password.

14. Is this password the same as any other passwords you've owned in the past, and can no longer use? Please circle the most appropriate answer.
*yes *no

We are investigating whether it is the general task of using passwords that cause the need for passwords to be reset. Please answer the following questions about the passwords that you use at work and away from work.

15. How many password systems have you used or passwords have you taken ownership of *in the last year*? **Not just your BT passwords.** People usually are unaware of quite how many passwords they have. It may help to work through the sorts of things that passwords are used for. Please look at the list of applications below, and next to each item on the list write down *how many* of this type of *application* you have passwords for.

If you use the same password for different applications, please count each of these applications *separately*.

*Resource Management systems (Whereabouts, expenses, etc.) ____
*Email systems (Outlook Exchange, talk21, etc.) ____
*Web sites (The Financial Times online, etc.) ____
*PC/Workstation (Windows, screensaver, etc.) ____
*Network/Server/Dialup (BTRA, Windows networking, ISP, etc.) ____
*Library/Reference (Blades, etc.) ____
*Others ____

Password Questionnaire

16. Normally, when you enter these passwords into the relevant systems, do you have to consciously try and remember the password or look it up, or does it come to you without thinking? The same password categories as in the question above are repeated below. Next to these categories, please write down *how many* of these applications you can *remember without thinking*.

*Resource Management systems (Whereabouts, expenses, etc.) ____
*Email systems (Outlook Exchange, talk21, etc.) ____
*Web sites (The Financial Times online, etc.) ____
*PC/Workstation (Windows, screensaver, etc.) ____
*Network/Server/Dialup (BTRA,Windows networking,ISP, etc.) ____
*Library/Reference (Blades, etc.) ____
*Others ____

17. Did you choose the *BT work-related password you last requested to be reset* yourself, or was it generated by the system (or by someone or something else) for you. Please circle the most appropriate answer.
*I chose it *It was chosen for me.

18. If you chose the password yourself - did you choose it using the same method that you use for choosing other passwords? Please circle the most appropriate answer.
*Yes *No

19. Roughly what proportion of your BT passwords are chosen using this method? Please write your answer below as a percentage.
*____%

20. It may be that the way passwords are chosen contributes to them needing to be reset. Please describe *in a general way* how you chose this password. If this question is too sensitive, or if you didn't choose your last password to be reset, please ignore this question.
*

21. How do you manage all your passwords? Please write the answer below.
*

Password Questionnaire

22. What caused the password management system you described above to break down for the BT work-related password you last requested to have reset?
*

To better understand the data collected in this survey, it would be necessary to collect similar data from people who have a different pattern of password resetting. To do this, it is necessary to match the second group of survey respondents as closely as possible to the group of survey respondents answering this questionnaire. To do this, we need to know a little about our respondents.

23. What is your age? Please circle the most appropriate answer.
*16-25 *26-35 *36-45 *46-55 *56-65 *66+

24. What is your gender? Please circle the most appropriate answer.
*Female *Male

25. For how many years have you been using computers?
*____ years.

26. What grade are you in BT? Please write your answer in the space provided below
*_____.

27. If you have any comments about this questionnaire (that you haven't already made on this document) please make them below.
*

Thank you for filling in this questionnaire.

Details for returning:

By post to: "Password" c/o Charles Brennan at
.....
or
By BTNet fax to:.....Charles Brennan at
.....

If you have any questions or comments, please email them to Sacha Brostoff at:
s.brostoff@cs.ucl.ac.uk

Password Questionnaire

TEAR OFF SLIP

Complete the sections for your name and contact details *only* if you are willing for Sacha to contact you for more information. Alternatively, e-mail your contact details to the address at the bottom of the slip.

To ensure the anonymity of your questionnaire responses, this slip may be detached and returned separately from the rest of the questionnaire.

I AM WILLING TO BE CONTACTED FOR MORE INFORMATION.

Name:

Email:

Phone:

THANK YOU - your time and effort so far are greatly appreciated. Please return the slip in the envelope provided, or using the instructions overleaf.

s.brostoff@cs.ucl.ac.uk [6]

Appendix 5 Feedback from experts

From expert1, Mary Ellen Zurko, a computer system architect and programmer at IBM, who publishes human factors work in the computer security field.

Hi Sacha, Thanks for inviting me to read and respond to your model work. You definately need a catchy name for it.

The Macro strengths of the model are that it encourages a full systems view of security and serious consideration of the implications of decisions at all levels. The downsides of this are that it doesn't obviously help someone Figure out what they can fix, based on what they have responsibility for. While the greatest improvements may come from organisational fixes, they may not be within the scope of the decision maker. So, it's good to provide a model that shows where the most impactful fixes would come from, but it will also need to be applied in situations where the most impactful fix (say, more money for anything) is simply not currently possible. You do allude to that issue a couple of time (for instance, at the end of 2.2.1), but I'm not sure how it would be integrated into a concrete use of the model.

I think it's quite useful to integrate human error into such a security model, and you do so quite well. I think there's less attention to the implications of having an active adversary (the aspect that makes security different from most other disciplines). For example, in 3.1, I'm not sure that that aspect isn't lost in a straight translation of a safety model. For instance, in the middle bullit in the Usability issues section, it's hard to think about "secure actions". The context matters so much. What does it mean to be transparent about the range of insecure actions the equipment allows?

On attacking latent vs. active failures (end of 2.1): Generally in security, it's good to pay attention to all aspects. You both want a system that is set up to be secure, and attention paid to making the human component secure as well (with both usability and consideration of morale issues). It's true you can't get rid of attempted external active failures, but you want to do what you can to minimize both latent and active failures. It's a belt and suspenders kind of approach.

It's not clear to me how surfing on company time can be an not-secure act, any more than reading email on company time is inately an insecure act. Many folks actually use the web in their work, and you just can't be sure any link is benign.

The page and picture refernces are off; fyi.

The description of routine violations is good.

2.2.2.4, page 8, SSO part of pwd authn mechs - well, it has been since Kerberos. You're closer to the core of the issue when you say "interoperation" as well. Kerberos is set up so that anyone who uses it can in fact have a SSO experience; if everything else uses Kerberos, and if the right cross domain trust is set up.

workable procedures and relevant training are not possible (to the extent you suggest) for all jobs. And it's not just the higher in the organisation that works against proceduralization. It's also the size of the organisation. In small orgs, everyone is expected to show greater initiative and diversity of skills.

One aspect that seems less well covered by your model is checking for compliance. This includes tiger and penetration teams, running tools like SATAN, running audits of physical security compliance, etc. I suppose not considering these would be a line management deficiency. Perhaps is a precondition/motivator - 3.3.

Which maybe points to another aspect of the model that might make it difficult to use. It's probably pretty hard for a non-security expert to translate from the abstractions of the model into concrete possibilities. It may be that you're just targeting intensive use by experts at this time, but that's not clear. Maybe that's what you get at in section 4.2, 2nd bullet.

On a related note, section 3.1, Communications section, channels not regularly used - not regularly actively checked?

I didn't understand Illusory correlation and Halo effects in Table 3 (though I think you had better explanations for them somewhere else in the text).

I think it would be very useful to the model to use it on a real case; not just docu-dramas based on snippets of real cases. I know how hard that can be, but I think it could offer useful insight into what kind of information is available, and what kind isn't, in reality. If the information isn't available for some aspects of the model, then what would that imply?

2nd to last paragrah in 3.1 - it would be interesting to see such a bar chart generated. I'm not convinced it's easily possible. And then there's the question about how truthfully any of the questions might be answered. Particularly if there is concern about liability issues.

The example in section 3.3.3 seems particularly weak, for a couple of reasons. First, availability aspects that generally fall under security have some sort of maliciousness as their cause (or they're found via error, but could be exploited by a malicious user). DDoS, buffer overflows that crash the system. It's hard to see how an adversary could make that happen. The other aspect that's questionable is to say that the organisation should not have opted to only support supported configurations. That's the kind of overall decision made all the time by organisations, and telling them they can't save resources that way can be a non-starter.

"Facts of the case" is a confusing section on the roll ups (table 5, etc.), because some of the facts are made up.

Section 4.1, Advantages, bullit 7, I'm still not convinced this points to the areas of security where the largest gains can (in practice) be made. None of your examples hone in on that aspect; they seem to work hard instead to cover lots/all areas.

2nd to last bullet - not obvious to me how this is achieved, since it doesn't seem to have been argued via an example anywhere.

Last bullet - I don't think that really helps a lot. Only if the model can speak to the needs of decision makers can it help. Every discipline can find experts who say that discipline should be given more resources.

4.2, 1st bullet; if this model can't be usefully used without trustworthy hw and sw, then it can't be usefully used, and it must be purely theoretical. That would be pretty depressing to me.

5 References, the Baker reference; actually, I don't think that paper was presented at NSPW '96. Dixie wrote it up for the NISSC panel based on several papers presented, and Marv brilliantly put it into the proceedings.

While I think section 6.1 is a good section to have, I think it's not well thought out yet. It's unfair to criticise other models about details of technology that weren't a concern when they were written, such as addressing BLP and multiple system authentication. It reminds me a bit of Pancho's NSPW '99 paper; if you shift the ground on a security model or mechanism, of course you can criticise it. It would be fair to say that the existing models don't cover broad areas of concerns, which is why another model might be useful.

Also, you seem to include only the most abstract security models, which are least likely to overlap with your concerns. Gasser's Building A Secure Computer System includes the user as part of its overall model, as I'm sure any consulting company's model would. And McGraw's latest book probably does as well. What about Toward a Secure System Engineering Methodology in NSPW '98? The presentation at that time was pretty comprehensive, though I don't remember if DARPA restrictions made it hard for them to update the paper.

Section 6.3, Check question 1, Was the breach forseeable? : wow; how can anyone answer that question? If the answer is yes, that means I was stupid.

It is nice to have a model that gives a place to slot usability failures.
Around page 46, it looks more like safety and less like security.
What environmental conditions could make a security breach more likely? Ones that make the system higher risk, like a high profile site or organisation, political, social, or cultural issues, well funded competition, etc.
Sales, Marketing, Publicity, page 48: what about followup concerns like reputation, fixing the holes, etc.
6.4 evaluation questions
1. I think I've sprinkled some of that about above. It gives a good overall view, short shrift to issues that arise from active adversaries, not a lot of help when the scope for change is constrained, and not a lot of help prioritizing (which is important).
2. It seems closest in spirit to other overall system models, like Gasser's and Saydjari's work. It does quite well in fleshing out and emphasizing usability, social, and organizaitonal aspects, perhaps to the detriment of technical security aspects.
3. There are a lot of issues with figuring out how to slot this into a corporations natural processes, which I think I've touched on. And there are so many issues, it's hard for the non-expert to know what to emphasize. It's hard to imagine a really comprehensive breach analysis of any breach that didn't make it to the cover of the NYTimes.
4. I think the biggest win of this model is the emphasis it gives to social and usability issues. It can be cited when trying to ensure that they are properly considered.
5. and 6. See above
Looking back to table 4, I can't Figure out the relationship between 1. Defences and the Products box. As a product person, I look on it as my responsibility to try to produce products that can provide security in configurations customers find desirable. So, while I fully agree that, for example, as in 3.2, all corporations should have a skilled and knowledgeable workforce, I would consider myself lazy if I made that a full precondition of anything I shipped, since I am well aware that that isn't always the case, and with changing times it's hard to keep it a constant.
Trying to Figure out how I might use Tables 1 - 4, I find myself wondering how I could use them to argue that foisting a security decision onto the user is a bad idea ("Do you want to run these potentially dangerous Word macros?"). That's the kind of usabilty problem I deal with regularly, and where I can possibly have some impact. Perhaps that is the Gulf of evaluation, but I would like to argue we shouldn't be bothering the user with stuff like that at all (we should make it safe/secure), rather than providing the poor user with even more feedback.

From expert 2, Dieter Gollmann, a hci and security researcher at Microsoft.

I had a quick look through Sacha's expert pack. One of the problems with security is that it is quite a multi-faceted area and I am definitely not an expert on the management side of security.
The statement 'In recent years, the security research community ...' at the start of the introduction is only true for some sectors of this community. (I am happy to take credit for changing Bruce Schneier's outlook on this matter ...) At Royal Holloway, students have been told for (at least) a decade that security is a people problem and that security problems are not solved by technology alone.

In terms of terminology, the differentiation between active failures, latent failures, and violations reminds a lot of work in the dependability community that uses terms like error, flaw, failure, etc. This framework has been promoted by IFIP WG 10.4 (Jean-Claude Laprie et al.) and, as with Reason's work, much of their background is in safety-critical systems.
In my understanding, Sacha's model is concerned with operational violations of given security policies but less with the security policies themselves. For example, in section 3.3.1 it is taken for granted that encryption should be used. (Security policy happens to be an overloaded term used in different contexts in the security community. My favourite reference on this subject is D.F. Sterne, On the Buzzword "Security Policy", Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy, pages 219-230.) Research into the reasons why security policies are not effective and into changing this situation is valuable indeed, but it is different from research into the 'security policies' enforced within an IT system. In this sense, the proposed model is complementary to existing security models.=20
The security models listed in appendix 6 do not seem to be relevant to Sacha's work. BLP, Biba, and Clark-Wilson are more concerned with security policies enforced within an IT system than with the type of security policies members of an organisation are asked to adhere to. It would be a mistake (by their proponents or their critics) to treat them as universal security models.

Appendix 6 The Password Manager (PM)

Become your own (Password) Manager!

From ISE Password Control

Please have your EIN and PIN ready.

Become your own (password) manager

Contents

Become your own (password) manager

Introduction and General Instructions

Use this document as a password management manual

- **Group the computer systems you use according to how often their passwords must be changed.**
- **Each time you *Change a password*, write a letter "C" in the relevant column so that you know which version the password is 'up to'.**
- **Each time you *Sign in with a system*, write a letter "S" in the relevant column so that you know which systems still think you're alive.**

General password tips

- To make things easier to remember, try making all the passwords in a group the same, but make each group different.
- Remember to choose a password of appropriate strength for the sensitivity of the different groups. Each group has instructions about how to do this.
- Book a quiet time in your diary for changing your passwords, so you don't have to do it in a rush.
- Try to remember each group's password 10 minutes to half an hour after you've changed it. This will make the password stronger in your memory, and act as a *Sign in* with the computer system.

Become your own (password) manager

Instructions for Passwords changed monthly

Use

STRONG passwords

with this group

1. **Use the first letters of each word** in a line from a favourite song or poem. If it's a well known line (such as the first line or the chorus) its easier to crack, so try to use a different line, or miss out the first word, or use a rare poem or song that few people know. This strategy makes a password that looks random, so that the hacker is forced to try all possible combinations. As an example, the second line from "Yesterday", by the Beatles.. *Now it looks as though they're here to stay* becomes *nilattht*s
2. Now change some of the letters for numbers or symbols. Do not choose the first or last letter for this, but letters from the middle of the password. *nilattht*s then might become *ni1@ttht*s. Some systems require passwords of this format, since it makes them harder to crack.

- For a top strength and memorable password
  - If it's a well known line (such as the first line or the chorus) its easier to crack, so try to use a different line, or miss out the first word, or use a song that's not at no.1 right now.
  - **Make the password 8 characters long**, as this is a requirement on some systems.

The grey part of the table >>

...opposite contains systems which are interconnected, so that changing the password on one of the systems will update the passwords of some of the other systems. If you change the password on the wrong system then the password updating programs will get confused and **you will require a password reset**. However the systems have been arranged in the table to make password changing easy.

- There is a white arrow in the great part of the table, pointing down.
- change the password of a system next to the **top of the arrow**
- all the systems below it in the grey section will be updated automatically
- systems above it will not be updated.
- Always change the password at the start of the arrow, and the others will be updated for you.

On systems with linked password changing (such as *NC access* and *CSS*), you must *Sign in* separately with each of the linked systems - it is not enough to just change the password on one of the systems.

Sign in with your systems once a month

- Many BT computer accounts will automatically be suspended if you don't Sign in with them every month, by logging in.
- Most systems with monthly password changing require monthly sign-ins.
- You can indicate which systems you use are like this, by placing a tick in the "Sign in Required" column next to the system's name.

Become your own (password) manager

Passwords changed monthly

Computers used for **very sensitive information**. Use a top strength password. **Never use these with the Internet** – they will be stolen by hackers and used to commit crime *in your name*.

Sign-in Required	System Name	Date (fill in below)	Password Hint (use pencil)
		Write down C or S below each time you (C) Change the password / Sign in to the system (S)	
	Change a password & all the ones below it on the arrow will be changed for you.		
	SMART		
✔	UK Connect		
✔	NC Access		
✔	CSS		
✔	COSMOSS		
✔	TSO		

Become your own (password) manager

Passwords changed monthly, continued

Sign-in Required	System Name	Date (fill in below) Write down C or S below each time you (C) Change the password / Sign in to the system (S)	Password Hint (use pencil)

Become your own (password) manager

Passwords changed monthly, continued

Sign-in Required	System Name	Date (fill in below) Write down C or S below each time you (C) Change the password / Sign in to the system (S)	Password Hint (use pencil)

Become your own (password) manager

Instructions for Passwords changed Quarterly

Use

MEDIUM strength passwords

with this group

1. **Put three or more words together**. This strategy uses the vast number of combinations of dictionary words in order to create a combination that is difficult to guess. Leave out the spaces between the words, and use only one case to make it easier to remember. Making the combination of words funny and unexpected also aids memorability. As an example *Hit a boss* becomes *hitaboss*.
2. Now change some of the letters for numbers or symbols. Do not choose the first or last letter for this, but letters from the middle of the password. *hitaboss* then might become *hit@6o$s*. Some systems require passwords of this format, since it makes them harder to crack.

The grey part of the table >>

... opposite contains systems which are interconnected, so that changing the password on one of the systems will update the passwords of some of the other systems. If you change the password on the wrong system then the password updating programs will get confused and **you will require a password reset**. However the systems have been arranged in the table to make password changing easy.

- There is a white arrow in the great part of the table, pointing down.
- change the password of a system next to the **top of the arrow**
- all the systems below it in the grey section will be updated automatically
- systems above it will not be updated.
- Always change the password at the start of the arrow, and the others will be updated for you.

On systems with linked password changing (such as *NC access* and *CSS*), you must *Sign in* separately with each of the linked systems - it is not enough to just change the password on one of the systems.

Sign in with your systems once a month

- Many BT computer accounts will automatically be suspended if you don't Sign in with them every month, by logging in.
- Most systems with **quarterly password** changing require **monthly sign-ins**.

You can indicate which systems you use are like this, by placing a tick in the "Sign in Required" column next to the system's name.

Become your own (password) manager

Passwords changed Quarterly

Computers used for **quite sensitive** information. Use a medium strength password. Try not to use these with the Internet.

Sign-in Required	System Name	Date (fill in below)	Password Hint (use pencil)
		Write down C or S below each time you (C) Change the password / Sign in to the system (S)	
	Change a password & all the ones below it on the arrow will be changed for you.		
	WHOOSH COLON		
✔	WHOOSH TITAN		
✔	NT DOMAIN		
✔	OUTLOOK		
✔	LAN		
✔	E Gatekeeper		
✔	CAMMS		

Become your own (password) manager

Passwords changed Quarterly, continued

Sign-in Required	System Name	Date (fill in below) Write down C or S below each time you (C) Change the password / Sign in to the system (S)	Password Hint (use pencil)

Become your own (password) manager

Passwords changed Quarterly, continued

Sign-in Required	System Name	Date (fill in below) Write down C or S below each time you (C) Change the password / Sign in to the system (S)	Password Hint (use pencil)

Become your own (password) manager

Instructions for Passwords changed Annually

Use

WEAK passwords

with this group

1. Pick as single, lower-case word that you can remember. Examples would be *password*, *chocolate*, *tottenham*.
2. Now change some of the letters for numbers or symbols. Do not choose the first or last letter for this, but letters from the middle of the password. *password* then might become pa$$word. Some systems require passwords of this format, since it makes them harder to crack.

Writing down passwords in this group is OK

- you do not have to write a hint
- you can write down the password itself
- as long as it is not the same as a password in a more sensitive group
- do not use the same passwords as systems with strong or medium strength passwords

Sign in with your systems once a QUARTER

- Many BT computer accounts with **annual password changing** will automatically be suspended if you don't Sign in with them every **quarter**, by logging in.

You can indicate which systems you use are like this, by placing a **tick** in the "Sign in Required" column next to the system's name.

Become your own (password) manager

Passwords changed Annually

Computers used for not very sensitive information. Use a password that's easy to remember.

Sign-in Required	System Name	Date (fill in below) Write down C or S below each time you (C) Change the password / Sign in to the system (S)	Password (use pencil)
✔	CDS		
✔	MCSS3		
✔	FPQ		

Become your own (password) manager

Instructions for Passwords never changed – important ones

Use

STRONG passwords

with this group

1. **Use the first letters of each word** in a line from a favourite song or poem. If it's a well known line (such as the first line or the chorus) its easier to crack, so try to use a different line, or miss out the first word, or use a rare poem or song that few people know. This strategy makes a password that looks random, so that the hacker is forced to try all possible combinations. As an example, the second line from "Yesterday", by the Beatles.. *Now it looks as though they're here to stay* becomes *nilattht*s.
2. Now change some of the letters for numbers or symbols. Do not choose the first or last letter for this, but letters from the middle of the password. *nilattht*s then might become *ni1@tth*s. Some systems require passwords of this format, since it makes them harder to crack.

Become your own (password) manager

Passwords never changed – important ones

Computers used for sensitive information. Use a different strong password than ones that have to be changed every month.

Sign-in Required	System Name	Date (fill in below) Write down C or S below each time you (C) Change the password / Sign in to the system (S)	Password Hint (use pencil)

Become your own (password) manager

Instructions for Passwords never changed – Mickey Mouse ones

Use

WEAK passwords

with this group

1. Pick as single, lower-case word that you can remember. Examples would be *password*, *chocolate*, *tottenham*
2. Now change some of the letters for numbers or symbols. Do not choose the first or last letter for this, but letters from the middle of the password. *password* then might become pa$$word. Some systems require passwords of this format, since it makes them harder to crack.

Writing down passwords in this group is OK

- you do not have to write a hint
- you can write down the password itself
- as long as it is not the same as a password in a more sensitive group
- do not use the same passwords as systems with strong or medium strength passwords

Become your own (password) manager

asswords never changed – Mickey Mouse ones

omputers used for **unimportant things**. Use a 'Mickey Mouse' strength password.

Sign-in Required	System Name	Date (fill in below) Write down C or S below each time you (C) Change the password / Sign in to the system (S)	Password (use pencil)

Become your own (password) manager

Blank

Become your own (password) manager

Different things passwords might be used for

One of the most difficult parts of dealing with passwords is remembering precisely how many of them you have! It is so difficult to do that people are often surprised that they have so many passwords. To help you remember which systems you have passwords for, look at the list below of different things passwords might be used for.

Look at each item in turn, and try to remember if you have any passwords for something like that. Then go on to the next item, and so on.

- "On/Off Button" passwords on PCs
- Network / Novell / LAN passwords
- Dial-up software passwords
- Email account passwords
- Internet Service Provider passwords
- Customer details databases
- calendar software
- jobs databases/scheduling software passwords
- digital libraries passwords
- invoicing system passwords
- account managing software
- customer service software
- purchasing system software
- financial system software
- web based computer systems
- security device passwords
- bulletin board passwords

Become your own (password) manager

Form for sharing passwords

Appendix 6 Password Manager

Become your own (password) manager

[form goes here]

Become your own (password) manager

Form for getting a password reset PIN

Become your own (password) manager

[form goes here]

Become your own (password) manager

Notes

[for your notes]

Contact details

In association