

RHODES UNIVERSITY

Where leaders learn

**Passphrase and Keystroke Dynamics Authentication:
Security and Usability**

By

Bhaveer Bhana

Passphrase and Keystroke Dynamics Authentication: Security and Usability

by

Bhaveer Bhana

g18B4469

Thesis

submitted in fulfilment of the requirements for the degree

Doctorate of Philosophy

in

Information Systems

in the

Faculty of Commerce

of

Rhodes University

Supervisor: **Prof. Stephen Flowerday**

2019

Abstract

It was found that employees spend a total 2.25 days within a 60 day period on password related activities. Another study found that over 85 days an average user will create 25 accounts with an average of 6.5 unique passwords. These numbers are expected to increase over time as more systems become available. In addition, the use of 6.5 unique passwords highlight that passwords are being reused which creates security concerns as multiple systems will be accessible by an unauthorised party if one of these passwords is leaked.

Current user authentication solutions either increase security or usability. When security increases, usability decreases, or vice versa. To add to this, stringent security protocols encourage unsecure behaviours by the user such as writing the password down on a piece of paper to remember it. It was found that passphrases require less cognitive effort than passwords and because passphrases are stronger than passwords, they don't need to be changed as frequently as passwords.

This study aimed to assess a two-tier user authentication solution that increases security and usability. The proposed solution uses passphrases in conjunction with keystroke dynamics to address this research problem.

The design science research approach was used to guide this study. The study's theoretical foundation includes three theories. The Shannon entropy formula was used to calculate the strength of passwords, passphrases and keystroke dynamics. The chunking theory assisted in assessing password and passphrase memorisation issues and the keystroke-level model was used to assess password and passphrase typing issues.

Two primary data collection methods were used to evaluate the findings and to ensure that gaps in the research were filled. A login assessment experiment collected data on user authentication and user-system interaction for passwords and passphrases. Plus, an expert review was conducted to verify findings and assess the research artefact in the form of a model.


The model can be used to assist with the implementation of a two-tier user authentication solution which involves passphrases and keystroke dynamics. There are a number of components that need to be considered to realise the benefits of this solution and ensure successful implementation.

Declaration

I, Mr Bhaveer Bhana, hereby declare that:

- The work in this thesis is my own work.
- This thesis has not previously been submitted in full or partial fulfilment of the requirements for an equivalent or higher qualification at any other educational institution.
- I am fully aware of the Rhodes University's policy on plagiarism and I have taken every precaution to comply with the regulations.
- I am fully aware of the Rhodes University's policy on research ethics and I have taken every precaution to comply with the regulations. Ethical clearance certificate number: **CIS18-4**

Name: Bhaveer Bhana

Signature: 

Date: 08/03/2020

Acknowledgements

I would firstly like to thank my supervisor, Professor Stephen Flowerday, for all guidance and advice, and significantly building my research skills and knowledge.

I would like to express my appreciation to all members of the expert review group for taking time out of their busy schedule to participate in my study and providing me with valuable feedback in a timely manner.

I am grateful to the proof readers for reviewing my writing skills and providing me with helpful and informative feedback.

I would like to thank all my friends and family for their support, encouragement and patience throughout my PhD degree.

I would also like to thank God for helping me cope with all the stress and giving me patience and knowledge to complete my PhD degree to the best of my ability.

Table of Contents

Chapter 1 – INTRODUCTION	1
1.1 Background.....	2
1.2 Current Research.....	4
1.3 Problem Statement	6
1.4 Research Question	7
1.4.1 Main Question.....	7
1.4.2 Sub-question 1	7
1.4.3 Sub-question 2.....	8
1.4.4 Sub-question 3.....	8
1.5 Objective of the Study	8
1.6 Significance of the Study.....	9
1.7 Theoretical Foundation.....	10
1.7.1 Keystroke-level Model.....	10
1.7.2 Chunking Theory.....	10
1.7.3 Shannon Entropy Theory	10
1.7.4 Section Summary.....	11
1.8 Research Methodology Overview.....	11
1.9 Research Propositions	13
1.10 Delimitations	13
1.11 Ethical Considerations.....	14
1.12 Research Contribution.....	14
1.13 Summary of Findings	16
1.14 Chapter Layout	16
Chapter 2 – METHODOLOGY	17
2.1 Introduction	18
2.2 Research Paradigm	18
2.3 Abductive Reasoning	18
2.4 Design Science	19
2.4.1 Design Science Guidelines	19
2.4.2 Design Science Application.....	21
2.5 Research Process.....	24
2.6 Data Collection.....	25
2.6.1 Primary Data Collection.....	25
2.6.2 Expert Review.....	32

2.7	Secondary Data Collection.....	33
2.8	Primary Data Analysis.....	34
2.8.1	Login Assessment.....	34
2.8.2	Expert Review.....	35
2.9	Secondary Data Analysis.....	35
2.10	Theoretical Foundation.....	35
2.10.1	Shannon Entropy Theory.....	36
2.10.2	Chunking Theory.....	36
2.10.3	Keystroke-level Model.....	37
2.11	Delimitations.....	38
2.12	Ethical Considerations.....	38
2.13	Conclusion.....	39
Chapter 3 – SECURITY.....		41
3.1	Introduction.....	42
3.2	User Authentication.....	43
3.2.1	One-tier Authentication Insufficient.....	43
3.2.2	Multi-tier Authentication.....	43
3.3	How Passwords are Cracked.....	44
3.4	Forms of Authentication.....	46
3.4.1	Knowledge.....	47
3.4.2	Forms of Knowledge.....	47
3.5	Passwords and Passphrases Defined.....	49
3.6	Password Policies.....	50
3.7	Strength Indicators.....	53
3.8	Summary of Password and Passphrase Components.....	54
3.9	Biometrics.....	55
3.9.1	Types of Biometrics.....	56
3.10	What is Keystroke Dynamics.....	56
3.10.1	Types of Keystroke Dynamics.....	58
3.10.2	Types of Trackers and Measures for Keystroke Dynamics.....	60
3.10.3	Leniency.....	61
3.10.4	Imposing Restrictions on Keystroke Dynamics.....	63
3.11	Summary of Keystroke Dynamics Components.....	65
3.12	Suggestion for Keystroke Dynamics Solution.....	66
3.13	Perceived Keystroke Dynamics Limitations Resolved.....	66

3.13.1	Typing Inconsistencies.....	66
3.13.2	Non-permanent Authentication.....	67
3.13.3	Reusing Recorded Typing Styles.....	68
3.13.4	Keyboard Layout.....	68
3.13.5	Physical State Influence on Keyboard Interaction.....	69
3.13.6	Alternative User Entry.....	69
3.14	Entropy of the Proposed Solution.....	70
3.14.1	Measuring Password and Passphrase Strength.....	70
3.14.2	Shannon Entropy Theory.....	71
3.14.3	Application of Shannon Entropy Theory.....	71
3.14.4	Entropy of Keystroke Dynamics.....	77
3.15	Summary of the Proposed Security Model.....	79
3.16	Conclusion.....	80
Chapter 4	– MEMORISATION.....	81
4.1	Introduction.....	82
4.2	Definition of Chunking.....	84
4.3	Memory Components.....	90
4.3.1	User.....	90
4.3.2	System.....	100
4.3.3	System Development Team.....	106
4.3.4	Password Composition.....	111
4.4	Passphrase and Keystroke Dynamics Memorisation Usability Model.....	116
4.5	Conclusion.....	117
Chapter 5	– TYPING.....	118
5.1	Introduction.....	119
5.2	Keystroke-level Model.....	120
5.3	Keystroke-level Model Application.....	120
5.4	Keystroke-level Model Scope.....	121
5.5	Usability Impact on Typing Different Password Types.....	122
5.6	Types of Typographical Errors.....	123
5.7	User Authentication Process.....	124
5.7.1	User Authentication from a User Perspective.....	124
5.7.2	User Authentication from a System Perspective.....	126
5.8	Typing Issues Affecting Usability.....	129
5.8.1	Association Between Errors and Keys Pressed.....	129

5.8.2	Association Between Errors and Typing Duration/Speed	130
5.8.3	Impact of Routine Tasks and Non-routine Tasks on Typing	131
5.8.4	Language Deviations: Punctuation and Grammar	133
5.8.5	Typing Effort.....	135
5.8.6	Hidden or Unhidden Text	136
5.8.7	Keyboard Exposure.....	137
5.9	Passphrase and Keystroke Dynamics Typing Usability Model.....	140
5.10	Conclusion	141
Chapter 6	– PROPOSED MODEL CONSTRUCTION	142
6.1	Introduction	143
6.2	Construction Approach for the Proposed Model	143
6.3	Literature Chapter Models.....	145
6.3.1	Proposed Two-tier User Authentication Security Model.....	146
6.3.2	Proposed Two-tier User Authentication Memory Usability Model	147
6.3.3	Proposed Two-tier User Authentication Typing Usability Model.....	147
6.4	Development of the Proposed Model	148
6.5	Part Model Considerations	150
6.5.1	Constructs.....	150
6.5.2	Associations.....	150
6.5.3	States.....	151
6.5.4	Events.....	151
6.6	Whole Model Considerations.....	152
6.6.1	Importance	152
6.6.2	Level	152
6.6.3	Novelty.....	153
6.6.4	Parsimony.....	153
6.6.5	Falsifiability	154
6.7	Proposed Model	154
6.8	Constructing the Proposed Model	155
6.8.1	Part Model Considerations	156
6.8.2	Whole Model Considerations	157
6.9	Conclusion	158
Chapter 7	– OVERARCHING THEORIES	160
7.1	Introduction	161
7.2	Theoretical Frameworks.....	161

7.3	Related Work on the AAA Framework and ISO 9241 Standards	162
7.3.1	Related Work on AAA Framework.....	162
7.3.2	Related Work on ISO 9241	163
7.4	Components of the AAA Framework and ISO 9241 Standards	164
7.4.1	AAA Framework Components	164
7.4.2	ISO 9241 Components.....	165
7.5	AAA Framework	165
7.5.1	Security – Authorisation	166
7.5.2	Security – Authentication.....	166
7.5.3	Security – Accounting	167
7.6	ISO 9241	167
7.6.1	Usability – Effectiveness	168
7.6.2	Usability – Efficiency	168
7.6.3	Usability – Satisfaction	169
7.7	Theories Summary	169
7.8	Research Propositions	171
7.9	Conclusion	172
Chapter 8 – FINDINGS AND DICUSSION		174
8.1	Introduction	175
8.2	Login Assessment Data Collection Approach.....	175
8.3	Login Assessment Data Analysis Approach	176
8.3.1	Data Analysis Process	177
8.3.2	Data Analysis Scoping	177
8.3.3	Constructs Evaluation	180
8.4	Login Assessment Results and Discussion	182
8.4.1	Security Evaluation	182
8.4.2	Memory Evaluation	185
8.4.3	Typing Evaluation.....	197
8.4.4	General Observations	204
8.4.5	Summary of Results.....	221
8.5	Model Updates	223
8.5.1	Updated Security Sub-model	224
8.5.2	Updated Memory Sub-model	225
8.5.3	Updated Typing Sub-model.....	226
8.5.4	Updated Proposed Model.....	227

8.6	Conclusion	228
Chapter 9 – MODEL EVALUATION		229
9.1	Introduction	230
9.2	Expert Review Approach.....	230
9.2.1	Expert Review Data Collection Approach.....	230
9.2.2	Expert Review Data Analysis Approach	232
9.3	Expert Review Results and Discussion	233
9.3.1	General Consensus	233
9.3.2	Security Feedback	234
9.3.3	Usability Feedback.....	235
9.3.4	Expert Suggestions.....	239
9.4	Research Model.....	241
9.4.1	Research Model Roadmap.....	241
9.4.2	Finalised Research Model.....	243
9.5	Conclusion	244
Chapter 10 – CONCLUSION.....		245
10.1	Introduction	246
10.2	Problem Area	246
10.3	Research Contribution.....	246
10.4	Theoretical Foundation.....	247
10.5	Research Questions.....	248
10.6	Research Design and Methodology.....	252
10.6.1	Problem Relevance.....	252
10.6.2	Design as a Search Process	252
10.6.3	Research Rigor	253
10.6.4	Design as an Artefact.....	253
10.6.5	Design Evaluation	253
10.6.6	Research Contributions.....	255
10.6.7	Communication of Research	255
10.7	Limitations of the Study	255
10.8	Future Research	256
10.9	Information Security Policy Recommendation	257
10.10	Concluding Summary	258
References.....		260
Appendix A.....		288

Appendix B.....	295
Appendix C	297

List of Figures

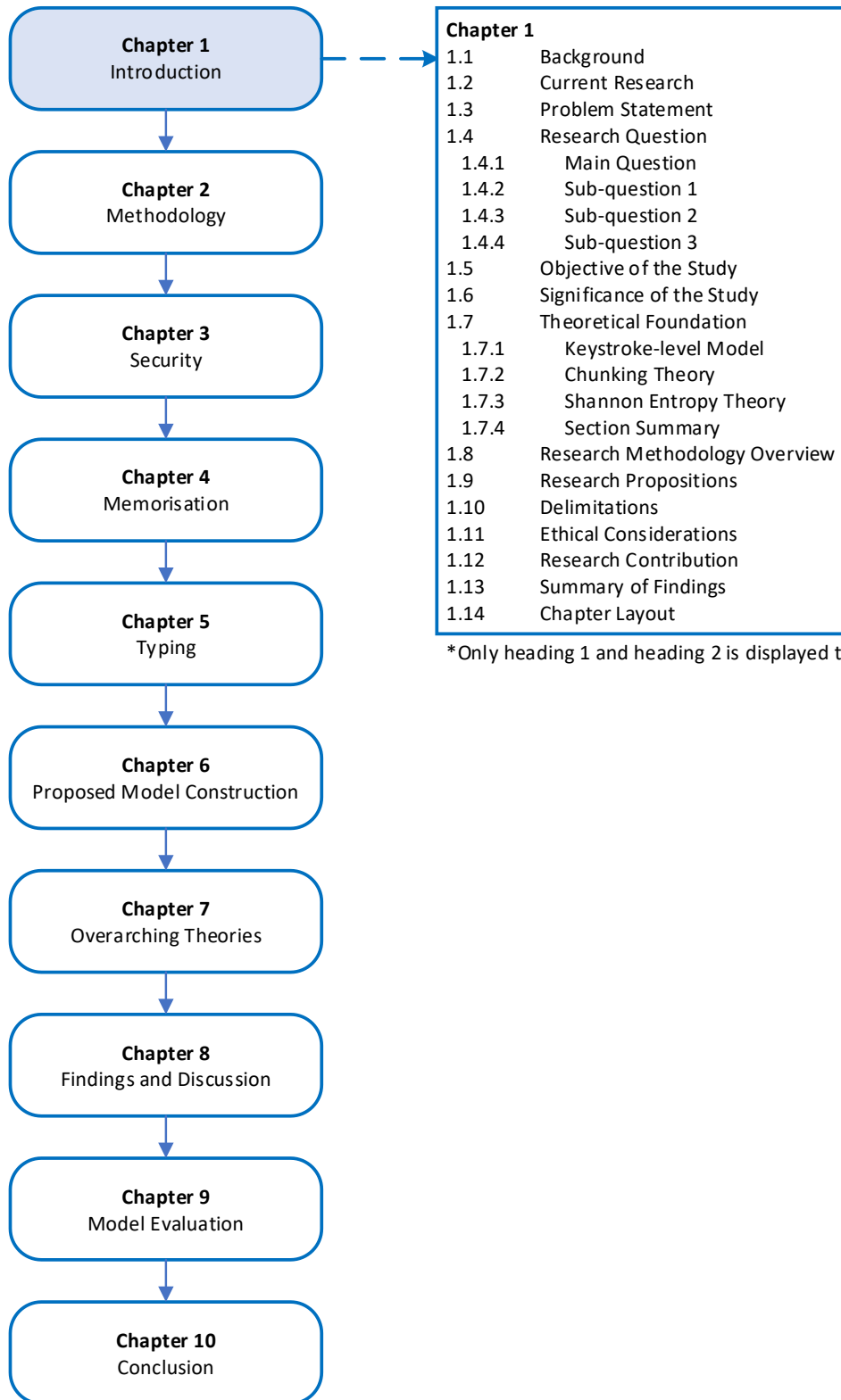
Figure 1-1: Typical Text-based User Authentication Process	3
Figure 1-2: Shannon Entropy Formula.....	11
Figure 1-3: Research Process.....	12
Figure 2-1: Information Systems Research Framework.....	20
Figure 2-2: Research Process.....	24
Figure 2-3: Participant Login Assessment Process.....	28
Figure 2-4: Login Assessment Screen Designs	29
Figure 2-5: Expert Review Process	31
Figure 3-1: Components of Passphrases	55
Figure 3-2: Mechanics Behind Keystroke Dynamics.....	57
Figure 3-3: Components of Keystroke Dynamics.....	65
Figure 3-4: Shannon Entropy Formula.....	71
Figure 3-5: Entropy of a Four-digit PIN.....	72
Figure 3-6: Entropy of a Passphrase	73
Figure 3-7: Entropy of Passwords	75
Figure 3-8: Proposed Two-tier User Authentication Security Model.....	79
Figure 4-1: Chunk Size, Numbers and Evolution.....	86
Figure 4-2: Passphrase and Keystroke Dynamics Memory Model.....	116
Figure 5-1: Research Summary.....	119
Figure 5-2: Typical Login Process from a User Perspective	124
Figure 5-3: Typical Two-tier Login Process Using OTP as the Second Tier.....	125
Figure 5-4: Typical Login Process from a User-System Perspective	126
Figure 5-5: Typing Usability Considerations for Proposed Two-tier Solution.....	140
Figure 6-1: Construction Approach of Proposed Model	145
Figure 6-2: Proposed Two-tier User Authentication Security Model.....	146
Figure 6-3: Proposed Two-tier User Authentication Memory Usability Model.....	147
Figure 6-4: Proposed Two-tier User Authentication Typing Usability Model.....	148
Figure 6-5: Model Assessment Considerations	149
Figure 6-6: Research Length vs Research Breadth.....	153
Figure 6-7: Proposed Model: Passphrases and Keystroke Dynamics.....	155
Figure 7-1: Guiding Theories.....	170
Figure 8-1: Login Assessment Website Screen Flow.....	176
Figure 8-2: Security Sub-model.....	178
Figure 8-3: Memory Sub-model.....	179

Figure 8-4: Typing Sub-model	180
Figure 8-5: Strength Indicators	184
Figure 8-6: Phonologically Similar Words	187
Figure 8-7: Password and Passphrase Length and Login Failures	189
Figure 8-8: User-generated Passphrases	191
Figure 8-9: Simple Username	192
Figure 8-10: Lenient Password Policy	193
Figure 8-11: Strength Indicator and Password Policy Alignment.....	194
Figure 8-12: Application of Chunking.....	196
Figure 8-13: Unfamiliar Sequence	197
Figure 8-14: Number of Keystrokes.....	199
Figure 8-15: SMS Language, Acronyms and Abbreviations.....	201
Figure 8-16: Password and Passphrase Entropy.....	205
Figure 8-17: Password and Passphrase Complexity	206
Figure 8-18: Similar Passwords and Passphrases Created by Users	207
Figure 8-19: Improvement Over Time.....	208
Figure 8-20: Days between Logins	210
Figure 8-21: Login Time Period	211
Figure 8-22: Password and Passphrase Rests.....	213
Figure 8-23: User Gender.....	217
Figure 8-24: Password and Passphrase Failures	218
Figure 8-25: Language in Passwords and Passphrases	219
Figure 8-26: User Preference	220
Figure 8-27: Updated Security Sub-model.....	224
Figure 8-28: Updated Memory Sub-model.....	225
Figure 8-29: Updated Typing Sub-model.....	226
Figure 8-30: Updated Proposed Model.....	227
Figure 9-1: Expert Review Process	232
Figure 9-2: Research Model Roadmap.....	242
Figure 9-3: Research Model	243

List of Tables

Table 2-1: Differences in Behavioural Sciences and Design Sciences	21
Table 2-2: Aspects of the Login Assessment.....	34
Table 3-1: Types of Knowledge-based Authentication.....	48
Table 3-2: Password and Passphrase Defined.....	50
Table 3-3: Types of Keystroke Dynamics	59
Table 3-4: Entropy of Keystroke Dynamics.....	78
Table 4-1: Number of Chunks	89
Table 4-2: Examples of Meaningful Datasets	93
Table 4-3: Relationships Between Password Policy and Strength Indicators.....	108
Table 5-1: Number of Keys Pressed.....	131
Table 7-1: Main Theory Foundation.....	171
Table 7-2: Research Propositions	172
Table 8-1: Construct and Proposition Evaluation.....	181
Table 8-2: Passphrase Dictionary.....	186
Table 8-3: Common Segments Across Passwords and Passphrases	190
Table 8-4: Keyboard Visualisation and Touch	202
Table 8-5: Keyboard Layout Exposure	203
Table 8-6: User Age and Failure Types.....	215
Table 8-7: User Age and Login Device Failures	216
Table 8-8: Summary of Results	221
Table 10-1: Chapter Sections that Addressed the Respective Research Questions ..	250

Chapter 1 – INTRODUCTION



*Only heading 1 and heading 2 is displayed to avoid clutter

1.1 Background

At the forefront of user security is user authentication (Cheng, Yang, Shao, & Liao, 2015). User authentication refers to the security methods used to verify a user before granting him/her access to secured content on a system. Several user authentication methods exist such as text (passwords and passphrases), images (clicking on specific areas of an image/images in the correct sequence), audio/voice recognition and physical scans (face, fingerprint and retinal recognition). The robustness and variety of user authentication methods have increased over time to a point where many systems utilise two-tier authentication (also known as two-factor authentication) to validate the authenticity of the user (Adham, Azodi, Desmedt, & Karaolis, 2013; Reese, et al., 2019). Two-tier or two-factor authentication means that two methods of authentication are required before a user can be granted access to the system (Milton, Ramakrishnan, & Das, 2016). This study referred to it as two-tier authentication. All methods of authentication can be classified under one of the following categories. These are discussed in more detail in Chapter 3, Section 3.4, Forms of Authentication:

- What you know: e.g. passwords
- What you have: e.g. access card and one-time pin (OTP)
- Who you are: e.g. fingerprint

It is important to note that text-based authentication will continue to be used in the future (Wang, He, Haibo, & Ping, 2016). This may mean that it is likely that text-based authentication will be used in conjunction with another authentication method.

This study focused on the use of a two-tier user authentication method where passphrases are used as the first tier of authentication, coupled with a keystroke authentication algorithm as the second tier of authentication. A passphrase can be defined as a sentence or phrase that is used as a password but does not include any uppercase, special or numeric characters. However, some researchers (Ayyagari, Lim, & Hoxha, 2019; Chethan, Siddappa, & Jayanna, 2020; Loos, Ogawa, & Crosby, 2019) defined a passphrase as a sentence that can incorporate uppercase, special or numeric characters. Chapter 3, Section 3.5, Passwords and Passphrases Defined, explains the specific characteristics of a passphrase. For the purpose of this study, a passphrase is referred to as a certain type of text-based authentication and a password is referred to as a conventional password that uses multiple character sets such as lowercase letters, uppercase letters, digits and special characters (also referred to as LUDS). Passwords and passphrases are grouped into a larger form of authentication which is referred to as

text-based authentication. The keystroke authentication algorithm (also referred to as keystroke dynamics) is a backend solution that records the keystroke patterns (using time as a metric) of the user when he/she inserts their text-based authentication into the system. This pattern is then used to validate the user's identity. Keystroke dynamics is classified as a behaviours biometric form of authentication. More information on behavioural biometrics can be found in Section 3.4, Forms of Authentication.

The study assessed security and usability in two-tier authentication using passphrases and keystroke dynamics to determine whether it can be used as an acceptable user authentication method. Usability is defined as a user's ease of use in executing system inputs and receiving the correct system outputs (Hornbæk & Law, 2007; Hsu, Lee, Quek, & Chen, 2018). In relation to this study, lack of usability refers to any negative user experiences during the user authentication phase of system interaction. For example, any additional or unnecessary time or effort spent by the user may create a negative user experience. Figure 1-1 below illustrates a typical text-based user authentication process (one/single-tier authentication) for most web-based systems. The "user experience" grouped shapes in Figure 1-1 are typically mandatory for user authentication.

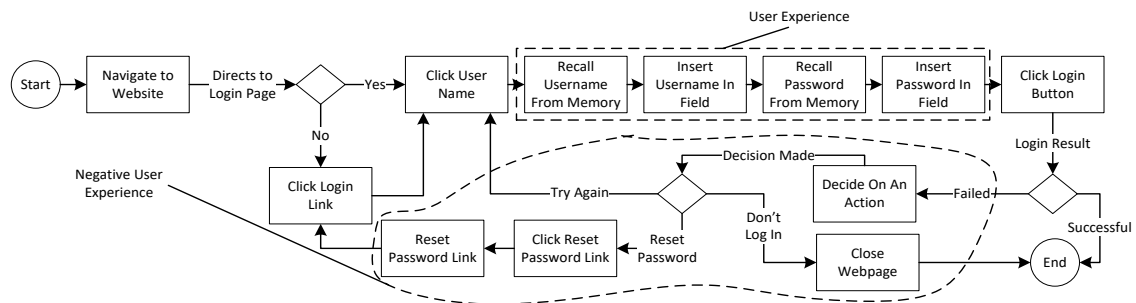


Figure 1-1: Typical Text-based User Authentication Process – Adapted from: (Dib & Ghazi, 2019; Keith, Shao, & Steinbart, 2009)

From a user perspective, the aim is to execute the processes which affect "user experience" (in Figure 1-1) as effortlessly and as quickly as possible (Wang, Lymberopoulos, & Liu, 2015). If the user fails to accomplish this, then the usability of the system is negatively affected. From a system development team perspective, their concern is more focused on the strength of the text-based authentication chosen by the user to ensure system security (Dib & Ghazi, 2019; Payne & Edwards, 2008). For the purpose of this study, a system development team will be the stakeholders used when referring to the stakeholder who designs the system, defines the password policy for the system, defines the information security policy (for business systems), develops the

system or updates the system. Although this role is usually fulfilled by the system development team on small projects or systems, larger organisations may have various stakeholders who are responsible for the security of the system. For example, a system administrator, a security analyst, a security policy administrator and/or a network architect.

Failure by the user to complete the processes that affect “user experience” in Figure 1-1 will result in the user having to follow the processes that trigger “negative user experience”. This will have a negative impact on the usability of the system. Consequently, most users will create a password that is easy to recall from memory and is not time-consuming to insert into the system (Keith et al., 2009). This has resulted in users creating weak passwords (Braunstein, 2015; Parsons, McCormac, Butavicius, & Ferguson, 2010; Renaud, 2019). A weak password in this context is defined as a password that is easy for another person to guess or to identify, using decryption software. Existing research aligned to this study is provided in the next section.

1.2 Current Research

To address the issue of users creating weak passwords, password policies have been imposed on user authentication (Keith et al., 2009; Kelley et al., 2012; Payne & Edwards, 2008; Renaud, 2019; Shay et al., 2014). Password policies can be explained as specific rules that a password must adhere to in order to be accepted by the system as a secure password (Shay et al., 2014). An example of a password policy is as follows: *The password created must contain one uppercase character, one special character and one numeric character* (Hussain, Atta, Bawany, & Qamar, 2018; Kelley et al., 2012; Shay et al., 2014). Password policies are intended to force the user to create a strong password that cannot be easily determined (manually through phishing or through brute force software attacks) by an unauthorised party. Logically, if passphrases and PINs (Personal Identification Number – i.e. numeric passwords) are accepted by a password policy, such policy should not be referred to as a password policy, as passphrases and PINs are not passwords. However, since the term “password policy” is popularly used across disciplines and people, this study will maintain the use of the term to refer to the rules governing the creation of a password, passphrase or PIN. In organisations a password policy forms part of the information security policy. An information security policy defines the governance and rules required to protect against unauthorised entry into the system.

Password policies vary across systems (Houshmand & Aggarwal, 2012; Kelley et al., 2012; Maoneke & Flowerday, 2019). Systems protecting important information tend to have more stringent password policies than others. For example, Facebook has a more lenient password policy than an online banking system. This is because most users are likely to experience less loss if an unauthorised party gains access to their Facebook account as opposed to their banking account, assuming that financial damage is greater than reputational damage in this example. The reason why stringent password policies are not assigned to all systems is that such password policies often have a negative impact on the user in terms of system usability.

Usability may result (but is not limited to) two possible scenarios: 1) The user may avoid using the system, or 2) the user may resort to unacceptable security behaviours to ensure that the password is not lost, for example writing the password down on a piece of paper (Kelley et al., 2012; Melicher et al., 2016). Therefore, when developing a system with a password policy, the security of the system should be considered as well as the possibility that users may avoid using the system due to password complexity. In other words, the benefits of using the system (desired purpose of using the system) should outweigh the costs (difficulty in creating a password that can be memorised and inserted into the system). This statement is supported by the research of Adams, Sasse, and Lunt (1997) and Stanton, Stam, Mastrangelo, and Jolton (2015). By incorporating a keystroke dynamics algorithm (as a second tier of authentication, the first tier being a passphrase), it may be possible to reduce the stringency of password policies, thus improving usability and increasing security.

This study focused on understanding the issues experienced by users and the system development team in the user authentication process. Issues include creating a password/passphrase that is difficult to remember, memorising the password/passphrase, and inserting the password/passphrase into the system. By addressing these issues through the adoption of passphrases and a keystroke dynamics algorithm, both system usability and system security may increase. The system development team can include passphrases in the password policy and information security policy and accommodate any reduction in security through the implementation of a keystroke pattern algorithm. Simultaneously, users can use passphrases to assist user memorisation and address the typing constraints created by current password policies. The next section explains the problem statement for this research study.

1.3 Problem Statement

Two interrelated problems exist which is discussed in this section. Strict system security protocols such as password policies have resulted in 1) user frustration in terms of memorising passwords, and 2) further user frustration in terms of typing the password. Current user authentication security measures focus more on system security with little consideration being given to the impact on usability. However, this has been tolerated by users as these measures protect them from security breaches such as having their passwords hacked. The two sub-problems are discussed in more detail below.

1. The enforcement of password policies and an information security policy by the system development team is intended to force users to create strong passwords in order to restrict unauthorised entry to a system. Although this has proven to be effective (Alomari & Thorpe, 2019; French, 2012; Shay et al., 2014), it has created negative implications for the usability of the system. These negative implications have resulted in unsecure actions/behaviour by users. For example, users have found it difficult to memorise the password created due to the stringent rules imposed by password policies. This has resulted in users creating common passwords which are easy to guess (Braunstein, 2015; Parsons et al., 2010). Some users even write the password down on a piece of paper or capture them digitally in a notepad text file (Braunstein, 2015; Parsons et al., 2010). Choong, Theofanos, and Liu (2014) state that employees spend a total of 2.25 days within a 60 day period on password-related activities. This includes activities such as inserting the password into the system and changing the password when necessary, including periodic password changes or using the “forgot password” option. In addition to memory issues that affect usability and security, typing is a major factor in password authentication usability.
2. The measures that affect usability in terms of typing the password include the amount of effort and time required to execute the action (Carstens, Mccauley-Bell, Malone, & Demara, 2014; Parsons et al., 2010). In the context of this research, effort refers to the number of keystrokes required to execute an action and time refers to the duration of completing such action. This becomes troublesome when special characters and numbers (which are required by most password policies) need to be used (Keith et al., 2009; Maoneke & Flowerday, 2019; Shay et al., 2014). To add to this, different keyboard layouts and key sizes also have an impact on typing (Choi, Jeong, Woo, Kang, & Hur, 2019; Keith et al., 2009; Shay et al., 2014). Another factor that needs to be considered is that

when a password is typed incorrectly, effort and time are affected as the user needs to repeat the action of inserting the password.

In summary, current research has failed to find a method which satisfies system security and system usability concerning passwords. This research study seeks to improve usability by allowing password policies to be more lenient by proposing the use of passphrases. This will allow users to create a text-based authentication passphrase that is easier to type and memorise than a conventional password. Simultaneously, the study attempted to improve security by introducing a keystroke dynamics algorithm, as a secondary tier of authentication, which has low to no impact on usability but strengthen security. The next section provides the research question.

1.4 Research Question

The research question, which is provided below, is broken down into three sub-questions. These sub-questions follow the research question.

1.4.1 Main Question

How can a two-tier user authentication solution involving passphrases and keystroke dynamics improve system usability without compromising system security?

The main research question was answered through the construction of a model. The model was constructed by addressing the following sub-questions.

1.4.2 Sub-question 1

What needs to be considered when ensuring the security of passphrases and a keystroke dynamics algorithm as a method of user authentication?

The strength of passwords is measured by the ability of an unauthorised entity to crack the password. To answer this sub-question, an understanding of what makes text-based authentication strong firstly needs to be identified. These metrics were then applied to passwords and passphrases to determine which is stronger in terms of security. The keystroke dynamics algorithm was also applied where possible (i.e. only text passwords) to determine the increase in security provided by this second tier of authentication. The intention is to determine how secure passphrases and keystroke dynamics are in comparison to conventional passwords. Once security is determined, usability is discussed, which leads to sub-question 2.

1.4.3 Sub-question 2

What factors in terms of system usability influence the memorisation of passphrases and may impact a keystroke dynamics algorithm?

The problem experienced by all non-biometric user authentication methods is the user's ability to recall the password from memory. This sub-question was addressed by understanding how users create and memorise passwords before determining how easy it is to memorise passphrases as opposed to conventional passwords. An assessment was also conducted to determine whether these factors were influenced by the keystroke dynamics algorithm. The second usability component to consider is the user's ability to input the password in the system.

1.4.4 Sub-question 3

What system input factors influence the use of passphrases and may impact the keystroke dynamics algorithm?

A user needs to remember his/her password. If they succeed in this, the next step is to ensure that the user can insert his/her password into the system as effortlessly as possible. This sub-question was answered by understanding the difficulties experienced by users when typing their password into the system. Once these difficulties are identified, the second part of answering this sub-question focused on determining how passphrases can address these difficulties, where possible. An assessment had to be conducted to determine whether the factors identified are influenced by the implementation of the keystroke dynamics algorithm.

Answering these three sub-questions assisted in addressing the main research question. The next section discusses the objectives of the study.

1.5 Objective of the Study

A model (artefact) was developed for the system development team to improve the usability and security of the password user authentication process. This primary objective is divided into smaller secondary objectives indicated below:

1. To determine how passphrases and keystroke dynamics meet the user authentication security requirements. This was addressed in the first sub-question.

2. To determine how passphrases and the keystroke pattern algorithm can address the usability issues experienced in the user authentication process. This was addressed in the second and third sub-question.
3. To develop and refine the two-tier authentication model, which simultaneously increases usability and security. This was done as an iterative approach through the research study.

Addressing the three sub-objectives allows for the development of a model which can be used to improve the usability of the password user authentication process while ensuring that the security of the system is not compromised. The model can then be used to improve the information security policy on websites, applications and organisations to allow the use of passphrases and implement a keystroke dynamics algorithm. The conclusion of the expressed objective introduces the significance of the study.

1.6 Significance of the Study

The significance of this study lies in the notion that the use of the two-tier user authentication model (research artefact) could lead to an increase in usability and an increase in security in the user authentication process. This was accomplished by improving the Information Technology (IT) security policy and user authentication process. As a result of the adoption of passphrases, system usability may increase, as users may need to spend less time and effort on the user authentication process (Banerjee & Woodard, 2012; Morimoto, Leyva, & Tula, 2018). Time and effort is reduced by decreasing the probability of the user forgetting the password created and having to either try again by retyping the password or select the “forgot password” option to send a request to the system to reset the password. In addition to user benefits, the system development team can increase security through the keystroke dynamics algorithm as a second level of authentication which has little or no negative impact on usability (Dutta, Madnick, & Joyce, 2016). The system development team is also able to reduce user security issues (Braunstein, 2015; Kelley et al., 2012; Melicher et al., 2016; Parsons et al., 2010). For example, if users are forced to create complicated passwords by a system’s password policy, users tend to record them on their computer or a piece of paper, or they reuse the password on multiple systems. In addition, the utilisation of keystroke dynamics can reduce certain instances of social engineering, as additional information on a user’s typing patterns will also need to be collected (Heartfield & Loukas, 2016). The next section focuses on the theoretical foundation of this research study.

1.7 Theoretical Foundation

This section provides a theoretical foundation for the study. Firstly, the Keystroke-level model is discussed, followed by the Chunking theory, and lastly, the Shannon Entropy theory. The Keystroke-level model and Chunking theory are both used to direct the research on usability in terms of the entering and memorisation of the password/passphrase.

1.7.1 Keystroke-level Model

The Keystroke-level model is a prediction tool used to measure the length of time taken by an expert user to execute a routine system task (John & Kieras, 1994; Jorritsma et al., 2015; Lee, Song, Ryu, Kim, & Kwon, 2015). In terms of this study, the executed routine task is the user authentication process. Since time has an impact on usability, the model was used to guide the discussion on the user's interaction with a typical login interface in terms of inserting the password/passphrase into the system.

1.7.2 Chunking Theory

The Chunking theory estimates the volume of content that an average person can hold in their short-term memory (Miller, 1956). An average person can remember three to five chunks of information (Cowan, 2010; Doumont, 2002). A chunk of information is referred to as a grouping of interrelated elements created by a person's associations attained through personal experience. The Chunking theory supports the efforts made in this study by supplementing the assumption that a user exerts more effort to memorise a password that adheres to stringent password policies as opposed to a password created from a more flexible password policy.

1.7.3 Shannon Entropy Theory

The Shannon Entropy theory quantifies the probability of the average number of attempts required to guess or determine the outcome. Since it is focused on discovering the number of yes/no questions that need to be asked before determining the answer, the units used to measure the strength of passwords/passphrases are in bits (i.e. binary). The number of bits in this theory is referred to as entropy. The higher the entropy, the greater the uncertainty of guessing the correct answer. Therefore, a higher entropy password/passphrase is more secure than a lower entropy password/passphrase. It should be noted that although the theory presents a numeric value to indicate the strength of a password/passphrase, it is actually presenting the best guess of the

password's/passphrase's strength, based on probability. The Shannon Entropy formula is presented in Figure 1-2 below.

$$H = - \sum p(x) \log p(x)$$

Figure 1-2: Shannon Entropy Formula (Shannon, 1948)

In Figure 1-2 above, p is the probability of making the correct selection from a known range and x is the total number of available options in the range. This formula was used to measure passwords/passphrases, where $p = 1$ and $x =$ the total number of character ranges that the user can select. The formula also considered the keystroke dynamics algorithm and its level of impact on security. The next section provides a summary of the theoretical foundation section.

1.7.4 Section Summary

The theoretical foundation for this research has assessed whether the proposed two-tier authentication solution does address the required security and usability concerns. Accordingly, the Shannon Entropy theory was used to assess whether the proposed solution addressed the security concerns, while the Keystroke-level model (for typing) and Chunking theory (for memorisation) was used to determine whether the proposed solution addressed the usability concerns. The next section presents the research methodology for this study.

1.8 Research Methodology Overview

Design science was selected after discovering that the approaches used by similar studies in this field (De Ru & Eloff, 1997; Keith et al., 2009; Nikora, Hunt, & Ryan, 2018; Shay et al., 2014) are aligned to the design science methodology. Design science is a problem-solving paradigm which “seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently accomplished” (Hevner, March, Park, & Ram, 2004, p. 76). The design science methodology consists of seven guidelines, all of which must be considered to apply this research paradigm effectively (Hevner et al., 2004). The seven design science guidelines are listed below.

1. Problem relevance
2. Design as a search process
3. Research rigour

4. Design as an artefact
5. Design evaluation
6. Research contributions
7. Communication of research

By considering the seven design science guidelines, a high level of confidence was assured in terms of the robustness of this research study. This also ensured that credible conclusions were reached. The next section refers to the way the data for this study was collected.

Figure 1-3 below graphically depicts how this research was conducted by considering the seven design science guidelines. The numbers between one and seven indicate the design science guideline followed for the collection and/or analysis of the respective activity. To ensure efficient utilisation of time, primary data and secondary data collection and analysis were conducted in parallel where possible (see Figure 1-3). As illustrated in Figure 1-3 by the arrows moving across primary data and secondary data, each segment of data collection and analysis was merged together to either support or contradict each other. A mixed methods approach was chosen for this study which included a login assessment experiment (quantitative data) and an expert review (qualitative data). The next section presents the delimitations of this study.

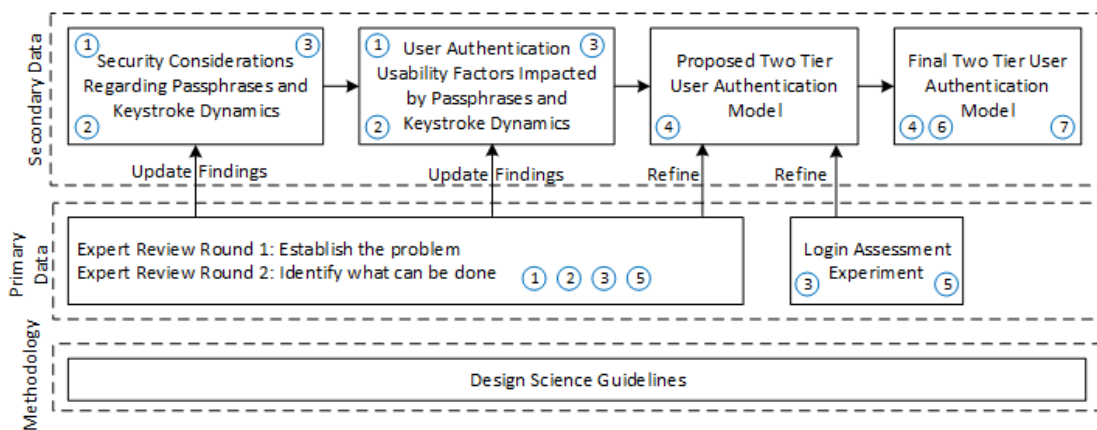


Figure 1-3: Research Process

More details of how this research was conducted can be found in Chapter 2. The next section provides this study's research propositions.

1.9 Research Propositions

Five research propositions were formulated based on the research undertaken while assessing whether the proposed solution addresses the research problem. The following research propositions were formulated:

1. Passphrases are more secure than passwords.
2. Keystroke dynamics supports passphrases better than passwords.
3. Passphrases are easier to remember than passwords.
4. Passphrases are easier to type than passwords.
5. Keystroke dynamics increases security and has little to no negative impact on usability.

More detail on the above research propositions can be found in Section 7.8, Research Propositions. The next section provides the delimitations of this study.

1.10 Delimitations

This study focused specifically on passwords and passphrases. Subsequently, the keystroke dynamics authentication was the only form of second-tier authentication that was explored. Although there are a number of methods for assessing password/passphrase strength, the Shannon Entropy theory was the only metric used as this theory was deemed to be best suited to this research study. In terms of usability, only short-term memory was assessed due to research time limitations.

The login assessment experiment was web-based using a convenient sample. When conducting the login assessment, participants were asked not to make use of any password/passphrase recall features on the browser or any other password/passphrase storage mechanism, including writing the password/passphrase down on paper. Since it is difficult to ensure this without inconveniencing participants, the study assumed that participants did comply with the rules of the login assessment when they conducted the assessment. To ensure this, participants had to state that they understood and would comply with all the rules of the login assessment. For the sake of convenience, all participants in the login assessment resided in South Africa. This did not affect the sample representation of the population (Etikan & Bala, 2017). It is also important to note that keystroke dynamics was not assessed using the login assessment experiment due to resource and time constraints.

The boundaries of this study have been emphasised through delimitations. The next section covers the ethical considerations of the study.

1.11 Ethical Considerations

Babbie (2005) states that it is the author's responsibility to comply with academic integrity and honesty, as well as to respect other people. Primary data collection efforts for this study only began once approval had been obtained from the Rhodes University Ethics Committee. Arifin (2018) and Punch (2006) lists categories of ethical issues which should be considered by researchers. Informed consent (Punch, 2006) was obtained by sharing all necessary information about the research with all potential participants before requesting participation. Participants were guaranteed anonymity (Punch, 2006) by constructing any questions posed to participants in a manner which ensured that no personal information was captured. The misuse of results (Punch, 2006) was mitigated by ensuring that information gathered from participants would only be used for the express purpose of this study. Holloway and Wheeler (1995) put strong emphasis on researchers having to ensure that their research is non-maleficent. To ensure non-maleficence in this study, users participating in the login assessment experiment were asked to create a new password and passphrase; one that they were not currently using or had used in the past for any system. This ensured that any passwords and passphrases collected could not be used to gain unauthorised access to participants' accounts. Participants were also allowed to refuse participation in the study at any time. It should also be noted that the information collected from participants was used for research purposes only. Accordingly, all these ethical issues were considered and the research complied in all respects. The next section provides the research contribution for this study.

1.12 Research Contribution

This section provides the finalised model after all the research was conducted. The model can be used to understand what needs to be considered to implement the two-tier user authentication solution. The model also assists in showing how the information security policy needs to be updated to support the two-tier user authentication solution. It is suggested that a passphrase character length of 16 to 18 is best suited to reduce the risk of login failure due to typing or memorisation.

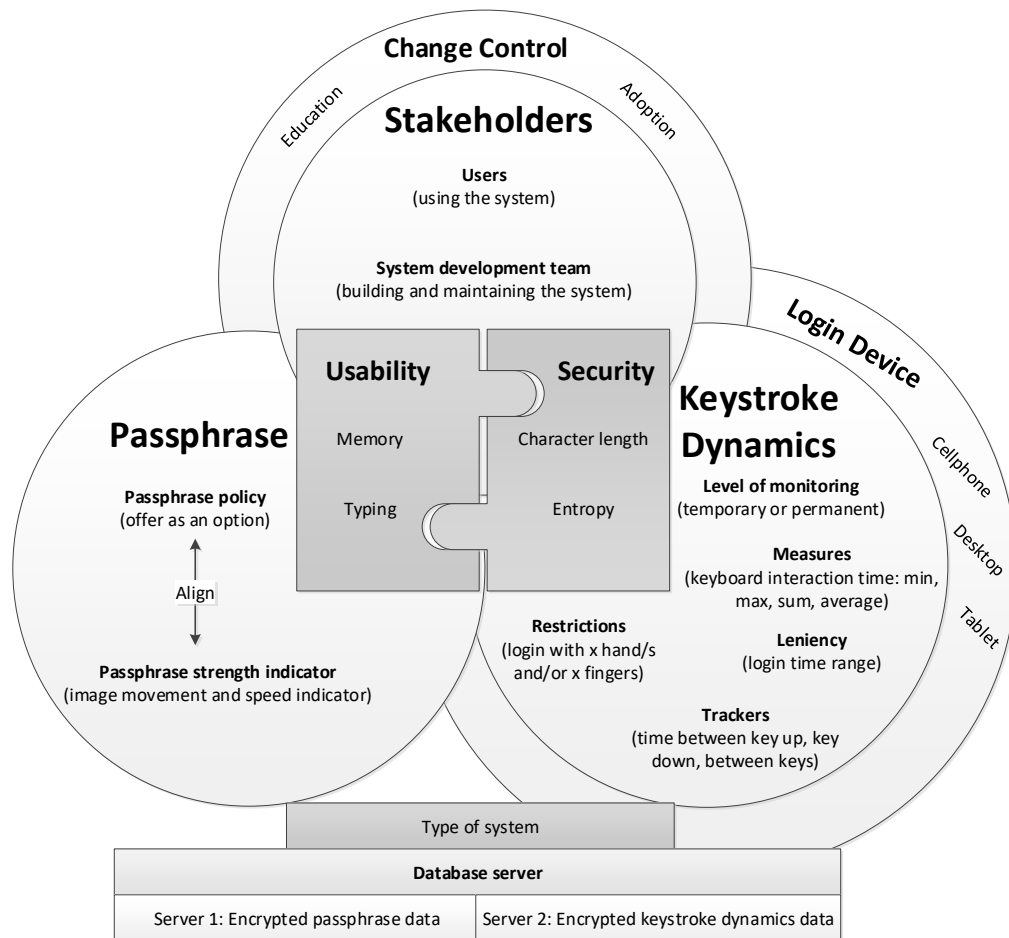


Figure 1-4: Research Model

The process of constructing and updating the research model is provided in Figure 1-5.

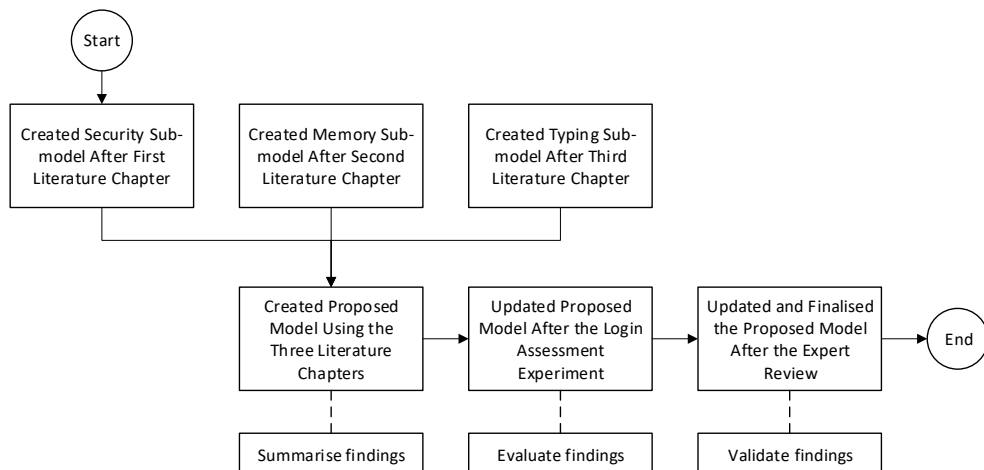


Figure 1-5: Research Model

Figure 1-5 illustrates that a sub-model was created at the end of each literature chapter. The sub-models were then used to create the proposed model. After the login

assessment experiment (i.e. findings were evaluated), the proposed model was updated based on findings from the data collected. However, it was still termed a proposed model as the model may need to be updated further after the expert review. After the expert review feedback was used to update the proposed model (i.e. findings were validated), it was termed the final model (see Figure 1-4). The next section of this chapter provides a summary of the findings.

1.13 Summary of Findings

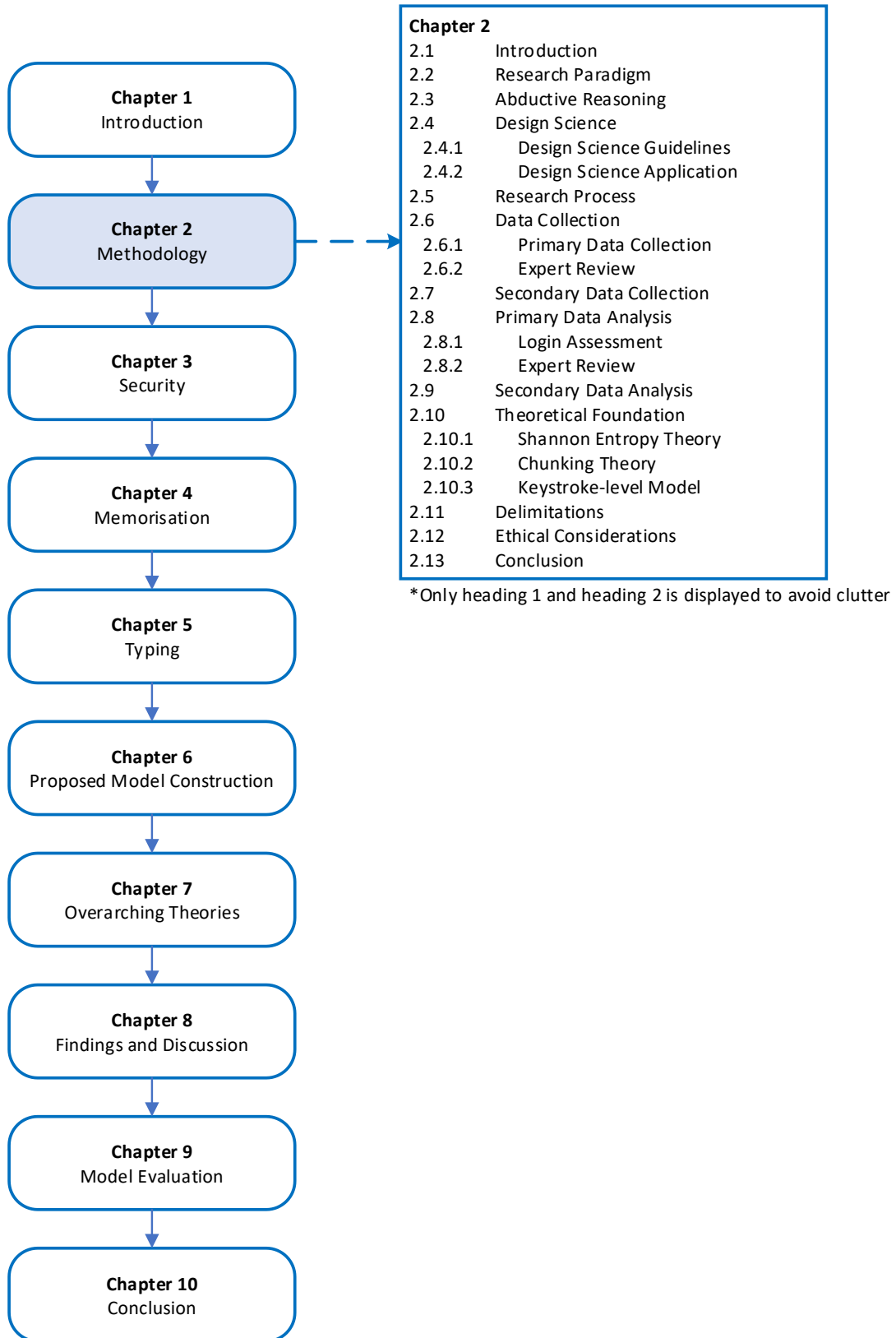
This study developed a model to assist in the implementation of a two-tier user authentication solution involving passphrases and keystroke dynamics. The literature review conducted in Chapters 3-5 assisted in developing the first draft of the model. This model was then evaluated by means of a login assessment experiment. Once the model had been updated, an expert review was conducted to validate the updated model.

It was subsequently found that the proposed solution could increase security and usability. However, a number of factors need to be considered when implementing the proposed solution. These include the fact that, for passphrases and keystroke dynamics, certain tools can be manipulated to improve security and/or usability. The last section of this chapter provides a breakdown of the chapters for this study.

1.14 Chapter Layout

Chapter 1 provides a summary of the study in terms of how the research was conducted. Chapter 2 focuses on the research methodology applied to this study. Chapter 3 is focused on measuring the strength of passwords and passphrases to address the security aspect of the user authentication process. Chapters 4 and 5 are devoted mainly to discussing the usability aspect of the study. Hence, Chapter 4 specifically focuses on the memorisation of passwords and passphrases and Chapter 5 on the user's ability to insert a password and passphrase into a system. Chapter 6 discusses the construction of the proposed research model. Chapter 7 maps the way in which the theories correlate with each other to drive the findings of this study. Chapter 8 includes a discussion on the findings of the expert review and the login assessment, as well as how these support the findings discussed in the literature chapters (Chapters 3, 4 and 5). Chapter 9 included the recommendations and focused on the construction and assessment of the artefact. The last chapter, Chapter 10, concluded the study. Chapter 2 shows the research approach taken for this study.

Chapter 2 – METHODOLOGY



2.1 Introduction

This chapter focused on the research methodology used for this study. The methodology in question was selected after gaining a thorough understanding of the research problem and the approaches used by other researchers (Geerts, 2011; Keith et al., 2009; Patas, Milicevic, & Goeken, 2011) who have addressed similar research problems. Such understanding was required because there are many ways of addressing a problem and reaching a conclusion (Hofstee, 2006; Poth, 2018). Hence, these preparatory efforts allowed for the most appropriate methodology to be selected.

The research paradigm is firstly discussed in this chapter which is followed by an explanation on abductive reasoning. Design science is then discussed. The research process for this study is then provided which is followed by the data collection techniques used for attaining primary data and secondary data. The data analysis techniques are then provided for primary data and secondary data. The theoretical foundation for this study is then discussed. Delimitations are then provided which is followed by a section on ethical considerations. The last section provides a conclusion for the chapter.

2.2 Research Paradigm

Two extremes exist when conducting research – interpretivism and positivism. Positivist research is focused on scientific explanations of events and results. Findings are usually objective and allow conclusions to be generalised. Interpretivism research is focused on a number of different interpretations of findings. Facts usually don't exist as research is conducted in a subjective manner.

This study took a pragmatic approach as this research assessed whether the overarching proposed solution can address a specific problem. In terms of this study, the proposed solution and hence the problem is as follows: A two-tier user authentication model which includes keystroke dynamics and passphrases can be used to increase system security and system usability in user authentication. The security aspect of this study leans more towards positivist research while the usability aspect of the study is focused more towards interpretivism research.

2.3 Abductive Reasoning

Research can be conducted from a number of different views. This influences the way data is analysed and how conclusions are drawn (Fellows & Liu, 2015; Hammond & Wellington, 2013; Walliman, 2011). Research views are commonly referred to by

researchers as "reasoning"; namely, inductive reasoning, deductive reasoning or adductive reasoning. There are clear differences between each of these views, which are indicated below:

- **Inductive reasoning** – The findings support the probable truth of the conclusion and therefore if the findings are realised, it is unlikely, but not certain, that the conclusion will be false (Duran & Şentürk, 2019; Hammond & Wellington, 2013; Walliman, 2011). For example; generalisations made from past events or experiences.
- **Deductive reasoning** – The findings support the conclusion so strongly that if the findings are realised, it is impossible for the conclusion to be uncertain or false (Duran & Şentürk, 2019; Hammond & Wellington, 2013; Walliman, 2011). For example; mathematical formulas that only accept objective inputs.
- **Abductive reasoning** – The findings support possible conclusions based on the best guess of an outcome (Fellows & Liu, 2015). In autumn and winter, the leaves fall off the trees.

This research used abductive reasoning. Abductive reasoning has two main requirements:

1. The researcher identifies a hypothesis, theory or model (Fellows & Liu, 2015). An overarching proposition (as a form of hypothesis) was formed that, using passphrases with keystroke dynamics as a two-tier authentication solution, can increase system security and system usability.
2. A specific problem needs to be identified which can be addressed by point 1 above (Fellows & Liu, 2015). The research problem identified refers to current authentication methods that cannot simultaneously address system security and system usability issues.

The next section explains the research approach.

2.4 Design Science

This section on design science firstly explained the methodology. The application of the methodology to this study was then discussed.

2.4.1 Design Science Guidelines

Hevner et al. (2004) provide a conceptual framework (see Figure 2-1 below) to support researchers in understanding, executing and evaluating research in the information systems discipline. All information system research includes three primary aspects –

people, business and technology; this is shown in the far left-hand box (environment) in Figure 2-1 (Indulska & Recker, 2008; Lasrado, Vatrappu, & Andersen, 2015). The box on the far left in Figure 2-1 is aimed at ensuring research relevance as it enables researchers to clearly define the problem or opportunity (Hevner et al., 2004). The box on the far right (knowledge base) in Figure 2-1 ensures the credibility of the study by guiding the researcher to utilise reliable and trustworthy material. Hevner et al. (2004) explain that information systems research is conducted in two phases; firstly, the development or criticism of an artefact or theory, and secondly, the assessment of this discovery using appropriate evaluation methods.

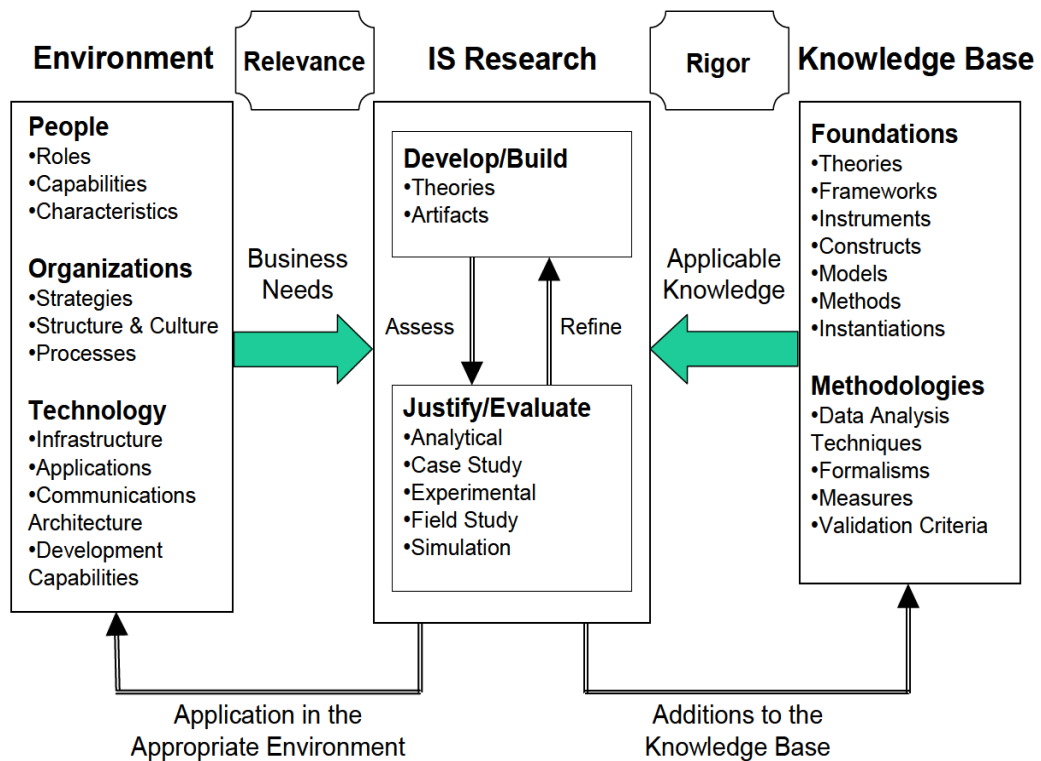


Figure 2-1: Information Systems Research Framework (Hevner et al., 2004)

Indulska and Recker (2008) and Lasrado et al. (2015) explain that there are two main paradigms that support the information system discipline; behavioural science and design science. Although these are two different approaches to research, both are important to the growth of the information system discipline. Table 2-1 differentiates the distinct attributes of the behavioural sciences paradigm from the attributes of the design science paradigm.

Table 2-1: Differences in Behavioural Sciences and Design Sciences (Hevner et al., 2004)

Design Sciences	Behavioural Sciences
Develop artefact to solve problem/s	Develop theories to explain problem/s
Improve reality	Understand reality
Truth – informs design	Utility – informs theory

Research guidelines have been developed for both the behavioural paradigm and the design paradigm. These guidelines are used by researchers to ensure that the research aimed at contributing to the information systems discipline is conducted correctly and ensure that correct conclusions are made.

The design paradigm is more closely aligned to this study than the behavioural paradigm, as it aims at the construction of an artefact in the form of a model and uses various theories to support its assumptions. For this reason, the design science guidelines were used to guide this research so that conclusions could be made with confidence.

2.4.2 Design Science Application

Design science is a problem-solving methodology which “seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently accomplished” (Hevner et al., 2004, p. 76). The methodology consists of seven guidelines. All seven guidelines must be accounted for in a study to ensure that the methodology is used correctly. Note that the seven guidelines do not have a chronological order; however, the researcher must ensure that all seven guidelines are accounted for at some point in the research (Hevner et al., 2004; Nikora et al., 2018). The section on design science application explains how the seven guidelines discussed above were applied to this study.

2.4.2.1 Guideline 1: Problem Relevance

The objective of design science research is to develop technology-based solutions for important and relevant business problems (Hevner et al., 2004; Lasrado et al., 2015). The problem identified is that the user authentication solutions currently utilised by systems are not sufficient for addressing both the usability and security issues simultaneously. This study focused on assessing the effectiveness of a proposed two-

tier solution which involved passphrases (first tier of authentication) and keystroke dynamics (second tier of authentication). The first round of the expert review was used to validate the research problem and understand the magnitude of the issue.

2.4.2.2 Guideline 2: Design as a Search Process

The search for an effective artefact requires using available means to reach desired ends while satisfying laws in the problem environment (Cronholm & Göbel, 2015; Hevner et al., 2004). Related literature in journals, conference proceedings, books and websites were used to collect information on system security from a user authentication perspective, as well as information on user authentication usability issues. The expert review was used to validate certain conclusions drawn from the secondary data which in turn were used to develop the proposed model. Feedback received from the expert review is expected to confirm the findings or make recommendations. This ensured that all findings were critically analysed and/or validated.

2.4.2.3 Guideline 3: Research Rigour

Design science research relies on the application of rigorous methods in both the construction and evaluation of the design artefact (Hevner et al., 2004; Nikora et al., 2018). Both primary data and secondary data were used to create and support all logical conclusions applied to the development of the artefact (the two-tier authentication solution). This is detailed in the two paragraphs below.

Primary data – The primary data and secondary data pertaining to the login assessment was used to build the proposed artefact. Subsequently, the expert review was used to confirm the research problem and receive direction and feedback on the potential benefits and drawbacks of the proposed two-tier solution, within the context of the problem being addressed. The proposed artefact was then presented to experts for review and for comment based on the impact of the proposed artefacts on the research problem. This ensured that all content used to build the artefact was validated by experts and refined accordingly.

Secondary data – Theories such as the Keystroke-level model (Jamaluddin & Revett, 2012; John & Kieras, 1994; Jorritsma et al., 2015; Lee et al., 2015), Chunking theory (Bošnjak & Brumen, 2016; España, 2016; Miller, 1956) and Shannon Entropy theory (Aguiar & Guedes, 2015; Arora, Hanmandlu, & Srivastavaa, 2015; Shannon, 1948) were used to guide the research, as well as to evaluate and validate any assumptions. In

addition, related work conducted by other researchers was used to develop and assess the effectiveness of the proposed solution.

2.4.2.4 Guideline 4: Design as an Artefact

Design science research must produce a viable artefact in the form of a construct, a model, a method, or an instantiation (Cronholm & Göbel, 2015; Hevner et al., 2004). This study intends to produce a two-tier authentication solution in the form of a model, which can be used to increase the security of user authentication while ensuring that usability does not become compromised.

2.4.2.5 Guideline 5: Design Evaluation

The utility, quality and efficacy of a design artefact must be rigorously demonstrated via well-executed evaluation methods (Cronholm & Göbel, 2015; Hevner et al., 2004). This is discussed in more detail under Section 10.6.5, Design Evaluation. Accordingly, the two-tier authentication model, as the proposed artefact, was validated by means of an expert review. This model was developed after considering both primary and secondary data. The proposed artefact was presented to the expert group (user experience experts and security experts) and the feedback obtained was used to make the necessary amendments to the proposed artefact before it was finalised. The login assessment experiment was also used to confirm secondary findings, conclusions, assumptions and propositions. Note, only confirmations pertaining to passwords, passphrases, typing and memory were confirmed through the login assessment. A keystroke dynamics algorithm was not applied to the login assessment.

2.4.2.6 Guideline 6: Research Contributions

Effective design science research must provide clear and verifiable contributions in the areas of the design artefact, design foundations and/or design methodologies (Hevner et al., 2004; Nikora et al., 2018). In this research, the design artefact takes the form of a two-tier authentication solution which aims to increase system security by reducing the likelihood of users resorting to behaviours that compromise the security of the system. The proposed solution also seeks to encourage users to create passphrases rather than passwords (increase in security) as they are easier to memorise and insert in the system (increase in usability). This may lead to less time being spent on forgotten passwords. Passphrases are also supported by the National Institute of Standards and Technology (NIST) (Burr, et al., 2017) when they are compared to passwords.

2.4.2.7 Guideline 7: Communication of Research

Design-science research must be presented effectively, both to technologically-oriented and management-oriented audiences (Hevner et al., 2004; Nikora et al., 2018). This study assists the system development team to improve the security and usability of the user authentication process by providing a model which includes components to be considered before implementing the two-tier authentication solution. In order to accomplish this, the findings of this study must be communicated to external readers (Cronholm & Göbel, 2015). To this end, the findings will be published in academic journals and conferences and made available to the public for future research. The thesis will also be accessible via the library at Rhodes University.

By ensuring that the seven design science guidelines are accounted for one may confidently state that robust research and credible conclusions were reached. The next section summarised the research process which entailed the application of the seven design science guidelines mentioned above.

2.5 Research Process

It is evident from the above discussion of the guidelines that they overlap across the study. To provide more clarity on the overlaps, Figure 2-2 below graphically depicts how this study in its entirety was used to support the construction of the proposed two-tier solution. The numbers between one and seven in Figure 2-2 indicate the design science guideline followed for the collection and/or analysis of the respective activity.

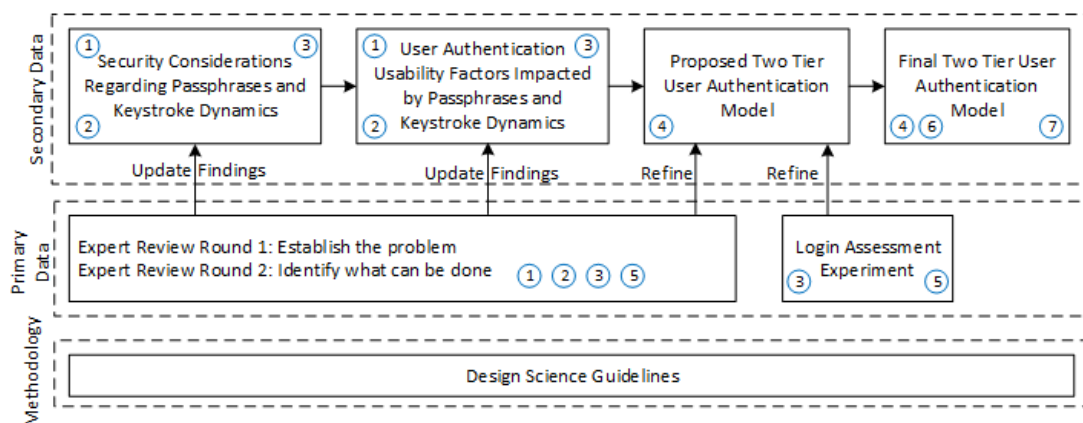


Figure 2-2: Research Process

As previously emphasised, specific design science guidelines are followed at different stages of the research study. However, all seven design science guidelines were accounted for throughout the study. To ensure efficient use of time, primary data and

secondary data collection and analysis was conducted in parallel where possible, as indicated in Figure 2-2. The next section discusses how the data for this study was collected.

2.6 Data Collection

Data can be collected and analysed using one of three main methods – quantitative, qualitative or mixed methods (a combination of qualitative and quantitative methods) (Morse & Niehaus, 2016). This research adopted a mixed methods approach to data collection and analysis. Accordingly, quantitative data collection and analysis were conducted through the login assessment, and qualitative data collection and analysis were conducted by means of the expert review and secondary data. Sections 2.6 to 2.9 discuss the mixed method approach adopted for this study in more detail.

2.6.1 Primary Data Collection

The primary data collection section includes a discussion on the login assessment and the expert review.

2.6.1.1 Login Assessment

The overall intention of a login assessment was to determine typing errors in passwords and passphrases, and the ability to recall conventional passwords and passphrases from memory. For this reason, a purposeful sample was used (Cresswell & Clark, 2011; Schoonenboom, Johnson, & Fröhlich, 2018) which in relation to this study required all participants to have prior login experience on other systems that utilise text-based user authentication. The login assessment thus assessed the following effects on usability:

- Memorisation of the password and passphrase
- Correctly inserting the password and passphrase into the system (typing)

Keystroke dynamics was not assessed in the login assessment. However, a number of measures were used to assess the above two effects on usability. These are discussed in the subsection, Login Assessment (Section 2.8.1) of the Section, Primary Data Analysis.

Based on research using similar experiments, 66 participants were deemed to be sufficient for this type of assessment approach (Chiasson, Forget, Stobert, Van Oorschot, & Biddle, 2009; Yang, Lindqvist, & Oulasvirta, 2014). A sample of 65–100 participants were targeted. This sample size was determined to be sufficient to collect valuable data that represents the population, based on similar studies (Chiasson et al.,

2009; Yang et al., 2014). A total of 123 participants started the login experiment however, only 112 participants completed the experiment (i.e. at least ten login interactions).

In order to collect sufficient quantitative data to identify trends in assessing login efficiency in participants, they had to interact with the system at least ten times. Studies in which similar experiments were conducted imposed a time limit on the assessment (Brill & Olmsted, 2016; McCarney, Barrera, Clark, Chiasson, & Van Oorschot, 2012); however, these experiments were performed on existing systems which had a high user–system interaction rate. Since this study was unable to use such an existing system owing to the type of data that had to be collected, all participants were requested to login to the system at least ten times (at least once per day) as part of the experiment.

The system used was developed for the sole purpose of collecting the data required for this research. The first iteration was the first contact with the participants. In initiating the research, participants were sent an email which included the following:

- A section explaining the research project.
- Instructions for the actions that the participants were required to perform.
- The duration of the experiment.
- An explanation on the way in which the data collected would be used.
- An assurance that participation was optional and participants could stop participating at any point during the experiment.
- To ensure that participants incurred no harm or damages, they were warned not to create passwords and passphrases that they were already using on other systems or had previously used. This instruction was stated in bold and in a slightly larger font to ensure that it was not overlooked.
- A link to the login assessment website with instructions on how to participate was distributed to all participants via email. To ensure that all participants had prior login experience, the email link was sent to Gmail and Yahoo email addresses, as these email systems require text login to access the email.
- A caveat was stated just above the link (mentioned above): “By clicking on the link, it is assumed that you have read the above and understand the rules of the experiment. Please reply to this email if you have any questions or concerns.”

A purposive sample was used (Cresswell & Clark, 2011; Schoonenboom et al., 2018) which in relation to this study, required all participants to have prior login experience on other systems that utilise text-based user authentication. For the sake of convenience,

email was selected as the primary communication method with participants. Email communication also ensured that the user had prior experience logging into systems with text-based authentication as the majority of email systems require a text-based user authentication method to gain access to emails.

Participants were instructed to create and login to two user interfaces. In order to control system usability, all interfaces that the participants interacted with had the same layout as the interfaces used by conventional systems. In addition, both user interfaces included two text-based authentications. Appendix A provides the screen flow and screenshots of the login assessment interface:

1. The first text-based authentication asked the participants to create and login to the system with a password. Although systems have different levels of stringency when it comes to creating a password (Houshmand & Aggarwal, 2012; Kelley et al., 2012; Yildirim & Mackie, 2019), an average stringency level password was assessed in this study. Stringent passwords are often found on systems where system security is paramount, such as online banking websites. An example of a stringent password would be that the password should include at least one special character, two digits, one uppercase letter, one lowercase letter, password length should be more than eight characters and no digits or letters can be consecutively repeated. Although these password rules do exist in reality, a more common rule set for password creation was used for this experiment. For this experiment a LUDS password had to be created. The system restricted the user from creating a password that did not fulfil the password parameters cited in bold above.
2. The second text-based authentication asked the participant to create and login to the system with a passphrase. By definition, a passphrase must be more than 16 characters long and cannot include any uppercase letters, digits or special characters (see Chapter 3, Section 3.5, Passwords and Passphrases Defined). Therefore, the system restricted the user from creating a passphrase that did not fulfil the passphrase parameters.

Participants were asked to create a password and a passphrase. After doing this, the system asked the user to login to the system using the password and passphrase they had created. Thereafter, participants were reminded via email to login to both the systems with the two text-based authentications that they had created at the first contact.

If a participant forgot their password or passphrase, he/she could create another. This was done by selecting the “forgot password” or “forgot passphrase” option and creating a new one. Logically, the experiment would end for the participant if they forgot both their password and passphrase at least once. However, offering the option to end the experiment early might have reduced participation and incurred inaccurate results. Therefore, a participant who forgot any password/passphrase throughout the duration of the experiment had to create a new password/passphrase or keep trying to login until the he/she was able to login successfully with their password and passphrase.

A rough design of the login assessment screens was developed to assist in the development of the login assessment. A high-level view of the sequence in which the participants interacted with these screens appears in Figure 2-3.

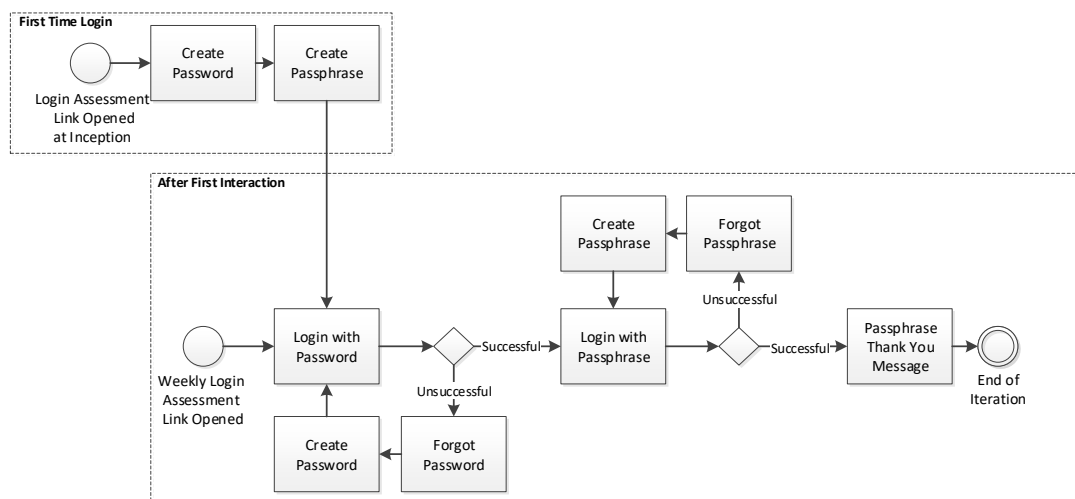


Figure 2-3: Participant Login Assessment Process

Figure 2-4 displays the eight screen designs the user interacted with during the login assessment. The login screen was designed to clearly indicate to a participant which login page they were on – password or passphrase. This was to ensure that the data collected was not skewed due to an error on the part of the participant in entering the correct password on the passphrase login screen, which would record the interaction as a login failure.

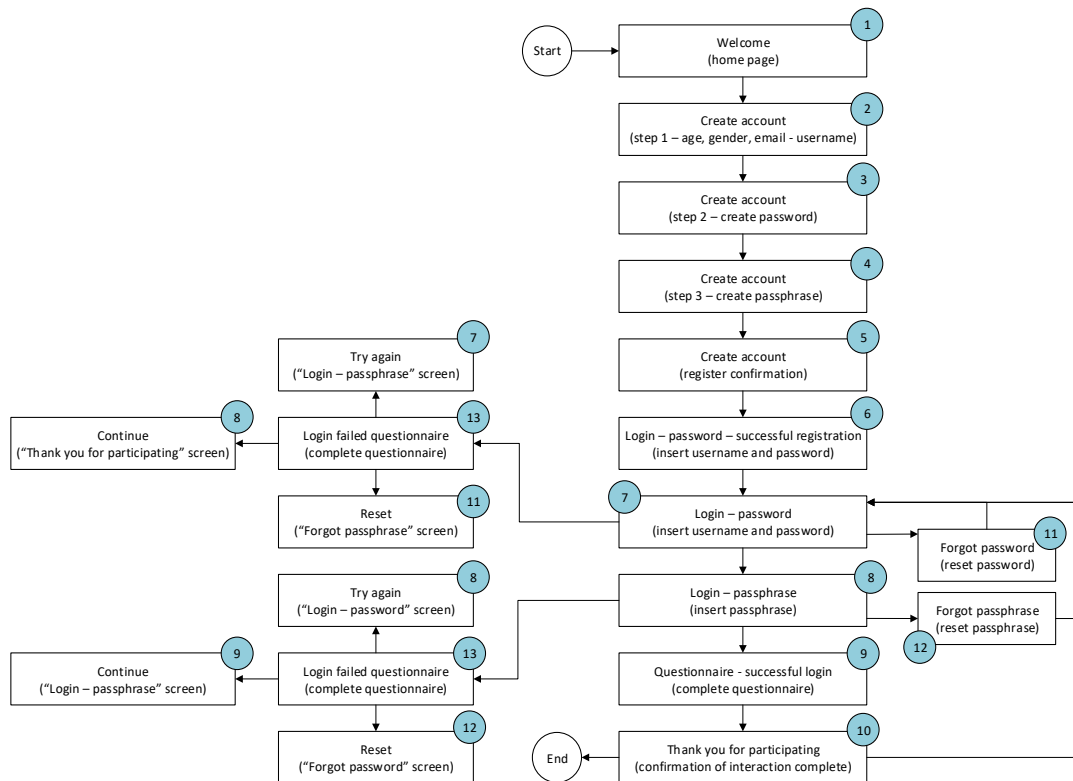


Figure 2-4: Login Assessment Screen Designs

Before the login assessment commenced the website (www.loginassessment18.co.za) was tested by a quality assurance system tester to confirm that the user interface was correct, and that the database would accurately record the necessary data. The interface had to be correct in terms of replicating a common login screen. This was important in order to ensure that usability was not affected by an uncommon login interface. The database checks were vital as the database was recording all the required data for the login assessment.

2.6.1.2 Expert Review

An expert review is a data collection method used to gain insights on specific areas from experts in the field (Molich & Jeffries, 2003; Zhang, Zhang, & Yang, 2016). Conclusions and assumptions collected from secondary data were presented to the experts to confirm, expand on or comment on. Questions were attached to the secondary data to guide feedback and discussion in order to ensure that relevant feedback was received.

The questions were constructed so as to encourage different points of view, the intention being to ensure that the areas reviewed by the experts were thoroughly critiqued. The questions were also used to control the relevancy of feedback to ensure that no off-topic feedback was collected. In order to ensure that the best conclusions in relation to this

study were collected, the expert review included two rounds. The research context and questions that were supplied to the experts are provided in the next section.

It should be noted that context and questions were constructed concisely to ensure that the experts did not lose interest as a result of the volume of reading material. All communication with experts was conducted via email and each email communication included the context and questions in relation to the proposed two-tier solution. The emails also requested that the experts contact the researcher via email if they required any clarification of the material supplied.

The selection of experts for an expert review is dependent on the research being undertaken (Barber, et al., 2015). It is important that relevant experts are chosen to ensure that the material presented to them is assessed effectively and that valuable comments are provided by the experts. In terms of this study, the experts were selected based on their educational background and their experience relevant to this study. Security experts had to have at least three years of front-end system security experience in the field and usability experts at least three years' experience in user experience design, which included login pages.

Although there is no rule which states the number of participants required for an expert review, it was found that it is dependent on three factors: 1) the structure of questions posed to the group, 2) the number of questions in each round, and 3) the volume of analysis expected to be conducted on the respective group of questions (Edwards, Dunlop, Mallick, & O'Callaghan, 2015; Hsu & Sandford, 2007). In this study, each round included one or two paragraphs to provide context to the participants, along with a few open-ended questions. Based on this layout and the study's goals, ten experts were deemed sufficient to obtain the correct volume and value of feedback from the group.

When attempting to search for sufficient numbers for the expert group, no fixed rule could be found for identifying the number of participants required for an expert review. However, based on similar research a number of factors were identified for consideration (Edwards et al., 2015; Hsu & Sandford, 2007). These factors can be grouped into three categories:

1. The structure of questions posed to the expert group.
2. The number of questions in each round and volume of expected feedback.
3. The amount of analysis expected to be conducted on the feedback received.

It is recommended that an expert review should not exceed 45 days. This minimises the risk of experts not responding and losing interest, or a decrease in the quality of feedback (John, Kadadevaramath, & Edinbarough, 2017; Ludwig, 1997). Taking these factors into consideration, the same participants were used in both rounds to limit the number of contradictions and ensure that all feedback was obtained within the set period. This study required a minimum of two rounds to assess the proposed solution. The breakdown of the two rounds is as follows.

1. Identification of the impact of the proposed concept on addressing the research problem.
2. Assessment for the degree to which the proposed artefact addressed the research problem.

Figure 2-5 below graphically depicts the way in which the expert review was applied to this study. In order to avoid confusion when reading Figure 2-5, the following terms need to be defined:

- **Round** – A round is a process that begins when the researcher makes contact with the experts and the experts in turn provide feedback to the researcher based on the researcher’s request.
- **Questions** – Questions were only provided at the beginning of each round. This was to ensure that relevant feedback was received from the experts.
- **Consolidate feedback** – The process of analysing feedback. This refers to the analysis process which involves a number of steps. Refer to Chapter 2 Section 2.8.2, Expert Review for further details.
- **Duration** – The length of one round.
- **Total duration** – The length of the expert review in its entirety.

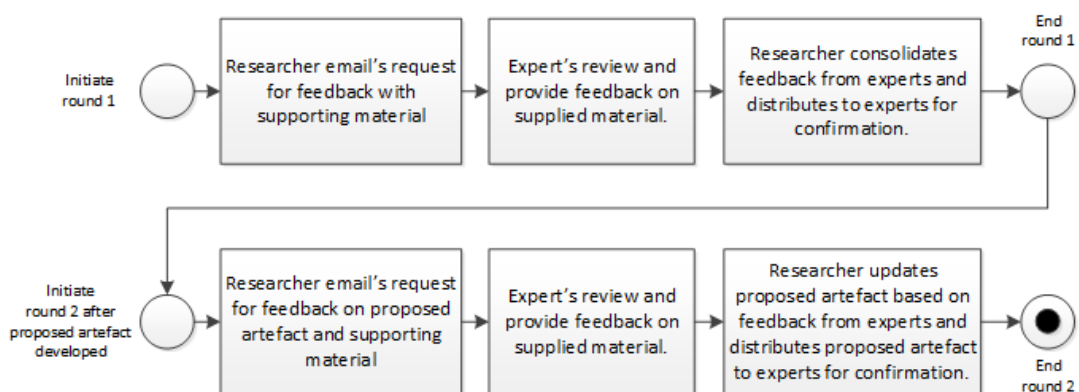


Figure 2-5: Expert Review Process

As Figure 2-5 indicates two rounds were conducted for the expert review. The initiation of round two was dependent on the completion of the proposed artefact. Now that the two rounds have been explained, the next section provides the content that was sent to the expert group for each round.

2.6.2 Expert Review

This section includes the material that was provided to the experts for each round of the expert review. In this study two rounds were required to ensure the robustness, completeness and accuracy of the research model:

1. **Round 1: Establish the problem (validate)** – One-tier authentication is not secure enough unless strong passwords are used, which has a negative impact on usability. Current two-tier authentication solutions provide security but negatively affect usability.
2. **Round 2: Refine the model** – Present the proposed artefact and amend it based on substantiated feedback until consensus is reached.

These are discussed further below.

2.6.2.1 Round 1: Problem and Proposed Solution

The problem this study aimed to solve related to the imbalance between system security and system usability. Current solutions identified either increased security and reduced usability or reduced security and increased usability.

A two-tier authentication method was proposed to resolve the problem mentioned above. The first tier of authentication allows the user to use a passphrase as a text-based user authentication method. This removes the need for a user to create a password that is more difficult to type and memorise when compared to passphrases, which do not include a combination of different character sets such as special characters, uppercase letters, lowercase letters and numbers. By addressing the typing and memorisation issues related to the user authentication process, the usability aspect of the user authentication process is improved.

The second tier of authentication is used to further increase security. This study proposes that keystroke dynamics be used as the second tier of authentication. Keystroke dynamics is a behavioural biometric authentication method which tracks a user's typing patterns in the backend of a system. Because keystroke dynamics is a backend solution, it has little to no impact on the usability, if implemented correctly. In addition, it provides a significant increase in security. If an unauthorised party wants

access to a system, he/she needs to not only get the user's passphrase but also determine how the user enters the passphrase into the system.

2.6.2.2 Round 1 Questions

1. Do you believe this solution will increase security and increase usability? If not, why?
2. What problems do you envision with this proposed solution that will jeopardise the proposed solution's capability of increasing both security and usability?

2.6.2.3 Round 2: Proposed Artefact

The proposed artefact took the form of a model. The model depicts a two-tier authentication method and the issues that should be considered when implementing this method from a security and usability perspective. The model was presented to the experts along with an explanation of the model to ensure that all the experts understood the model before commenting.

2.6.2.4 Round 2 Questions

1. Why do you or don't you believe that this model will increase security?
2. Why do you or don't you believe that this model will increase usability?
3. Why do you or don't you believe that this model will be accepted?
4. Do you have any further comments or recommendations?

The secondary data collection for this study is discussed in the following sections.

2.7 Secondary Data Collection

The secondary data for this study used related literature contained in journals, conference proceedings and books, and also considered relevant theories and methodologies. In addition, the frameworks, models, techniques and standards used related to user security, system security, password and passphrase memorisation, password and passphrase creation thinking process and user system input experiences. These areas were identified by assessing the tags attached to journal articles and conference proceedings that were closely related to this study. In addition, the assessment considered only research efforts conducted in the past 10 years to ensure relevancy. This is an important factor to consider when researching material in the technological discipline, owing to the fast and constant innovation in cyber security (specifically in authentication and hacking methods) going on in the environment.

This concludes the discussion on secondary data and the section on data collection. The next section discusses the way the data collected was analysed.

2.8 Primary Data Analysis

The login assessment is firstly discussed, followed by the analysis of the data collected from the expert review.

2.8.1 Login Assessment

The intention of the login assessment was to identify which text-based authentication method – passwords or passphrases – has a larger impact on system usability without weakening security. The keystroke dynamics algorithm was not assessed in the login assessment. Security was assessed by analysing the number of users that create stronger than required passwords and/or passphrases based on the common password/passphrase strength indicator and an uncommon password/ passphrase strength indicator (the runny bunny). From a usability perspective, the data collected from the login assessment was used to assess the participants’ ability to recall the text-based authentication they had created (memorisation assessment) and the ability to correctly insert the text-based authentication into the system (typing assessment). Table 2-2 below tabulates the tools used to assess memory and typing proficiency.

Table 2-2: Aspects of the Login Assessment

Memory assessment tools	Typing assessment tools
Login failed (a question was asked to identify whether this was a typing error or user-memory error)	Login failed (a question was asked to identify whether this was a typing error or user-memory error)
“Forgot password”/“Forgot passphrase” option was selected	

In Table 2-2 above, the “login failed” assessment is classified under memory assessment and typing assessment. Realistically, if a user’s login failed it could have been a memory or a typing issue. Unfortunately, this is difficult to identify, therefore, the participant was asked a simple question in order to ascertain whether the login failure resulted from a user memory or typing error. In addition, the participants were asked another simple question regarding what device was used to login, i.e. a computer, a cellphone or a tablet.

This was done to attempt to discover whether some devices have a higher likelihood of typing errors occurring than others.

2.8.2 Expert Review

When feedback was received from experts after each round, the analysis was performed in the following order:

1. Feedback was grouped in order to remove duplication.
2. The grouped feedback was then summarised.
3. The summarised grouped feedback was then shared with experts to avoid the risk of misinterpreting the feedback received from the experts during steps 1 and 2 of this analysis process.

The feedback also had a large influence on how the proposed artefact was developed. Once the proposed artefact was developed, the second round was initiated and once feedback had been received by the expert group on the proposed artefact, the artefact was finalised by incorporating the agreed changes to the model. The proposed artefact then became the final artefact. That being said, round two was dependent on the completion of the proposed artefact, which was only ready for review once the other primary data and secondary data had been collected and analysed. The next section explains how the secondary data collected was analysed.

2.9 Secondary Data Analysis

All secondary data collected was analysed inductively. Conclusions reached by other researchers were assessed to understand why their research agreed or disagreed with relevant theories in the respective discipline. In order to comply with this approach, critical thinking (Oates, 2006; Pilgrim, Vasinda, Bledsoe, & Martinez, 2019) needs to be applied when assessing past research efforts. The following section states the delimitations of this study.

2.10 Theoretical Foundation

Three main theories were used as a foundation to conduct this study: The Shannon Entropy theory, the Chunking theory and the Keystroke-level model. These are discussed in more detail in the following sections.

2.10.1 Shannon Entropy Theory

To ensure a comprehensive understanding of why this theory was selected to assess security, one important aspect of the Shannon Entropy theory has to be discussed. It should be noted that although a number of theories and methods have been proposed to assess text-based authentication strength, all seem to have been criticised for lacking various aspects.

In spite of similar criticisms, the Shannon Entropy theory is deemed sufficiently flexible to apply to both passphrases and keystroke dynamics, which is what is required by this study. For this reason, it is important to mention that the results generated by the Shannon Entropy formula should only be used as a rough estimate to gauge authentication strength.

It is important to note the use of a rough estimate here, as this is required in order to maintain assessment simplicity. Simplicity is required to ensure a consistent assessment. It is possible to obtain a more precise assessment. However, the intention of the security portion of this study is merely to determine an estimated level of security and then compare it to benchmarked text-based authentication. As an example, a more accurate assessment of security would consider the probability of certain characters being used more than others and which characters users are most likely to start their text-based authentication with. These probabilities are not considered in this study as they cannot be generalised to the security assessment of keystroke dynamics. In other words, if a detailed assessment were performed it would not be possible to compare passwords and passphrases and keystroke dynamics with the same theory and metric.

Chapter 3 discusses various components and levels of keystroke dynamics. The modification of these components (add or remove) and levels (change the levels) has a direct impact on security. Hence, an average value was used to indicate the general influence of these changes on the security of a system.

2.10.2 Chunking Theory

The Chunking theory holds that people have less trouble recalling from memory something to which they can relate based on past experiences and/or level of exposure. For example, you are more likely to recall what jacket a person was wearing yesterday if you own the same jacket, as opposed to seeing a jacket on a person with which you have no past link.

Accordingly, the Chunking theory was used firstly to understand how a user creates a text-based authentication password. The theory was then used to determine the extent to which password policies restrict users from creating one that would be easy for them to recall from memory.

The intention was to understand what system restrictions could be removed in order to improve the likelihood of a user recalling the text-based authentication from memory. Once this understanding was gained it was possible to determine how many of these restrictions could be removed.

2.10.3 Keystroke-level Model

The Keystroke-level model is a prediction tool used to measure the length of time taken by an expert user to execute a routine system task (John & Kieras, 1994; Jorritsma et al., 2015; Lee et al., 2015). In terms of this research, the routine task to be executed is the user authentication process.

The Keystroke-level model takes six factors into consideration:

- **K** – Number of keys pressed
- **P** – Pointing to on-screen targets
- **H** – Hands to keyboard and mouse
- **D** – Drawing a line to assess mouse movement
- **M** – Mental preparation required
- **R** – Response time of the system

This was found to be an acceptable theory for assessing usability of user-keyboard interactions where the user enters various forms of text-based authentications, the reason being that system usability is directly affected by the amount of effort required to execute certain system tasks (Komogortsev, Mueller, Tamir, & Feldman, 2009; Leino, Todi, Oulasvirta, & Kurimo, 2019). Effort in the case of the Keystroke-level model is represented as time. It should be noted that the factor “drawing a line to assess mouse movement” and “pointing to on-screen targets” was excluded as it relates to usability effects relating to mouse movements which are not required for text-based authentication.

The four factors listed above (“drawing a line to assess mouse movement” and “pointing to on-screen targets” excluded from the six factors) were used to assess how the use of various forms of text-based authentication, all running a keystroke dynamics algorithm,

influence these factors. Since all factors are important for usability, no weights were added to indicate the importance of the Keystroke-level model factors.

2.11 Delimitations

The delimitations of a study allow the researcher to clearly define the scope of the research by stating what will and will not be considered (Hofstee, 2006; Saini, Kaur, & Bhatia, 2018). This study focused specifically on password and passphrase authentication. Other forms of authentication were not assessed for the first tier of authentication. The keystroke dynamics authentication is the only form of second tier authentication that was explored. Although there are a number of methods for assessing password and passphrase strength, the Shannon Entropy theory was the only metric used as this theory is best suited to this study while still yielding realistic results.

The login assessment was web-based to make participation more convenient. When conducting the login assessment, participants were asked not to make use of any password/passphrase recall features on the browser or any other password/passphrase storage mechanism, including writing the password/passphrase down on paper. Since this is difficult to ensure, the study assumed that participants complied with the rules of the login assessment when they conducted the assessment and stated that they understood and would comply with the rules of the login assessment. For convenience, participants in the login assessment were all resident in South Africa (Etikan & Bala, 2017). This did not affect the sample representation of the population. The login assessment also did not include assessment of the keystroke dynamics algorithm due to the volume of data required to be collected and reported on. This would have resulted in a too large scope for this research study.

The boundaries of this research have been emphasised by the delimitations mentioned above. The next section covers the ethical considerations of the study.

2.12 Ethical Considerations

Babbie (2005) explains that it is the researcher's responsibility to comply with academic integrity and honesty, as well as to respect other people while conducting any form of research. Arifin (2018), Hammersley and Traianou (2012), Punch (2006) and Russell, Hogan, and Kenny (2012) add to Babbie's (2005) explanation by listing categories of ethical issues that should be considered by researchers. These ethical considerations are listed below.

- Informed consent

- Obtained by sharing all necessary information about the research with all potential participants before requesting participation.
- Confidentiality and anonymity
 - Questions posed to participants were constructed in a manner that ensures no personal information is captured.
- Ownership of data and conclusions
 - All data collected, used and generated for this research acknowledged the respective stakeholders.
- Use and misuse of results
 - The misuse of results was mitigated by ensuring that information gathered from participants was only used for the intended purpose of the research.
- Honesty and trust
 - This was obtained by ensuring that all study participants were clearly informed of about the research, the data collected and how it would be used.
 - Information collected from participants was used for research purposes only.
 - Participants were allowed to refuse participation in the research at any time.
- Harm and risk
 - No primary data was collected for this study until approval was obtained from the Rhodes University's Ethics Committee.
- Non-maleficent (Holloway & Wheeler, 1995)
 - Users participating in the login assessment were asked only to use passwords and passphrases that they were not currently using or had used in the past. This ensured that any passwords and passphrases collected could not be used to gain unauthorised entry to a participant's accounts.

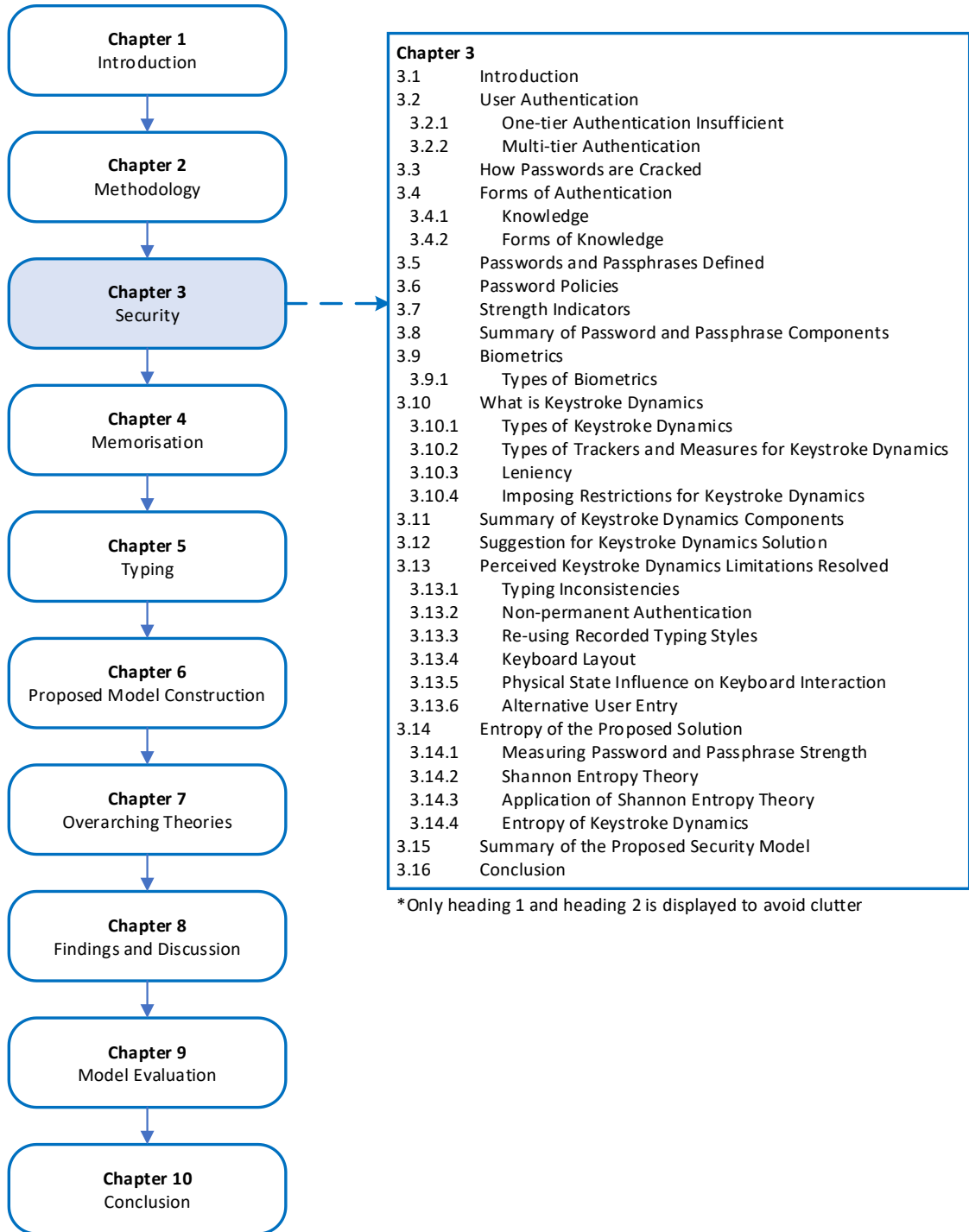
The above-mentioned ethical issues were considered and the research undertaken complied in all respects. The next chapter is the first literature chapter (Chapter 3) of this study, focusing on system security of text-based user authentication.

2.13 Conclusion

This chapter was aimed at explaining how this study was conducted. In summary, this study took an abductive reasoning approach while following the design science

guidelines. A pragmatic research paradigm was best suited for this research. The following were used as driving theories to guide research findings: the Shannon entropy model, the Chunking theory and the Keystroke-level model. Primary data collection came from a login assessment experiment which included 112 participants, and an expert review which involved a total of 10 experts; five usability experts and five security experts. Delimitations were also presented to clearly define the scope of this study. Lastly, ethical considerations were listed to ensure that the research was conducted in an ethical manner.

Chapter 3 – SECURITY



3.1 Introduction

Computers allow for large volumes of data to be stored and retrieved. Although storing and retrieving data from a computer has given rise to a number of benefits, it is also linked to certain issues such as security breaches. One aspect of securing a system is restricting users from accessing unauthorised data. The system development team attempts to address this issue by introducing passwords as a form of user authentication. Passwords force a user to input a specific set of characters, in the correct sequence, before the system gives the user access to authorised content. Security continues to be an issue for many systems. Over the years, hackers have become smarter due to easier access to knowledge. Hacking tools have also become more sophisticated and easier to use (Abinaya & Sigappi, 2018; Braunstein, 2015).

Although it may be possible to secure a system at the front end, the system development team needs to consider system usability. Payne and Edwards (2008) and Dib and Ghazi (2019) explain that this had always been a problem and continues to grow as technology and systems proliferate. Users are now using more systems than in the past which has required them to memorise more login details. Bear in mind that the problem this study is essentially aiming to resolve is addressing the ineffective current solutions which either increase system security and reduce usability or decrease security and increase usability. Usability is discussed in more depth in Chapters 4 and 5. This chapter focused more on understanding the security aspects of the issue. However, high-level impacts on usability were referenced throughout the chapter since they are closely related to security.

The definition of different tiers of authentication is firstly discussed. This is followed by a brief explanation of how passwords are cracked. Forms of authentication are then provided which is followed by a definition of passwords and passphrases. This is followed by a section on password policies and subsequently a section on strength indicators. A short summary of the password and passphrase components is then provided. Types of biometrics are discussed, followed by an explanation of keystroke dynamics. A summary of the components of keystroke dynamics is then discussed before providing a suggestion for the best way to implement keystroke dynamics. Resolutions for the keystroke dynamic's limitations identified are then provided. The entropy of passwords, passphrases and keystroke dynamics is then discussed before introducing the security model for this study. The last section summarises the discussion in this chapter.

3.2 User Authentication

This section firstly explains why one-tier authentication is no longer effective as a security protocol before moving on to a discussion of multi-tier authentication as a more viable solution.

3.2.1 One-tier Authentication Insufficient

There was a time when one-tier of authentication was sufficient. However, due to increases in hackers' knowledge and access to faster hardware and software, one-tier authentication is insufficient to ensure the security of a system (Alguliyev, Aliguliyev, & Yusifov, 2018; Raghavan, Desai, & Rajkumar, 2017). Although there are a number of different types of authentication, Chiasson et al. (2009), Patel and Jha (2015) and Payne and Edwards (2008) explain that a single tier of authentication is simply insufficient to prevent unauthorised access to a system without significantly affecting system usability in a negative way. Many researchers are exploring merging two forms of authentication without having much of an impact on the user. The difficulties faced with current two-tier authentication solutions are discussed in the next section.

3.2.2 Multi-tier Authentication

The definition of multi-tier authentication refers to the number of times a user needs to verify his/her identity before access is granted (Muhamad, et al., 2019). The intention of multi-tier authentication is to provide multiple layers of protection in order to make it difficult for any unauthorised entities to gain access to the system. A number of user authentication methods exist. Multi-tier authentication simply utilises various user authentication methods in conjunction with one another. A common example is an online banking system where a user needs to login to the system online (first tier of authentication) and then inserts an OTP (sent to their cellphone or email address) if they want to transact on the system (second tier of authentication).

In this study, research is conducted into a two-tier authentication approach where passphrases are used as the first tier of authentication and keystroke dynamics are used as the second form of authentication. When combining different forms of authentication, it is important to consider their impact on usability and security. This assessment is important as some combined authentications complement each other, while others may not work as well together.

A number of studies have conducted research into two-tier authentication and some even go as far as three-tier authentication (Dmitrienko, Liebchen, Rossow, & Sadeghi, 2014; Wang & Wang, 2015). Although adding more tiers of authentication may be seen as a simple solution for increasing security, having multiple authentication methods reduces usability. Research has focused on merging a number of combined user authentication methods to address the usability aspect of user authentication (Siddiqui, Abdullah, Khan, & Alghamdi, 2014; Wang & Wang, 2015). Accordingly, no research was found that focuses specifically on the customisation of password policies, and the utilisation of passphrases and keystroke dynamics as a method of authentication. In addition, studies that have focused on keystroke dynamics as a method of authentication have not considered other input devices such as tablets, laptops and mobile phones (De Ru & Eloff, 1997). The next section explains how passwords are cracked.

3.3 How Passwords are Cracked

Hackers are people who try to gain access to unauthorised systems. The number of hackers has increased over the years. This has to do with the population having easy access to information on learning how to hack and with hacking software becoming more user friendly (Abinaya & Sigappi, 2018; Bhivgade, Bhusari, Kuthe, Jiddewar, & Dubey, 2014). This has required steps to be taken to strengthen user authentication. Various methods of user authentication have been explored, ranging from passwords to pattern recognition, image passwords, and even biometrics. Currently, biometrics seem to be effective in terms of security, however, they are not ubiquitous. Although biometric hardware is costly, there has been a wide proliferation of this hardware over the past few years, especially in mobile devices such as mobile phones and tablets. Nevertheless, there are those in the population who do not have access to this hardware, especially low-income earners. Another effective security method is multi-tier authentication. However, this was found to have a negative impact on usability due to the sheer iterations of user authentication required (Dmitrienko et al., 2014; Fujita, Inomata, & Kashiwazaki, 2019).

Before explaining the Shannon Entropy theory, it is important to have a basic technical understanding of a typical password cracking process. This process also applies to cracking passphrases. There is a common misperception created by the entertainment industry that a password can be cracked by guessing a password character by character (Gagneja & Jaimes, 2017; Gordon, 2010). In many spy movies an agent would begin with the first character and run through a list of all possible characters the user could

have used. Once this character is identified, the same process is applied to the second character and the process is repeated until all characters of the password are identified. This approach would allow any password to be cracked in a couple of minutes at most. In reality, a password can only be cracked by identifying all characters at the same time. This makes it much more difficult for an attacker, as a perfect match to the password needs to be achieved.

In the past it was acceptable for passwords to be created with no minimum requirements. However, as a result of advances in hacking software and hardware, system development teams were required to add certain security measures to prevent attackers from successfully guessing a user's password. These minimum requirements are character-based and are known as password policies. Password policies reduce the probability of a hacker guessing a user's password; i.e. without password policies, it would be possible for a user to create a password that is only three characters in length and consists of only numbers. This would take a maximum of N guesses (3^{10} where 3 = number of characters and 10 = number of options per character).

The following method was used to assess the entropy of a password. We assumed that lowercase letters will be tried first by a hacker, as these are usually the most common characters used in a password (Abinaya & Sigappi, 2018; Jansen, 2004; Ur et al., 2015). If a hacker is aware that the systems password policy requires a minimum character length of six characters, then the hacker will most likely begin by trying all combinations of passwords that include six characters. This is common, as hackers are aware that users usually try to meet the minimum requirements of a password policy (Boonkrong, 2012; Vittori, 2019).

From the above discussion on password cracking it can be seen that there are a number of uncertain variables that need to be considered. The list below provides a few common variables that researchers have identified which could influence the entropy of a password.

- Types of hardware an attacker is using (linking graphics processing units (GPUs) together increases the speed of guessing passwords)
- Types of software the hacker is using (certain software can prioritise specific passwords based on data provided by the hacker, e.g. user date of birth, name, surname, preferred language and city)
- Hackers' knowledge of the password policy (excludes guessing password that do not align to password policy)

- Hackers' knowledge of the user (phishing can be used to prioritise password guessing types)
- Type of password the user creates (minimum compliance to password policy or exceeding password policy requirement)

From this list above it can be seen that password entropy is influenced by a number of different unknown variables. This is the reason why a comprehensive and accurate method for assessing password/passphrase strength has yet to be discovered. Many researchers have proposed methods which have been disproved in the past. Therefore, it is important to note that the Shannon Entropy theory used in this study should be seen as an estimated indication of password strength. The next section identifies the forms of authentication that can be used to protect users from hackers.

3.4 Forms of Authentication

Monrose and Rubin (2000), Pansa and Chomsiri (2018), Sawant, Nagargoje, Bora, Shelke, and Borate (2013) and Teh, Teoh, and Yue (2013) classify forms of authentication into three groupings. These groupings were created based on what a user uses to authenticate themselves on a system. The three groupings are listed below.

1. **Knowledge** – Authentication by verifying what a person knows (e.g. a password, passphrase or personal identification number (PIN)).
2. **Token** – Authentication by verifying what a person owns (e.g. access card or OTP, a designated subscriber identification module (SIM) card).
3. **Biometrics** – Authentication by verifying personal traits (e.g. fingerprint scan or retinal scan).

Every form of authentication that currently exists can be organised into one of the above categories and each provides different levels of security and usability, i.e. some forms of authentication have high security and low usability, while other forms of authentication offer lower security and higher usability. In order to understand how the proposed solution to the problem being addressed in this study (passphrases and keystroke dynamics) fits into the above categories, further research is needed on the above categories. At this point it can be said that passphrases are grouped into the knowledge category and researchers have classified keystroke dynamics into the biometrics category (Sawant et al., 2013; Teh et al., 2013). These two authentication groups are discussed in more detail below. Passwords and passphrases are classified as a

knowledge form of authentication. This type of authentication is discussed in the next section.

3.4.1 Knowledge

This part of the chapter provides a discussion on the first tier of authentication, passwords/passphrases. The first section identifies the different types of knowledge as a form of authentication and the next section identifies various types of text-based authentication. This is followed by the next section that defines the difference between passwords and passphrases. This is followed by a discussion on password policies and then strength indicators. The components of passwords/passphrases that influence security are then summarised in a diagram to conclude this section of the chapter.

3.4.2 Forms of Knowledge

All forms of authentication in this group (knowledge) can be classified according to methods of input. These categories are as follows:

- **Text (typing)** – keyboard
- **Selection (clicking)** – mouse, finger or hand.
- **Movement (patterns)** – mouse, finger or hand

Table 3-1 below presents a number of knowledge-based authentications.

Table 3-1: Types of Knowledge-based Authentication

Authentication Type	Description	Knowledge Types
Passwords	Combination of different types of characters, i.e. letters, numbers and special characters (Fatima, Siddiqui, Umar, & Khan, 2019; Scarfone & Souppaya, 2009).	Text
Passphrases	Sequence of words (Dooley, 2018; Scarfone & Souppaya, 2009).	Text
Pass-algorithm	User enters the next sequence of a set of characters presented by the system. Example: the system will present ABC _ _ _ and the user will have to enter DEF. The main advantage of this method is that the pass-algorithm will be different on every login (Gao, Kim, & Udayan, 2018; Haskett, 1984).	Text
Cognitive passwords	A series of personal questions that only the specifically authorised user would be able to answer correctly. Nowadays, this form of knowledge authentication is used in the form of security questions if a user forgets their password (Gao et al., 2018; Zviran & Haga, 1990).	Text
One-time pins (OTP)	Primarily used as a second tier of authentication or a temporary password if a user forgets their login credentials. An OTP is a password randomly generated by the system that normally has an expiration period and can only be used once (Brostoff & Sasse, 2000; Jermyn, Mayer, Monroe, Reiter, & Rubin, 1999; Saleh & Mashhour, 2018).	Text
PassPoints	System presents the user with an image/s and the user needs to click on certain areas of the image in the correct sequence (Davis, Monroe, & Reiter, 2004; Meng, Zhub, Liac, Hand, & Lie, 2019)	Selection
Graphical passwords	Commonly found on touch-screen devices where the user has to draw a specific pattern that matches the pattern defined by the user on registration (Gao et al., 2018).	Movement

Table 3-1 above provides some examples of the three forms of knowledge-based authentication. This study focused on text-based knowledge authentication. More specifically, the study assessed whether passphrases can be used instead of passwords and whether passphrases can be used in conjunction with keystroke dynamics as the second tier of authentication. It is important to assess whether passphrases can replace passwords as the majority of systems are utilising password authentication. This is

creating usability issues as systems are forcing users to create passwords that are difficult to remember and to insert into the system. The next section introduces the two types of knowledge-based authentication that were the focus of this study – passwords and passphrases.

3.5 Passwords and Passphrases Defined

This section is necessary as current research seems to make assumptions based on the difference between passwords and passphrases. This was discovered when it was found that researchers have an inconsistent view of what exactly defines a passphrase when compared to a password (Dooley, 2018; Holstein, 2006; Scarfone & Souppaya, 2009; Turan, Barker, Burr, & Chen, 2010; Zviran & Haga, 1993). This section clearly defines the difference between a password and a passphrase.

A conventional password can be as simple as “password” to a more complicated form such as “P@\$5w0rd”. A passphrase on the other hand is a sequence of words. No specific criteria were found that clearly define the difference between a password and passphrase. However, this study requires a clear differentiation between passwords and passphrases in order to make the most accurate assessment possible regarding the difference in strength between a password and a passphrase. The National Institute of Standards and Technology (NIST) also did not provide a detailed difference between passwords and passphrases (Burr, et al., 2017). Table 3-2 below provides examples of various text-based authentications which are classified as a password or passphrase. This is the best method to create and ensure clarity in the rules that define a password and a passphrase. Table 3-2 provides an example of a password or passphrase and the rule that defines it. Table 3-2 was formulated based on research conducted by Novoselov, Kudashev, Shchemelinin, Kremnev, and Lavrentyeva (2018) and Shay et al. (2014).

Table 3-2: Password and Passphrase Defined – Adapted from: (Novoselov et al., 2018; Shay et al., 2014)

No.	User Authentication Examples	Authentication Type	Rule
1	Password	Password	One word
2	P@\$5w0rd123	Password	Any special character
3	myfacebookpassword	Passphrase	More than one word
4	MyFacebookPassword	Password	Any uppercase characters to separate the words in the phrase
5	My_Facebook_Password	Password	Any special character to separate the words in the phrase
6	mypass	Password	More than one word but only 6 characters long
7	MyFacebookPassword2015	Password	Any numbers added to a phrase
8	MyF@cebookP@\$W0rd	Password	Any special characters added to a phrase
9	strongfacebookpassword	Passphrase	More than 16 characters long

Based on Table 3-2, a passphrase is defined by three strict rules:

- The passphrase must have more than one word.
- The passphrase must be 16 or more characters in length.
- The passphrase cannot have any special characters, uppercase letters or numbers in it.

All three of the above bullets must be accounted for in order for a text-based authentication to be labelled a passphrase. If the above criteria for passphrases are not met, the text-based authentication can be labelled as a password. The next section discusses how password policies influence security from an authentication perspective.

3.6 Password Policies

This section aims to identify what factors should be considered in terms of usability and security when implementing a password policy on a system. There are a number of different methods for applying a password policy. However, there are a number of factors that need to be considered before a password policy is implemented.

Consideration of these factors will assist in determining how the password policy should be implemented on a system.

Users seldom change their passwords as they perceive it to be a frustrating process in terms of time spent creating one and effort exerted to memorise the newly created password (Houshmand & Aggarwal, 2012; Rajkumar, Dhurka, & Kayathri, 2016). Users may be aware of the importance of creating a secure password; however, studies have found that users do not know how to create strong passwords (Adams et al., 1997; Charoen, Raman, & Olfman, 2008; Sannicolas-Rocca, Schooley, & Spears, 2014; Schulze, 2018).

Researchers have found that password policies have an influence on how a user creates a password (Alomari & Thorpe, 2019; Bhivgade et al., 2014). Password policies form part of the information security policy. Password policies are rules enforced by the system development team on a user when creating a password. For example: the password created must be a LUDS password. Password policies can be implemented by the system development team in a mandatory or optional manner. A mandatory password policy would not allow a user to create a password that does not comply with the system's password policy. Optional password policies merely indicate the strength of the password created to the user, even though the system will accept any password the user creates. Alternatively, a system can have no password policy. However, this is not advisable from a system security perspective.

Bhivgade et al. (2014) and Golla and Dürmuth (2018) found that users regarded a password meter which measures the password strength as important. Currently, many systems provide an indication of the password strength but do not communicate the reason for the strength result to the user. Some users can identify how the password result is derived by trying a few passwords and monitoring changes in the strength indicator. If the password policy rules are indicated in text, the user can also determine how the password indicator works based on the level of compliance to the password policy. The password policy rules are usually provided on systems that apply a mandatory password policy.

From a usability and security perspective, it is important that the user understands how the password policy assesses the strength of the password created. It may be advisable to go as far as providing the user with the reasons why the password created is weak or strong. For example, a created password results in an indication of medium strength. In this case, the system displays the result together with the reason, for example "medium

strength password was created as no uppercase letter was included in the password". In this method, the user understands why the password created is not strong and the system also suggests what the user needs to do to change that.

Having a password policy that is too strict often results in users creating passwords that just meet the password policy rules instead of exceeding them. This also makes it easier for hackers to tailor a password guessing attack to exclude passwords that do not conform to the password policy. This means that fewer password guesses are required, which leads to less time required to guess the password, i.e. the password attack can be completed faster. For example, if the password policy only excludes numeric passwords from being created, the password cracking algorithm can be set to exclude such passwords. This reduces the number of password guesses required, as the password guessing list has been reduced due to the numeric password exclusions. It should also be noted that stringent password policies have been found to annoy users (Bhivgade et al., 2014; Hussain et al., 2018).

Although password policies have improved security to an extent, they do have limitations. Password policies do not reduce the reuse of passwords across different systems or the security risks attached to the meaningful information users add to the passwords that they create (Weir, Aggarwal, Collins, & Stern, 2010; Yildirim & Mackie, 2019). Sahin, Lychev, and Wagner (2015) explain that the imposition of strict password policies can make it easier for hackers to crack the password. For example, if the hacker understands what passwords are restricted because of the password policy, filters can be added to the password cracking software that reduces the amount of time the cracking software takes to match the password. The stricter the password policy the more usability is reduced (Alomari & Thorpe, 2019; Buhrmester, Kwang, & Gosling, 2011; Maini, Kimmatt, Cunningham, & Vaughan, 2013). A proposed solution for creating the best password policy was a constantly evolving password policy attached to an enormous database (Wimberly & Liebrock, 2011). This database would store every single stolen password on the internet and update the password policy accordingly. Schechter, Herley, and Mitzenmacher (2010) suggest a solution for unreasonable or unreliable password policies whereby users are allowed to blacklist password policies that they deem to have negative implications for security or usability. However, for security purposes, this approach would only be possible for public domain password policies. Another password policy approach which is now deemed obsolete is the analyse-and-modify approach

where a dictionary check was run on a password to indicate its strength (Chiasson et al., 2009; Kuo, Romanosky, & Cranor, 2016).

In summary, it is advised that a lenient password policy be enforced but the system should continue to indicate to the user that although the password complies with the password policy and the password will be accepted, the password is of a medium strength. In this way, a user has the option to create a stronger password than the medium strength password. This applies to passphrases as well. The next section explains the impact of strength indicators and their influence on a user creating stronger passwords/passphrases.

3.7 Strength Indicators

Simply displaying the strength of a created password/passphrase can have a sizable effect on usability and can even give rise to an increase in security (Bhivgade et al., 2014; Golla & Dürmuth, 2018). Systems usually align the strength indicator with the password policy. This alignment, regardless of whether alignment is mandatory or optional, can be presented in a number of ways, for example graphically, through text, or a combination of both.

The data used to indicate the strength of a password/passphrase is also an important factor to consider. Most systems use rules in the backend of a system to assess the number of character sets used and the length of the password/passphrase created and then assign a value to the password/passphrase. Based on the value, it is assigned a status, for example weak, medium or strong, which is usually displayed to the user. However, users do not understand what this actually means. Renaud and Zimmerman (2017) and Golla and Dürmuth (2018) performed a study where instead of providing a status to indicate the strength of a password, the estimated amount of time it would take a hacker to crack the password was displayed. This could influence users to create stronger passwords/passphrases as they were presented with valuable information rather than just a status that did not clearly indicate the consequences to the user.

Bhivgade et al. (2014) took a more pleasant approach to influencing a user to create a strong password. A bunny was presented on the password creation screen, which began to dance as a password was created. The speed at which the bunny danced was aligned to the strength of the password created by the user – as the password created became stronger, so the bunny would dance faster. In this study, it was found that some users

were curious to find out how fast the bunny could dance. This influenced them to create a stronger password than they would have if the bunny had not been there.

The presentation of password strength also has an impact on the password a user creates (Bhivgade et al., 2014; Golla & Dürmuth, 2018). The positioning of the strength indicator for the password/passphrase on the screen is also an important factor to consider in terms of security. The user should not need to search for the strength indicator but it should attract the user's attention. Some users who do not automatically see the indicator may assume that no indicator is provided which may lead to a weak password/passphrase being created. Therefore, it is suggested that the strength indicator be positioned closely to the field where the user inserts the new password/passphrase. Further, the strength indicator should stand out to ensure that the strength indicator is not camouflaged by the background or other components on the screen.

Lastly, a strength indicator may be synchronous or asynchronous. Asynchronous strength indicators inform the user after the password/passphrase has been created using a button (usually "check password" or "create account") that initiates the communication of the created password strength. Synchronous strength indicators provide live feedback to the user as they are typing the password/passphrase into the system. Bhivgade et al. (2014) and Golla and Dürmuth (2018) found that this was the best approach for indicating password strength to a user. It was found that some users played with the strength indicator merely to understand how it works (Althubaiti, 2017; Bhivgade et al., 2014), for example the impact of adding an additional special character or a number to the password. This method also gives the user immediate feedback on the strength of the password/passphrase created as opposed to going back and forward with an asynchronous strength indicator approach. The next section summarises the components of passwords and passphrases identified from the discussions above.

3.8 Summary of Password and Passphrase Components

Figure 3-1 below graphically summarises the discussion above on passwords and passphrases.

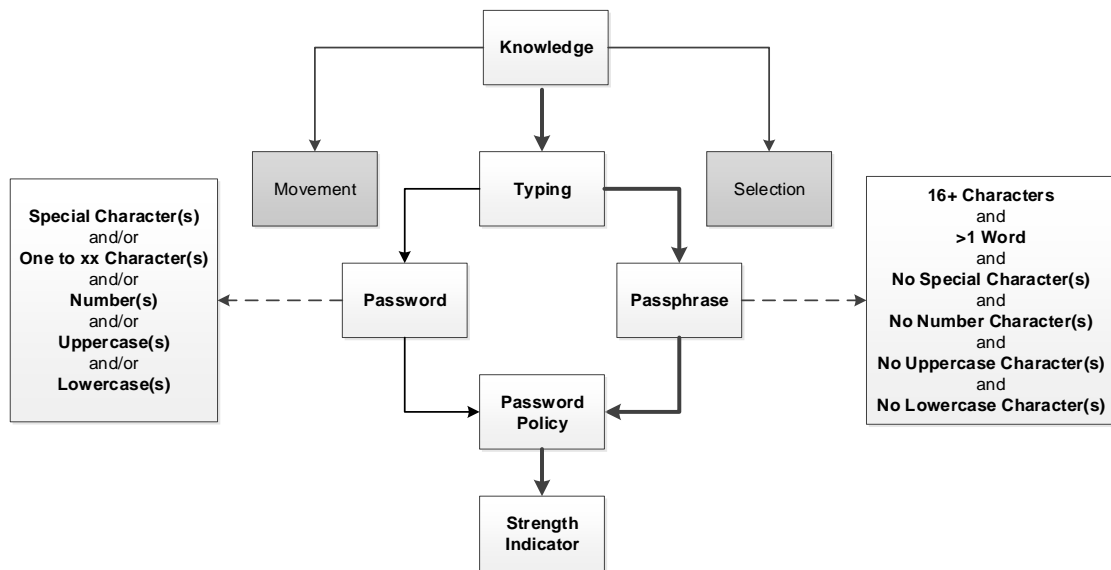


Figure 3-1: Components of Passphrases

The greyed-out blocks in Figure 3-1 indicate the areas in the knowledge category of authentication that are beyond the scope of this study. Passwords and passphrases are text-based authentication methods that fall into the knowledge category of user authentication. In this category, there are three subcategories – movement, typing and selection. Both passwords and passphrases form part of the typing sub-category of knowledge-based user authentication. The two blocks at the bottom of Figure 3-1 (“password policy” and “strength indicator”) depict the characteristics that both define and differentiate a password from a passphrase. Each of these characteristics either has a positive or negative impact on usability on different levels.

This concludes the discussion on the components of the first tier of the proposed solution. The next section focuses on the second tier of authentication with regard to the proposed solution.

3.9 Biometrics

This section includes more detail on biometrics as a form of authentication. Recall, biometrics includes methods of authentication classified under “what we are”, such as fingerprints. Keystroke dynamics is classified as a behavioural biometric form of authentication. A brief explanation of keystroke dynamics and how it works is then provided. Different types of keystroke dynamics exist and these are discussed in the next section. Trackers and measures for keystroke dynamics are then discussed, followed by the levels of leniency that can be set for keystroke dynamics. This is followed by a discussion of the restrictions that can be imposed to assist keystroke dynamic

accuracy and security. A diagram was then constructed to summarise the findings of the above components of keystroke dynamics. This is followed by a section discussing a few implementation considerations. Lastly, ways in which the components can be used to address some limitations and concerns that researchers have raised regarding keystroke dynamics are discussed.

3.9.1 Types of Biometrics

In Section 3.4, Forms of Authentication, it was found that biometrics is a form of authentication. Two forms of biometrics exist – physical biometrics and behavioural biometrics. This aligns to the definition of biometrics by Allen and Komandur (2019) and Sawant et al. (2013) where they explain that biometrics is a method of authentication using a person's characteristics or traits. Physical biometrics utilises a person's unique physical characteristics to authenticate themselves on a system (Allen & Komandur, 2019; Teh et al., 2013). Examples of physical biometrics are voice recognition, fingerprint scanners and facial recognition. Behavioural biometrics, on the other hand, is a method of authentication that observes how a person executes certain tasks which can uniquely identify them. Examples of behavioural biometrics are gait recognition (monitors walking patterns) and keystroke dynamics (second tier of authentication for the proposed solution).

3.10 What is Keystroke Dynamics

This section provides a basic explanation of how keystroke dynamics typically works. It is best explained using a diagrammatic example. Figure 3-2 illustrates the steps involved in a typical keystroke dynamic setup. Note that Figure 3-2 was derived from research papers explaining the keystroke dynamics backend process (Abinaya & Sigappi, 2018; Banerjee & Woodard, 2012; Bergadano, Gunetti, & Picardi, 2002; Giot, El-Abed, & Rosenberger, 2011). The steps below provide a description of the numbers included in Figure 3-2.

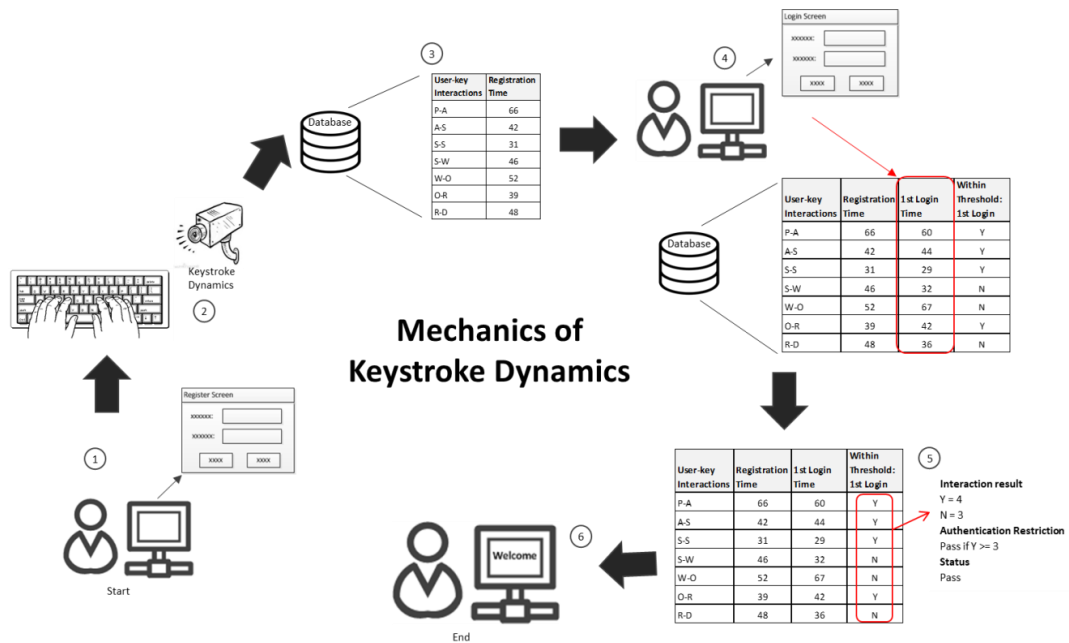


Figure 3-2: Mechanics Behind Keystroke Dynamics – Adapted from: (Banerjee & Woodard, 2012; Bergadano et al., 2002; Daribay, Obaidat, & Krishna, 2019; Giot et al., 2011)

1. A user navigates to a system and is presented with the system user authentication page (login page).
2. The user inserts their password/passphrase on the registration page of the system. As the user is inserting their password/passphrase into the system, in the background the system is monitoring (in milliseconds) the way in which the user is typing the password/passphrase into the system. Note that the system monitors the user interaction through the keyboard. In other words, in keystroke dynamics authentication, the way in which the user interacts with the keyboard is used as the tool for user authentication.
3. As the user inserts their password/passphrase (in step 2), the system records the user's interaction with the keyboard and saves it to the database. A number of interaction sets can be recorded (this is discussed in detail in Section 3.10.2, Types of Trackers and Measures for Keystroke Dynamics) by the system, based on how the keystroke dynamics algorithm was programmed.
4. During login, each user-keyboard interaction set is matched to the user's interaction set that was recorded during registration. In order to keep this example simple, the interaction sets that have already been recorded by the system will be the ones collected when the user registered on the system.

5. Depending on how the keystroke algorithm was programmed, a certain amount or all of the interaction sets need to be matched. A detailed discussion of this can be found in Section 3.10.3, Leniency, in this chapter.
6. The user has successfully been authenticated and is granted access to the system.

Now that a better basic description of the way keystroke dynamics works has been provided, the next few sections discuss where keystroke dynamics fits into the user authentication world and what needs to be considered when setting up a keystroke dynamic algorithm on a system. The next section identifies the different types of keystroke dynamics.

3.10.1 Types of Keystroke Dynamics

As discussed above, keystroke dynamics forms part of behavioural biometrics. According to researchers, two types of keystroke dynamics exist, static and non-static (Banerjee & Woodard, 2012; Monrose & Rubin, 2000; Saini et al., 2018; Teh et al., 2013). However, before these two types can be explained, a basic understanding of keystroke dynamics is required.

Keystroke dynamics, also referred to in research articles as keystroke recognition, is a backend security solution. Keystroke dynamics is a type of algorithm that monitors user-keyboard interaction on a system. Various measures exist for monitoring a user's keyboard interactions. For example, the length of time a user holds down a certain key on the keyboard and the length of time between the keys being pressed on the keyboard. These are discussed in more detail in Section 3.10.2, Types of Trackers and Measures for Keystroke Dynamics.

Following the brief explanation of keystroke dynamics, it is important to differentiate between the two types of keystroke dynamics:

- **Static** – Static keystroke dynamics measures user-keyboard interaction at specific states. This can be on a specific screen, such as a login page.
- **Non-static** – Non-static keystroke dynamics measures user-keyboard interaction continuously throughout the system. Monitoring occurs from the time a user begins to interact on the system (i.e. login page) to the point where they log off the system.

Although researchers only refer to two types of keystroke dynamics, the two types presented above are the two extreme types. In order to accommodate a mixture of both,

a third type of keystroke dynamic should be introduced. For the purposes of this study, this was referred to as semi-static keystroke dynamics.

Semi-static keystroke dynamics combines static and non-static characteristics. It measures user–keyboard interaction for a specific period of time or at specific states. For example, keyboard interaction can be monitored at the login page and for the first 10 minutes after successful login on a system.

All three types of keystroke dynamics have a number of advantages and disadvantages. A non-exhaustive list of these advantages and disadvantages is presented in Table 3-3 below.

Table 3-3: Types of Keystroke Dynamics – Adapted from: (Banerjee & Woodard, 2012; Monroe & Rubin, 2000; Saini et al., 2018; Teh et al., 2013)

	Static	Semi-static	Non-static
System Speed	Fast	Moderate	Slow
Security	Low	Moderate	High
Authentication Accuracy	Low	Moderate	High

Constantly running the keystroke dynamic algorithm in the background may significantly slow down the response time of the system which may have a negative impact on usability. That being said, although static keystroke dynamics may not have much of an impact on system response time, it does have a reduced level of security as opposed to non-static keystroke dynamics. This is because the former will not accommodate the identification of breaches in login after a user has been successfully authenticated. For example, if a user successfully logs in and either forgets to log off or walks away from their device, an unauthorised person may gain access to the system. Non-static keystroke dynamics has the capability to identify such threats. In terms of the accuracy of the keystroke dynamic algorithm, non-static keystroke dynamics will have a more accurate view of the user as more user–keyboard interaction data is being collected. This can create a more lenient interaction monitoring approach that can accommodate changes in user–keyboard interaction behaviour. For example, the longer a user interacts with the system, the slower their typing patterns become.

Based on the above, it may seem that a non-static keystroke dynamic algorithm offers the best combination. However, no specific type of keystroke dynamic is better than another. Accordingly, the system development team should carefully analyse which is

the best type of keystroke dynamic for the system based on the purpose, content and user interaction of the system.

The intention of this discussion was to understand how flexible keystroke dynamics can be and not which type of keystroke dynamic is best. Since this study proposed that keystroke dynamics be monitored on the login page, static keystroke dynamics should be utilised. The next section focuses on the types of trackers and measures available when developing a keystroke dynamics algorithm.

3.10.2 Types of Trackers and Measures for Keystroke Dynamics

The previous section explained that there are three types of keystroke dynamics. In addition, there are different types of trackers that span all three types of keystroke dynamics. This section discusses the types of trackers associated with keystroke dynamics.

Raul, Shankarmani, and Joshi (2019) and Teh et al. (2013) explain that there are two main states trackers use to authenticate the user–keyboard interaction of a specific user – dwell time and flight time. Dwell time refers to the user pressing a key and flight time relates to the time taken between pressing the keys.

The following trackers can be used to record user–keyboard interaction.

- **Tracker 1** – Key Down 1 to Key Up 1
- **Tracker 2** – Key Up 1 to Key Down 2
- **Tracker 3** – Key Down 2 to Key Up 2
- **Tracker 4** – Key Down 1 to Key Down 2
- **Tracker 5** – Key Up 1 to Key Up 2
- **Tracker 6** – Key Down 1 to Key Up 2

In the example above, dwell time trackers are classified as the time between key down and key up (trackers 1, 3 and 6). Flight time trackers are classified as key down and key down, and key release and key release (trackers 2, 4 and 5). These trackers can be recorded by the system as computers have the capability to record time in milliseconds. Therefore, recording user–keyboard interaction has a high accuracy rate.

Monrose, Reiter, and Wetzel (2002) explain that in addition to simply measuring duration of time between specific interactions (the six trackers listed above), other methods are also possible, for example average time, standard deviation, minimum time and maximum time. For the purposes of this study, these are referred to as the types of

measures of trackers. It should be noted that hold and press keys can have the same trackers but different measures. For example, on a traditional keyboard the user needs to hold down the “shift” key and select the “4” key to get a “\$” sign.

Research has also been conducted to increase system security further by measuring user–mouse interaction (Ponkshe & Chole, 2015) and user pressure on keys (Banerjee & Woodard, 2012; Teh et al., 2013). Banerjee and Woodard (2012) and Tsai and Shihb (2019) agree that keyboard pressure is possible; they explain that many traditional keyboards do not have the hardware function to track user pressure on the keyboard. At this time, the use of keyboard pressure as a prospective security measure may have more of a negative impact on usability. However, it is still possible to have the keystroke dynamic algorithm allow for keyboard pressure measures as an optional security protocol. This security protocol should only be mandatory if the system identifies that the user is interacting with a pressure sensing keyboard. This provides the user with the option to increase their personal system security by purchasing and using a pressure sensor keyboard when interacting with the system.

Banerjee and Woodard (2012) explain an additional measure which can be used to measure the length of time to type a specified number of characters. This may even provide an opportunity for a system to remove the text-based authentication process and provide keystroke dynamics as the only authentication method. This can be accomplished by the system simply asking the user to type a random sentence or short paragraph displayed to them by the system in order to authenticate themselves. This removes the need for a user to remember a password/passphrase or pattern for authentication. However, keystroke dynamics does not seem to have evolved enough yet to cater for such an approach effectively.

3.10.3 Leniency

The previous section discussed the various methods that can be used to track user–keyboard interaction. This section explains how the trackers can be used to alter the leniency of security protocols. This is important as a too stringent security protocol for keystroke dynamics may have a negative impact on usability as the system may prevent the authorised user from accessing the system. Alternatively, setting security protocols for keystroke dynamics that are too lenient may compromise the system. One of the big criticisms of keystroke dynamics is that it does not make allowances for user drowsiness, fatigue, injury or user–system interaction on an unfamiliar device (Carstens et al., 2014;

Hayes, 2016). Setting a keystroke dynamic algorithm to the correct leniency levels can, to a certain extent, prevent these user behaviour changes from affecting usability.

Any form of leniency can be organised into one of the following categories:

- Measurement-specific leniency
- Tracker-specific leniency
- Optional vs mandatory leniency
- User selective leniency

Tracker-specific leniency and measure-specific leniency are discussed together as the reasoning behind this discussion relates to both forms of leniency. The previous section discussed the various types of trackers and measures that can be used for keystroke dynamics. Although having all trackers and measures present in a keystroke dynamic solution significantly increases security, this may have a negative impact on usability. This is because keystroke dynamic algorithms are not yet mature enough to ensure that a user can be successfully identified with a 100% success rate. Therefore, the more trackers and measures included in a keystroke dynamic algorithm, the greater the risk of negatively affecting usability. On the other hand, having too few trackers and measures will reduce the security level of the system. It is therefore important to select the correct combination of trackers and measures to achieve the best balance between security and usability.

In terms of the tracker-specific leniency and measurement-specific leniency discussed above, it is important to select the correct trackers and measures to find the correct balance between usability and security. Mandatory vs optional leniency is an approach that may complement tracker-specific leniency and measurement-specific leniency. Mandatory vs optional leniency can allow for all types of trackers and measures to be included in a keystroke dynamic algorithm. The algorithm can then specify which trackers and/or measures need to be passed (mandatory leniency). Alternatively, the algorithm can state that of the full list of trackers and measures, a certain percentage needs to be passed (optional leniency). For example, out of a total of six measures, any four of the six measures need to be passed. This approach makes room for failure, which mitigates the risk of reduced usability if the system blocks an authorised user from entering the system because of minor deviations in user–keyboard interaction.

User selection leniency is another option that has not yet been extensively researched. This option would provide the user with an interface to select which tracker and

measurement options they would prefer. This allows the user to toggle their security leniency for keystroke dynamics without having to strengthen their password or passphrase. This should be offered to the user on the user registration page. Thereafter a link on the login screen can be provided that allows the user to change the security leniency level at any time. One benefit of this approach is that by providing the user with the option to drive user authentication protocols to a certain extent, their level of frustration may change if authentication fails. The reason for this is that full blame cannot be directed at the system as the user has already been given the option to reduce the leniency of the keystroke dynamics algorithm. The user must, however, be given a basic explanation of keystroke dynamics before allowing them to modify security leniency for keystroke dynamics. In order to reduce the risk of the user misinterpreting or not understanding the explanation, which may result in a negative usability experience, a simple slider can be offered for the user to toggle the security leniency for keystroke dynamics. The slider can provide three, five or X options to increase or reduce security leniency. Although the minimum security leniency offered to the user can include no keystroke dynamic trackers and/or measures, it is advisable that it has a few keystroke dynamic trackers and/or measures included in it. This is to prevent the user from completely deactivating the keystroke dynamics algorithm.

The next section discusses how restrictions can be imposed to assist keystroke dynamic accuracy and security.

3.10.4 Imposing Restrictions on Keystroke Dynamics

Restrictions can be used to limit the risk of authentication failure resulting from user handicaps and to increase the accuracy of authentication. Restrictions can be imposed by a system running a keystroke dynamics algorithm. These are usually applied on login screens where the system will instruct the user to use only a portion of the keyboard during a part of or the entire authentication phase (Babaeizadeh, Bakhtiari, & Maarof, 2014; Raul et al., 2019; Teh et al., 2013). For example, during the authentication phase a user should only use the number (digits) keys or only lowercase letter. For the purposes of this study, this is referred to as keyboard restrictions. However, this restriction may be difficult to impose if the keystroke dynamics algorithm is running behind a login page. Also, if a user's password includes numbers and letters, it will not be possible to have such a restriction. To address this limitation, it is also possible to impose physical restrictions on the user (Raul et al., 2019; Teh et al., 2013). The system

can, for example, instruct the user to only use one hand or one finger during a portion of or the full authentication process.

A system that is running a keystroke dynamic algorithm assumes that the user is utilising two hands when interacting with the system. If a user injures his/her hand and cannot use it to interact with the system, the system may identify the user as an unauthorised party as his/her typing pattern has changed. For example; he/she is now only able to use one hand to interact with the system. Although leniency levels assist with these types of potential failures, they do reduce the level of security. Hence, restrictions are another option for accommodating these unfortunate user events.

Restrictions also allow for more accurate authentication as they limit the number of possibilities. For example, with no restriction the keystroke dynamics algorithm should accommodate a user who wants to use one hand, two hands or even one finger to interact with the system.

Users can select their restrictions on the registration page; for example, only use one finger, the right hand or two hands. However, this may reduce security as the hacker can now filter the guesses of different speeds of login based on the restriction the user selected. An alternative approach would be to only select the restriction on the sign-up screen of a system and explain to the user that they need to remember the restriction selected. Although this increases security it may have a negative impact on usability as the user now has to remember an additional element. The next section summarises the components of keystroke dynamics identified above.

3.11 Summary of Keystroke Dynamics Components

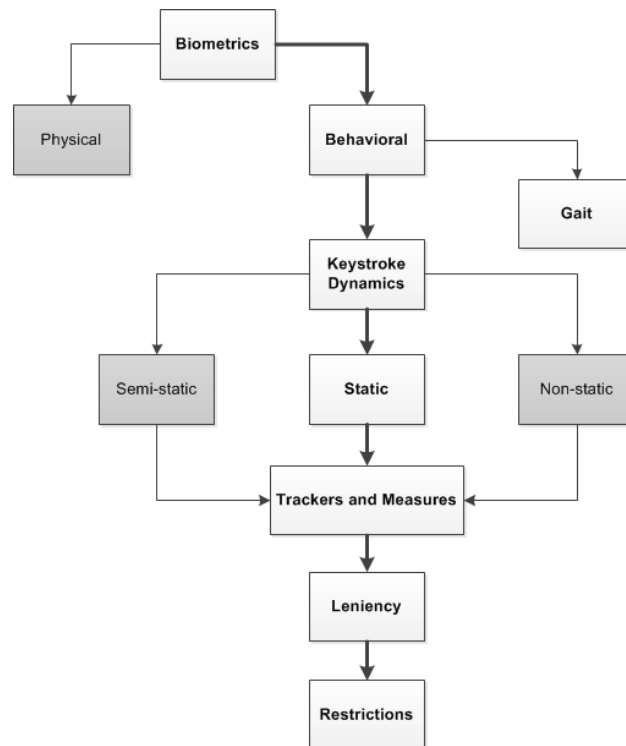


Figure 3-3: Components of Keystroke Dynamics

Figure 3-3 graphically summarises the above discussion on keystroke dynamics. Specific blocks in Figure 3-3 have been greyed out to indicate the areas that are beyond the scope for this study. Nevertheless, it is important to show where this study fits into the spectrum of user authentication.

Keystroke dynamics is a user authentication method which is classified as a form of biometric authentication. All forms of biometric authentication can be classified under behavioural biometrics or physical biometrics, with keystroke dynamics being classified as a behavioural-type biometric because this seeks to authenticate a user based on specific behavioural characteristics. Two examples have been provided in Figure 3-3 – gait authentication and keystroke dynamics. Three states of keystroke dynamics exist – static, semi-static and non-static, all three of which have the capability of utilising the same forms of trackers and measures. All leniencies and restrictions can also be used by all three states of keystroke dynamics. Furthermore, all the components of keystroke dynamics (indicated in the keystroke dynamics grouping in Figure 3-3 above) can be manipulated to tweak the levels of usability and security.

The next section provides a few suggestions that could assist keystroke dynamics solutions.

3.12 Suggestion for Keystroke Dynamics Solution

It is suggested that the strength of the second tier of authentication be adaptive. Levels of security are provided for each user profile and when sufficient data is collected on the user's typing patterns, the security group for that user increases. This may also encourage users to utilise the system more to reduce security threats. In addition, buy-in from the system development team may also be easier to obtain as a result of system security improvements. Users can also choose what level of authentication they would prefer.

If the user selects a level that is above the recommendation, an OTP is used to ensure that they do not get locked out of their account. Offering this option to the user shifts some control over security and usability to them with a minimum impact on usability and a more positive impact on security. The following section provides a response to a limitation or concerns that researchers have raised regarding keystroke dynamics.

3.13 Perceived Keystroke Dynamics Limitations Resolved

A number of concerns regarding keystroke dynamics have been raised by researchers in the discipline. This section discusses how some of these concerns can be addressed temporarily or permanently. Where concerns cannot be resolved an alternative temporary fix is suggested until a more permanent solution is found. The first concern refers to typing inconsistencies.

3.13.1 Typing Inconsistencies

Carstens et al. (2014) and Hayes (2016) explain that keystroke dynamics do not accommodate for injuries, casual typing, fatigue, temporary incapacity (e.g. using one hand) or distractions. Leniency can be used to accommodate some of these changes in behaviour but it is important not to set the leniency level too high as security will then be reduced, which will increase the likelihood of unauthorised entry to the system. Restrictions can also be used to accommodate for hand or wrist injuries, or temporary incapacities (Crawford & Ahmadzadeh, 2017). This is possible if the keystroke dynamics algorithm is set up in a manner that can record a user's keyboard interaction using different restrictions. For example, the keystroke algorithm may have three forms of data restrictions assigned to one user – the user's keyboard interaction using one hand (data restriction 1), both hands (data restriction 2) and one finger (data restriction 3). The user can then use any of the three restrictions to gain access to the system. Note, however, that this will reduce security as now an unauthorised party can guess one of three

interactions instead of just one. In terms of setting up a keystroke dynamics algorithm, restrictions can be an “and” or an “or”. Another concern that researchers have raised relates to the non-permanent nature of keystroke dynamics.

3.13.2 Non-permanent Authentication

Kasiani and Yusuf (2019) and Teh et al. (2013) state that the keystroke dynamics algorithm is not as permanent as physical biometric authentication, as a user's typing pattern can change over time, i.e. a user becomes more proficient as they are exposed to a standard layout keyboard. This is classified as a disadvantage, perhaps because it makes it difficult to set up a keystroke algorithm. However, it is not entirely a disadvantage. The problem with physical biometrics is that if an unauthorised party finds a method to mimic a user's body parts used for physical biometric user authentication then the system is compromised forever and that body part will most likely never be able to be used again for user authentication. The reasoning behind this is because a user's fingerprint or retina remains the same for years. It is recommended that passwords be changed on a regular basis to avoid unauthorised parties from gaining access to a system (Houshmand & Aggarwal, 2012; Jansen, 2004; Rajkumar et al., 2016). The same concept can be applied to keystroke dynamics. Therefore, it is advantageous for keystroke dynamics that users' typing patterns do change over time. However, it does have minor complications when setting up the keystroke dynamic algorithm.

Research has been conducted on allowing the keystroke dynamic algorithm to adapt to natural changes in user typing behaviour after a period of time or a number of specific user interactions with the system (Abinaya & Sigappi, 2018; Epp, Lippold, & Mandryk, 2011). However, this does become complicated as users have differing amounts of keyboard interaction time per week. Although an average improvement can be added to the keystroke dynamics algorithm, it may not accommodate all users. For example, User A is a plumber and interacts with a keyboard an average of two hours per day but only uses the system running the keystroke dynamic algorithm for one hour per day. By contrast, User B is a programmer who interacts with a keyboard on average eight hours a day and uses the system running the keystroke dynamic algorithm 30 minutes per day. The system running the keystroke dynamic algorithm will identify that User A's typing proficiency is increasing faster than User B's, although User B is exposed to a keyboard four times (2/8) more per day than User A.

An alternative approach which may be more successful is developing the keystroke dynamics algorithm to identify minor improvements in users' keyboard interaction (Kim,

Kim, & Kang, 2018; Pleva, Bours, Ondáš, & Juhár, 2017). Once a certain improvement threshold is met, the user's keyboard interaction data is updated. It should be noted that the thresholds need to be sensitive enough to identify that it is the same user and not a user with a similar typing pattern. The keystroke dynamics algorithm also needs to accommodate reductions in typing proficiency displayed by a temporarily incapacitated user, who has not interacted with a keyboard for a short period of time.

Another approach which is similar to the above is to create user proficiency categories. If the user's keyboard interaction is correctly matched to their classification on registration, then access to the system is granted. Examples of categories could be Expert (0 to 300 milliseconds), Intermediate (301 to 500 milliseconds) and Beginner (501 to 900 milliseconds). A three-category classification may be too weak in terms of security; however, the keystroke dynamics algorithm is flexible enough to create any desired number of categories. Another concern that is raised by research is the reuse of captured typing styles.

3.13.3 Reusing Recorded Typing Styles

Teh et al. (2013) discuss the fact that a user's typing style is stored in a system, resulting in it being replicated to gain access to another system with the same keystroke dynamic algorithm. It may be gathered from the above discussion that there are a number of different segments which are incorporated in a keystroke dynamics algorithm that can be set up in various ways. It is important that the same keystroke dynamic is not used in another system. If it is unavoidable then another tier of authentication should be included in combination with the keystroke dynamics algorithm. Another common concern raised by researchers (Šidlauskas, 2018) relates to the difficulty of accommodating numerous keyboard layouts.

3.13.4 Keyboard Layout

Banerjee and Woodard (2012) and Kasiani and Yusuf (2019) explain that keyboard layout and familiarity with specific keyboards have an impact on a user's typing style. A system may prevent an authorised user from accessing the system because he/she may be using a keyboard that is new or unfamiliar to them. This unwanted result has a negative impact on usability. Since the system cannot differentiate between keyboard layouts, alternative solutions need to be explored. Key down measures can still be used in conjunction with restrictions and limitations but this is only effective to a certain extent without compromising security. In addition, key down will not be responsive if the user

accesses the system through a touchscreen device such as a tablet or mobile phone. Although touchscreens are beyond the scope of this study, it is important to discuss potential solutions at a high level as this is currently one of the biggest drawbacks of keystroke dynamics.

One solution which has already been mentioned is having a user indicate that they are logging in with a desktop computer or laptop, tablet or mobile phone. The system then records three different user–keyboard interactions for one user account – desktop computer or laptop (user–keyboard interaction 1), another for tablets (user–keyboard interaction 2) and mobile phone (user–keyboard interaction 3). Restrictions and limitations can then be used to accommodate a user using an unfamiliar touchscreen device, such as his/her friend’s device, to access the system.

Another solution that is suggested for use with the first proposed solution to accommodate any failure, is providing the user with the option to bypass the keystroke dynamics authentication when trying to login to the system with an unfamiliar device. In this case an OTP is sent to the user’s email address or mobile device. Researchers (Jadhav, Kulkarni, Shelar, Shinde, & Dharwadkar, 2017) have also raised concerns about keystroke dynamics being unable to accommodate a user’s varying physical states during authentication.

3.13.5 Physical State Influence on Keyboard Interaction

Banerjee and Woodard (2012) mention that a user’s posture has an impact on his/her typing style and thus prevent him/her from accessing the system. A user’s posture differs when sitting, standing, walking or lying down. Accordingly, leniency can be used to take into consideration changes in posture when a user interacts with the system. Although posture has an impact on user–keyboard interaction (Karwowski, Eberts, Salvendy, & Noland., 2007), it does not have sufficient influence to disregard leniency as an effective solution. The next section discusses the last limitation of keystroke dynamics, alternative user entries.

3.13.6 Alternative User Entry

Raul et al. (2019) and Sawant et al. (2013) explain that keystroke dynamics does not make it possible for a friend or family member to login to a system on behalf of the user during an emergency. Although this is an unacceptable scenario in terms of system security, it is highly possible that such an event may occur. For this reason, a solution needs to be provided for this occurrence. The acceptable solution in this regard would

be to send the user an OTP. The user could then send the OTP to another user to login to their account. When the OTP is entered into the system, the keystroke dynamics algorithm will not run thus ensuring that the other user is not blocked from accessing the system because of a change in typing behaviour.

This concludes the section on the perceived limitations of keystroke dynamics. The next section introduces the Shannon Entropy theory.

3.14 Entropy of the Proposed Solution

This section of the chapter is focused on assessing the entropy of the proposed solution. Firstly, an explanation of the Shannon Entropy theory is provided before the entropy assessment is conducted to identify the security strength of the proposed solution. The entropy of three different types of text-based authentication is provided for comparability. This is followed by a summary of the findings on the entropy of text-based authentication. The entropy of keystroke dynamics is then assessed to conclude this section of the chapter.

3.14.1 Measuring Password and Passphrase Strength

A number of methods have been proposed for measuring password strength. The common characteristic of all the methods is that it is based on probability. One of four methods seem to appear in password measuring research papers: 1) The Shannon Entropy theory has been used to quantify password strength by presenting a variable which represents the number of guesses required to obtain a password (Aguiar & Guedes, 2015; Arora et al., 2015; Houshmand & Aggarwal, 2012; Kelley et al., 2012). 2) The probabilistic context-free grammar approach believes that guesses for passwords should be based on the prioritisation of password templates. That is, users often add a word first followed by two digits (Melicher et al., 2016). 3) The Markova model uses specific characters based on the user and then guesses the next character based on the previous characters (context characters) (He, et al., 2019; Marechal, 2008; Narayanan & Shmatikov, 2005). 4) A method which does not have a term attached to it entails running through dictionary entries that include special character replacements. For example, the letter “a” is replaced with “@” and the letter “s” is replaced with “\$” (Melicher et al., 2016).

3.14.2 Shannon Entropy Theory

Shannon Entropy is an information theory which assigns a value (represented in bits) to the number of guesses required to successfully obtain specific information. The formula used to calculate this action is presented in Figure 3-4 below.

$$H = - \sum p(x) \log p(x)$$

Figure 3-4: Shannon Entropy Formula (Shannon, 1948)

In Figure 3-4 above, p is the probability of making the correct selection from a known range and x is the total number of available options in the range. A coin toss is used as an example. There is a 50% chance the coin will land on heads and a 50% chance the coin will land on tails, therefore $p = 0.5$. Since there are only two possible outcomes, heads or tails, $x = 2$. Using the formula in Figure 3-4, the entropy of a single coin toss is 1 bit. The next section explains how the Shannon Entropy formula has been applied.

3.14.3 Application of Shannon Entropy Theory

Researchers (Greene, Kelsey, & Franklin, 2016; Houshmand & Aggarwal, 2012; Kelley et al., 2012) have applied the Shannon Entropy theory to estimate the entropy of certain passwords. However, the Shannon Entropy formula needs to be applied in a specific way in order for it to consider password length. It is important for this study to consider password length so a comparison can be made between passwords and passphrases.

A caveat needs to be provided; no measuring tool exists at this point in time to accurately identify the strength of a password owing to the number of uncertain factors and their probability of occurring. Plenty of models have been developed and proposed but further research assessing these models has found that certain aspects were not considered (Greene et al., 2016; Hingmire & Saliya, 2017; Houshmand & Aggarwal, 2012). The reason for this difficulty is the measurement of a number of random and unknown variables, internal and external to the hacking process, Shannon Entropy theory is merely used as an indication of password and passphrase strength for the purposes of this study. The next few sections illustrate the way in which the Shannon Entropy formula was applied to various types of passwords and passphrases.

3.14.3.1 Entropy of a PIN

The only information that the Shannon Entropy formula requires to calculate the entropy of a password is the total number of options in a set. In terms of passwords, the total

number of options in a set would be the total number of characters in the set. For example, if the password can only include numbers (i.e. a PIN), the total number of characters in a set would be 10 (0 to 9) per character in the password. Figure 3-5 graphically describes this example with the four-digit PIN “2493”. Figure 3-5 indicates that for each character in a four-digit PIN, 1 of the 10 options of characters is selected.

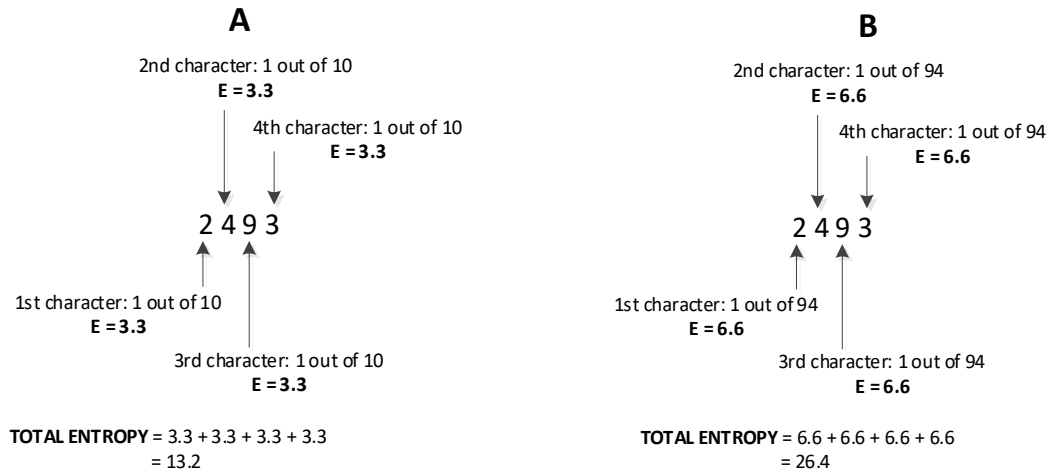


Figure 3-5: Entropy of a Four-digit PIN – Adapted from: (Shannon, 1948)

In the example in Figure 3-5, “2” was selected for the first character from a character range of 10 characters (0 to 9). The Shannon Entropy formula is then applied to each character of the PIN. Once the entropy is calculated for each character of the PIN, the entropy results are added together to obtain a final entropy level. This grand total indicates the level of difficulty involved in cracking the PIN created. The higher the result of the total entropy, the more difficult it is to crack the PIN. This method is indicated in Figure 3-5 where “E” stands for entropy. This figure also displays the entropy for each character. The grand total entropy calculation and result is also provided at the bottom of Figure 3-5 (“total entropy”).

Figure 3-5 is separated into two diagrams to cater for two scenarios; scenario A and scenario B. This is indicated in Figure 3-5 above the diagrams. In Figure 3-5 scenario A, a user is required to create a four-digit PIN with only numbers. The probability of each character is ranked from 1 to 10 which results in the entropy of the four-digit PIN being 13.2 units (3.3 x 4). In scenario A, the user is restricted to creating a PIN using only numeric characters.

In scenario B, the user has the option to create any password with any character set, however they chose to create a four-character length password with only numeric characters. Because the user had the freedom to select any character set, the character

range is no longer 10 (0 to 9) but is now 94 (26 uppercase letters, 26, lowercase letters, 32 special characters and 10 numeric characters). This makes the entropy of scenario B 26.4 units.

As shown in Figure 3-5 above based on the total entropy, there is a vast difference between scenario A and scenario B. The intention of this example was to assess the security impact of the hacker being aware of a password policy imposed on the user as opposed to having no password policy. The entropy of a passphrase is illustrated next.

3.14.3.2 Entropy of a Passphrase

The next example calculates the entropy of a passphrase. Although a passphrase also includes one character set, the character set is larger, and the password character length is longer than the password used in the example in Figure 3-5. A passphrase is used as an example in Figure 3-6 to determine the entropy of the passphrase, “jackspassphrases”.

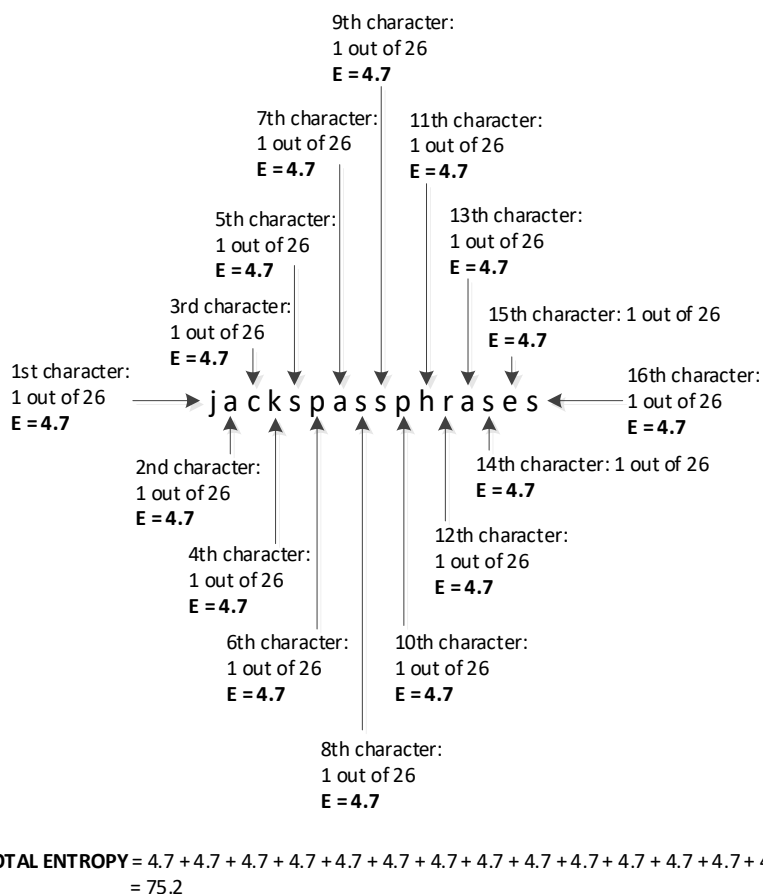


Figure 3-6: Entropy of a Passphrase – Adapted from: (Shannon, 1948)

The passphrase example in Figure 3-6 uses only lowercase letters (character set) and has a length of 16 characters. Since only one character set is used for the passphrase,

the result of the Shannon Entropy formula applied to each character in the passphrase is the same amount of entropy. As Figure 3-6 indicates, the entropy of each individual character is 4.7 units. The individual character entropy results are then added together to determine the total entropy for the passphrase. In terms of this passphrase, the total entropy is 75.2 units.

When comparing the entropy for the PIN, "2493" (entropy = 13.2 units) and the passphrase, "jackspassphrase" (entropy = 75.2 units), the passphrase has a much higher entropy than the PIN. Even with the user freedom that results in a numeric password being created in scenario B in Figure 3-5, the entropy was 26.4 units. This indicates that password length has an influence on the strength of the password. However, the examples provided in Figure 3-5 and Figure 3-6 included only one character set (numeric characters in the Figure 3-5 example and lowercase characters in the Figure 3-6 example). The next section illustrates the entropy of a typical password that includes multiple character sets.

3.14.3.3 Entropy of a Password

Realistically, passwords often include multiple character sets whether it be by user choice or whether enforced by a password policy.

The following are considered to be character sets:

- Special characters
- Uppercase letters
- Lowercase letters
- Numbers

The example in Figure 3-7 uses a variety of character sets in a password and indicates how the Shannon Entropy formula is applied to such a password.

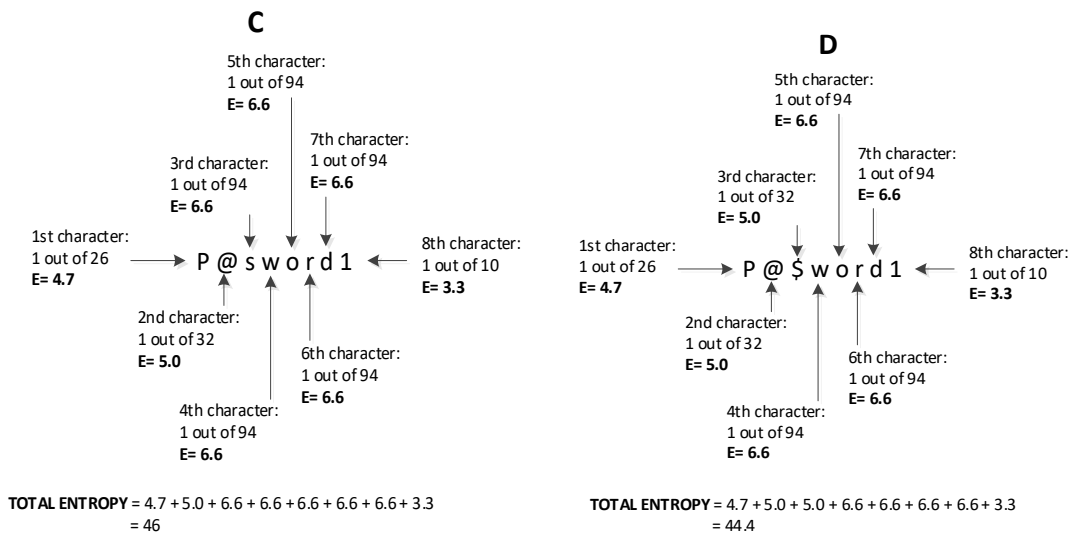


Figure 3-7: Entropy of Passwords – Adapted from: (Shannon, 1948)

For the purpose of explaining the example provided in Figure 3-7, the password in scenario C was created to comply with the following password policy:

1. The password must be more than 7 characters long.
2. It must include at least 1 special character.
3. It must include at least 1 uppercase letter.
4. It must have 1 numeric character.

The password in Figure 3-7 uses an uppercase character to satisfy point 3 of the password policy above. Since the first character of the password is an uppercase letter, the total character set for uppercase letters is 26 characters. Therefore, the entropy for the uppercase character set is calculated, which equates to an entropy of 4.7 units. However, a hacker would not know which character position in the password the uppercase letter appears (i.e. in this case, the password position where the uppercase letter appears is in the first character of the password). However, based on the password policy a hacker knows that the password needs to have at least one uppercase letter. For this reason, for one character slot of the password example in Figure 3-7, scenario C is calculated as one character of a possible 26 (the total possibilities of uppercase letters). The same logic is applied to the next character, which is a special character. In terms of special characters, there are a total of 32 special characters that are accepted for passwords (Kuka & Bahiti, 2018; Matta & Pant, 2018; Yoon & Kee-Young, 2011). In this case the total possibilities are 1 of 32 since the password policy requires at least one character (i.e. slot) of the password to include a special character. This makes the entropy for the second slot/character in the password 5.0 units. The same is applicable to the eighth character of the password in scenario C. Since the password policy requires

at least one number in the password, the entropy of the eighth character is 3.3 units. The remaining characters in scenario C are unknown as a user is not forced to select a specific character set. For this reason, it is assumed that the user will select one of 94-character possibilities (26 uppercase letters, 26 lowercase letters, 32 special characters and 10 numeric characters). This means that for every remaining character slot in scenario C, the entropy is 6.6 units. When summing up all the entropy for each character slot in scenario C, the total entropy is 46 units.

The next example aims to indicate the security impact of a minor change of the password policy used in Figure 3-7, scenario C. The change is indicated in bold text below. The following password policy is enforced for scenario D in Figure 3-7.

1. The password must be more than 7 characters long.
- 2. It must include at least 2 special characters.**
3. It must include at least 1 uppercase letter.
4. It must have 1 numeric character.

This password policy requires two special characters to appear in the password. In scenario D, “s” is replaced with “\$”. By enforcing this password policy, an additional password character slot is expected to be occupied by an additional special character. This means that an additional slot where the user had a choice of 94 characters now only has a choice of 32 special characters. This has resulted in the total entropy being 44.4 units. This shows that the password in scenario D is 1.6 units weaker than the password in scenario C and, thus, that password strength is aligned to the amount of predictability of the user in selecting certain characters. Therefore, a too stringent password policy may in fact weaken security instead of strengthening it. However, making password policies more stringent, for example, enforcing rules such as saying that a user cannot repeat characters or the password cannot be the same character as the username, may increase security but will compromise usability. A summary of the above entropy results is discussed in the next section.

3.14.3.4 Summary of Entropy

As indicated in Figure 3-7, the entropy is calculated for each character of the PIN/password/passphrase and then totalled to determine the total entropy. In summary, the total entropy for each of the PIN/password/passphrase examples provided above is listed below:

- PIN: “2493” = 13.2 units (scenario A)
- Numeric password: “2493” = 26.4 units (scenario B)

- Passphrase: “jackspassphrase” = 75.2 units
- Password: “P@sword1” = 46 units (scenario C)
- Password: “P@\$word1” = 44.4 units (scenario D)

Based on the above examples and their associated amount of entropy, it is clear that the strength of the numeric password is much lower than the amount of entropy for the password and passphrase. It can also be seen that the passphrase example is stronger than the password examples. In addition, it was found that password policies that are too stringent, increases the predictability of user character choices which weakens the strength of passwords. In addition, passwords have a negative impact on usability, as this is affected by user challenges related to memorising complex passwords as well as difficulty in typing such passwords. These challenges are discussed in more detail in Chapters 4 and 5.

The above discussion focused on the entropy of the first tier of authentication. The next section discusses the entropy of the second tier of authentication; keystroke dynamics.

3.14.4 Entropy of Keystroke Dynamics

It is difficult to estimate the entropy of the keystroke dynamics algorithm because there are a number of variables to consider and the entropy amount can vary significantly based on how the keystroke dynamics algorithm is set up. For this reason, the entropy provided in this study for the keystroke dynamics algorithm merely indicates how each element of the keystroke dynamics algorithm affects security and usability. The level of entropy for keystroke dynamics is indicated in Table 3-4 below.

In Table 3-4 entropy is calculated on each component of a keystroke dynamics algorithm. In Table 3-4 the keystroke dynamics entropy was assessed by comparing a passphrase and a password. This is an important factor as the more keys the user is required to type on the keyboard (i.e. the more user-keyboard interaction), the higher the entropy will be for the keystroke dynamics algorithm.

Table 3-4: Entropy of Keystroke Dynamics (Shannon, 1948)

Measures	Options	P@sword1		jackspassphrases	
		Option Set	Entropy	Option Set	Entropy
Types	- Non-static - Semi-static - Static	3	1.6	3	1.6
Trackers	- Key Down 1 to Key Up 1 - Key Up 1 to Key Down 2 - Key Down 2 to Key Up 2 - Key Down 1 to Key Down 2 - Key Up 1 to Key Up 2 - Key Down 1 to Key Up 2	6 x 11 = 66	6	6 x 16= 96	6.6
Measures	- Sum - Min - Max - Mean	4	2	4	2
Leniency	- Measurement-specific leniency - Tracker-specific leniency - Optional vs mandatory leniency - User selective leniency	4	2	4	2
Restrictions	- Hand - Fingers	2	1	2	1
TOTAL			12.6		13.2

The entropy is calculated by the number of options available for each component. Since the tracker component has a direct impact on user–keyboard interaction, the entropy for the trackers should be calculated to take this interaction into consideration. The total number of trackers available (a total of six) is multiplied by the number of keys the user needs to select on the keyboard in order to insert the password/passphrase without any errors. In the case of the password and passphrase example in Table 3-4, the password requires 11 keys to be selected and the passphrase requires a total of 16 keys. The entropy is calculated on the result (number of tracker options multiplied by the number of keys pressed) for both the password and passphrase.

The overall entropy in terms of keystroke dynamics is 12.6 and 13.2 for the password and the passphrase respectively, as indicated in Table 3-4. Although there is a minor difference between a password and a passphrase, it can be seen that passphrases complement keystroke dynamics slightly more than passwords. It should be acknowledged that due to the continuous evolution of the keystroke dynamics algorithm (the algorithm can strengthen security based on the volume of keyboard interaction data

that's collected on a user) applied to a system, the entropy has the potential to become stronger (over time) than the results provided in this research.

Passphrases may be more user friendly than passwords (an assessment of this was conducted in Chapters 4 and 5, which addressed the usability issues attached to passwords). The next section summarises the discussion above regarding the proposed two-tier authentication.

3.15 Summary of the Proposed Security Model

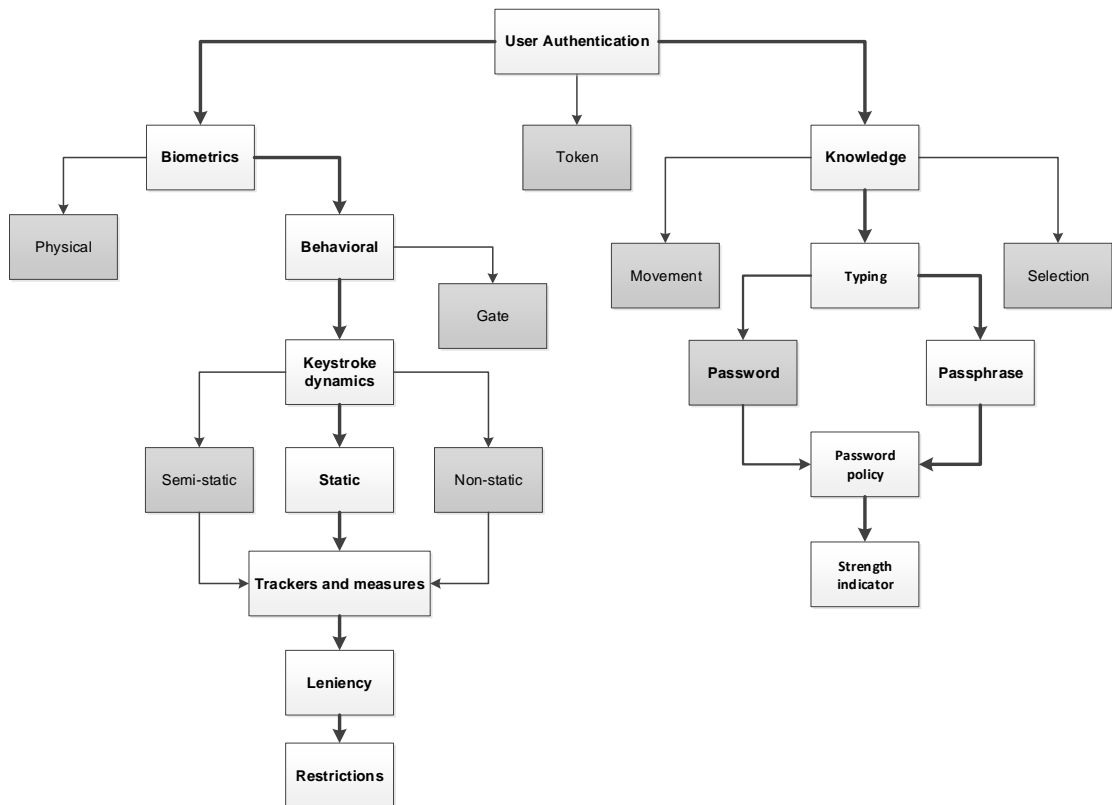


Figure 3-8: Proposed Two-tier User Authentication Security Model

Figure 3-8 illustrates the merging of the diagrams in Figure 3-1 and Figure 3-3. Figure 3-8 graphically depicts all the components that need to be considered when implementing the proposed two-tier authentication solution from a security perspective. Two user authentications are indicated – the path of consideration for passphrases (knowledge) and the path of consideration for keystroke dynamics (biometrics). The last section of this chapter provides concluding remarks.

3.16 Conclusion

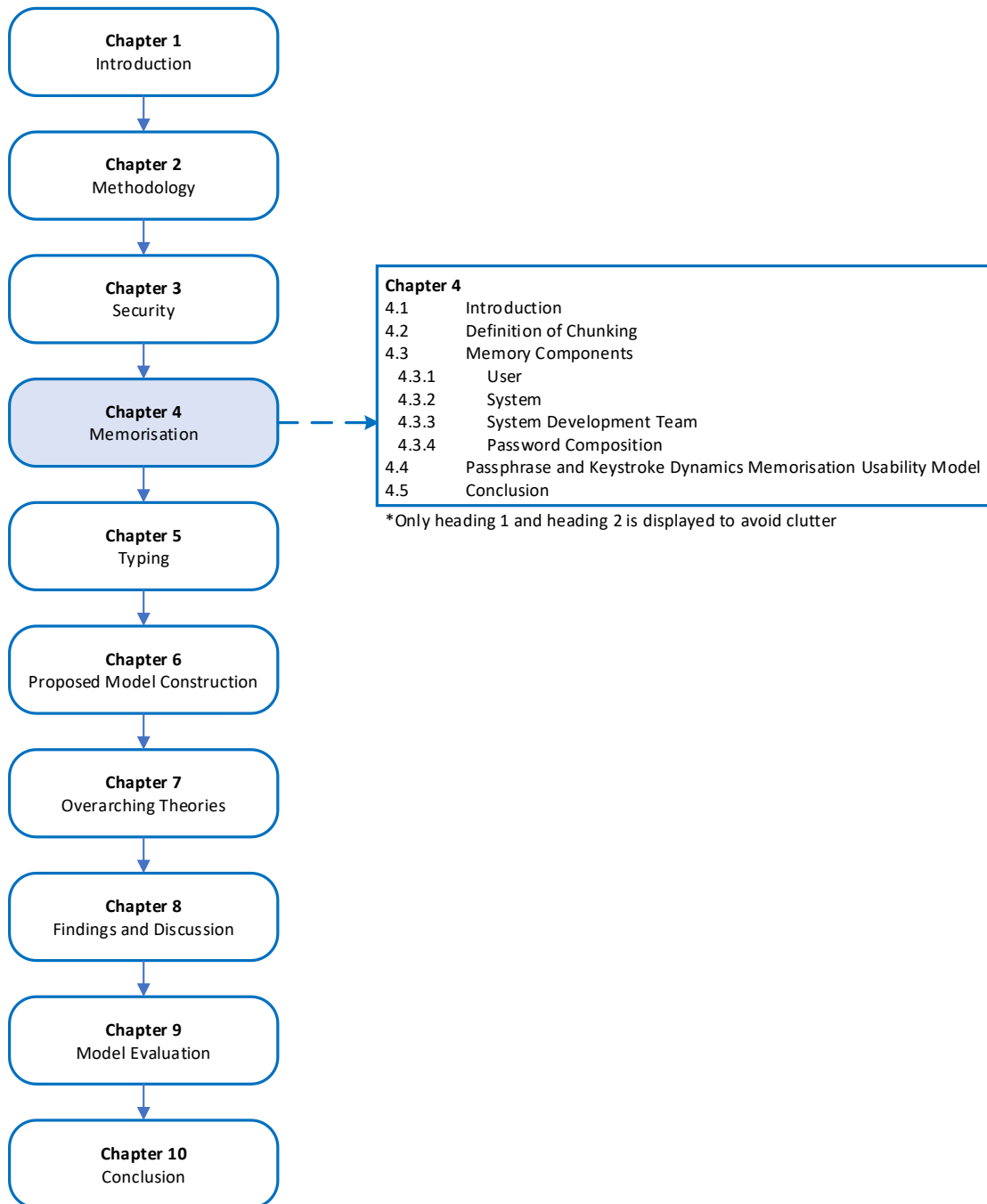
This research study attempts to address the problem of finding a user authentication solution that can achieve increased security and usability in relation to current text-based user authentication solutions. The solution proposed by the study is a two-tier authentication solution, incorporating passphrases (first tier of authentication) and keystroke dynamics (second tier of authentication). In order to validate the effectiveness of this proposed solution, the research should focus on two main aspects – security and usability. This chapter focused on the security aspect of this research and aimed to determine whether the proposed solution has the capability to improve security while maintaining an acceptable level of usability.

In this chapter, based on the Shannon Entropy formula it was found that passphrases are stronger than passwords. Keystroke dynamics further complements passphrases because as user–keyboard interactions increase, more security and accuracy can be provided. Since passphrases were found to monitor keyboard interactions better than passwords, passphrases work better with the keystroke dynamics algorithm than typical passwords. The components of keystroke dynamics that were identified in this chapter can be used, excluded or modified to incrementally increase security and/or increase usability.

In summary, the proposed solution has the capability of addressing the research problem from a security perspective. However, the solution needs to be tailored based on the situation in which the proposed solution needs to be implemented. The different components of passphrases and keystroke dynamics may be adjusted to support the user’s current requirements.

This chapter focused mainly on the security aspect of the proposed solution. The following chapter focuses more on the memorisation of passwords and passphrases as a usability issue with regard to user authentication.

Chapter 4 – MEMORISATION



4.1 Introduction

Although a number of authentication methods have been proposed and utilised, some researchers predict that text-based authentication will remain the primary form of system authentication for the majority of the population for the foreseeable future (Bhivgade et al., 2014; Obeidallah, Ahmad, Farouq, & Awad, 2015). Bhivgade et al. (2014) and Naor, Pinkas, and Ronen (2019) support this prediction by explaining that text-based authentication is easy to implement on a system. Users seem to understand how it works, it can be easily changed and it is a relatively fast method of authentication. A survey conducted around the world found that 58% of users indicated that passwords are their preferred method of authentication (Carstens et al., 2014; Schulze, 2018). That being said, there are some security and usability constraints concerning passwords. This is perhaps the reason why a number of other authentication methods have been proposed.

Ibrahim (2015) argues that graphical passwords are easier to memorise than textual passwords. He further explains that people recall items easier if presented to them visually as opposed to textually (Ibrahim, 2015). However, he does explain that graphical passwords demand more processing power and storage space than text passwords (Ibrahim, 2015). That being said, usability is improved through the use of graphical passwords. However, usability is simultaneously reduced as a result of effects on system performance. Ibrahim (2015) concludes that graphical passwords are appropriate for accessing personal devices offline. However, if graphical passwords are used online usability may be further reduced as it requires computer processing power and an acceptable internet speed. Some users may also consider more data usage as unacceptable. Since graphical passwords use more data, this further affects usability in a negatively manner. Therefore, until data costs reduce and system enhancements increase, textual passwords are a more effective user authentication method than graphical passwords. That being said, current authentication methods, including passwords, have not managed to simultaneously address the usability and security issues associated with user authentication. Albeit, the findings of a study conducted by Carstens et al. (2014) and Naor et al. (2019) indicate that although there are usability issues with all current user authentication methods, text-based authentication is still the authentication method preferred by users and system development teams. This may be because it provides the best balance between security and usability as opposed to other user authentication methods.

Almalki, Chatterjee, and Roy (2019) and Wright, Patrick, and Biddle (2012) explain that the goal of user authentication is to make it simple for the user to authenticate him/herself and also to ensure the security of the system. Forget and Biddle (2008) claim that a user authentication solution has yet to be discovered that satisfies system security and user memorability. Since user memorability forms part of authentication usability, this claim by Forget and Biddle (2008) resonates with the problem identified for this study. It should be remembered that the research problem that this study aims to address is the imbalance between system security and usability in terms of user authentication. Current authentication solutions either increase security and decrease usability, or decrease security and increase usability. There is thus a security-usability trade-off (in terms of memory) with passwords (Woo & Mirkovic, 2018; Wright et al., 2012). It is important to note that a proposed solution is a two-tier user authentication method involving passphrases (as the first tier of authentication) and keystroke dynamics (as the second tier of authentication). It was expected that memory has little to no impact on keystroke dynamics as this is more focused on behavioural interaction with the system. However, memory may be affected if the user needs to remember certain system interaction behaviours. For example, the password or passphrase needs to be inserted using only the user's right hand. Therefore, this chapter has two main goals:

1. Determine whether addressing memory issues with passphrases reduces security.
2. If security is reduced as a result of point 1 above, then is it possible for keystroke dynamics to address this reduction in security.

Zhang and McDowell (2009) found that 53% of users had a password character length of more than six characters in 1999. By 2006, this had increased from 53% to 82% (Zhang & McDowell, 2009) and may have been the reason why memorability issues in text-based authentication arose. This increase in password character length may not have been deliberate on the user's part as many password policies may have forced users to create a password with a character length of more than six characters. In addition, password policies also require different character sets to be used (Maoneke & Flowerday, 2019; Zhang & McDowell, 2009). This, in addition to the increase in password character length, may have further affected memorability.

Blanchard, Malaingre, and Selker (2018) and Carstens et al. (2014) states that passwords are pointless if they cannot be remembered by anyone. Users struggle to remember complex passwords which results in the creation of passwords that are easy

to remember but weak in terms of security (Renaud, 2019; Wiedenbeck, Waters, Birget, Brodskiy, & Memon, 2005), and making the passwords easy for attackers to crack (Proctor, Lien, Vu, Schultz, & Salvendy, 2002). When restrictions are imposed that ensures that users create a strong password, unwanted user behaviours are created such as writing the password down (Parsons et al., 2010; Woo & Mirkovic, 2018) and using the same password for multiple systems (Carstens et al., 2014). This is because users struggle to recall such strong passwords from memory. Although forcing users to create unrealistically strong passwords may create the perception of an increase in security, it ends up creating other security issues such as writing the password down on a piece of paper. In addition, usability is reduced as users struggle to remember these strong passwords. The next section includes the theoretical discussion for this chapter.

4.2 Definition of Chunking

Miller (1956) explains that people memorise information in what he termed “chunks”. Chunks are an abstract definition of storage. A chunk cannot simply be defined as a character or word or even a sentence; the amount of information that can be held in a chunk of data is based on a person’s personal experiences (Harte & Law, 2019; Keith et al., 2009; Schuessler, 2017). Accordingly, a chunk is a meaningful set of related items. Therefore, the data in a chunk can be big or small and can grow over time. This is explained in more detail below. Carstens et al. (2014) and Strock, Rougier, and Hinaut (2019) explain that the data in short-term memory converts to long-term memory if it is retained for a sufficient period in short-term memory. They do not specify how long this process takes on average. However, this study did not consider long-term memory as this would have required the research to be conducted for a longer period of time. This study focused on short-term memory and considered this sufficient as most users would change their password/passphrase if their short-term memory fails them in recalling the password/passphrase. If a user fails to memorise a password, usability is immediately affected. The process of moving content from the short-term memory to the long-term memory differs from person to person. The speed at which content is transposed from short-term memory to long-term memory is also based on the person’s ability to create relationships between the content and the amount of rehearsal the person is exposed to regarding that content (Guo, Wan, Wan, Zhu, & Shi, 2013; Strock et al., 2019). Research by De Munari, Cozzutti, and Romero-Naranjo (2016) found that there was no difference between short-term and long-term memory. However, the Chunking theory proves that this cannot be true as the number of chunks it takes a person to remember an item may

decrease as he/she becomes more and more exposed to the item to be remembered (Cowan, 2010; Doumont, 2002; Schuessler, 2017).

The Chunking theory, which was developed by Miller (1956), estimates the volume of content an average person can recall from short-term memory. He explains that individuals store information in a limited number of chunks. If information cannot fit into these chunks, then it is forgotten. He further explains that the number of chunks it takes a person to remember an item can reduce as he/she becomes more exposed to the item to be remembered.

Research has been conducted to assess whether the Chunking theory remains valid (Carstens et al., 2014; Cowan, 2010; Doumont, 2002; Keith et al., 2009; Schuessler, 2017). The original theory explains that an average person can memorise five to nine chunks of information (Miller, 1956). Recent studies have discovered that an average person can remember only three to five chunks (Carstens et al., 2014; Cowan, 2010; Doumont, 2002; España, 2016; Keith et al., 2009). Two major explanations testify to this:

1. Individuals are currently capable of memorising less information than in the past.
2. The size of a chunk has increased over the years and therefore the number of chunks has reduced.

In the above numbered list, point 2, which explains the difference between chunk numbers and chunk size, is best explained graphically, as in Figure 4-1.

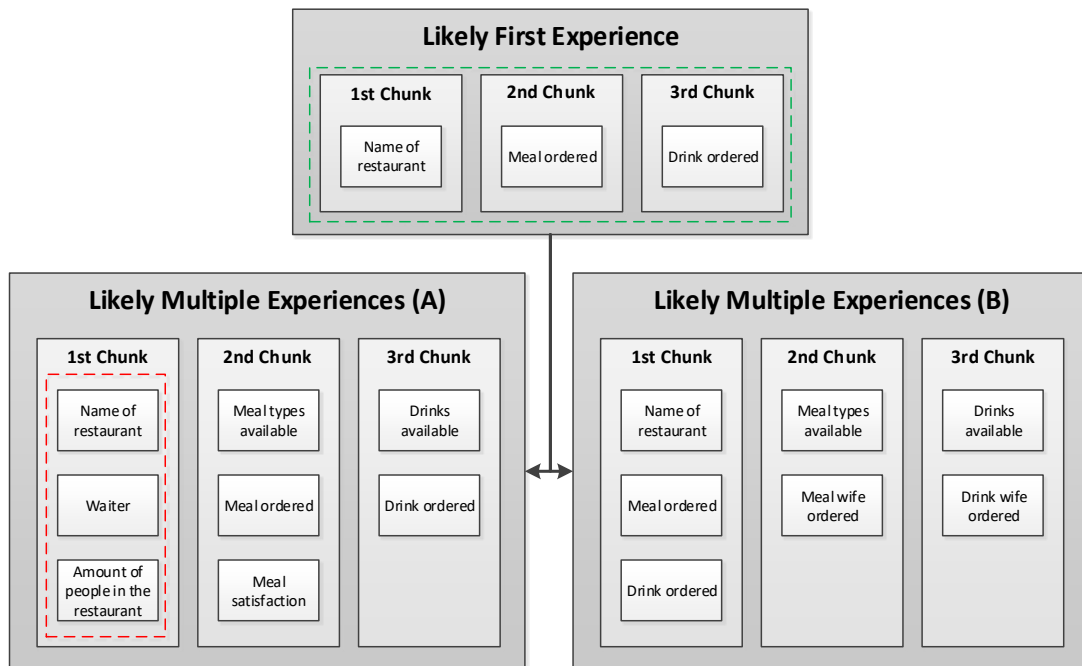


Figure 4-1: Chunk Size, Numbers and Evolution

For example, when a person goes to a Spur restaurant for the first time, he/she requires three chunks of information to remember what they ordered. This is indicated in the “first likely experience” panel in Figure 4-1. The first chunk is used to remember the name of the restaurant, the second chunk what meal was ordered and the third chunk what drink was ordered. The type of drink that was ordered would possibly be included in the third chunk or might form a fourth chunk.

Over time, as the individual has more exposure to eating at a Spur restaurant, the chunks required to memorise the first experience would reduce. To keep this explanation simple, it is assumed that the individual orders the same meal every time he/she goes to Spur. As indicated in the “likely multiple experiences (B)” panel, “1st chunk” group in Figure 4-1. The “likely multiple experiences (B)” panel in Figure 4-1 depicts another method a user may take to memorise information. Based on Figure 4-1, a chunk of information is referred to as a grouping of interrelated elements created by each person’s associations gained from personal experience. New exposures often require more chunks than routine experiences. Lastly, different people memorise information in different chunk formats.

Another example is the “Mugg&Bean” restaurant. “Mugg&Bean” may be seen as one chunk of information even though there are three words and nine characters. This is easy to remember as the individual may associate with the brand if he/she is aware of

and repeatedly exposed to the brand's existence. However, someone who is unfamiliar with the brand, such as a tourist, would use seven chunks to remember "Mugg&Bean": the first chunk being the word "Mug", the second chunk involving remembering that "Mugg" is spelt with two "g's", the third chunk being to remember "&", the fourth chunk being to recall that "&" is a symbol and not the full word "and", the fifth chunk being to remember "bean", the sixth chunk being to remember that there are no spaces between the words "Mugg&Bean" and the seventh chunk being to recall that the characters "M" and "B" are capitalised.

Huh, Kim, Bobba, Bashir, and Beznosov (2015) provide an example of the Chunking theory that many can relate to. In almost all marketing advertisements the contact number is separated into chunks to assist viewers to remember the contact number with less effort. For example; 011 555 4242. In this example; the advertiser has allowed the viewer the possibility of only using three chunks of data to memorise the contact number. If the contact number were presented to the viewers with no spaces (i.e. 0115554242), some viewers might separate it with spaces and some viewers might try to memorise the contact number by using each individual digit, i.e. ten chunks would be required to memorise the contact number.

Jablon (1996) found that an average person struggles to write down a ten-digit cellphone number. This means that the average person struggles to memorise ten characters of a one-character set. Despite strong passwords demanding more memory capacity than a ten-digit cellphone number, users are expected to memorise them to ensure the security of the system, although it could be argued that cellphone numbers are randomly generated and not person generated. Nonetheless, this illustrates that strong passwords are already difficult to memorise as they require memory capacity that is beyond the limits of an average person's memory. In addition, passwords may need to become stronger in the future due to advances in hacking tools. Simply forcing a user to memorise a more complex password to address this will continue to be a more ineffective user authentication solution. Alternative user authentication solutions therefore need to be explored that consider human memory capacity to ensure that usability is not compromised when security is increased. Another example to explain the Chunking theory is provided below.

If the Chunking theory is applied to passwords, it is expected that users will take more effort to recall passwords that comply with a stringent password policy as opposed to a more lenient password policy. For example, a stringent password policy requires the

creation of a password that contains one uppercase letter, one lowercase letter, two numbers and one special character. Accordingly, the user creates the password, "P@ssword2017". The following number of chunks is required to remember the password (each number in the list below indicates one chunk):

1. Password
2. 2017
3. <<the word "password">> <<the year "2017">>
4. "P" is capitalised
5. All other alphabetic characters are lower lowercase
6. The "a" was replaced with an "@" symbol

The above list indicates that a user might require a maximum of six chunks to remember this password. In addition to the six chunks required, a user often logs in to multiple systems every day (Florencio & Herley, 2007; Schuessler, 2017). For this reason, at least one more chunk is required for the user to remember that the password created is for a specific system, assuming that the user has a different password for each system. The next section assesses the memory impact when using a passphrase.

The passphrase example used to assess the number of chunks required to memorise a passphrase is developed based on the characteristics of passphrases (i.e. 16 characters or more, more than one word as well as only lowercase letters). This method is applied to ensure that a valid example is used for this study. The passphrase "spursthebestintheworld" is used in the example to assess the chunking impact of a common passphrase. The following chunks are used to remember the passphrase.

1. "spursthebest"
2. "ishebest"
3. "intheworld"

It should be noted that users have different learning styles (Gilbert & Swanier, 2008; Ma, Gong, Gao, & Xiang, 2017), therefore the Chunking theory should only be used as an estimated indication of an average person's memory capacity. In reality, a user may recall the password/passphrase from memory with more or less ease. As indicated in the list above, three chunks would be used to memorise the passphrase example. Table 4-1 below indicates the different chunks that could be used to memorise the passphrase, "spursthebestintheworld" and the password "P@ssword2017".

Table 4-1: Number of Chunks

	User A	User B	User C	User D
Chunk count	<ol style="list-style-type: none"> 1. "spur" 2. "steak" 3. "is" 4. "the" 5. "best" 6. "in" 7. "the" 8. "world" 	<ol style="list-style-type: none"> 1. "spurst steak" 2. "isthebest" 3. "intheworld" 	<ol style="list-style-type: none"> 1. "spurst steak isthebest intheworld" 	<ol style="list-style-type: none"> 1. P@ssword2017 2. password 3. 2017 4. The "a" in password is an "@" sign 5. The "p" in password is capitalised
Total number of chunks	8	3	1	5

Table 4-1 above indicates that User A would use eight chunks to remember the passphrase, "spurst steak isthebest intheworld". User B, on the other hand, would require three chunks and User C would require one chunk to recall the passphrase. In the assessment of the password, "P@ssword2017" above, six chunks were used to memorise the password. When comparing the quantity of chunks required by the passphrase example used in Table 4-1 (User A, User B and User C) and the password example ("P@ssword2017") above (User D, for the purpose of this explanation), it can be seen that User D may have less difficulty memorising the password than User A may have with the passphrase. However, User B and User C will use less effort memorising the passphrase than User D. Although User A does require more chunks to remember the passphrase, it is unlikely that a user would create a passphrase that requires memorisation to such an extent. It should also be noted that the more the user uses the passphrase, the more likely they will be to move from a User B state to a User C state. For example; the user will move from requiring three chunks to remember the passphrase to only requiring one chunk of memory.

The restaurant and meal example above assisted in understanding the journey a person takes when memorising specific aspects and how the effort of memorising such aspects is reduced through frequent exposure to similar or related experiences. This is supported by Carstens et al. (2014) and Mogire, Ogawa, Minas, Auernheimer, and Crosby (2018) who found that mixing digits and letters in a chunk made it more difficult to memorise

and recall as opposed to having only numbers or only letters in a chunk. They also found that memorisation and recall are easier if a person can make sense of the data. For example; meaningful passwords/passphrase are easier to remember than a set of random characters. This should apply to passphrases as well. In addition, Wright et al. (2012) discovered that users found it easier to recall whole words in their passwords. These findings indicate that the Chunking theory compliments passphrases.

The next section introduces the components of user authentication that may have a negative effect on memory required for user authentication.

4.3 Memory Components

The main section of this chapter is on the components required to support password or passphrase memorisation. Memory components are divided into four main categories:

- **User** – How a user can improve the memorisation of passwords or passphrases.
- **System** – How a system can assist a user to memorise passwords or passphrases.
- **System development team** – Aspects of user authentication that should be considered by the team that is designing, developing and maintaining the system.
- **Password composition** – Approaches to constructing passwords or passphrases that assist with memorisation.

These four memory components were derived from subcomponents that influence the memorisation of passwords and passphrases in user authentication. These memory components are discussed in more detail below, starting with the user component.

4.3.1 User

This section aims to determine what a user can do to assist password or passphrase memorisation. The first section provides a discussion on adding personal data to passwords and passphrases. This is followed by a discussion of user awareness and then user errors. The last section explains how users currently apply the Chunking theory.

4.3.1.1 Personal Data

The first section on personal data explains the association between personal data and the Chunking theory. The next section discusses the memory impact of associative data over new data. This is followed by a section on how users incorporate personal data into passwords. The last section discusses the different types of meaningful data.

4.3.1.1.1 Why Personal Data

A study conducted in 2007 found that 76% of participants included personal data in their passwords (Tamil, Othman, Abidin, Idris, & Zakaria, 2007; Vittori, 2019). A similar study conducted by Gafni, Pavel, Margolin, and Weiss (2017) found that 65% of participants believe that including personal data in passwords makes them more vulnerable to cyber-attacks however, 71% of participants still included personal data in their passwords. Zhang and McDowell (2009) performed a study that attempted to find out what personal data is often added to passwords and found that only 10% of the participants created passwords that included random strings. Personal data is added to passwords so that the password can be easily recalled from memory (Bhivgade et al., 2014; Chakraborty & Nguyen, 2018; Keith et al., 2009; Vittori, 2019; Zhang & McDowell, 2009). Carstens et al. (2014) and Vittori (2019) found that adding personal data to passwords reduces the amount of memory error. Therefore, from a usability aspect, adding personal data to passwords can assist in increasing usability by addressing some memorisation issues experienced by users. However, Vittori (2019) and Wright et al. (2012) explain that adding personal information to passwords makes a person more vulnerable to security attacks such as social engineering and phishing.

Kaiser and Reichenbach (2002) performed an experiment to assess the amount of information an average person can recall from short-term memory. Accordingly, participants were presented with an image which displayed a pattern of dots and were then required to replicate the dot sequence. Results showed that participants could successfully memorise a maximum of six dot patterns before they started guessing. Another notable discovery by Kaiser and Reichenbach (2002) was that adding attributes to the pattern significantly increases the participant's ability to memorise the pattern. For example, creating a pattern on a person's face was easier for the participant to remember as they could assign known location to the sequence, for example eyes, nose, ear and then mouth. This may be because users have previous experience with these graphical images which makes it easier to comprehend the pattern.

The Chunking theory also aligns to the way in which passwords and passphrases are created by users. This has been discussed above in relation to the fact that users create passwords based on descriptive aspects that they can relate to such as their first car, or their date of birth. The Chunking theory explains that as the information becomes more and more familiar to the person, more information can be included in a memory chunk. If the information is already familiar to the person such as personal information, less

memory capacity is required. The next section explains how users include personal data in passwords.

4.3.1.1.2 Use of Personal Data in Passwords

Adams and Sasse (1999); Woo and Mirkovic, (2018); and Zviran and Haga (1993) attempted to understand how a user creates a password. They found that users have various methods for creating passwords. This finding is beneficial to security as previously discussed; users tend to create a password based on personal experience to assist in memorising the password. In other words, users create meaningful passwords. The same approach is expected to be taken when users create passphrases. If users were to use similar methods to create passwords, phishing attacks would be more successful than they are today. However, as personal experiences are used in creating passwords, various methods/formats are used, which increases security. An example of users using different methods to create a password is as follows:

- User A creates the following password based on personal experience: <first car name><year born>.
- User B creates the following password based on personal experience: <day born><pets name>.
- User C creates the following password based on personal experience: <year born><first car name>.
- User D creates the following password based on personal experience: <first car name><year born>.

User A and User B have completely different methods for creating a password, while User C decides to create a password by starting with a number and then text. Although User A and User B use exactly the same method for creating a password, the passwords may differ as they are based on their personal experience. For example, User A's password may be ford1990 while User D's password may be toyota1984.

Since users have different methods for creating passwords, security is increased and accounts are less susceptible to phishing attacks. Therefore, even if an attacker has access to a user's personal data, they need to determine how the personal data is incorporated into the password. The same applies to passphrases. In addition, an attacker needs to know how the password or passphrase is inserted by the user because of keystroke dynamics. The last section on personal data explains the different types of meaningful data.

4.3.1.1.3 Types of Meaningful Data

Meaningfulness in a password or passphrase is what makes it easy to remember (Blanchard, 2019; Carstens et al., 2014; Parsons et al., 2010; Zhang & McDowell, 2009). Meaningfulness is defined by the number of associations a person can link to the word (Carstens et al., 2014). Therefore, meaningful words are different for different users as users have different experiences throughout their life. These experiences define the number of associations for a specific word. As discussed above, adding personal data to a password or passphrase assists memorability and also aligns to the Chunking theory. However, data can still be meaningful to a person while not being directly personal. Table 4-2 below provides some examples of different forms of meaningful data.

Table 4-2: Examples of Meaningful Datasets – Adapted from: (Blanchard, 2019; Carstens et al., 2014)

Primary Personal Data	Secondary Personal Data	Indirect Personal Data
First car	Wife's car	Dictionary words
ID number	Father's ID	Countries/Cities
Date of birth	Child's date of birth	Public holidays
Favourite band	Best friend's favourite band	Musicians

Table 4-2 above shows that meaningful data can be organised into one of three categories. The “primary personal data” column can be defined as data related to personal experiences. The “secondary personal data”, on the other hand, would be the “primary personal data” of someone else but you are linked to this person through your personal experiences. “Indirect personal data” can be defined as data, information or knowledge gained by a person through personal experiences but who is not directly related to them.

Based on the above discussion, it is evident that although adding personal data to passwords or passphrases makes it easier to recall them from memory, it does pose a security risk in the form of social engineering and phishing attacks. The research which led to this discovery (Parsons et al., 2010; Zhang & McDowell, 2009), did not mention what kind of personal data they were referring to; however, examples were provided which can be organised into the “primary personal data” category in Table 4-2. From

this it can be seen that the three categories of meaningful data proposed in Table 4-2 may have different levels of security. Each grouping in Table 4-2 also has a different volume of data per person, which may correlate to the different levels of security. The greater the volume of data, the lower the probability of an attacker finding the personal data selected by the person. Therefore, from a security perspective, “indirect personal data” would be more secure than “secondary personal data”, and “secondary personal data” would be more secure than “primary personal data”. However, from a usability point of view (in terms of memorability), “primary personal data” is easier to memorise than “secondary personal data” and “secondary personal data” is easier to recall from memory than “indirect personal data”.

Although Parsons et al. (2010) and Yildirim and Mackie (2019) argue that what makes a password strong is the meaninglessness present in the password, meaningful passwords may be allowed as they assist users to remember the password and therefore increasing usability. Based on the above discussion, meaningful data should be allowed to be included in passwords and passphrases. However, when considering the categories of meaningful data in Table 4-2, the “primary personal data” category should be avoided as this category is most likely to result in a social engineering or phishing attack. Blanchard (2019) and Carstens et al. (2014) suggest that passwords may be easier to remember if a part of them includes something meaningful. Based on this, the only time “primary personal data” should be used in a password or passphrase is when it includes both “primary personal data” and meaningless data.

These suggestions may slightly reduce security; however, they do maintain an acceptable level of usability as they support password and passphrase memorability. The keystroke dynamics algorithm compensates for this slight reduction in security as an attacker also needs to know how the password or passphrase is inserted by the user, which cannot be identified from a user’s personal/meaningful data. The next section discusses the impact of user awareness on usability and security.

4.3.1.2 User Awareness

In this section, the ineffectiveness of current solutions is firstly discussed, followed by user awareness provided by the system. The last section explains the impact of a user being aware of the potential risks associated with having a weak password.

4.3.1.2.1 Ineffective Solutions

Adams and Sasse (1999) and Woo and Mirkovic (2018) found that users are not well informed on how to create a strong password, which has resulted in security issues. Users need to understand how passwords and passphrases should be created, rather than merely preventing users from creating weak passwords or passphrases. System development teams have attempted to address this problem by adding restrictions that force the user to create a strong password. However, this increased the likelihood of the password being written down which compromises security. Renaud (2019) and Yan, Blackwell, Anderson, and Grant (2004) discovered that users are more tempted to write down a password that is complex to avoid the risk of forgetting it. The realisation of such a risk also has a negative impact on usability. Restrictions can be used but the more stringent the restrictions the greater the risk of the user compromising security by writing down the password/passphrase (Adams and Sasse, 1999; Parsons et al., 2010; Yan et al., 2004). Therefore, it is important to inform the user on how to create a strong password and passphrase. The next section discusses methods for providing immediate user awareness.

4.3.1.2.2 Synchronous User Awareness

Adams and Sasse (1999) and Woo and Mirkovic (2018) explain that in addition to educating users on the importance and risks of user authentication, they should also be taught how to ensure security. While many systems inform users as to whether a password/passphrase is weak or strong using a strength indicator, they are absent on other systems. With systems that do provide an indicator, most merely indicate the strength of the password without giving an explanation of the result. Users then need to discover for themselves through trial and error how the strength indicator derives the result. This has negative implications for usability.

Carstens et al. (2014), Furnell (2007) and Renaud (2019) state that systems have inconsistent password strength indicators. For example, one system may indicate that the password is strong while another system may indicate that the same password is weak. In addition, most systems do not provide the dynamics of how the strength of a password is calculated, i.e. they do not state why the password created by the user is deemed weak. Reasons may include the password only having one special character; or each special character = 2 security points, each numeric character = 1 security point. The system then informs the user that a password must have a minimum of six security points to be accepted by the system. As these rules are programmed in the back-end of

the system, system development teams should consider presenting these rules (in a user-friendly manner) to the user instead of leaving the user to figure them out for themselves. The assessment is usually based on theoretical findings, included in the information security policy or it may merely be created at the discretion of the system development team.

System development teams may argue that the strength indicator is based on the content held or services provided by the system. For example, a theoretically accepted moderate password may be viewed as a strong password on a system where minor damage can be incurred by a user, while the same password may be labelled weak on a system that holds sensitive data such as online banking. That being said, these inconsistencies in strength indicators have resulted in users having to create different passwords for different systems. Although this may not be viewed as a negative result from a security perspective, usability in terms of memory is severely compromised, even more so when the user needs to modify a password in order to make a password stronger based on what the system defines as strong.

Passphrases may be easier for the user to comprehend, as password strength is based mainly on character length (Burr, et al., 2017). This is in contrast to passwords where password strength is defined in terms of the character set counts included in a password (Burr, et al., 2017). Therefore, usability may be higher for passphrase strength indication than password strength indication. Additionally, most systems do not accept or regard passphrases as an acceptably secure text-based authentication method. Nevertheless, as was shown in the previous chapter, passphrases can be even more secure than the conventionally secure password. It is suggested that strength indicators be based on theoretical findings, it should inform the user why their password is weak and allow users to create passphrases. The last section on user awareness focuses on the importance of creating user awareness of the risks of creating a weak password.

4.3.1.2.3 User Awareness of Potential Damage

Parsons et al. (2010), Woo and Mirkovic (2018) and Zhang and McDowell (2009) explain that users do not understand the repercussions associated with having their passwords cracked. They predict that if users were more informed of the potential damage attached to creating weak passwords both their behaviour and their attitudes might change, which would reduce frustration and likewise increase usability. Although creating awareness by communicating the potential risk or damage might change user's attitudes it could also create unnecessary anxiety which might lead to the user avoiding the system

altogether. To counter this imbalance, it is advised that user awareness be created around keystroke dynamics so they know there are additional security protocols in place, even though it could be argued that users might create weak passwords. Nevertheless, they would most likely maintain some of the security control, i.e. creating a strong password. This is justified below.

The above discussion aligns to the Rational Choice theory (Adanali, 2017) and the Protection Motivation theory (Kothe, et al., 2019; Zhang & McDowell, 2009). The Rational Choice theory explains that a person will only perform a task if some kind of benefit is perceived (Adanali, 2017). If the Rational Choice theory is applied to user authentication, the task would be creating a strong password or passphrase. The perceived benefit in this regard would be the reduction of the risk of damages if the password/passphrase were hacked. The Protection Motivation theory is similar to the Rational Choice theory and states that users have a fear or perception of vulnerability and weigh this against the response cost incurred if the risk turns into an issue (Kothe, et al., 2019; Zhang & McDowell, 2009). Based on the Protection Motivation theory, it is suggested that users be informed of the potential damage relating to the creation of weak passwords and also what the system is doing to reduce this risk. The next section provides a discussion on user error.

4.3.1.3 User Error

This section has been separated into three subsections, namely, user limitations, user productivity and password creation mental models. User limitations are discussed first.

4.3.1.3.1 User Limitations

A study conducted by Lewis (2003) found that 65% of security incidents in organisations were related to human error while only 3% were related to hacking attacks. Renaud (2019) and Zhang and McDowell (2009) claim that human error is a high risk factor for security. Sasse (2003) and Woo and Mirkovic (2018) provides a somewhat different view by explaining that human limitations create security threats. While a large volume of research states that educating users can increase security, few researchers have mentioned that humans have limitations and thus current security protocols need to take human limitations into consideration. For example, if an average person can memorise a maximum of eight characters, password/passphrase policies should adhere to this limitation and avoid requiring users to create passwords/passphrases that exceeds this limitation. Passphrases are able to support such human limitations while the keystroke

dynamics algorithm can compensate for the security loss arising from these human limitations. The next section discusses user motivation.

4.3.1.3.2 User Productivity

Forget and Biddle (2008) and Renaud (2019) claim that one of the reasons why users create weak passwords can be attributed to a lack of motivation on the part of the user. Hussain et al. (2018) and Zhang and McDowell (2009) support this statement by mentioning that users view passwords as an annoying barrier. They explain that users view passwords as an overhead cost as they do not increase productivity and merely delay the intended use of the system. While users may understand that a password is a preventative measure to reduce risk, the reason for such a negative opinion of passwords may be because password security protocols are currently too stringent. Braz and Robert (2006) explain that at times a user needs to login to a system that they are only going to use for a short period of time. For example, a user logs into his/her email to check whether they have received a specific email. In this case, the intended action time to complete the desired task is shorter than the login time required to login to the system. The proposed solution of combining passphrases with keystroke dynamics allows the user to enjoy the usability benefits of a one-tier text-based authentication method while compensating for the lack of security attached to many one-tier authentication methods. The next section provides another approach to ameliorate user errors.

4.3.1.3.3 Password Creation Mental Models

Carstens et al. (2014) suggest that mental models of password creation be used by users when creating passwords as it is an effective method for managing a portfolio of passwords. These models comprise rules that can be followed by the user. An example of a user's mental model for password creation is indicated below:

- The first "a" in a password should be converted to "@".
- Any numbers added to a password should be included at the end.
- Any second "s" in a password should be replaced with a "\$"
- The first letter of every password should be capitalised.
- Password cannot be longer than 14 characters.
- Password should always include the word "the".
- Passwords for financial systems should include the word "money".

This means that the user only has to memorise the rules of the password creation mental model and the word for each system. Then the user simply converts the word based on

the rules of the model. Although it is still not recommended, the user can write down the instructions for each system on a piece of paper as a third party will still not be able to gain access to the system even if they get hold of the piece of paper. This is because the word still needs to be converted based on the password creation mental model rules.

Password creation mental models may not be as important for the solution proposed by this work, as passphrases only demand the use of a one-character set. However, they may assist the user in managing a portfolio of passwords and passphrases. The last three bullet points in the list above can be used for passphrases. The last section on user errors explains users' application of the Chunking theory.

4.3.1.3.4 Application of Chunking

Wright et al. (2012) found that the participants in their study did not apply the concept of chunking correctly, attempting to remember the first letter of the word instead of chunking words together. Accordingly, if chunking is to be used effectively a change in mindset is required. Passphrases are more likely to indirectly encourage this mind-set change as opposed to passwords as passwords are more difficult to separate into chunks.

4.3.1.4 Muscle Memory

Muscle memory is achieved when information is embedded in a person to such an extent that minimal cognitive effort is required in order to access information to execute a task (Skovgaard, Almquist, & Bangsbo, 2018). In other words, information has moved from short-term memory to long-term memory and then surpassed long-term memory and moved to muscle memory. This is best explained with an example. When a person first learns how to drive a car, the driving instructor provides instructions on the procedures required to drive. For example, before changing gears, push in the clutch. At this point the learner driver holds this information in short-term memory. After a few lessons (practice through repetition) the instructions for driving moves into long-term memory. At this point the learner driver is aware that when a gear change is required, he/she needs to push in the clutch. After a few months of consistent driving by the learner driver, he/she gets to a point where they can subconsciously change gears, by pushing in the clutch. This state from a memory perspective is known as muscle memory. This process is supported by research on information processing theory (Arthur & McMahan, 2018; Heuer & Sanders, 1989). Similar to the learner driver, user's experience the same process with passwords. Passwords used consistently for a certain period of time will eventually be accessible through a user's muscle memory.

According to the heuristic-systematic model of information processing (Eslami & Ghasemaghahi, 2018; Trumbo, 2002), this process of movement between memory states (short-term, long-term and muscle memory) is accomplished in two ways:

- Heuristic processing – This requires the least amount of cognitive effort by the individual. The person would accept the information consciously or sub-consciously and accept it for what it is.
- Systematic processing – The reliability of the information received by a person requires confirmation. Individuals seek confirmation through logical self-assessment or from additional sources. Systematic processing results in a faster move of information from short-term memory to long-term memory and then to muscle memory.

Based on user authentication security protocols, conventional passwords are recommended to be changed after a certain period of time due to security risks (Victor, Dogonyaro, Victor, Meshach, & Ayobami, 2018). This has resulted in users (who comply to this security requirement) not having enough time to get the password to muscle memory. If a password moves to a user's muscle memory, usability improves from a password memorisation point of view. The proposed solution does not require passphrases to be changed due to compensation from the keystroke dynamics algorithm. Most users use systematic processing to create and memorise passphrases. Therefore, the passphrase may have a faster journey to muscle memory than passwords. Additionally, the passphrase in muscle memory can be used for longer than passwords as passwords need to be changed frequently. Long term memory does not form part of the scope of this study due to research time constraints. However, from the research conducted it is possible that the proposed solution complements long term memory and muscle memory better than conventional passwords. Further research is required to confirm these findings.

The next section discusses the impact a system has on memory in terms of user authentication.

4.3.2 System

This section focuses on understanding how a system can assist users to memorise passwords and passphrases. The first section discusses the impact of having multiple passwords for different systems on the user's memory. The next section focuses on the usability and security impact of changing passwords frequently, followed by a section on persuasive passwords. The last section discusses the difference between user-

generated passwords and system-generated passwords in terms of usability and security.

4.3.2.1 Multiple Accounts, Multiple Passwords

This section is split into two: an exploration of the usability issues associated with having a portfolio of passwords and a discussion on potential solutions to this issue.

4.3.2.1.1 Usability Issues Attached to a Portfolio of Passwords

An average user has 25 accounts and types eight passwords daily (Florencio & Herley, 2007). Florencio and Herley (2007) found that an average user has 6.5 passwords and each password is used on an average of 3.9 different sites. To further impact usability in a negative manner, researchers advise that users need to use different passwords for different sites (Bhivgade et al., 2014; Blanchard, 2019; Carstens et al., 2014). The primary reason for this rule is to minimise the risk of damage (Renaud, 2019; Zhang & McDowell, 2009). Accordingly, if an attacker were to discover a password, they would only be able to access one system using the stolen password (limited amount of damage) as opposed to multiple systems if the user had used the same password for multiple systems (increased amount of damage). Currently, systems cannot enforce this restriction, as systems (especially public systems) cannot verify whether the user is using the same password on another system. Therefore, this is the responsibility of the users rather than the system.

If users comply with the security recommendation to have a different password for every system they access, then the total number of passwords a user has to remember will add strain on a user's memory (Blanchard, 2019; Carstens et al., 2014). Over time, technology will provide more systems that attract users where login details are required. Therefore, this statistic is expected to increase and exacerbate the memory issue attached to user authentication. Even if users attempt to comply with this rule, most create passwords that, while different, are similar to each other (Keith et al., 2009; Renaud, 2019). This slightly increases security, as attackers are aware of this approach and if one password is compromised, focus will be prioritised on guessing similar passwords. This would eventually apply to passphrases as well if users start adopting them.

As discussed above, users often use the same password or similar passwords for all the systems they login to. However, recommending that users use different passwords for different systems may increase security but reduce usability, as users now need to

remember multiple passwords and know which password is used for which system. Some users have resorted to a password management tool to keep track of their passwords. A password management tool is simply a system that stores all a user's password and is accessible with a primary password. This too; can also be used to hold users' passphrases. Bhivgade et al. (2014) and Pearman, Zhang, Bauer, Christin, and Cranor (2019) warn of the danger of using a password management tool, explaining that if a hacker were to discover the password for the password management tool, they would also be able to gain access to all the other passwords the user has stored in their password management tool. It should be acknowledged that a two tier authentication solution can be used for a number of password management tool to increase security. However, this will also negatively impact usability as the user will need to authenticate themselves twice before gaining access to their passwords and passphrases. Nonetheless, this defeats the object of having different passwords for different systems. The intention of different passwords for different system is to spread the risk. If a hacker gains access to one user's password, they can only access one system instead of having access to multiple systems. Usability is also negatively affected as the user now needs to login to a secondary system, first to retrieve the password and then login to the intended system. Using multiple password management tools to disperse the risk of having all passwords compromised may slightly increase security (when compared to one password management tool); however, it also negatively affects usability. Users then need to remember which management tool was used for the system they are trying to access. Having this additional information to remember negatively affects usability. Accordingly, password management tools are an ineffective solution to address the password/passphrase portfolio issue faced by users.

In terms of the proposed solution, it is recommended that users have different passphrases for different systems. However, users can afford to apply a bit more leniency and use passphrases that are similar. This option is provided to assist memorability as an attacker still needs to determine how the passphrase is inserted into the system if the passphrase is compromised. The next section discusses potential solutions (not specific to the proposed solution) to address the usability and security issue of having different passwords/passphrases for different systems.

4.3.2.1.2 Potential Solutions

A number of solutions have been recommended to address the problem discussed in the section above. This section discusses these solutions to determine if any solutions are viable from a security and usability perspective.

Carstens et al. (2014) used the Chunking theory to support their suggestion of having different passwords for different systems. They proposed that users should maintain a common part of the password across all passwords. This reduces the amount of memory required to remember a variety of different passwords as one or two chunks are always included in majority of the passwords. This method also complies with the recommended rule of having different passwords for different systems. The same method can be applied for passphrases, using a specific word/s in all passphrases. This assists in mentally filtering the list of all phrases the user may have by merely looking at the phrases that include the specific word/s. For example, a user may make their own rule that the words “isthe” will be used as a common chunk of information in all of their passphrases. The user knows to only try passphrases which include “isthe” if they are unsure what passphrase was used for the specific system. The slight reduction in security because of a common part being in all passphrases can be covered by the keystroke dynamics algorithm.

Wright et al. (2012) assessed whether a master acronym would be effective as a trigger for a specific password for a portfolio of passwords. They suggested that a word be used in a password for different systems and an acronym be developed by taking the first letter of each password. This may trigger the user’s memory by providing them with a hint on the word chosen for the password. When testing this method, they found that participants used the incorrect words at times with the same letter when trying to recall the password from memory. This may be because common words were used. If uncommon words were used the rate of success may have been higher. That being said, it seems that the acronym did assist to a certain extent. The use of an acronym can be applied to passphrases; however, the word chosen for the acronym should be used to trigger the entire passphrase. For example, the letter “T” appears in the acronym which stands for “Terminator”. This may trigger the user to identify that the passphrase is a quote from the Terminator movie; “illbebacksarah”.

In terms of passphrases, a master passphrase can be used instead of an acronym. However, it should be noted that a user also needs to constantly update their mental acronym or master passphrase when a new passphrase is created or a brand new set

of master passphrases or acronyms are created, which may negatively affect memorability. In addition, a user may still struggle to match the correct password/passphrase to the correct system, an issue which these methods do not address. However, if these master passphrases/acronyms are created in a way that is cryptic to anyone else who reads them, then they can be written down. Therefore, even if the master passphrase/acronym is compromised by a third party, it adds no value to them. The next section discusses the impact of changing passwords frequently.

4.3.2.2 Frequency of Password Changing

Carstens et al. (2014) and Woo and Mirkovic (2018) conducted a survey to identify whether users changed their passwords on a regular basis. They found that 69% of participants indicated that they never changed their passwords. In addition, 33% of participants stated that the passwords that were changed were changed back to a previously used password. Another study found that 44% of participants changed their password once a year (Bryant & Campbell, 2006). Keith et al. (2009) and Woo and Mirkovic (2018) also found that many users who do comply with the password changing rule create different but similar passwords. The rule of changing passwords regularly is to address the risk of giving an attacker enough time/attempts to discover the password. It also limits the amount of time an attacker can use a stolen password.

Essentially, the password change rule is a precautionary rule that is difficult to enforce (Schulze, 2018; Zhang & McDowell, 2009). However, this rule should be excused if a password is strong enough to withstand an attack or at least to extend the duration of maintaining the same password before a change is required. Owing to advances in hacking technology over the years, however, the speed with which passwords are hacked may increase. This requires stronger passwords which may in turn have a negative usability impact, as users now have to remember a more difficult password (longer or more complex). This is also applicable to passphrases and is a good reason to introduce a second tier of authentication. The proposed two-tier solution allows a user to keep the same passphrase as the attacker still needs to identify how the user inserts the passphrase (for the keystroke dynamics tier) if the passphrase does become compromised. And since a person's typing patterns do change, this can be used instead of a change in a passphrase. The system development team should then ensure that the keystroke dynamics algorithm accommodates the change in typing patterns.

4.3.2.3 Persuasive Passwords

Forget and Biddle (2008) and Loos and Crosby (2018) suggest that systems should use persuasive password methods for their users. This would entail a balance between system-generated passwords and user-generated passwords. Kävrestad, Eriksson, and Nohlberg (2019) and Keith et al. (2009) state that user-generated passwords have a more positive impact on memorability than system-generated passwords. However, the former generally result in weaker passwords as the control of password strength is shifted from the system to the user. While systems can restrict users from creating weak passwords, this can only be done with certain types of weak passwords, for example by imposing character-set usage as opposed to certain content restrictions on passwords such as adding personal information into passwords. While common user authentication systems merely inform the user if a password is too weak and requests them to create a stronger password, some systems go further to state what the password is missing.

Persuasive passwords, on the other hand, firstly allow the user to create a password of their choosing. If the password is weak, the system converts the password to a stronger one and then proposes a list of stronger passwords from which the user may choose. The user can then either use one of the passwords proposed by the system or create a new one.

An example of a system converting a weak password to a strong password is as follows: "password" is the weak password created by the user. The system, for example, converts any "a" to "@" and any "s" to "\$" until a minimum of two special characters are included in the password. A random two-digit number may also be added to the end of the password. In terms of passphrases, words and English phrases can be used to strengthen the passphrase created by the user. However, it should be noted that it is more complex to set up than a persuasive password solution. That being said, it may still be effective in providing users with alternate options which could assist password memorisation better than the password the user created. Whatever the case, this method, although possible, is not as effective with passphrases as with passwords.

In the above case it should be noted that usability is negatively affected since the user does not create the password they want, assuming that the password created was too weak. However, the password persuasive method removes some of the negative effects on usability, i.e. the fact that users had to restart the authentication process and create a new password which resulted in extra time, effort and brainpower/creativity on the part of the user. This method allows the user to simply select a password proposed by the

system and thus avoids repeating the process of creating another password and the risk of having the password rejected by the system again. The next section briefly discusses the impact on memory of user-generated passwords when compared to system-generated passwords.

4.3.2.4 User-generated Passwords Vs System-generated Passwords

Keith et al. (2009) state that user-generated passwords place less strain on a user's memory than system-generated passwords. However, the latter usually comply with the system's password policy as they are created by the system and provided to the user. Therefore, the latter offers more security than the former. However, system-generated passwords are often not user friendly which leads to users resorting to behaviours that compromise security such as writing them down on a piece of paper. For this reason, it is suggested that users be allowed to create their own passwords. The same recommendation is made for passphrases. The next section discusses the way in which the system development team can assist users to memorise passwords.

4.3.3 System Development Team

This section seeks to determine what a system development team can do to assist a user to memorise passwords and passphrases. The first section discusses the impact of usernames on the memorisation of passwords/passphrases, while the second section discusses password policies and strength indicators. The following section is focused on why the system development team should consider the Chunking theory when deciding on a user authentication approach. The final section explains the different types of memory recall.

4.3.3.1 Usernames

Adams and Sasse (1999) and Li, Wang, and Sun (2018) found that usernames also have an impact of memorisation as users tend to use memory capacity to recall usernames and passwords. Although both are usually required for logging into a system, users have to focus more of their memory capacity on the password/passphrase. Therefore, it is important for the system development team to allow for or provide an easy-to-remember username. It is suggested that the username be something that the user can recall from memory with little effort. The system development team could also remind the user of the username format on the screen. For example, a line could be added under the username input field which states "(email address)" or "(name_surname)". The next

section discusses the memory and security impact of password policies and strength indicators.

4.3.3.2 Password Policies and Strength Indicators

The two most common approaches used by many system development teams to assist users to create strong passwords are password policies and strength indicators. Guidelines on how these should be implemented are normally included in an organisation's information security policy.

Password policies may be defined as system rule/s that restrict the user from creating specific passwords/passphrases. The intention of a password policy is to ensure that all users create a strong password for system authentication. An example of a password policy would be that the password must include one uppercase letter, two numbers, one special character and must be at least six characters long. The system restricts the user from creating an account until the password/passphrase created complies with the password policy criteria. A password policy is normally included in the registration (i.e. password creation screen) screen of a system and is included in the algorithm that validates whether or not the user complies with the password policy of the system. Many password policies have a negative impact on memorability as they are too stringent. Hence, keystroke dynamics would allow password policies to be less stringent as it offers additional security, which in turn would increase memorability, resulting in increased usability.

Strength indicators (also known as password strength indicators) are presented to the user merely to assist them in creating a strong password, with their main intention being to indicate the strength of the password/passphrase they created. Strength indicators are often synchronous and therefore update as the user inserts each character of the desired password/passphrase into the respective field. Strength indicators can be displayed in a number of forms, for example as a bar or simply as text.

Table 4-3 tabulates the different scenarios in which the system development team can include or exclude password policies and strength indicators.

Table 4-3: Relationships Between Password Policy and Strength Indicators

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Password Policy	Included	Excluded	Excluded	Included
Strength Indicators	Included	Excluded	Included	Excluded

From Table 4-3, it can be gathered that password policies can be used together with strength indicators (scenario 1). This is the recommended method as the user is able to understand the rules of the password policy imposed on the system. It may happen that no password policy or strength indicator is included (scenario 2); however, this is not recommended as the user is free to create any password/passphrase they deem fit. For example, even a password such as "1" would be accepted by the system on registration. Scenario 3 involves only a strength indicator being used in a system. In this scenario the system merely suggests that the password/passphrase strength is low or high but does not restrict the user from creating a weak password/passphrase. Although scenario 4 is possible, it is not recommended as it would most likely result in a negative user experience. In terms of this scenario, the system development team has imposed a password policy in the backend of the system and merely restricts the user from registering without indicating what was wrong with the password/passphrase they created.

It is also important to note that a correlation between the password policy and the strength indicator is not necessary. However, misalignment may cause confusion for the user. Althubaiti (2017) and Bhivgade et al. (2014) found that users are interested in the rules behind the strength indicators. If a user fails to understand these rules, a negative user experience may be a result, especially if there is no correlation between the password policy and the strength indicator. The system development team should prevent users from having to use unnecessary brain power when trying to determine the password policy using the strength indicator as he/she should be more focused on creating a password/passphrase that they can recall from memory. The next section explains why the system development team has to consider the Chunking theory when developing a user authentication approach.

4.3.3.3 Chunking Considerations

Wijayarathna and Arachchilage (2019) and Wright et al. (2012) state that password security best practices tend to focus on security as opposed to usability, not

understanding that other security issues may occur if usability is not considered. Carstens et al. (2014) and Mwagwabi, McGill, and Dixon (2018) add to this, stating that they believe that cognitive theories should be considered when ensuring the security of systems. They also predict that authentication systems will be more robust if they are designed in line with the way in which users create and memorise passwords. Carstens et al. (2014) and Mwagwabi, McGill, and Dixon (2018) goes into more detail, mentioning that few guidelines have been developed and even fewer implemented that educate users on how to create meaningful but strong passwords. Forget and Biddle (2008) suggest in this regard that the Chunking theory should be considered when developing user authentication systems.

Carstens et al. (2014) and Nicholson, Vlachokyriakos, Coventry, Briggs, and Olivie (2018) propose a list of password guidelines, which are indicated below:

- Use two to four chunks of data.
- Include 10 to 22 characters.
- Contain multiple character sets (numbers, letters and special characters).
- Words cannot be repeated.
- No dictionary words.
- No proper nouns such as names of people, places or things.
- No personal information that is easily obtained such as ID number and date of birth.

Although these guidelines significantly increase security, they likewise have a negative effect on usability because the user has to apply more time and effort to think of a password that conforms to these requirements. In addition, the password will probably be difficult to memorise owing to the number of rules that have to be addressed. However, from a user educational perspective, it would be valuable to create awareness about how users can make their passwords stronger. Therefore, it is recommended that rules similar to those mentioned be communicated, but not enforced by the system. For example, do not restrict access to the system if a user's password does not align to the password guidelines. Restrictions can also be made less stringent, for example by communicating the password guidelines to the user and explaining that at least three guidelines must be adhered to, to gain access to the system. Woo and Mirkovic (2018) and Zhang and McDowell (2009) support this approach by explaining the importance of providing guidance to a user on creating a password. In addition, it is important to align strength indicators to the guidelines and indicate why the password created is weak as

opposed to merely informing users that their password is weak. This also applies to passphrases. All these considerations should be documented in an organisation's information security policy to ensure consistency across systems in the business.

4.3.3.4 Memory Recall

Wright et al. (2012) explain that there are three types of memory recall with regard to information previously acquired by a person. These are stated below.

- **Recognition** – A person is presented with information and must then determine whether they recognise the information or not. In terms of passwords/passphrases, this would be a scenario in which the user is presented a list of five passwords/passphrases, for example, and they are required to select the one password/passphrase that they created. However, this situation is unlikely due to the high security risk associated with such an approach.
- **Cued recall** – Information triggered from memory based on some form of assistance. From a password/passphrase perspective, this is password/passphrase hints created by the user when they created the password/passphrase.
- **Uncued recall** – Information called from memory with no assistance. Uncued recall is currently included in the user authentication process of most systems. The user is expected to recall the password/passphrase from memory without any assistance.

Based on the above component of memory recall, it can be seen that although recognition is not an effective approach, cued recall is worth exploring. It is clear that uncued recall tends to create memory issues for users memorising passwords/passphrases. Hence, authentication solutions should be implemented that provide the user with the option of taking advantage of the cued recall component of memory recall.

Password hints are generally conveyed to the user by offering them a text-based option for creating a hint for the password to assist them to recall the password from memory. The system then presents this password hint to the user if they indicate to the system that it is required. The same approach can be applied to passphrases. Revett and Bahaa (2012) explored the possibility of offering users the ability to provide visual password hints. They argue that humans recall items from memory better using visual cues as opposed to text-based information. Johnson (2014) and Pearman, Zhang, Bauer, Christin, and Cranor (2019) supports this by explaining that although computer

icons have an image and a text label attached to them, many users have become so familiar with the icon images that they do not need to read the label in order to identify the icon to be selected. Hence, users should be allowed to add one or more images to the system to assist them in recalling the password/passphrase. However, even though this is optional, usability may be affected as a user needs to search for an image, save it to their computer and then upload it on the system. This requires additional time and effort to avoid a risk that may or may not occur and thus will negatively affect usability. For this reason, users should be allowed to simply draw an image/s on the system with their mouse. This prevents them from having to switch to an alternative screen in order to create a password/passphrase hint.

In addition, systems usually only allow a user to create one hint as opposed to multiple hints. Accordingly, users should be allowed to provide one or many password/passphrase hints. These hints can then be provided to the user randomly or the user can flip through them. In order to restrict anyone from viewing a user's password/passphrase hint/hints, a second tier of authentication should be added which perhaps sends a verification pin to the user's email address or cellphone. However, it should be noted that this would have a negative impact on usability as additional actions would be required by the user.

In summary, it is suggested that visual hints (uploading of an image or simply drawing an image with a mouse) be offered to the user as an option in addition to text-based hints. Additionally, one or more password/passphrase hints should be allowed to assist users to recall passwords/passphrases. However, there should be a restriction on the number of these hints to ensure the user does not reveal too much. The next section is the final section of this chapter and is focused on the way passwords and passphrases can be created to best assist memorisation.

4.3.4 Password Composition

This section on password composition focuses on discussions relating to constructing passwords/passphrases to assist with password/passphrase memorisation. The first section discusses the use of multiple character sets before discussing abbreviating passphrases and the impact of this on security and usability. The next section discusses the use of phonologically similar words in passwords, while the last section discusses the impact of password length on security and usability.

4.3.4.1 Multiple Character Sets

Complex passwords are recommended for security purposes and are sometimes enforced through password policies to ensure that a certain level of security is maintained by users (Braz & Robert, 2006; Yıldırım & Mackie, 2019). Brumen (2019) and Vu, Tai, Bhargav, Schultz, and Proctor (2004) suggest that a password should include different character sets. However, they note that this does create memory issues for users which result in login errors. They go on to explain that a complex password may only be a temporary memory issue as, over time, it will not demand much memory. In addition, the frequency of password use also has an impact on memorability. If a user only uses a system once a month as opposed to a system used daily, the password used less frequently may take more time to memorise than the one used more frequently. While this may be true, passwords should be changed constantly for security reasons (Woo & Mirkovic, 2018 and Zhang & McDowell, 2009).

The problem with adding multiple character sets to a password is that there is not always a personal link between modifying a password by adding numbers or special characters. However, numbers may be easier for the user to relate to than special characters if used to represent personal experiences, i.e. meaningful numbers; for example, a user using their house number or special date such as a date of birth or anniversary date. It should be noted that this is not advisable as this data is often available to hackers or can be collected via a phishing attack.

With regard to special characters, a user can create a personal word-converter cheat sheet which they follow for any password created. An example of a word-converter cheat sheet would be as follows: convert any "a" to "@" and any "s" to "\$". However, this negatively affects memorability as the user now has to memorise this password cheat sheet. In addition, these converted dictionary words are more likely to be incorporated in a password cracking attack if the hacker understands the password policy applied to the system, i.e. all password guesses include one or more special characters if the password policy for the system restricts users from accessing the system without such a requirement.

Since there are a number of memorability issues associated to a password with multiple character sets, passphrases are recommended in preference to passwords from a memorability perspective with the keystroke dynamics algorithm further increasing security. The next section discusses the impact of abbreviating passphrases in terms of security and usability.

4.3.4.2 Abbreviating Passphrases

Based on their research, Forget and Biddle (2008) suggest that passphrases be used; however, the phrase selected by the user should be abbreviated as this would seem to assist with memorability. This may even increase security, as abbreviation converts the phrase to a less commonly used phrase. However, these findings contradict the Chunking theory. If the Chunking theory is applied to this recommendation, then additional chunks are required to remember how the passphrase was abbreviated in addition to the passphrase itself. Because not all words can be abbreviated, the user will need to remember which words they abbreviated if at all. Consequently, it is not recommended that passphrases be abbreviated owing to the strain it places on the user's memory. The next section discusses the usability and security impact of phonologically similar words.

4.3.4.3 Phonologically Similar Words

Wright et al. (2012) conducted a study to test certain aspects of passwords and their impact on memorability. They discovered that users found a password easier to remember if it included semantics, for example, related words such as eyes and sunglasses, or Brazil and soccer. The participants of the study also mentioned that verbs were easier to recall from memory. These results were supported by stating that they align to the findings of the original study (Anderson, Wagovich, & Brown, 2019; Deese, 1959). Passwords do not usually include such an approach; because of the multi-character requirement, users generally create passwords using one word and modify that word to align with the password requirements. For example; the word "toyota" is selected and converted to "Toyot@1997". With current password requirements, adding two or more words to a password and still aligning them to the password requirements severely affects the memorability of the password. Passphrases, on the other hand, support semantics and verbs and may be included in a passphrase with little creativity from a user. Examples of passphrases which include semantics and verb/s would be "watchstarwarsdvd" or "raceredferrari". Therefore, when prompting users to create a passphrase, it should be suggested but not required that they create one that includes semantics and verb/s to assist with memorability.

Phonologically similar words are more difficult to differentiate than non-phonologically similar words (Anderson, Wagovich, & Brown, 2019; Keith et al., 2009). This confuses the user regarding the word/s selected; for example, "there" and "their", "lite" and "light", and "threw" and "through". It is therefore suggested that such phonologically similar

words be avoided as they create memory issues. The following and last sub-section of this section discusses password length.

4.3.4.4 Password Length

A study conducted by Bhanbhro, Hassan, Nizamani, Bakhsh, & Alassafi (2018) and Proctor et al. (2002) found that simply increasing character length will increase security. In addition, different character sets also have the ability to increase security but these are best avoided because of their negative impact on usability.

Carstens et al. (2014) and Hou, Wei, Wang, Wang, and Xu (2018) suggest that users' passwords should be of different lengths. This avoids the risk of having multiple passwords compromised. They explain that when hackers crack one password, they usually focus their attacks based on the password found, i.e. password length, characters used and various versions of the word/s used in the password. Therefore, Carstens et al. (2014) and Hou, Wei, Wang, Wang, and Xu (2018) suggest that different password lengths be used to avoid the risk of additional damage. This also applies to passphrases. In the case of text-based authentication, this may have a negative impact on usability, as additional characters are required to memorise some passwords/passphrases. Thus, additional character memorisation is required as systems should still maintain a minimum password length requirement for security purposes. Since users usually aim to meet the minimum password requirements for a system to assist memorisation, some passwords may need to exceed minimum password requirements.

In terms of passphrases, memorisation for a change in character length may not be much of an issue as passphrases are unlikely to have the same character length. In fact, attempting to align the character length of passphrases may negatively affect memorability. For example, if "marvelsuperherouniverse" is shortened to "marvelsuperherouni" an extra chunk of information is required to remember that "universe" has been shortened to "uni". Alternatively, an extra chunk may be required if the passphrase is extended to change the password length, for example "marvelsuperherouniverse" is extended to "bestmarvelsuperherouniverse". Therefore, the same memory capacity is required whether the passphrase is reduced or extended. However, if the passphrase is extended, the passphrase will be more secure.

This concludes the main section of this chapter in which the user authentication components that affect the memorability of passwords and the application of these

components in the context of the proposed solution were discussed. The next section provides a summary of the components in a graphical presentation that may assist memorability for the proposed two-tier user authentication solution involving passphrases and keystroke dynamics.

4.4 Passphrase and Keystroke Dynamics Memorisation Usability Model

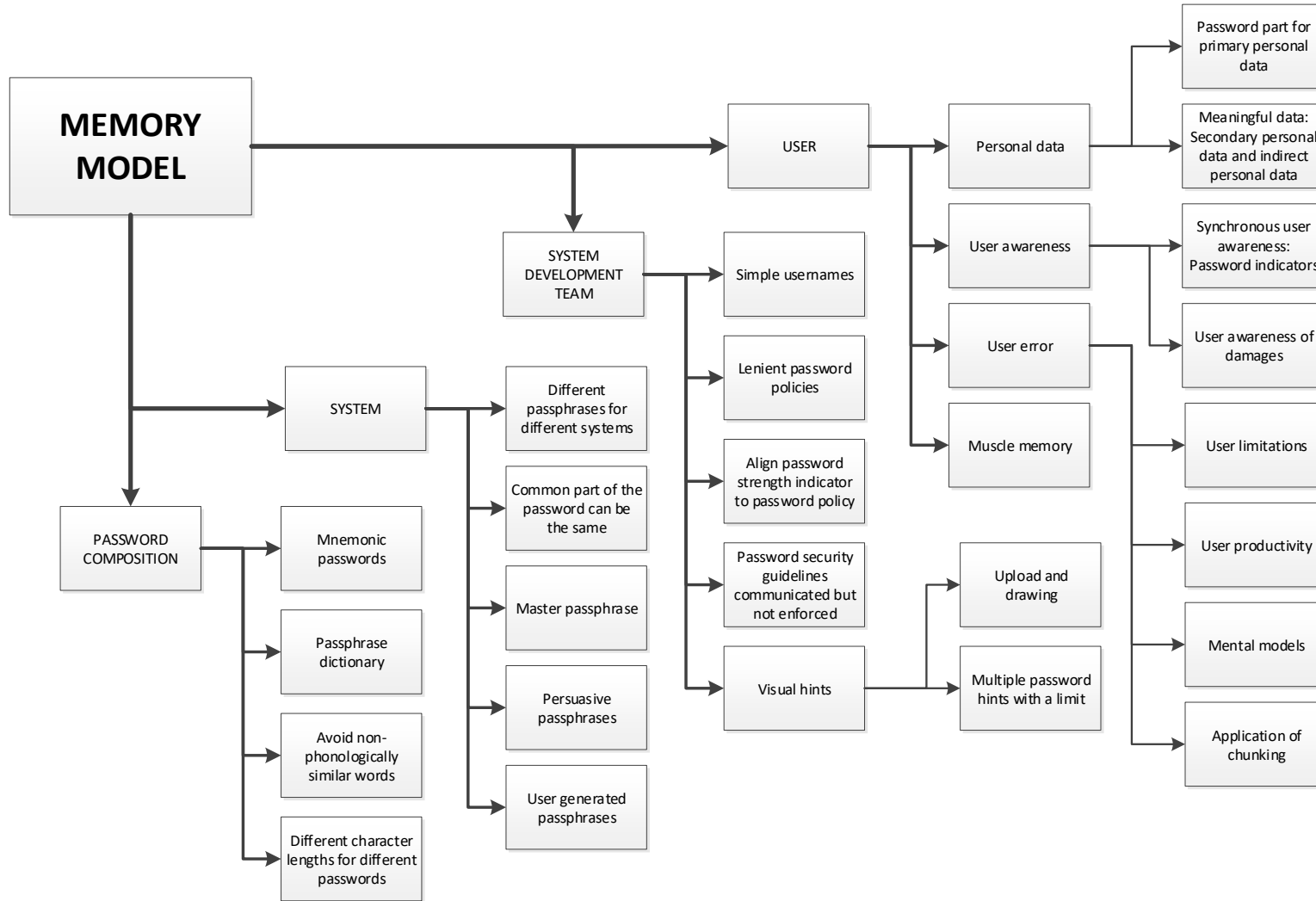


Figure 4-2: Passphrase and Keystroke Dynamics Memory Model

Figure 4-2 above graphically provides a summative view of the discussion on the memory components that have the potential to positively influence the memorability of passphrases when passphrases are supported by keystroke dynamics. Therefore, these components may be used to assist the memorability aspect of usability for the proposed solution without compromising security.

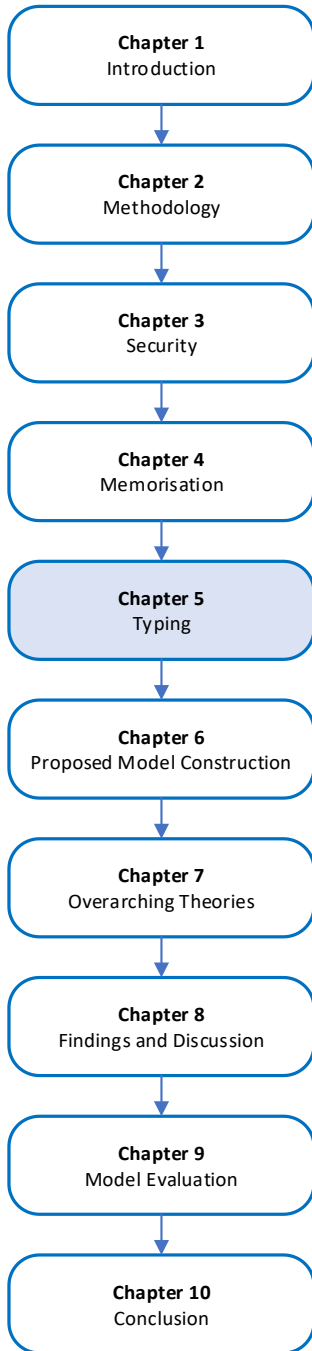
4.5 Conclusion

This study proposes a two-tier user authentication solution which incorporates passphrases and keystroke dynamics. In order to assess the effectiveness of the proposed solution, in-depth research is required on two aspects – security and usability. Two considerations should be assessed regarding usability – memory and typing. This chapter has focused on the memorisation aspect of usability.

Based on the existing literature it may be argued that passphrases are easier to memorise than passwords. It was discovered that leniency can be provided on various security protocols that have an effect on memory as the keystroke dynamics algorithm can make up for the security lost to leniency. Four main components exist which can assist password/passphrase memorisation. The four main components that can influence memory are users, systems, the system development team and password/passphrase composition. It is important that all these components work together to support the user to better memorise passwords/passphrases.

This chapter focused discussions and findings on the memorability aspect of usability in terms of the proposed solution. The next chapter is focused more on the second aspect of usability – the typing of passwords/passphrases as a usability issue with regard to user authentication.

Chapter 5 – TYPING



Chapter 5	
5.1	Introduction
5.2	Keystroke-level Model
5.3	Keystroke-level Model Application
5.4	Keystroke-level Model Scope
5.5	Usability Impact on Typing Different Password Types
5.6	Types of Typographical Errors
5.7	User Authentication Process
5.7.1	User Authentication from a User Perspective
5.7.2	User authentication from a System Perspective
5.8	Typing Issues Affecting Usability
5.8.1	Association between Errors and Keys Pressed
5.8.2	Association between Errors and Typing Duration/Speed
5.8.3	Impact of Routine Tasks and Non-routine Tasks on Typing
5.8.4	Language Deviations: Punctuation and Grammar
5.8.5	Typing Effort
5.8.6	Hidden or Unhidden Text
5.8.7	Keyboard Exposure
5.9	Passphrase and Keystroke Dynamics Typing Usability Model
5.10	Conclusion

*Only heading 1 and heading 2 is displayed to avoid clutter

5.1 Introduction

Typing errors occur regardless of the level of expertise of users. Kozak, Krzanowski, Cichocka, and Hartley (2015) found that 28% of their participants made at least one error while typing. Koester and Mankowski (2015) also found that considering the average of all keys pressed on a keyboard by a user, 18% of keys pressed are typed unintentionally by the user (i.e. in error). This estimates the risk of a typing error occurring once in every five keys pressed, thus negatively influencing usability, as typing errors are an unwanted occurrence that should be avoided as much as possible. Avoidance of typing errors is important to this study, as the proposed solution required users to authenticate themselves by typing a passphrase. Wixom and Todd (2005) and Woo and Mirkovic (2018) assessed the usability of login interfaces using the Technology Acceptance model. They found that if a user fails to login to a system regardless of whether it is due to typing or memory errors, negative perceptions are created about the “ease of use” and “usefulness” of the system. This in turn negatively affects the usability of the system.

Figure 5-1 below depicts the aim of this study. The areas highlighted in grey indicate the part of the research study that this chapter focused on.

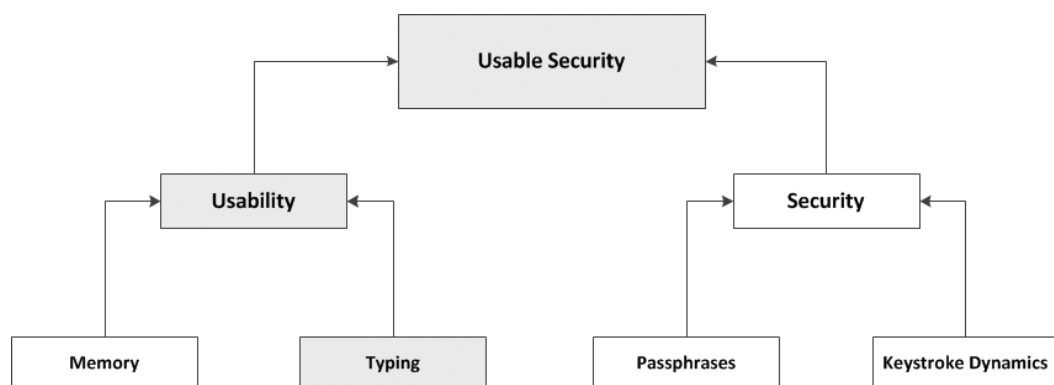


Figure 5-1: Research Summary

This chapter is focused on identifying the factors affecting typing usability and assessing these factors against the proposed solution. The intention of this approach is to determine the extent to which the proposed solution can address the factors influencing typing usability. The guiding theory for this chapter – the Keystroke-level model (Lee et al., 2015; John & Kieras, 1994; Jorritsma et al., 2015) – is firstly discussed. A brief explanation of the impact of typing on different text-based authentication types is followed. Types of common typing errors are then discussed. The user authentication process from a user perspective is then provided which is followed by a discussion on user authentication from a system perspective. The main topic of this chapter, typing

aspects of usability, is provided. This is followed by the construction of a model derived from the findings on the typing aspects of usability. Lastly, a conclusion to this chapter is provided. An explanation of the theory used to direct this chapter is discussed next.

5.2 Keystroke-level Model

The Keystroke-level model (Lee et al., 2015; John & Kieras, 1994; Jorritsma et al., 2015) was used to guide this chapter to its research findings that focus mainly on typing usability. This section firstly discusses the application of the Keystroke-level model to this study, followed by a section on defining the scope of the Keystroke-level model for this study.

5.3 Keystroke-level Model Application

The Keystroke-level model is a tool used to measure the duration taken by an expert user to execute a routine system task (Lee et al., 2015; John & Kieras, 1994; Jorritsma et al., 2015). In this research, the routine task being executed is the user authentication process. The Keystroke-level model ensures that all components that may affect usability during the user's interaction with the system are observed. The model consists of six assessments, namely:

- **K** – Number of keys pressed
- **P** – Pointing to on-screen targets (beyond the scope of this research)
- **H** – Hands to keyboard and mouse
- **D** – Drawing a line to assess mouse movement (beyond the scope of this research)
- **M** – Mental preparation required
- **R** – Response time of the system

Two assessments are beyond the scope of this research study (pointing to on-screen targets and drawing a line to assess mouse movement) as they relate to the user actions required before a user inserts their password/passphrase, i.e. navigating to the login screen, finding the fields to insert the username and then moving to the password/passphrase field. However, it should be noted that the process whereby the user navigates to the login screen in order to enter a password/passphrase needs to be considered for the proposed solution when comparing it to other forms of user authentication (i.e. biometrics and graphical passwords). This study did not consider the two assessments as this research was focused on text-based user authentication only. However, the reason it should be considered when comparing the proposed solution to

other forms of user authentication is that some user authentication methods do not require the user to provide a username: for example physical biometric authentication (Lupu & Valeriu, 2014); devices that assume a one-to-one relationship between user and system/device (Shay et al., 2016), i.e. only one user uses the system/device; and cellphones and tablets where no username is requested to access the device, merely a password/passphrase/PIN.

Since time has an impact on usability, the Keystroke-level model was used to guide the discussion on the user's interaction with a typical login interface in terms of inserting the password/passphrase into the system. Four of the six assessments from the Keystroke-level model above were used to assist in the identification of the issues influencing these four assessments and determine whether the proposed two-tier solution (passphrases and the keystroke dynamics algorithm) can address these issues. The model also supported the development of the login assessment fieldwork, which is discussed in more detail in the data collection methods section in Chapter 2.

It is important to note that passwords yielding higher usability are not very secure (Hussain et al., 2018), while passwords yielding a lower level of usability are generally more secure (Gahlot & Gupta, 2016). Therefore, a balance needs to be found between usability and security. The same applies to passphrases. The Keystroke-level model uses time (in duration) as the variable for assessing usability (Giot et al., 2011; Leino et al., 2019). The longer the amount of time taken to execute a task, the less usability is achieved. Therefore, the Keystroke-level model sees a correlation between effort and time. It explains that the more time that is required to complete a task, the more effort is required by the user (Banerjee & Woodard, 2012; Leino et al., 2019), as the executable actions (components of the Keystroke-level model) are seen as the effort required to be exerted by the user. The following section clearly indicates the scope of the use of the Keystroke-level model in this study.

5.4 Keystroke-level Model Scope

Since the scope of this study includes the usability impact of text-based authentication, user–mouse interactions have no direct impact on the usability of text-based authentication, hence, this assessment was not considered for this study. A measure for “pointing to on-screen targets” in the Keystroke-level model was also not assessed as it relates to user–mouse interaction. The same reasoning applies to the “drawing a line to assess mouse movement” assessment as this item is directly related to user–

mouse interaction. However, it should be emphasised that these assessments need to be considered when comparing the proposed solution to non-text-based user authentication methods as discussed above.

Although “response time of the system” does not directly affect passwords/passphrases, it was considered for this study. This is important as the keystroke dynamics algorithm may have a negative impact on system response time. This is especially significant when a number of user–keyboard interactions are being monitored and/or the user system performance is low. For this reason, user authentication would affect system response time. “Number of keys pressed” and “mental preparation required” directly affects usability from a password/passphrase perspective and is, therefore, considered. Now that the application scope of the Keystroke-level model in this study has been defined, the next section discusses the usability impact on typing common password types.

5.5 Usability Impact on Typing Different Password Types

Shay et al. (2016) created four password composition categories based on the common types of character sets used in passwords:

- "comp8" which represents a typical password that includes eight characters with at least one character from each character set (special character, digit, uppercase letter and lowercase letter).
- "basic12, basic16, basic20" which includes any character set as long as the password is 12 characters long (basic12), 16 characters long (basic16) or 20 characters (basic20) long.
- "2word12, 2word16" are passwords which must include two words and must be 12 characters (2word12) or 16 characters (2word16) long.
- "3class12, 3class16" are passwords that must include any 3 character sets and must be 12 characters (3class12) and 16 characters (3class16) in length.

From a typing perspective, Shay et al. (2016) found that 3class16 passwords and basic20 passwords took longer to type than comp8 passwords. Shay et al. (2016) does not explain why their research findings identified that comp8 passwords are easier to type. It is possible that participants in their study were not asked to use a new password. Since comp8 passwords are commonly required for most user authentications at the time this study was conducted, participants may have used a comp8 password that they were

familiar with typing. This is explained in more detail in the Section 5.8.3 Impact of Routine Tasks and Non-routine Tasks on Typing.

Shay et al. (2016) also found that 3class16 passwords are more difficult to type than any other password composition category in the above list. From a security point of view, 3class16 passwords would be more secure than basic20 passwords, and basic20 passwords would be more secure than comp8 passwords. When comparing these passwords from a security perspective, it can be seen that the more secure a password is, the more difficult it is to type. Although comp8 passwords were easier to type than basic20 and 3class16, it has the lowest security of the three password types (basic20, 3class16 and comp8). Alternatively, the most secure of the three password categories being compared is 3class16 and it is also the most difficult to type. The basic20 password category offers a balance between security (more secure than comp8 but less secure than 3class16) and usability from a typing perspective (easier to type than 3class16 but more difficult to type than comp8). Since basic20 is the closest comparison to a passphrase and comp8s are the most commonly used passwords, it would seem that passphrases offer a balance between security and usability. In this study, the second tier of authentication (keystroke dynamics algorithm) can offer further security while maintaining usability. The next section discusses the common types of typographical errors created by the user.

5.6 Types of Typographical Errors

The risk of users making errors needs to be reduced as much as possible. When a user makes an error during the user authentication process, it often results in the user having to repeat an action/s or at least part of the action. Moreover, the error requires the user to spend additional effort and time in either redoing the task or attempting to rectify the error. This additional time and effort, which could have been avoided, then results in a negative usability experience. Keith et al. (2009) identify three types of typographical error:

1. **Substitution errors** – A user mistakenly presses a nearby key instead of the targeted key on the keyboard.
2. **Temporal errors** – Transposing hardcopy text into a system (e.g. data capturing).
3. **Execution errors** – Inserting more or fewer keys than intended.

Points 1 and 3 above are the most common types of errors that users make when typing a password/passphrase (Cox, Cox, & Cox, 2017; Keith et al., 2009; Paz & Granollers, 2019). Point 2 is not applicable to typing passwords/passphrases as users should not be copying a password or passphrase from anywhere but should be retrieving it from memory. The next section discusses the user authentication process.

5.7 User Authentication Process

This section consists of two subsections. Firstly, the user authentication process from a user perspective is discussed, followed by a section focusing on the user authentication process from a system perspective.

5.7.1 User Authentication from a User Perspective

A user transitions through a number of mental and physical states when authenticating themselves on a system. Figure 5-2 indicates the typical steps a user passes through during a typical text-based user authentication process involving passwords. This also applies to passphrases. Login pages from popular websites such as Facebook and Skype were used to assist the mapping of the process in Figure 5-2.

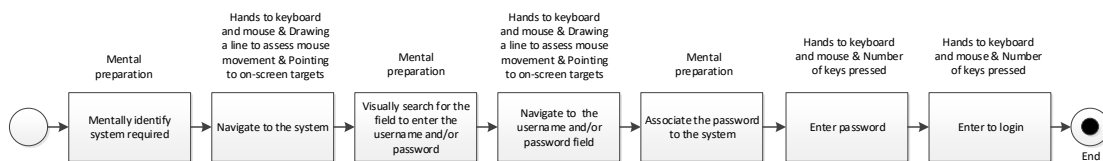


Figure 5-2: Typical Login Process from a User Perspective

In Figure 5-2, the categories in the Keystroke-level model (Lee et al., 2015; John & Kieras, 1994; Jorritsma et al., 2015) were applied to the different phases of the login process from a user perspective (text above the block in Figure 5-2). Firstly, the user identifies the system they are trying to gain access to and then navigates to that particular system. The user then looks for the field in which they need to enter their username and password. Once this field is visually identified by the user, he/she navigates to this field. The user then mentally associates the password with the system he/she is trying to gain access to before entering the password into the system. Once the password is inserted, the user presses the “enter” key on the keyboard or the login button to inform the system that the process of entering the username and password is complete.

The process depicted in Figure 5-2 is commonly found regardless of whether or not a user authenticates himself/herself with a password or a passphrase. The keystroke dynamics algorithm can only influence usability when the user interacts with the system,

as the algorithm is a system-based approach aimed at increasing security and usability. A more detailed discussion on the impact of this algorithm on usability is provided in Section 5.7.2, User Authentication from a System Perspective.

As previously mentioned, users like to complete the user authentication process as effortlessly and in as timely a fashion as possible. Figure 5-2 also depicts the process a user has to perform when authenticating themselves through the proposed two-tier user authentication solution, using a passphrase with the keystroke dynamics algorithm running in the background. This is because the proposed solution does not affect the authentication process from a user perspective. In order to illustrate the significant usability impact that the proposed solution can offer, Figure 5-3 graphically depicts a common two-tier process where a password (or passphrase) is used as the first tier of authentication and an OTP (one-time pin) is used as the second tier.

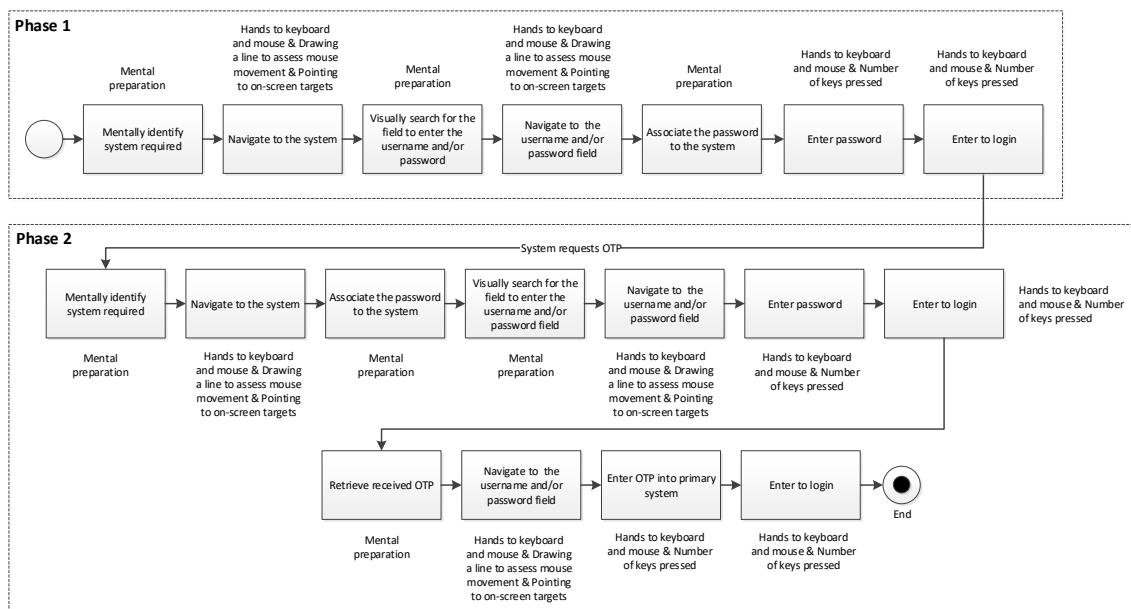


Figure 5-3: Typical Two-tier Login Process Using OTP as the Second Tier

Figure 5-3 illustrates the process a user has to undergo in order to authenticate themselves on a common two-tier authentication solution which uses a password and an OTP that is sent to their email or cellphone. The entire process in Figure 5-3 is separated into two phases. The first phase groups the processes for the first tier of authentication and the second phase groups the processes for the second tier of authentication. As may be seen, the second tier of authentication (OTP) demands a significant amount of time and effort on the part of the user. The diagram in Figure 5-3 shown that the user has to switch between three different input devices to login to the system. The first being

the mouse to navigate to the username and password field, the second is the keyboard and the third is a cellphone or a tablet to retrieve the OTP. This switching has a negative impact on usability. The proposed two-tier user authentication solution (illustrated in Figure 5-2) has a much lower impact on usability by reducing the number of steps a user needs to perform to authenticate themselves.

The next section provides a typical one-tier user authentication process from a system perspective.

5.7.2 User Authentication from a System Perspective

There are certain actions a system has to perform throughout the user authentication process, with the system response time component of the Keystroke-level model having the most impact on user authentication from a system viewpoint. System response time may be defined as the speed of executing a request made either by the system or the user (Bellamy, John, & Kogan, 2011; Riedl & Fischer, 2018). Logically, the faster the response time of the system the greater the satisfaction experienced by the user and concomitantly an increase in usability.

Figure 5-4 graphically illustrates the system actions required to assist the user through the user authentication process. Figure 5-4 is an extension of Figure 5-2, which displayed the actions a user performs throughout the login process.

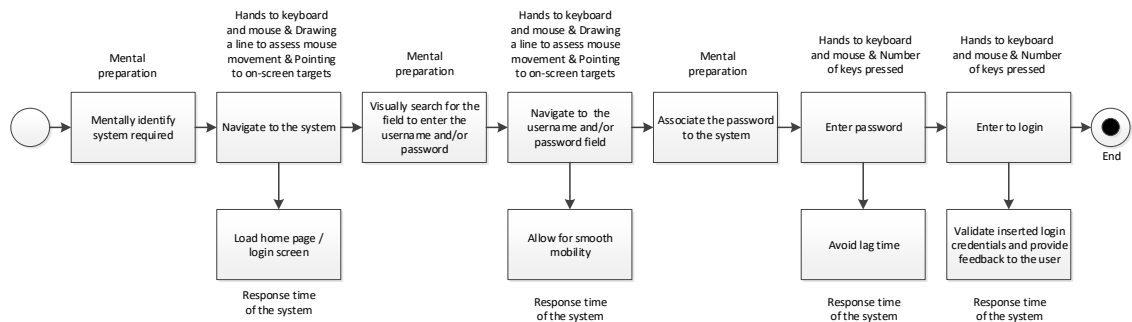


Figure 5-4: Typical Login Process from a User-System Perspective

The first action a system has to perform for the user during the user authentication process is displaying the login screen so that the user has an interface to login to the system. Usability is negatively affected if the loading time of the login screen is slow or the login screen loads incorrectly.

The second time the system assists the user with the login process is when the user needs to navigate to the username and password field on the login screen. In order to avoid negative usability, the system should ensure that the user can navigate to the

username and password field as smoothly as possible. It is important that the team that develops the login screen ensures that the user can find the username and password field on the login screen easily otherwise usability may be affected.

The first and second system support discussed above are important but are beyond the scope of this study. The third system support is partially within scope for this study, i.e. when the user enters the username and password – in this study the entering of the password is important. Usability is negatively affected if there is a lag time between the point that the user enters a character on the keyboard and the point where the system displays the character insertion on the screen. This time should be reduced as much as possible to ensure a positive user experience regarding the user-system interaction of the user inserting the username and password into the designated field on the login screen.

It should be noted that system response time will be similar for passwords and passphrases. System lag should not differ when different keys are pressed, only if the speed of typing changes (Liu, 2013; Priva, 2010). However, strictly speaking, since it should be possible to type passphrases faster than passwords because there is no character switching (see Section 5.8.2 Association between Errors and Typing Duration/Speed and Section 5.8.3 Impact of Routine Tasks and Non-routine Tasks on Typing), system lag time has a higher risk of occurring with passphrases than passwords. Although this is unlikely to create system response time delays, this is possible if applications are running that consume a lot of the system's speed. This needs to be noted when implementing a keystroke dynamics algorithm as it may create system response time delays for the user when typing the password/passphrase. This is likely to occur if the keystroke dynamics algorithm is monitoring a number of different user–keyboard interactions.

Once the user has entered the username and password, he/she presses the “enter” button on the keyboard or clicks the login button on the login screen to inform the system that the password/passphrase has been inserted. The system must subsequently ensure that two conditions are fulfilled to avoid the user having a negative user experience in this phase (Archibald & Ferguson, 2017; Rajanen & Rajanen, 2017) namely:

1. Provide feedback to the user as soon as possible. Feedback should indicate that the password/passphrase entered was correct or incorrect.

2. Ensure that the feedback to the user is correct, i.e. the system needs to provide reliable results.

This is the last form of support that the system can provide for users in the user authentication process. It is important to note that secondary flows such as the user selecting the “forgot password” (or “forgot passphrase”) option also require system support to ensure that no negative usability is experienced by the user.

In terms of the proposed solution, the second tier of authentication (keystroke dynamics algorithm) may have a large impact on system response time. Alshehri, Coenen, and Bollegala (2018) and Banerjee and Woodard (2012) state that the data collection and analysis required by the keystroke dynamic algorithm require a large volume of processing power (if a high level of user-keyboard interaction monitoring was applied), which may create performance issues on the system. This has been found to be the case when accessing a system through a device that cannot process large quantities of data such as a mobile phone or tablet (Mittal & Singh, 2016; Perera, Zaslavsky, Christen, Salehi, & Georgakopoulos, 2012). However; Alshehri et al. (2018) and Perera et al. (2012) explain that the processing power of mobile devices is becoming stronger and stronger over time.

A more acceptable approach for users would be if the keystroke dynamics algorithm were only applied to part of the system (such as the login screen) or for a specific duration (e.g. monitoring user interactions on the system for 1 minute at 30 minute intervals). However, a greater negative impact on usability would be experienced if the keystroke dynamics algorithm were programmed to run consistently throughout the user–system interaction lifespan. It is important for the system development team to consider this when implementing the keystroke dynamics algorithm. Two solutions are applicable in this case:

1. The system development team can reduce the number of user interaction elements monitored by the keystroke dynamics algorithm. This ensures that the keystroke dynamics algorithm does not negatively affect system response time.
2. The duration of the monitoring may be reduced. For example, reduce the level of monitoring from full user–system interaction to merely login screen monitoring. It should be noted that these two reductions would increase usability but reduce security.

In summary, it was found that passphrases have a higher risk of creating system lag than passwords owing to typing speed. However, this only has an impact if the keystroke

dynamics algorithm requires a lot of processing power. In terms of the impact of the keystroke dynamics algorithm on system response time, the number of monitoring mechanisms and duration of monitoring can have a significant impact on system response time (Lin, Liu, & Lee, 2018). The faster the system's response time, the greater the positive impact experienced in terms of usability. The next section introduces the various typing issues that have an impact on usability.

5.8 Typing Issues Affecting Usability

The intention of this section is to identify typing issues that affect usability. Each of these issues is assessed against the proposed two-tier solution (passphrases and keystroke algorithm) in order to understand their impact on the proposed solution. The first typing issue discussed is the association between errors and keys pressed. This is followed by the association between errors and typing duration/speed. The impact of routine tasks and non-routine tasks on typing is then discussed. The next section discusses the impact of language deviations on typing usability, followed by the typing effort associated with typing usability. This is followed by a section on hiding and displaying the text that the user types. The final section discusses the typing usability impact on keyboard exposure. The first typing issue affecting usability is the association between errors and keys pressed.

5.8.1 Association Between Errors and Keys Pressed

Keith et al. (2009) and Saleh and Mashhour (2018) explain that the more keys that are pressed by the user, the higher the probability of an error occurring. However, if a certain sequence of keys is pressed on a regular basis, then this sequence has a lower probability of errors than an unfamiliar sequence. Therefore, it is more accurate to state that the risk of typing errors is higher when a sequence of keys unfamiliar to the user is pressed. Rieger (2004) supports this statement by explaining what she terms "automatic activation". This is a situation where a user becomes so familiar with typing a specific word or phrase that they require little effort to do so. She adds that this repetition increases the speed and accuracy of typing this word or phrase.

These findings complement passphrases as opposed to passwords. If a user has converted a word to align to a common password policy requiring special characters and numbers, then the risk of error increases as this is an uncommonly typed word. For example, the user is most likely to make an error typing the password "b@33" than "bass". Similarly, users who frequently type numeric groups of characters are less likely

to make a typing error (Lin & Wu, 2011). For example, year of birth 1989, ID number 890522 and student number 2008035. Therefore, passwords that include these number groups at the beginning or end, as opposed to numbers included in words, are more likely to be typed correctly. For example, "fox1989" or "890522fox" are less likely to be typed incorrectly than a password such as, "b33s" or "1ll". This is because the user has experience typing these character sequences outside and in addition to typing the sequence for their password.

In terms of the keystroke dynamics algorithm, the fewer false positive results generated by the keystroke dynamics algorithm, the greater the usability. A false positive is a term used to describe an unwanted result of the keystroke dynamics algorithm where an authorised user is refused access even though inserting the correct password/passphrase (Ponkshe & Chole, 2015). This is caused by the authorised user inserting the password/passphrase outside the acceptable threshold set by the keystroke dynamics algorithm (Ponkshe & Chole, 2015), thus producing an unwanted result from a keystroke dynamics algorithm perspective and negatively affecting the usability of the authentication process. The reason for the negative impact on usability is that this scenario results in the user having to insert the password/passphrase again.

In summary, repetition of actions reduces the probability of errors. Since alphabetical characters on a keyboard are used more than special characters and sometimes even numeric characters, passphrases are less likely to create typing errors than passwords. This is because passphrases include only one-character set as opposed to passwords. In addition, the character set used for passphrases is the character set used most often by users. Therefore, switching character sets increases the likelihood of errors. The Keystroke-level model category related to the findings in this section is the "number of keys pressed".

5.8.2 Association Between Errors and Typing Duration/Speed

Research by Shay et al. (2016) highlights two main success criteria for assessing the success of typing:

- Time taken to type the password/passphrase
- Risk of errors

Both items in the list above have an opposite effect on the other. If any errors occur when typing the password/passphrase (risk of errors), the time taken to type the password/passphrase is extended. Likewise, the time taken to type the

password/passphrase may be increased to reduce the risk of typing an incorrect key (risk of error), i.e. typing slower to avoid pressing an incorrect key on the keyboard.

There is not a vast difference between the number of keys a user presses on a keyboard for a password and a passphrase. This is because passwords require the shift and caps lock keys to toggle between character sets. When adding up the number of times the user needs to press these toggle keys it is found to closely equate to a passphrase which has more characters than a password. Table 5-1 indicates this using an example of a typical password and a passphrase. In addition, the keyboard typing actions required to generate special characters may disrupt the typing flow of a user, leading to a reduction in typing speed.

Table 5-1: Number of Keys Pressed

Example	Typing Actions on Keyboard	Number of Keys Pressed
P@ssw0rd	Caps on, p, caps off, shift, 2 (@), s, s, w, 0, r, d	11
thisismy passphrase	t, h, i, s, i, s, m, y, p, a, s, s, p, h, r, a, s, e	18

A second tier of authentication reduces the time taken to type, as no user–keyboard interaction is required for the second tier. Therefore, usability is increased, as the risk of errors does not apply to the second tier (keystroke dynamics algorithm). However, if a user tries to avoid errors by typing a password/passphrase slower than usual this may create an authentication error if the keystroke dynamics algorithm is too stringent. Therefore, the system development team should take this possibility into consideration when setting up the keystroke dynamics threshold and updating the information security policy to ensure that this is documented accordingly. The next section discusses the impact of routine tasks and non-routine tasks on typing.

5.8.3 Impact of Routine Tasks and Non-routine Tasks on Typing

Capture errors occur when a routine task is taken over by a non-routine activity (Kasiani & Yusuf, 2019; Parsons et al., 2010). However, a more correct statement would be that the risk of errors increases when non-routine activities are carried out rather than routine activities. This is most likely because the patterns of a routine task are practised over time, resulting in increased typing accuracy or reduced execution time, and in some cases both.

Parsons et al. (2010) explain a different type of error where a user subconsciously performs an action because it looks similar, but not the same, as a routinely performed action. The action is performed quickly by the user because he/she believes it is a routine activity that he/she is familiar with. This type of error is common when a user decides to speed up the completion time of executing a task. One example of such an error is when a user constantly clicks the next button without reading the instructions when installing an application.

It should be mentioned that typing behaviour may change due to a lack of short-term exposure to a familiar keyboard layout (Giot et al., 2011). Then the number of character sets included in the user's password/passphrase becomes an important factor in order to avoid false positive results from occurring. The following scenario is used as an example to explain this statement. A user usually accesses a system on a daily basis using a desktop. Suddenly he has to travel abroad for a month and uses his cellphone or tablet to login to the system throughout the trip. On his return, owing to his lack of exposure to desktop computers, the user's desktop keyboard interaction changes. Therefore, the risk of false positives increases when the user logs on with a desktop computer after a period of non-exposure.

Since the keyboard layout for lowercase letters is similar on desktops and mobile phones, there is less risk of false positives as the user's interaction will most likely not reduce as much as the interaction when other character sets are required. Therefore, a password including multiple character sets may require a user to switch between multiple keyboard layouts when using multiple devices to login to a system such as a desktop, tablet or cellphone. Accordingly, a user using one-character set of text-based authentication such as lowercase letters for inserting a passphrase is less likely to result in false positives than when using passwords including multiple character sets. This is because the keyboard layout for lowercase letters is similar for desktops, tablets and cellphones.

The above discussion relates to the following categories of the Keystroke-level model:

- **Number of keys pressed** – Time increases due to new task as opposed to a routine task.
- **Mental preparation required** – Increases due to new task.
- **Hands to keyboard and mouse** – Increases due to new task as a different position may be required.

The next section discusses the influence of language deviations on typing speed and errors.

5.8.4 Language Deviations: Punctuation and Grammar

It is difficult to understand the logical reason/s for excluding spaces in passwords and passphrases. Ritter and Sue (2007) merely explain that spaces should be avoided in passwords/passphrases to avoid errors, providing no further explanation as to why they should be avoided. In fact, allowing spaces to be used in passwords/passphrases would provide the user with an additional character, thus increasing the number of possible characters that could be included in a password/passphrase. This would mean that an increased number of guesses would be required to identify the password/passphrase as a larger range of characters is at the user's disposal.

Another possible reason why spaces are not allowed in passwords/passphrases is to prevent users from using a phrase as a password (Keith, Shao, & Steinbart, 2007) and this may render the passwords/passphrases more susceptible to a dictionary attack. It is advisable that the use of spaces be excluded from passwords and passphrases as the security threats associated to hacking may outweigh the security benefits of providing the user with the option of using an additional character.

Shay et al. (2016) found that it is not common practice to add spaces in passphrases. Keith et al. (2007) found that passphrases had a higher level of typing errors than passwords. When passphrases are used, users tend not to separate words with a space or simply use underscore (“_”) to separate words. Since the user might be conditioned to typing the phrase with spaces, this change in typing pattern may create typing errors if the user completely ignores the spaces when typing the passphrase. However, using an underscore as a substitute key for the spacebar breaks the rules of a passphrase being of one character set. Therefore, the substitute key for the keyboard should be from the same character set as the passphrase (i.e. a lowercase letter).

The substitute key may also reduce the risk of errors as now a user's typing pattern in a specific phrase does not change as much as if they were to completely ignore one or more keys (i.e. the spacebar between words in a phrase). In order to ensure a subtle change in typing patterns, a substitute key for the spacebar should be a key that is close to the spacebar. For example, the “x” key on the keyboard can be used as a substitute for the spacebar as it is close to the spacebar on a conventional keyboard layout. Therefore, typing patterns are less affected. In addition, “x” is a letter that does not

usually appear in many commonly used words. Therefore, a user would find it easier to read the passphrase typed that replaced the spacebar with an “x”. Nonetheless, it should be noted that errors may still occur as the typing pattern has slightly changed (i.e. “x” replaced with a spacebar). However, the risk of errors is reduced if the user uses the “x” key as a substitute for the spacebar as opposed to any of the other keys on the keyboard.

In addition to reducing the risk of errors, replacing the spacebar key with another character is recommended from a security perspective. It was found in Chapter 2 that the more characters that are added the more passphrases are strengthened, while still allowing the user to type the same number of keys as they would have if the user had typed the phrase with spaces. When applying the Chunking theory to this suggestion, a maximum of one chunk is required for the user to remember to replace all spaces with the “x” character. The same number of chunks (one chunk) is required if the user needs to remember that the password/passphrase needs to be inserted without spaces.

Some users may include acronyms or shorter versions of a word in their password/passphrase. If the findings above are applied to this approach to password/passphrase creation, a positive or negative result may be obtained based on user exposure. Although shortening words may save time, which is positive for usability, errors may occur as typing patterns are different from the conventional (plain language) typing of the word. That being said, if the user has a high level of exposure to typing the shortened version of the word (i.e. they use the shortened word often), then the risk of error is reduced. Bošnjak, Sreš, and Brumen (2018) and Mahapatra and Magesh (2015) suggest that common acronyms should be avoided in passwords/passphrases as the majority of them are included in dictionary attacks. Bošnjak, Sreš, and Brumen (2018) and Saevanee, Clarke, and Furnell (2011) add that common SMS language is also included in dictionary attacks. They also explain that user SMS language can be monitored on public forums, blogs and social networks. These words can then be added to the dictionary attack list.

From a usability and security perspective, it is also encouraged to include words from other languages in a password/passphrase. However, Bošnjak, Sreš, and Brumen (2018) and Clark and Arakia (2011) warn that dictionary attacks do not only include English dictionary words but also common acronyms and SMS language words (e.g. “gr8” and “luv”). Therefore, multiple languages can be mixed with different language acronyms. However, caution is advised as this is only encouraged if the user has

exposure to such words. If exposure is low, then the risk of typing errors occurring would be higher.

The discussion above affects two areas of the Keystroke-level model:

- **Mental preparation required** – The user must understand that the spacebar needs to be avoided and a substitute key is required. The user also needs to decide what substitute key to use. It was recommended that “x” be used as the substitute key.
- **Hands to keyboard and mouse** – Hand position on the keyboard may change based on the user’s decision of the substitute key to replace the spacebar.

The next section discusses the relationship between typing effort and risk of errors.

5.8.5 Typing Effort

As previously discussed, the more keys a user has to press on the keyboard, the greater the risk of an error occurring. This risk also increases the unfamiliar keystroke patterns that need to be typed by the user. When the user makes an error, additional time is spent on redoing the process by correcting the error; in this case, retyping the password/passphrase. This negatively affects usability as the user now spends more than twice the amount of time to login to the system.

Keith et al. (2009) and Renaud (2019) explain that people prefer to reduce effort as much as possible when using a system. Since user authentication processes are perceived by most users as a nuisance (Botha, Furnell, & Clarke, 2009), as they are regarded as a non-tangible benefit by the user, this process should be as effortless as possible.

Passwords may require more effort to type than passphrases due to the break in pattern when switching between different character sets. In addition, the user has to press additional keys when switching character sets, which are not part of the password characters but are required to display the respective character. For example; in order for a user to include the “@” character in their password, the user has to select the “shift” key and then the number “2” key on a typical keyboard. Although only one special character is included in the password, the user is required to interact with two keys on the keyboard to include this special character in their password (“shift” and “2”). Not only does this increase the risk of error but it also requires the user to put in twice the effort (the two keys pressed) for a single reward (one character included in the password).

These findings relate to the number of keys pressed category of the Keystroke-level model, as switching between certain character sets requires twice as many keys to be pressed by the user. The next section discusses the effect on usability of hiding and showing text typed by the user.

5.8.6 Hidden or Unhidden Text

This section discusses the graphical user interfaces that does and does not display the password/passphrase while a user is typing the password/passphrase and the impact this has on usability. The intention of this system function is to prevent other people, physically close to the user, from seeing the password/passphrase on screen (De Luca, Von Zezschwitz, Pichler, & Hussmann, 2013; Varalakshmi, 2015). This could be a person or even a security camera focused on a user's screen. In some locations, user privacy is not guaranteed in terms of others seeing their screen (Lin & Wu, 2011; Yildirim & Mackie, 2019). For this reason, the user should have the option to toggle between displaying and not displaying what is being typed. The drawback of having hidden text when typing in a password/passphrase is that a user cannot identify any mistyped characters then or afterwards. In some cases it is possible to flash (unhide and then hide again) a key immediately pressed by the user for one second before hiding it (De Luca et al., 2013; Varalakshmi, 2015).

When text is hidden while typing, passphrases may be less prone to errors than conventional passwords. When passwords contain special characters and combinations of uppercase and lowercase characters, they may become difficult for the user to track. This is because an initiation key is required to switch between upper case characters, numbers and special characters. In addition, initiation keys usually differ in different devices such as cellphones and tablets (Liu, Dillon, & Zhang, 2017). Therefore, if the password is not flashed, there is a greater risk of the user mistyping a character and failing to see that it needs to be rectified, as opposed to a passphrase which does not require any initiation keys.

The default option should be set on hidden text to avoid the user forgetting to activate the hidden text functionality. There is a risk that if the user does not realise that the hidden text functionality is on while typing, it may result in an incorrect password/passphrase being typed as the screen will not display the keys pressed by the user. However, it is more damaging for the users if they think the hidden text functionality is on when it is not, and another party manages to glance at the entire password/passphrase or even part of the password/passphrase. It is therefore

recommended that hidden text toggling be provided, however it should be defaulted to on (i.e. hidden text). In addition, from a usability perspective, it is suggested that this toggle option be located close to the password/passphrase field so that the user can find it and use it easily if required.

In terms of the Keystroke-level model, two categories are influenced by the hidden text discussion above:

- **Hands to keyboard and mouse** – Systems can provide an option for the user to toggle between showing and hiding the text typed in the password/passphrase field. Switching between keyboard and mouse is required if the user opts to verify whether the password/passphrase typed was inserted correctly in the password/passphrase field. This is an additional step which might be seen as negatively affecting usability because of the additional time required to switch between keyboard and mouse, merely to verify that the password/passphrase typed is correct. However, uncertain users can save time by avoiding the risk of having to login again if the password/passphrase was mistyped. This can be done by merely validating that the password/passphrase typed was correct before selecting the “login” button to indicate to the system that the password/passphrase insertion process has been completed. Recall that this is an optional process provided for users who are unsure as to whether the known password/passphrase was typed correctly.
- **Response time of the system** – If a time lag occurs when toggling between showing and hiding the password/passphrase usability may be affected.

The next section discusses how keyboard layout and visibility can influence typing usability.

5.8.7 Keyboard Exposure

It is important to ensure the user has a clear view of the keyboard when typing at all times. A user should be able to toggle quickly between sight-of-screen and sight-of-keyboard. Kim, Yib, and Yoonc (2019) and Weiss, Hollan, and Borchers (2010) explain that in addition to visually having sight of the keys, a user's lack of touch can impair typing abilities which may result in typing errors. Most typing errors occur because of users switching between devices (Kim, Aulck, Thamsuwan, Bartha, & Johnson, 2014; Park & Han, 2010). Accordingly, device switching errors occur between desktop computers and laptops, which have analogue keyboards, and touch screen cellphones and tablets,

which have a digital keyboard (Findlater, Froehlich, Fattal, Wobbrock, & Dastyar, 2013; Yazdi, Negahban, Cavuoto, & Megahed, 2019).

It should also be noted that from a user device exposure point of view, risk of errors is reduced based on the amount of prior exposure the user has had interacting with the keyboard layout of a specific device. For example, if User A does not interact with a desktop computer as much as User B, then User A's risk of error will be higher than User B for desktop computers. This assumes that the desktop computers have the same keyboard layout. Likewise, if User A interacts more with a cellphone than User B, then User A's risk of error on a cellphone is lower than User B's. This may suggest that User A's usability will be higher if he/she can login with their cellphone as opposed to a desktop computer. However, this will only be true on a system that is typing-heavy as these findings are specific to keyboard interaction. Usability may reduce if cursor movement is required. Research by Findlater et al. (2013) and Yazdi et al. (2019) aligns to this discussion as they found that age influences touch screen interaction. Since older users grew up in an era when only analogue keyboards were available, younger users have a lower level of errors on touchscreen keyboards than older users (Findlater et al., 2013; Yazdi et al., 2019). This finding remains true even though overall exposure to keyboards is higher among older users than younger users who have been exposed to touchscreen keyboards. However, Bi, Ouyang, and Zhai (2014), Findlater and Wobbrock (2012) and Vertanen, Memmi, Emge, Reyat, and Kristensson (2015) argue that users typing on a touchscreen cellphone have a lower risk of errors as only two fingers (two thumbs) are usually used to interact with the digital keyboard. These authors further explain that subconscious concentration is improved as a user only has to focus on manoeuvring two fingers as opposed to the ten fingers used on a typical desktop keyboard.

From a keystroke dynamics perspective, the total time of all key-ups and key-downs may be higher on an analogue keyboard than a digital keyboard. However, response time on a digital keyboard will be slower than an analogue keyboard as the former requires more processing power (Liu, 2013; Priva, 2010; Yazdi et al., 2019). This may not be an issue unless the computer is running multiple applications that demand a lot of processing power thus affecting the response time of the digital keyboard. Therefore, if a digital keyboard is running a keystroke dynamics algorithm that is monitoring a number of user-keyboard interactions simultaneously, the responsiveness of the digital keyboard may be affected. This results in a reduction in usability. There may be a delay between the key pressed and the character being displayed on screen, and the readings of the user-

keyboard interaction being monitored by the keystroke dynamics algorithm may be inconsistent. It is important for the system development team to understand the processing power of devices before implementing the correct keystroke dynamics algorithm. Keystroke dynamics are flexible enough to add additional user–keyboard interaction measures at a later stage, usually without affecting the current measures. Therefore, as computers become faster over time, the keystroke dynamics algorithm can evolve with the increases in computer performance.

In terms of the Keystroke-level model, the category influenced by the discussion above is the “hands to keyboard and mouse” measurement. This process includes the user’s touch on the keyboard in relation to errors typed.

5.8.7.1 Lack of Muscle Memory Opportunity

Muscle memory was discussed in Chapter 4, Section 4.3.1.4, Muscle Memory. Muscle memory is achieved when information is embedded in a person so well that the person requires little cognitive effort to recall that information from memory (Skovgaard et al., 2018). Muscle memory can be used for typing a password/passphrase. Muscle memory is utilised for typing when a user subconsciously remembers the typing pattern when inserting their password/passphrase. A user would rely on the typing pattern (and/or keyboard visibility) as much as or even more than the actual password/passphrase characters. The issue is that passwords are required to be changed regularly, which prevents the user from reaching a state to take advantage of muscle memory. The proposed solution does not require a passphrase to be changed as frequently and therefore, users have a better opportunity to take advantage of muscle memory. This section concludes the discussion on the typing aspects affecting usability. The next section summarises the findings discussed above in the form of a model.

5.9 Passphrase and Keystroke Dynamics Typing Usability Model

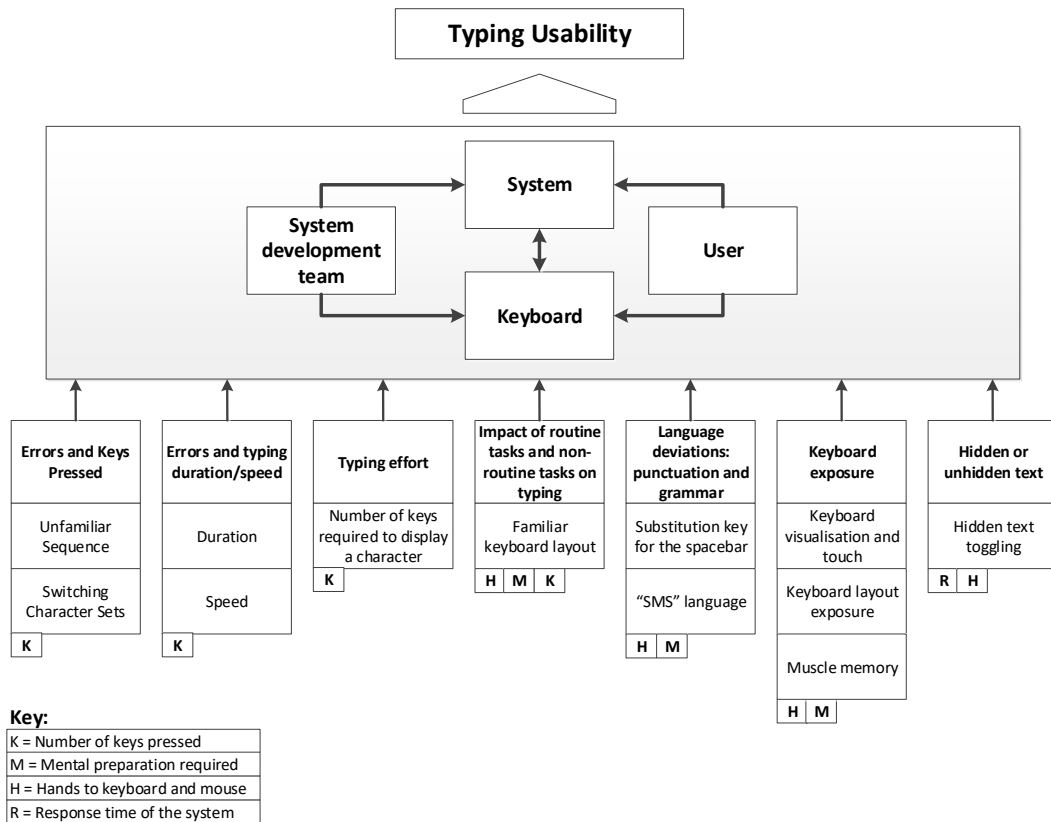


Figure 5-5: Typing Usability Considerations for Proposed Two-tier Solution

The model in Figure 5-5 is derived from the above discussion. The model indicates that seven components influence typing usability using seven arrows that point upwards into the grey rectangle. Within this triangular shaped border, the diagram shows that addressing the seven components affecting typing usability requires synergy between the system, the user (user and system development team) and the keyboard/s used by the user.

Owing to the breadth of the seven components, subcomponents are provided below these seven main components. The intention of these subcomponents is to depict exactly which parts of the main components influence typing usability for the proposed solution. The model consequently provides the components that must be considered to address typing usability issues for a two-tier solution, incorporating passphrases and keystroke dynamics.

The Keystroke-level model assessment items are also depicted on the model by the letters; "K", "M", "H" and "R" (key also provided in the model). The Keystroke-level model

provides certain assessment aspects that need to be accounted for to measure usability. The Keystroke-level model letters indicate which Keystroke-level assessment is addressed by the main components (seven components) of typing usability. The purpose of including the Keystroke-level model assessment items in the typing usability model is to ensure that the four components (“K”, “M”, “H” and “R”) have been considered. As illustrated in Figure 5-5, different Keystroke-level model assessment items are associated to different typing usability components.

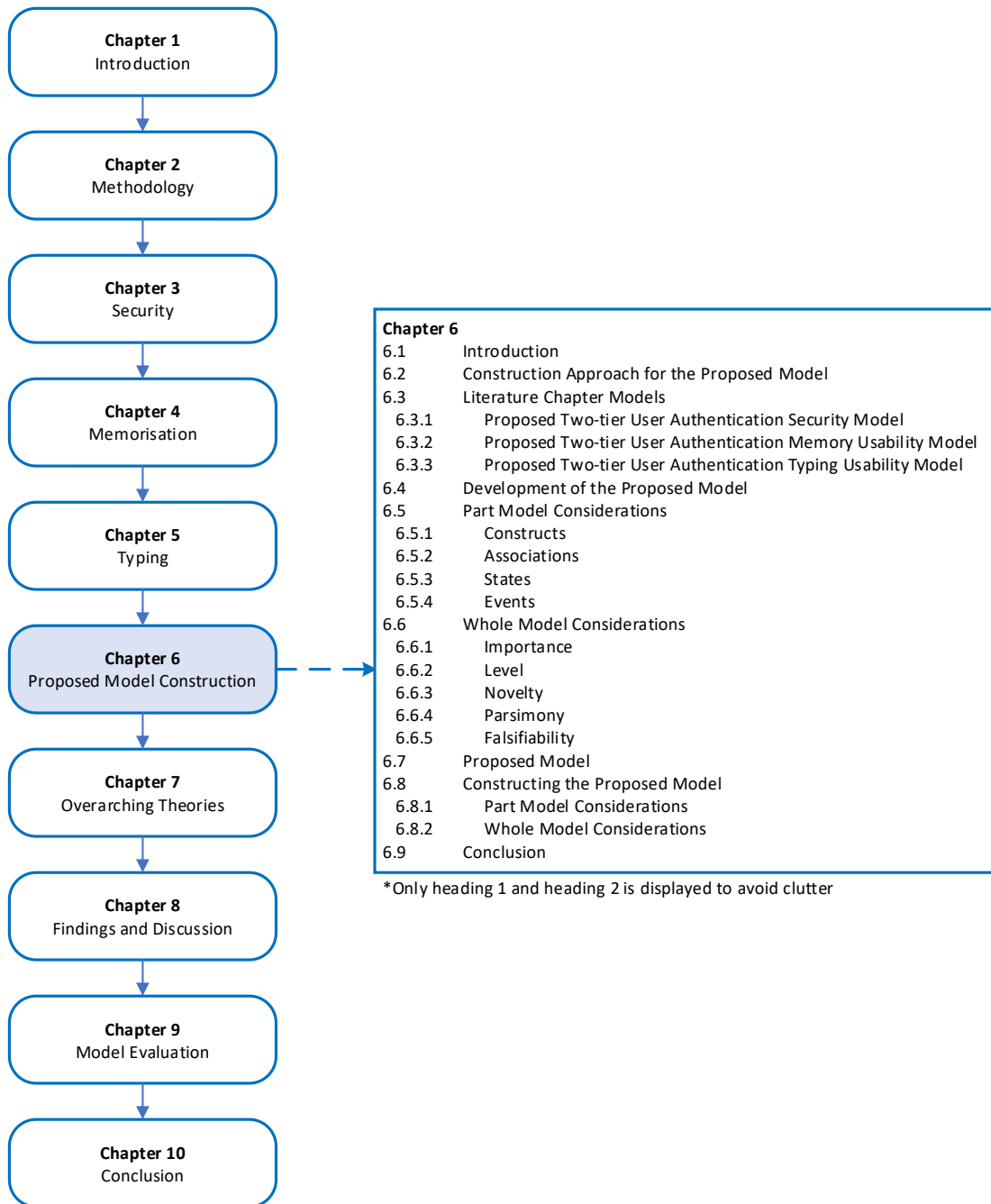
It can also be seen that some Keystroke-level model assessment items repeat across the typing usability components. This emphasises that some typing usability components require multiple solutions to address a specific Keystroke-level model assessment item.

The assessment aspects that have no impact on typing were filtered out as the scope of this research does not require the assessment of the other aspects, as discussed in the Keystroke-level model scope section above. It was noted that the filtered-out assessments of the Keystroke-level model need to be considered when comparing the proposed solution with non-text-based user authentication processes such as biometrics. The last section of this chapter provides a conclusion of this chapter that focused on typing usability.

5.10 Conclusion

This chapter focused on answering the typing usability aspect of the research question. The Keystroke-level model was used as a guide to identify components affecting typing usability aligned to text-based user authentication. It was found that three main components can influence typing usability in a number of different ways: the user (1), the system (2) and the keyboard (3). Since these components overlap, an increase in typing usability demands a collaborative effort from all three components. From a user perspective, exposure to typing certain keys and sequences influences typing usability. In terms of the system, any activities that jeopardise system response time should be avoided as this negatively affects usability. From a keyboard point of view, the keyboard layout and level of user exposure to these layouts seem to have a significant impact on typing usability. Considering all three components will ensure an increase in typing usability for text-based user authentication. Now that the last literature chapter has been presented, the next chapter focuses on the construction of the proposed model.

Chapter 6 – PROPOSED MODEL CONSTRUCTION



6.1 Introduction

This chapter introduces the proposed model for this research study. The proposed model is derived from the findings uncovered in the literature chapters and is a draft of the solution that seeks to address the research problem of the study.

The approach used to construct the proposed model is firstly discussed. The next section provides the models derived from the findings in the literature chapters. This is followed by a discussion on how the proposed model is developed. The next section presents the proposed model which is followed by a section explaining what was considered when constructing the model. The last section provides a conclusion for this chapter.

It should be noted that the evolution of the model (the draft and updates to the research model) is referred to as the proposed model until the proposed model has been finalised. i.e. Drafted from literature, tested with the login assessment experiment, evaluated through an expert review and then finalised.

6.2 Construction Approach for the Proposed Model

Robinson, Arbez, Birta, Tolk, and Wagner (2015) explain that most models can be constructed in a number of ways and still communicate the same solution. They emphasise that it is important to identify the audience and ensure that the intention is accurately depicted in the model. The model in this study is intended for use by system development teams for development guidance and consideration (the information security policy should be updated to accommodate this solution), as well as by researchers aiming to assess and expand on the model's findings. The model can also be used to inform information security policies on user authentication.

The construction of the proposed model was assisted by previous empirical literature and findings. This section is intended to explain the approach taken in constructing the proposed model. The proposed solution directed research for this study to focus on assessing whether the proposed solution is able to address the identified research problem. It should be borne in mind that the research problem identified that current user authentication solutions either focus on security and compromise usability or addresses usability but simultaneously compromise security. This study aims to identify whether the proposed solution can address this problem, i.e. can the two-tier user authentication solution involving passphrases and keystroke dynamics offer a high level of security and usability?

The literature chapters focused on understanding and assessing the effectiveness of the proposed solution in terms of the two issues that current user authentication solutions cannot simultaneously address: security and usability. The first literature chapter, Chapter 3, focused on research and findings relating to the security aspect of the proposed solution, while Chapters 4 and 5 both focused on the usability aspect to assess the proposed solution. Keith et al. (2007) found that with text-based user authentication, two areas need to be addressed; effect on user memory (1) and effect on user's typing (2). Subsequently, a model was constructed at the end of each chapter (Chapters 3, 4 and 5) to summarise the important findings uncovered.

The construction of the proposed solution involved merging the three models from the literature chapters into one, as consolidating all three models addresses all the aspects that had to be assessed from a secondary data perspective. In this chapter the construction of the proposed model by merging the three models created in the literature chapters is described. It is important that the following was considered when merging the models (Jabareen, 2009):

- Remove redundancy in content.
- Correct level of detail taken from all three models.
 - The proposed model must not be overwhelmed with information.
 - Too little detail would make the model too generic or not allow any benefit to be realised.
- The areas of security and usability or if both are influenced or affected, had to be clearly depicted and then illustrated on the merged model.

Figure 6-1 graphically illustrates the process involved in constructing the proposed model.

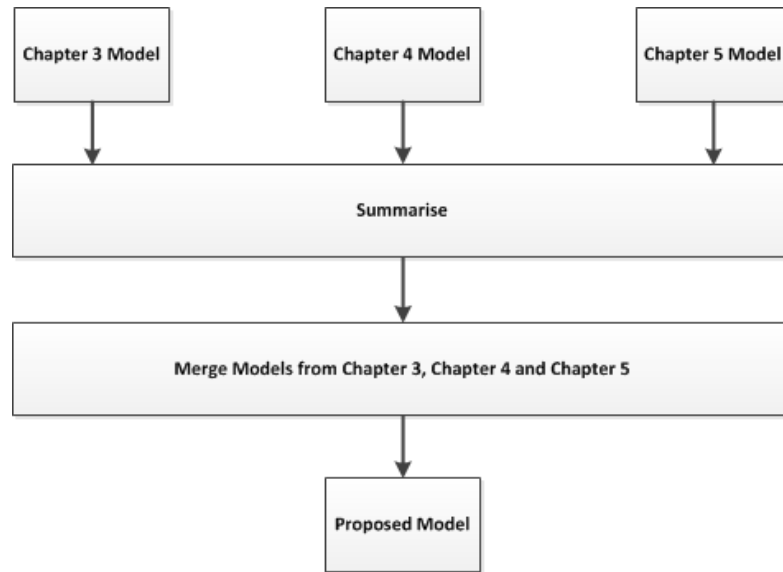


Figure 6-1: Construction Approach of Proposed Model

As Figure 6-1 shows, the models constructed in Chapters 3, 4 and 5 were merged to form the proposed model. However, prior to merging a summary step was required to ensure that the models worked together correctly to address the research problem. This included the efforts suggested by Jabareen (2009) that were mentioned above. This summary process involved ensuring that all the above considerations were addressed. Once the proposed model was developed from the literature, it had to be evaluated and validated.

Two main evaluations and validations were conducted:

1. Login assessment.
 - Findings from the login assessment resulted in updates to the proposed model.
2. Review of the proposed model by experts in security and usability.
 - Feedback received from experts regarding the review of the proposed model may require the model to be updated.

The next section includes the models derived from each of the three literature chapters as this provides the foundation for the construction of the proposed model.

6.3 Literature Chapter Models

This section recaps on the models constructed in the literature chapters, as these were used to construct the proposed model. The first section recaps on the model constructed to focus on the security aspect of this research study.

6.3.1 Proposed Two-tier User Authentication Security Model

The first model was created in the first literature chapter (Chapter 3) to determine the security implementations of the proposed solution. This model is illustrated in Figure 6-2: proposed two-tier user authentication security model.

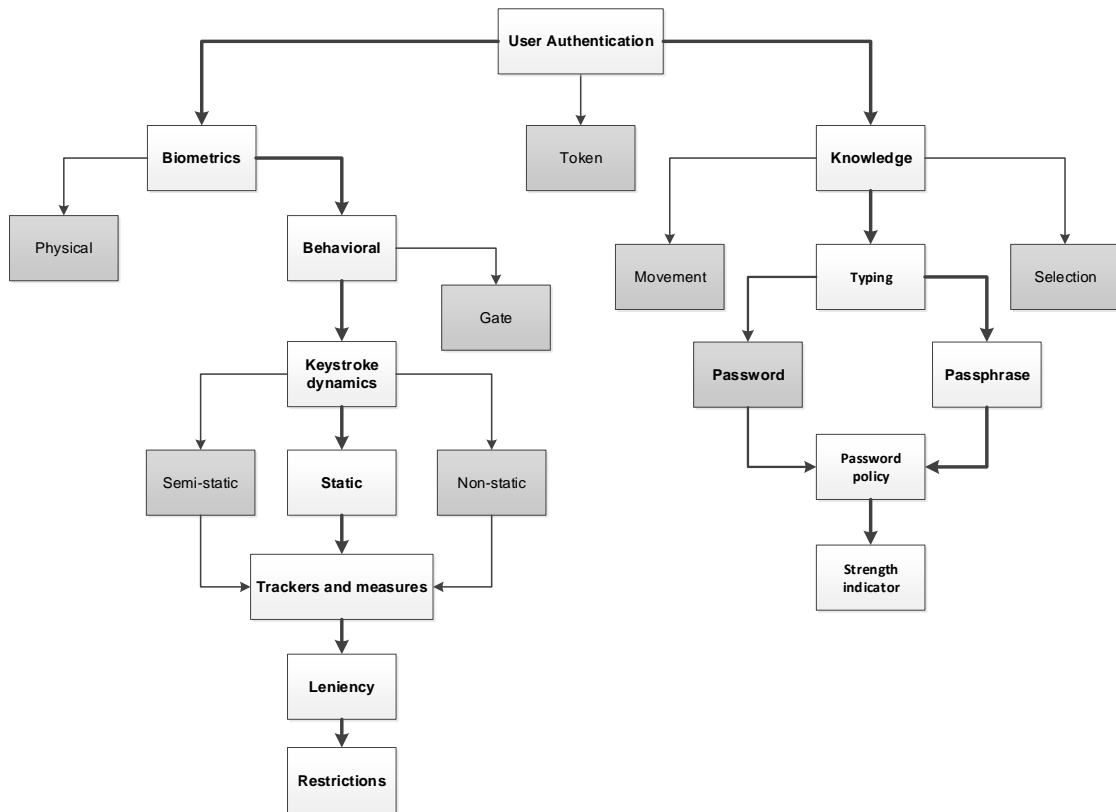


Figure 6-2: Proposed Two-tier User Authentication Security Model

Figure 6-2 shows that any form of user authentication can be classified into one of the three main categories – biometrics, tokens or knowledge. Keystroke dynamics forms part of behavioural biometrics, which in turn is a subcategory of biometric user authentication. Similarly, passphrases form part of the typing subcategory of knowledge. With regard to passphrases, two elements can influence security – password policies and strength indicators. In terms of keystroke dynamics three elements can influence security – trackers and measures, leniency and restrictions. All three of these elements can be monitored by the keystroke dynamics algorithm, once-off (static), ongoing (non-static) and a combination of both (semi-static). The grey blocks indicate areas that are beyond the scope of this research study. The next section explains the memory model to address the research problem of the study.

6.3.2 Proposed Two-tier User Authentication Memory Usability Model

The proposed two-tier user authentication memory usability model was intended to provide the aspects that may be considered to assist users to memorise passphrases better.

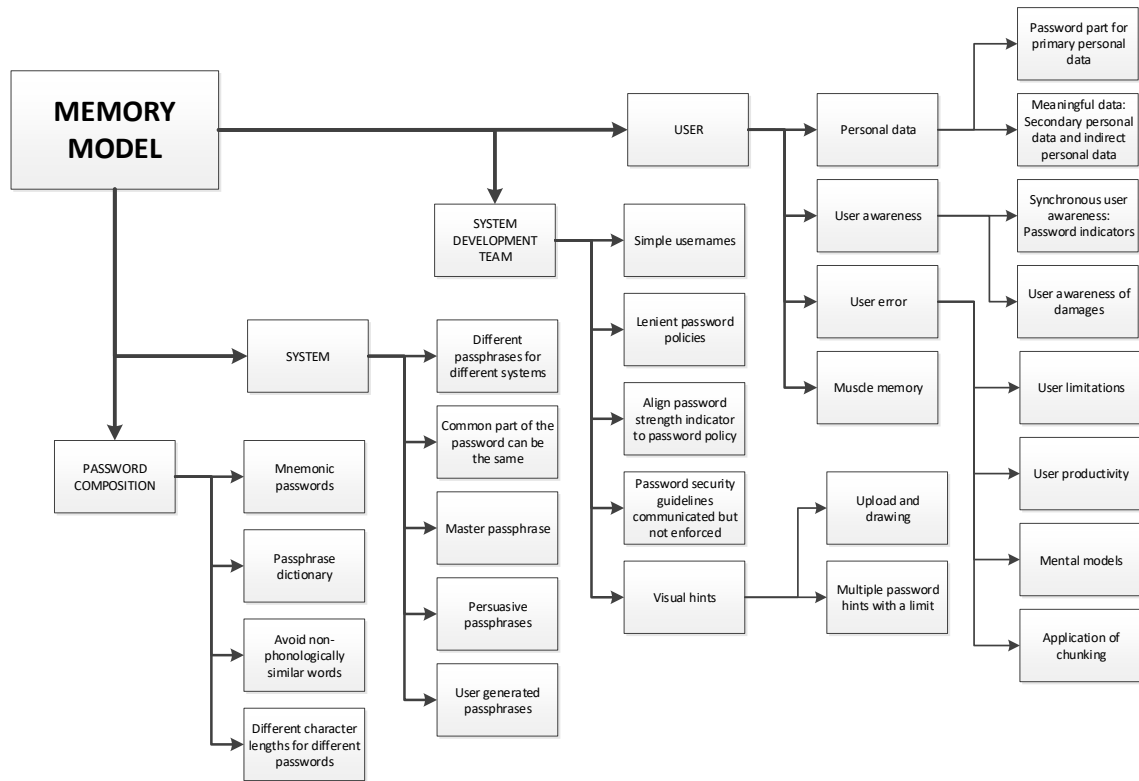


Figure 6-3: Proposed Two-tier User Authentication Memory Usability Model

The proposed two-tier user authentication memory usability model in Figure 6-3 aims to mitigate the negative security behaviours created by users in an attempt to avoid forgetting their passphrase. All these considerations can be mapped back to one of four memory elements that describe how the considerations can be influenced by memory. The following memory elements may influence the memorisation of passphrases: the user, the system development team, the system and password/passphrase composition. The next section discusses the typing model to address the research problem.

6.3.3 Proposed Two-tier User Authentication Typing Usability Model

Typing is the second part of usability that needs to be addressed to ensure the successful implementation of the proposed solution. The proposed two-tier user authentication typing usability model is depicted in Figure 6-4.

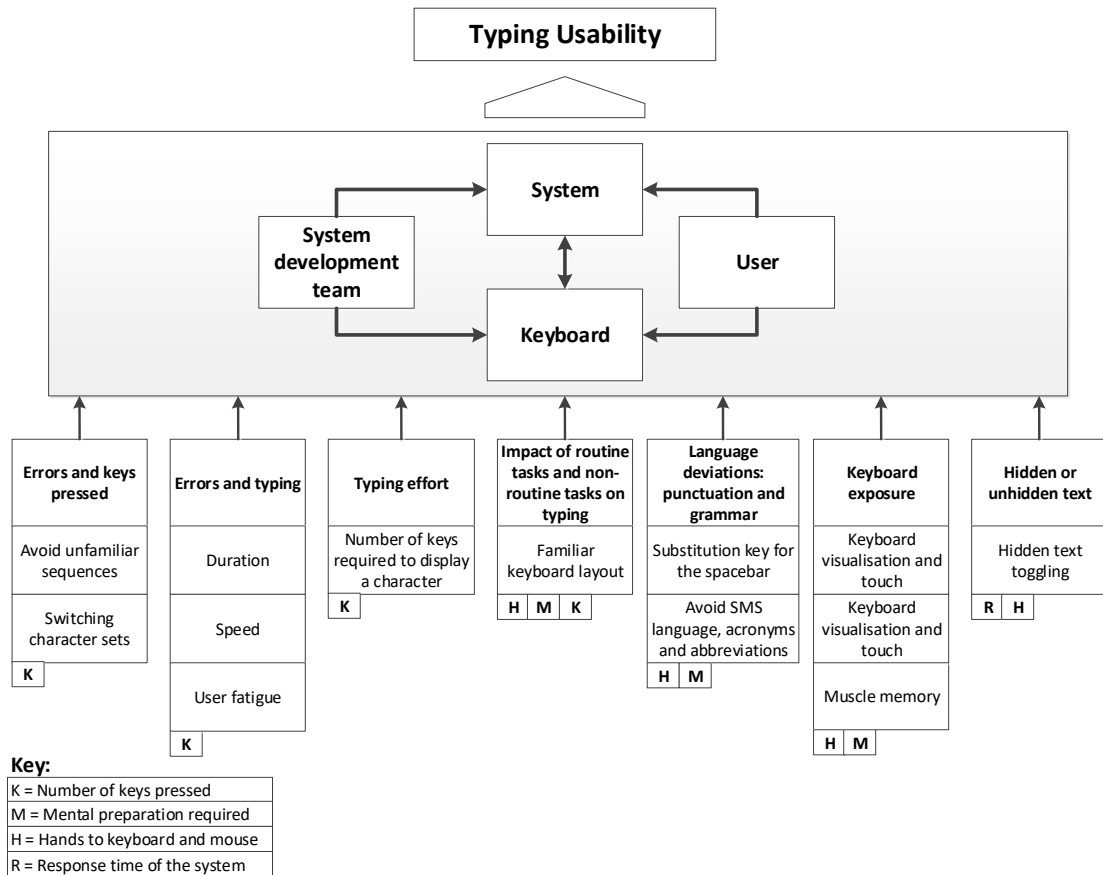


Figure 6-4: Proposed Two-tier User Authentication Typing Usability Model

The key at the bottom of Figure 6-4 was used to ensure that all aspects of the Keystroke-level model were accounted for to ensure that the typing elements addressed the entire typing process. The grey block shows significant associations between the four elements of typing usability for this study: system, user, keyboard and system development team. The blocks below the grey box indicate the considerations that influence typing usability.

Now that on all the models derived from the literature chapters have been reviewed, the following section discusses the way in which the proposed model was constructed.

6.4 Development of the Proposed Model

The model proposed for this study was constructed by merging the three models developed at the end of each literature chapter, namely the:

- Security model
- Memory usability model
- Typing usability model

It was not possible to merge these three models by simply joining them together. Guidelines had to be considered to ensure that the model merger process produced a high quality model. Weber (2012) provides guidelines for evaluating theories in the information systems discipline. These guidelines were deemed to be best suited for constructing the proposed model. Despite the use of these guidelines, the proposed model was still reviewed by experts with sufficient knowledge of the system security and system usability domain. Once the proposed model had been constructed, the findings of the login assessment were used to update the proposed model before it was presented to experts for review.

Weber (2012) explains that the evaluation firstly needs to be conducted on the model parts. Once the parts are correct, the consolidated model should be evaluated. This logic ensures that only high-quality models are consolidated because the part models do not affect the quality of the combined model. Figure 6-5 graphically depicts the factors that should be considered when evaluating the model parts, as well as the combined model. This diagram was used to guide the construction of the proposed model.

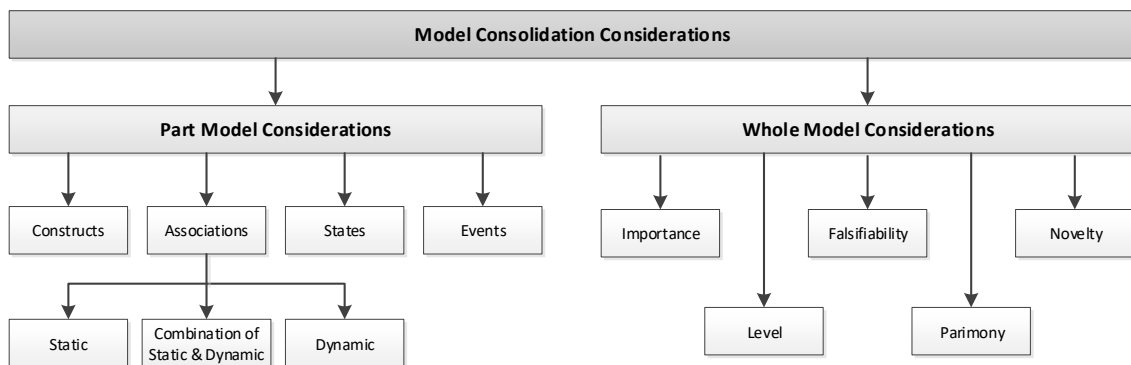


Figure 6-5: Model Assessment Considerations – Adapted from: (Weber, 2012)

As illustrated in Figure 6-5, four considerations (constructs, associations, states and events) were taken into account when assessing the model parts and five considerations (importance, level, falsifiability, parimony and novelty) were accounted for during the consolidation process. The four considerations for the part models were firstly discussed, followed by an explanation of the five considerations for the whole models. The considerations that were applied to the development of the three models in the literature chapters were then reviewed. Finally, once all three part models had been assessed, the considerations for merging the three part models were discussed.

6.5 Part Model Considerations

In this section, the four considerations for part models were discussed individually and then applied to the three-part models constructed in each literature chapter. The four considerations for assessing part models were as follows (Weber, 2012):

- Constructs
- Associations
- States
- Events

The first part model consideration discussed is constructs.

6.5.1 Constructs

Weber (2012) explains that constructs are attributes that provide more context to a generalised class of things. For example, for the class, “person”, the attributes could be gender, race and eye colour. Weber (2012) used the popular Technology Acceptance Model (TAM) as an example to clarify the concept of constructs. Examples of constructs include the components of the TAM model such as perceived usefulness and perceived ease of use. Weber (2012) further explains that if any attribute of a class is unclear, then the class itself may be unclear. Backus, Ferriere, and Zinab (2015) emphasise the importance of removing ambiguity in models. This can be done by clearly indicating the attributes of all classes in a model. However, removing ambiguity can significantly expand the size of the model and provide unnecessary detail, which ends up drowning the important aspects of the model. Any attributes depicted in the model must be used sparingly to ensure the model's conciseness.

6.5.2 Associations

Three types of association exist, each of which can be depicted by different levels of detail. Weber (2012) explains that although more detail is better, at times it might not be possible or necessary to depict a higher level of detail with regard to associations. Unlike constructs, the level of detail for associations is usually determined by the amount of information the person constructing the model has, concerning the construct links (Jilani, Usman, & Nadeem, 2011). For example, in forex trading, an association could be created between currency price and news even though the way in which they influence each other may not be known. The difference in levels of detail is explained below.

- **Static associations** – Constructs are associated but do not influence each other.

- **Low level of detail** – Association between constructs are indicated but does not show how they influence each other.
- **Moderate level of detail** – One construct negatively or positively relates to the other construct.
- **High level of detail** – The amount that the one construct has influences the other construct.
- **Dynamic associations** – A change in one construct will result in a change in its associative construct.
 - **Lowest level of detail** – Relationship between constructs is shown by groupings but with no links between constructs. This may be because of uncertainty.
 - **Low level of detail** – An association is shown but no direction is provided, i.e. flows or dependencies.
 - **Moderate level of detail** – Association and direction are provided.
 - **High level of detail** – The amount of influence one construct has over the other is depicted, as well as whether it is a positive or a negative influence.
- **Combination of static and dynamic associations.**

It is common for different levels of detail in terms of associations between constructs to appear in a model. High levels of detail should be the aim but it might not be necessary to depict this. The researcher constructing the model needs to determine whether any value is added by depicting high levels of detail on the model. The next section discusses states.

6.5.3 States

The state of each construct must be defined. Weber (2012) explain that states are the disclosure of all known values of a construct. The various known combinations of values must also be disclosed. This practice works well for defining the scope in which this model can be applied, for example the TAM will not have a state where users accept a technology. This state would fall outside the boundary of the model.

6.5.4 Events

An event is an occurrence that is realised before a state or after a state (Weber, 2012). It can also be a combination of both (Jilani et al., 2011). In other words, a state can be seen as a result of an event. Therefore, every event must be related to one or more

events. For example, consider a situation where Kyle's teacher shouts at him for not doing his homework. In this example, the events will be Kyle not completing his homework (event 1) and the teacher shouting at Kyle (event 2). The following states may be realised: Kyle gets upset (state 1), Kyle feels a sense of regret (state 2) or Kyle isn't worried by the teacher's reaction (state 3).

6.6 Whole Model Considerations

The previous section ensured that the model parts provided were of the correct quality. To ensure that this quality is maintained when merging the models, the following should be taken into account (Weber, 2012):

- Importance
- Level
- Novelty
- Parsimony
- Falsifiability

Each of the above points are discussed in more detail below. Importance is the first consideration pertaining to the whole model to be discussed.

6.6.1 Importance

The importance of the model must be realised in the model. Importance can be emphasised by ensuring that the model is able to provide the reader with sufficient understanding of the effect on the research problem (Hofstee, 2006; Poth, 2018). In terms of this study, the research problem relates to current user authentication approaches that result in an imbalance between security and usability. The model for this study addresses this problem through a two-tier user authentication approach involving passphrases and keystroke dynamics.

6.6.2 Level

Weber (2012) explains that models are presented by researchers on one of two levels: length or breadth. It is difficult for a model to cover both length and breadth and therefore the researcher needs to determine which is most appropriate to address the research problem. Figure 6-6 graphically differentiates the difference between length and breadth.

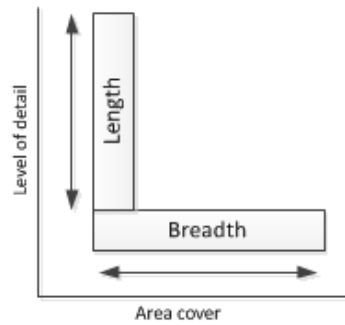


Figure 6-6: Research Length vs Research Breadth

In Figure 6-6, it can be seen that breadth requires a large amount of detail whereas length demands generalisation that covers multiple considerations. With regard to models that focus on breadth, predications are more detailed and accurate. Likewise, models that focus on length cover a larger research area but are not usually well defined.

6.6.3 Novelty

The model must have a certain level of uniqueness and significance (Hofstee, 2006; Poth, 2018). If not, its contribution to the body of research is minor or possibly unnecessary. Weber (2012) emphasises that novelty will result from the way other researchers view the value of the model. This also provides a good indication of the likelihood of the model being accepted in well-known publications. Citing of the model in reputable publications is important as it helps to communicate the model to other researchers. This will allow the model to be rigorously tested and/or enhanced where possible.

6.6.4 Parsimony

Weber (2012) explains that high quality models ensure that resources are utilised efficiently. Therefore, waste needs to be identified and removed where possible. In certain cases where waste cannot be avoided, reassurance should be provided that it was unavoidable, and the benefits gained outweighed the waste incurred. Weber (2012) adds that a model that makes effective use of resources makes it easier to explain and thus, easier to comprehend. Weber (2012) recommends that the model should include no more than seven constructs and seven associations in order to ensure that parsimony has been achieved; not that this suggestion by Weber (2012) was used as a guideline. i.e. the finalised model resulted in having nine constructs.

6.6.5 Falsifiability

Weber (2012) explains that it is hard to prove a model as a fact as it is difficult to assess all combinations of states, associations and events. Knight and Cross (2012) support Weber (2012), explaining that a model is the “best” answer to a problem or problems using the resources that the model claims to utilise. Weber (2012) continues, explaining that because of this models must ensure that predictions are accurate enough to align with the results when tested. This increases the likelihood of the model being accepted by researchers. Additionally, a model accepted by researchers in the identified areas encourages the model to be evaluated in other areas that were not originally stated. This allows the model to grow and evolve through future research by others.

Falsifiability concludes the discussion on whole models. Now that the considerations for part models and whole models have been discussed, the next section focuses on the results of applying the above-mentioned considerations.

6.7 Proposed Model

The proposed model was constructed by taking the above considerations for part models and whole models into account and subsequently guiding the merger of the three sub-models created in the literature chapters (Figures 6-2, 6-3 and 6-3). Figure 6-7 below provides the proposed model for this research study.

In Figure 6-7, the letters A and C in circles are not part of the model. C1 to C5 illustrates that there are five constructs in the model, while the letters A1 to A3 indicate the corresponding attributes for those constructs. The security puzzle piece and the usability puzzle piece indicate that balance is achieved between security and usability by the three constructs working together, namely, stakeholders, passphrases and keystroke dynamics. The attributes for each of these constructs are provided below the construct names. These are the main attributes of the constructs that have an influence on achieving balance between security and usability respectively.

Three main theories (Shannon Entropy theory, Chunking theory and Keystroke-level model) assisted in constructing the proposed model. The parts of the proposed model that these theories supported are as follows:

- **Shannon Entropy theory** – Support the security (C5), keystroke dynamics (C3) and passphrase (C2) aspect of the proposed model.

- **Chunking theory** – Provided assistance on the usability – memory (C4) part of the proposed model as well as the stakeholder portion (C1).
- **Keystroke-level model** – Provided support on the usability – typing (C4) aspect of the proposed model and the stakeholder part (C1).

These three theories played a vital role in the construction of the proposed model. These theories were also used extensively in the login assessment experiment (results from the experiment can be found in section 8.4, Login Assessment Results and Discussion).

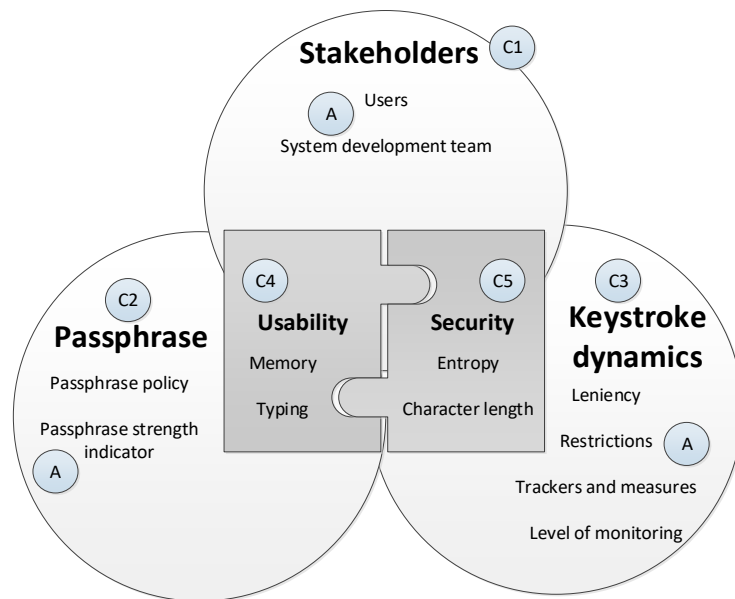


Figure 6-7: Proposed Model: Passphrases and Keystroke Dynamics

Now that an explanation has been provided on the proposed model, the next section explains how the proposed model was constructed by taking into account certain considerations.

6.8 Constructing the Proposed Model

The previous section introduced the proposed model. This section is intended to provide a breakdown of how the proposed model was constructed. It focuses on the considerations proposed by Weber (2012) that should be taken into account when merging models into one consolidated model. This section discusses the part model considerations before moving on to the whole model considerations.

6.8.1 Part Model Considerations

Figure 6-7 was constructed after taking into account the considerations of part models and whole models. The way in which the part model considerations were accounted for is discussed below.

- **Constructs** – Attributes that provide more context to a class of things.
 - The detailed constructs were excluded from Figure 6-7.
 - The high-level constructs were grouped and summarised where possible.
 - Common constructs across the three-part models were merged to remove redundancies.
- **Associations** – Static associations indicate the relationship between constructs, and dynamic associations indicate that a change in one construct influences another construct, either in one direction or in two directions.
 - It was found that the merger of the three sub-models required updating the associations by using both static associations and dynamic associations.
 - Static associations with a moderate level of detail were identified for the security aspect in Figure 6-7. The relationship between components was determined and a positive and negative effect was identified.
 - Dynamics associations with a low level of detail were identified for the usability aspect in Figure 6-7. Association is shown but no direction is provided.
- **States and events** – Events are an occurrence which results in a state. Although these are separate considerations, because of their close relation to each other, it is best to discuss them together.
 - Security
 - After a valid user enters a passphrase (event), the keystroke dynamics algorithm successfully authenticates the user (state).
 - After a valid user enters a passphrase (event), the keystroke dynamics algorithm failed to authenticate the user (state).
 - After a non-permitted user enters a passphrase (event), the keystroke dynamics algorithm successfully authenticates the user (state).

- After a non-permitted user enters a passphrase (event), the keystroke dynamics algorithm failed to authenticate the user (state).
- Usability (memory)
 - A user forgets their passphrase (state) and the system allows the user to reset their passphrase (event).
 - A user recalls their passphrase (state) and the system permits access to the user (event).
- Usability (typing)
 - A user correctly types their passphrase (state) and the system permits access into the system (event).
 - A user incorrectly types their passphrase (state) and the system denies access into the system (event).

The above provided all considerations for part models. The next section provides a discussion on the considerations for the model as a whole.

6.8.2 Whole Model Considerations

The following whole model considerations must be applied to ensure that the three sub-models created in each literature chapter were consolidated correctly. The following whole model considerations need to be taken into account: importance, level, novelty, parsimony and falsifiability. The first whole model consideration that is discussed is importance.

6.8.2.1 Importance

The importance of the model must be understood (Weber, 2012), as it relates closely to the research problem this model attempts to address. The proposed model illustrates factors that need to be considered to address security and usability issues. Usability has been separated into memorisation and typing.

6.8.2.2 Level

The correct level must be determined for the model (Weber, 2012). Figure 6-6 illustrates what is meant by length and breadth. Because this research study was structured around determining how the two-tier user authentication system could address security and usability concerns regarding user authentication, the research leaned more towards the length of the research area as opposed to the breadth. If, for example, this study focused on identifying an appropriate solution to address the research problem, then the

study would tend to focus more on the breadth of the research area. It is important to note that the specific details were summarised in the proposed model as this prevented the creation of a level of complexity that might create confusion and take attention away from the primary components of the proposed model.

6.8.2.3 Novelty

A certain level of uniqueness must exist (Weber, 2012). Prior to this study, a model that incorporated passphrases and keystroke dynamics as a two-tier user authentication method to address security and usability did not exist.

6.8.2.4 Parsimony

The model must ensure that there is no wastage of resources (Weber, 2012). This was ensured by firstly assessing whether the sub-methods that made up the entire proposed solution were relevant. Secondly, all relevant solutions were measured to determine whether the benefit outweighed the costs/effort.

6.8.2.5 Falsifiability

There must be room for improvement in the model resulting from further research (Weber, 2012). The scope of this study had inherent restrictions which, it is assumed, may be solved through larger data collection sets and improvements in technological performance. Additional findings after assessment of those assumptions being realised may disprove this model or find areas for improvement that may require the model to be updated.

The whole model considerations above ensured that the process of merging the three sub-models, as derived from the literature chapters, was completed correctly. The next section provides a conclusion for this chapter.

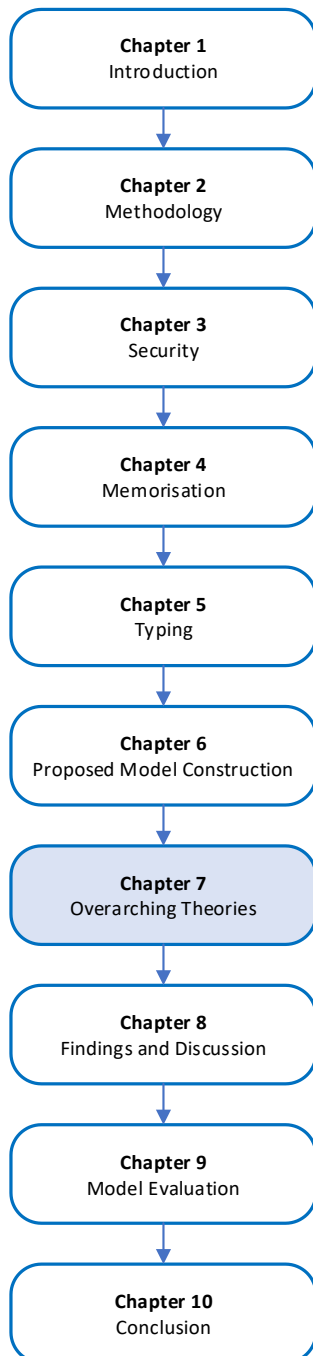
6.9 Conclusion

Chapter 6 aimed at constructing the proposed artefact for this study in the form of a model. The chapter focused on determining how best to merge the three models created from the literature chapters into one model. Subsequently, the sub-models were evaluated on their own prior to the merger, and then evaluated as a whole thereafter.

The proposed model was constructed, which is followed by the evaluation. The proposed model was used to guide the primary data collection process through the stage of refining the proposed model to its final form. The proposed model was evaluated and then updated based on findings from the login assessment experiment (see Chapter 8)

and then updated again after the expert review feedback was received (see Chapter 9). Before this is provided, the overarching theories used to assist the construction of the proposed model are presented in Chapter 7.

Chapter 7 – OVERARCHING THEORIES



Chapter 7	
7.1	Introduction
7.2	Theoretical Frameworks
7.3	Related Work on the AAA framework and ISO 9241 Standards
7.3.1	AAA Framework
7.3.2	ISO 9241
7.4	Components of the AAA framework and ISO 9241 Standards
7.4.1	AAA Framework Components
7.4.2	ISO 9241 Components
7.5	AAA Framework
7.5.1	Security – Authorization
7.5.2	Security – Authentication
7.5.3	Security – Accounting
7.6	ISO 9241
7.6.1	Usability – Effectiveness
7.6.2	Usability – Efficiency
7.6.3	Usability – Satisfaction
7.7	Theories Summary
7.8	Research Propositions
7.9	Conclusion

*Only heading 1 and heading 2 is displayed to avoid clutter

7.1 Introduction

The main aim of this chapter is to discuss the theoretical foundation of the study. Accordingly, certain theories were used to guide the research so as to ensure that the findings and conclusions were credible and could inform the empirical work. The theories focused on increasing security (Greene et al., 2016; Houshmand & Aggarwal, 2012; Shannon, 1948) and increasing usability (John & Kieras, 1994; Miller, 1956).

This chapter is introduced at this stage as a thorough understanding is required on how the three main theories (Shannon Entropy theory, Chunking theory and Keystroke-level model) were applied to this research. This was provided in Chapters 2, 3, 4 and 5. The theories used for this research are discussed in the next section. This is followed by an introduction to the overarching theories on usability and security. The next two sections form the main part of this chapter and discuss the way the supporting theories were used in conjunction with the main theories on usability and security. A summary section is then provided which is followed by a conclusion to this chapter.

7.2 Theoretical Frameworks

The three literature chapters used three theories to arrive at the findings and conclusions that focused on security and usability in user authentication. The following theories were used:

- First literature chapter (Chapter 3) on security used the Shannon Entropy theory (Shannon, 1948).
- Second literature chapter (Chapter 4) on usability in terms of the memorisation of passwords and passphrases used the Chunking theory (Miller, 1956).
- Third literature chapter (Chapter 5) focused on usability with regard to the user's ability to type the password/passphrase into the system. This chapter used the Keystroke-level model (John & Kieras, 1994) to guide the discussion and the findings.

These theories were used to guide the findings pertaining to the respective areas. However, these theories must be mapped back to the research problem to assess how security and usability have been addressed.

The AAA (Authentication, Authorisation and Accounting) framework (Alhassan & Quaye, 2017; Demchenko, et al., 2011; Jiang, 2018; Santuka, Banga, & Carroll, 2010) and the ISO 9241 standard (Ehteshami, Sadoughi, Saeedbakhsh, & Isfahani, 2013; Moumane,

Idri, & Abran, 2016) was used to assess whether the three theories (Shannon Entropy, Chunking and keystroke-level model) adequately contributed to the security (AAA framework) and usability (ISO 9241 standard) aspects of this research study. The AAA framework was originally developed for network security. However, it has also been applied to other aspects of system security (Garcia, Zarca, Hernández-Ramos, Bernabe, & Gómez, 2019; Islam & Atwood, 2006; Rensing, Karsten, & Stiller, 2002). ISO 9241 is a usability standard that provides guidelines on what needs to be considered to address usability. These are discussed further in the next section.

7.3 Related Work on the AAA Framework and ISO 9241 Standards

This section discusses how other researchers have used the AAA framework and the ISO 9241 standard in their studies and how these theories were used in this study. The AAA framework is firstly discussed, followed by a discussion on the ISO 9241 standard.

7.3.1 Related Work on AAA Framework

Toapanta, Maflab, and Orizagac (2018) developed a security model to identify security vulnerabilities. Construction of the model was guided by the AAA framework and the CIA (Confidentiality, Integrity and Accessibility) triad. Figure 7-1 displays the CIA triad.

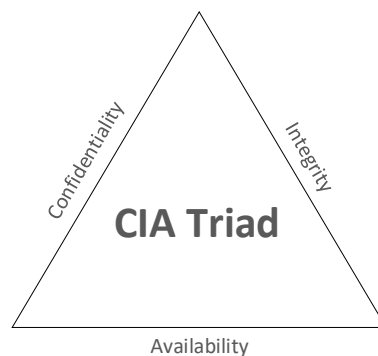


Figure 7-1: CIA Triad (Šidlauskas, 2018)

The following definitions are provided for each component of the CIA triad (Šidlauskas, 2018):

- Accessibility – Aims at ensuring that the technological services are available when required.
- Confidentiality – Refers to restricting access to information or actions that people do not have the authorisation to view or execute.

- Integrity – Relates to ensuring that data that is not meant to be manipulated remains unchanged.

Both Alhassan and Quaye (2017) and Jiang (2018) used the AAA framework to discuss various types of security threats and also referred to the CIA triad and AAA framework. Toapanta et al. (2018) explain that the AAA framework considers different types of security such as technical (software), administrative and even hardware.

Zheng, Cheng, Zhang, Zhao, and Wang (2018) used the AAA framework to guide the construction of a security model where the user provides an eight-digit password on registration. When a user logs into the system, the user is presented with a matrix created by the researcher, that will request a user to insert a different password during login based on an eight digit password created on registration. Although the research focused on the authentication component of the AAA framework, the other components (authorisation and accounting) were not considered. A security model was created in the first literature chapter (Chapter 3) of this study, using the Shannon Entropy theory (Shannon, 1948) to determine whether passphrases and keystroke dynamics algorithm satisfy the authentication component of the AAA framework.

7.3.2 Related Work on ISO 9241

Burkhard and Koch (2012) used ISO 9241 to guide the development of a methodology for assessing usability in elderly users' interactions with keyboard sizes on touch screen devices. Pradnyana, and Dantes (2019) used ISO 9241 to assess improvements in usability of a crowdsourcing system in Indonesia. Kim et al. (2014) also used ISO 9241 to assess the effect of keyboard size on usability, posture and productivity. The latter subsequently recommend that to reduce usability issues, a key should be larger than 16 mm on any touch screen keyboard. Kim, Aulck, Bartha, Harper, and Johnson (2012) previously conducted a similar study to Kim et al. (2014), however, in this study they assessed the productivity of touchscreen keyboards vs analogue keyboards using the ISO 9241 standard. They found that analogue keyboards are easier to use than touchscreen keyboards. Arthana, Horcher (2018) conducted an investigative study to assess cognitive efforts required for keyboard usability on mobile devices. The ISO 9241 standard was used to determine whether usability had been satisfied. Horcher (2018) consequently found that current mobile keyboards fail on usability for all components of this standard. König, Hofmann, and Bruder (2012) used the ISO 9241 usability standard to support the development of a user interface for a highly complex air traffic control system. It was vital that this system be developed by ensuring that little cognitive effort

is required by the user and to make use of efficient visual resources. König et al. (2012) subsequently found that while ISO 9241 was effective in assisting the development of the air traffic control system, a usability criteria had to be defined by considering the users' tasks.

The above-mentioned researchers focused on usability in relation to keyboard size. Horcher (2018) and König et al. (2012) add the concept of concentration to user–keyboard and user–system interaction as the main usability factor. In this study, the usability perspective is twofold: firstly, focusing on the ability of the user to memorise a password/passphrase and secondly, user–keyboard interaction in terms of typing that password/passphrase into the system.

The researchers above also covered touchscreen keyboards and analogue keyboards. This is important as the study allows for both types of keyboard interaction, noting that there is a difference in usability. This study proposes that passphrases are easier to type than passwords on both types of keyboard.

7.4 Components of the AAA Framework and ISO 9241 Standards

This section introduces the two main theories of this study. The AAA framework is firstly discussed which focuses primarily on security. The next section relates to the ISO 9241 standard which relates to usability.

7.4.1 AAA Framework Components

The AAA framework explains that three aspects of security must be considered. These are discussed further below.

- **Authentication** – Authentication is the system's way of giving the user an opportunity to validate that they are who they claim to be. For example, the user provides the system with a username and password/passphrase. If the data provided by the user matches the data held by the system, then the user is authorised to enter the system.
- **Authorisation** – Authorisation ensures that the user only gains access to the functionality and data that they are allowed to access on the system. For example, a user may view certain content but, based on their security user group, they may not download the content.

- **Accounting** – Accounting refers to the system resources a user uses when accessing the system with logs being kept on such data; for example how much data was used by the user, how much data was sent and received, and the length of time the user was on the system.

The next section discusses the second main theory in the form of a standard, ISO 9241.

7.4.2 ISO 9241 Components

ISO 9241 also provides three key factors that need to be considered for usability:

- **Effectiveness** – Measures the user's required approach to achieve a goal or goals.
- **Efficiency** – Ensures that the goal achieved outweighs the resources used to assist the user to achieve that goal.
- **Satisfaction** – The level of acceptance of the system by the user and anyone else affected by the system.

The next section explains how this study is associated to the AAA framework from a security perspective.

7.5 AAA Framework

The Shannon Entropy theory (Shannon, 1948; Yona & Diggavi, 2017) is the theory that supports the AAA framework in this study. Muthiya, Padvi, Patil, and Patil (2017) explain that the Shannon Entropy theory can be used to quantifiably measure password and passphrase attacks on a generalised scale. For example, certain characters are more commonly used than others however, the Shannon Entropy theory treats all characters as equally probable. Because of the nature of this study, the Shannon Entropy theory was nevertheless still found to be the best tool for assessing the strength of passphrases and conventional passwords. The main aim of the use of the Shannon Entropy theory was to compare the strength of passphrases and passwords. A strength estimate that does not cater for the rare password/passphrase attacks is an acceptable exclusion as the keystroke dynamics algorithm would likely protect the user from the rare password/passphrase attacks not considered by the Shannon Entropy theory.

The components of the AAA framework were mentioned in the previous section. This section provides a more in-depth discussion on each component of the AAA framework and the way the findings of this study were affected by them. The first section provides

a discussion on the authorisation component which is followed by the authentication component. The last section explains the accounting component of the AAA framework.

7.5.1 Security – Authorisation

The proposed solution does not have an influence on authorisation since it targets user authentication. Other security protocols (beyond the scope of this research) would address this security requirement.

7.5.2 Security – Authentication

This research study contributes largely to the authentication component of security as the proposed solution aims to improve the user authentication process using passphrases and keystroke dynamics. Passphrases are recommended to a user as an option during the user authentication process. Using the Shannon Entropy formula (Shannon, 1948), it was found that passphrases are stronger than conventional passwords. The findings from Chapters 3 and 4 show that passphrases are easier to memorise than passwords and will, consequently, discourage users from performing unsecure acts to remember them such as writing them down or creating a common or weak password.

The Shannon Entropy formula (Aguilar & Guedes, 2015; Arora et al., 2015; Shannon, 1948) was also applied to keystroke dynamics. Keystroke dynamics adds another layer to authentication security by requiring the user to not only recall the passphrase from memory but also to type the passphrase into the system correctly. At the same time, the keystroke dynamics algorithm monitors the user's typing patterns to validate that the correct user is inserting the password/passphrase. Using keystroke dynamics, even if an unauthorised party retrieves a user's password/passphrase, by whatever means; they will still have to insert the password/passphrase in the same manner (pattern and speed) that the authorised user applies when entering the password/passphrase into the system.

The Shannon Entropy formula (Aguilar & Guedes, 2015; Arora et al., 2015; Shannon, 1948) was also used to assess whether keystroke dynamics were more effective, from a security perspective, than passwords or passphrases. Passphrases and keystroke dynamics complement each other, as a keystroke dynamics algorithm assessing a passphrase input is likely to create fewer false positives than when assessing a password input, which involves character switching (Ponkshe & Chole, 2015). False positives, in the context of user authentication keystroke dynamics, occur when a

legitimate user enters the correct password or passphrase but does not enter it as he/she does normally. Hence, the keystroke dynamics algorithm may restrict a legitimate user's access to the system.

7.5.3 Security – Accounting

Accounting security, from a user authentication process perspective, reduces the number of system resources as the use of passphrases reduces the risk of login failures resulting in the user spending less time on the user authentication screen. This was found with the assistance of the Chunking theory (Miller, 1956), the Keystroke-level model (John & Kieras, 1994), the login assessment experiment and the expert review.

A keystroke dynamics algorithm collects additional data on the user in terms of how they interact with the system. When applying the Shannon Entropy formula to this it was found that user–system interaction data can be used to further increase security (Shannon, 1948). However, it should be noted that the keystroke dynamics algorithm puts additional strain on system resources as more processing power is required to run the algorithm.

Because the proposed solution provides for both an increase and a decrease in system resources, these cancel each other out and hence “no impact” on system resources would be experienced. Further research should be undertaken to assess the magnitude of the increase versus the decrease to determine whether a “no impact” conclusion is valid.

The next section explains how this study is associated to ISO 9241 from a usability point of view.

7.6 ISO 9241

The previous section discussed the first main theory regarding security. This section is focused on the other main theory, namely, the ISO 9241 standard. The supporting theories relating to the main theory are the Chunking theory and the Keystroke-level model. In this section, each component of the ISO 9241 usability standard is discussed separately, starting with the effectiveness component. The efficiency component is then discussed followed by the satisfaction component of the ISO 9241. Each section explains how this study influences the respective component of ISO 9241.

7.6.1 Usability – Effectiveness

Usability effectiveness can be measured by assessing whether an approach adopted by a user has the ability to achieve the intended goal. In terms of this research, the goal is the authentication of the user and the approach is the solution proposed by this study (a two-tier user authentication solution involving passphrases and keystroke dynamics). The second literature chapter (Chapter 4) found, using the Chunking theory, that passphrases are easier to remember than conventional passwords (Bošnjak & Brumen, 2016; España, 2016; Miller, 1956). Huh et al. (2015) applied the Chunking theory to assess the memorisation of OTPs, while Marquardson (2012) focused their study on assessing the memory impact of certain password policies. In addition to the fact that passphrases assist in the memorisation of content for user authentication, the Keystroke-level model (John & Kieras, 1994; Jorritsma et al., 2015; Lee et al., 2015) was used to support findings on inserting a passphrase into the system. With the assistance of the Keystroke-level model, passphrases were found to be easier to type than typical passwords with less risk of typographical errors occurring. Quezada et al. (2017) used the Keystroke-level model to assess the interactions of users with autism with a system interface specifically designed for such users.

7.6.2 Usability – Efficiency

The ISO 9241 efficiency component in this study relates to the user's ability to seamlessly authenticate themselves before gaining access to a system. Efficiency refers to successfully executing a task while using the least amount of time and effort. From a memorisation perspective, the Chunking theory (Bošnjak & Brumen, 2016; Carstens, Malone, & Mccauley-Bell, 2007; España, 2016; Forget & Biddle, 2008; Miller, 1956; Zhang, Luo, Akkaladevi, & Ziegelmayer, 2009) was used to discover that less effort is required by the user to memorise a passphrase as opposed to a conventional password. If a user can successfully memorise a passphrase, the next step is inserting the passphrase into the system.

With guidance from the Keystroke-level model (John & Kieras, 1994; Jorritsma et al., 2015; Lee et al., 2015), it was discovered that the user expends less effort typing passphrases than passwords, mainly because there is no character switching. This also has a larger impact when typing on a mobile device such as a cellphone or tablet. Sulaiman, Lokman, and Hussien (2017) also used the Keystroke-level model to assess the usability impact of a technical solution aimed at improving user–system interaction on mobile web browsers. Character switching also increases the risk of typographical

errors, which results in login failure and the user having to login again, thereby reducing efficiency.

7.6.3 Usability – Satisfaction

Satisfaction from a usability perspective relates to the level of acceptance of the system by the user. Firstly, offering users the option to use a passphrase as opposed to a password to increase text-based authentication strength will influence the user's perception that the system is now more secure (Colnago et al., 2018; Davis, Bagozzi, & Warshaw, 1989). Secondly, using the Chunking theory (Miller, 1956) it was found that passphrases are easier to remember than conventional passwords and using the Keystroke-level model (Jorritsma et al., 2015; Lee et al., 2015; John & Kieras, 1994), it was found that passphrases are easier to type than passwords. This is mainly due to the absence of character switching. Because passphrases are easier to memorise and type, user satisfaction is increased. Satisfaction is further increased by the user knowing that the system is more secure as a result of passphrases and keystroke dynamics (Montesdioca & Maçada, 2015).

It was mentioned earlier in this chapter that the obtaining of false positives poses a risk with keystroke dynamics. If false positives are obtained, user satisfaction is affected. However, in Chapters 3 and 5 it was suggested that leniency should be set at high until more user–system interaction data is collected on the user. When a large volume of user–system interaction data has been collected, a more stringent keystroke dynamics algorithm can be applied. The next section summarises the influence of this study on the components of the AAA framework and the ISO 9241 usability standard.

7.7 Theories Summary

The above sections of this chapter explained the findings obtained using the supporting theories. These findings were then applied to the components of the two main theories (AAA framework and ISO 9241) to assess whether the findings achieved the research goal pertaining to increasing security and usability during the user authentication phase.

Figure 7-2 graphically depicts the theories used to guide the research efforts and findings in this research study. Figure 7-2 also illustrates the associations between these theories based on this research study.

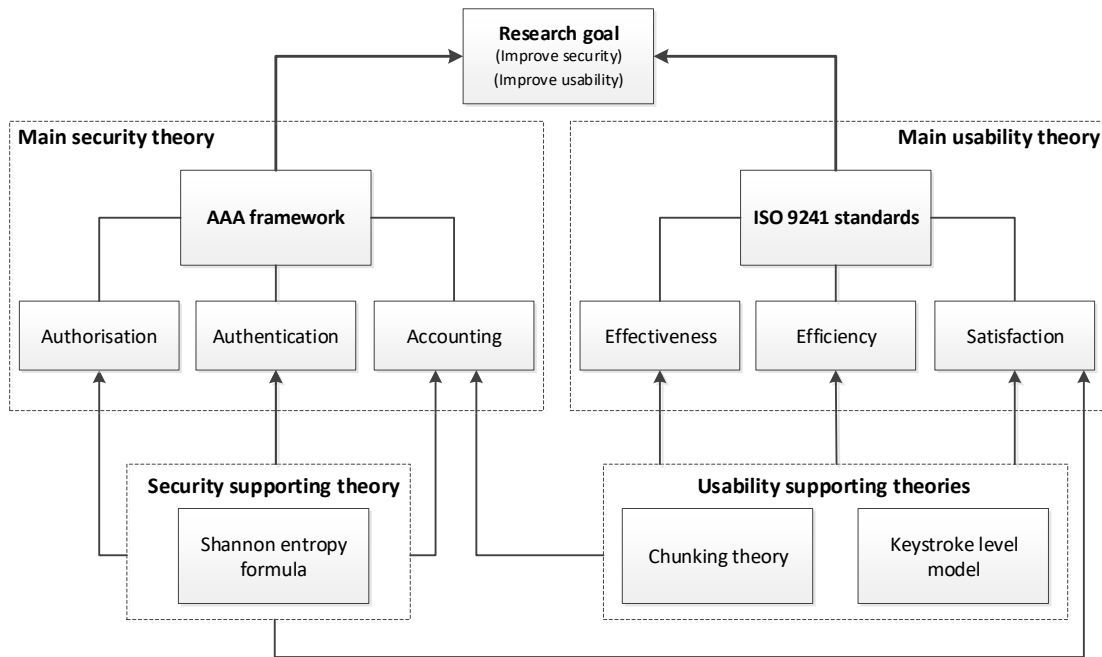


Figure 7-2: Guiding Theories

In Figure 7-2, the first block indicates the goals of this study, which are to simultaneously increase security and usability in the user authentication process. The two arrows pointing to the research goal illustrates that the two main theories used to assess the proposed solution addressed the stated research goal. The components of the two main theories are linked to other theories (third row of boxes from the top of Figure 7-2) and boxed in with the dotted lines to illustrate the two main theories and their components. These theories include the AAA framework used mainly to assess security and ISO 9241 used mainly to assess the usability aspect of this research. The supporting theories have also been grouped together with a dotted box. The Shannon Entropy theory is grouped under the theory supporting security, and the Chunking theory and the Keystroke-level model are grouped under the theories supporting usability. The arrows from the supporting theories to the components of the two main theories depict the relationship between the supporting theories and the main theories. These links were based on how the supporting theories were used to identify certain findings for this study.

Figure 7-2 illustrates how all the theories are associated to each other and were used to assess and guide findings. Table 7-1 summarises the overall influence of the study's findings on the two main theories at a high level.

Table 7-1: Main Theory Foundation

AAA Framework		ISO 9241	
Security components	Research impact	Usability components	Research impact
Authorisation	No impact	Effectiveness	Increase
Authentication	Increase	Efficiency	Increase
Accounting	No impact (cancellation)	Satisfaction	Increase

The metrics used in Table 7-1 indicates whether the findings increased, decreased or had no impact on security and usability based on the main theories. This indicates which components were influenced to increase security (indicated in the AAA framework components) and usability (indicated in the ISO 9241 components). The last section provides a conclusion for this chapter.

7.8 Research Propositions

Five research propositions were identified. The assessment of these five propositions assisted in answering the sub-research questions for this study. Propositions, in the context of this study, are predictions which require confirmation. The propositions for this study were confirmed through the primary and secondary research. Table 7-2 provides the research propositions for this study and also maps the propositions to the research sub-questions and indicates where the propositions were mainly assessed (“proposition measures column”).

Table 7-2: Research Propositions

Propositions	Research sub-question/s (key below)	Measures
1. Passphrases are more secure than passwords.	1	- Chapter 3 - Expert review
2. Keystroke dynamics supports passphrases better than passwords.	1, 2 and 3	- Chapters 3, 4 and 5 - Expert review
3. Passphrases are easier to remember than passwords.	2	- Chapter 4 - Expert review - Login assessment experiment
4. Passphrases are easier to type than passwords.	3	- Chapter 5 - Expert review - Login assessment experiment
5. Keystroke dynamics increases security and has little to no negative impact on usability.	1, 2, 3	- Chapters 3, 4 and 5 - Expert review
Research sub-questions		
1. What needs to be considered when ensuring the security of passphrases and keystroke dynamics algorithm as a method of user authentication?		
2. What factors in terms of system usability influence the memorisation of passphrases and may influence a keystroke dynamics algorithm?		
3. What system input factors influence the use of passphrases and may affect the keystroke algorithm?		

The above table indicates that secondary data gleaned from the previous chapters were used to assess the propositions. These were confirmed by the primary data assessment methods – the login assessment experiment and the expert review. The login assessment findings are discussed in more detail in the next chapter.

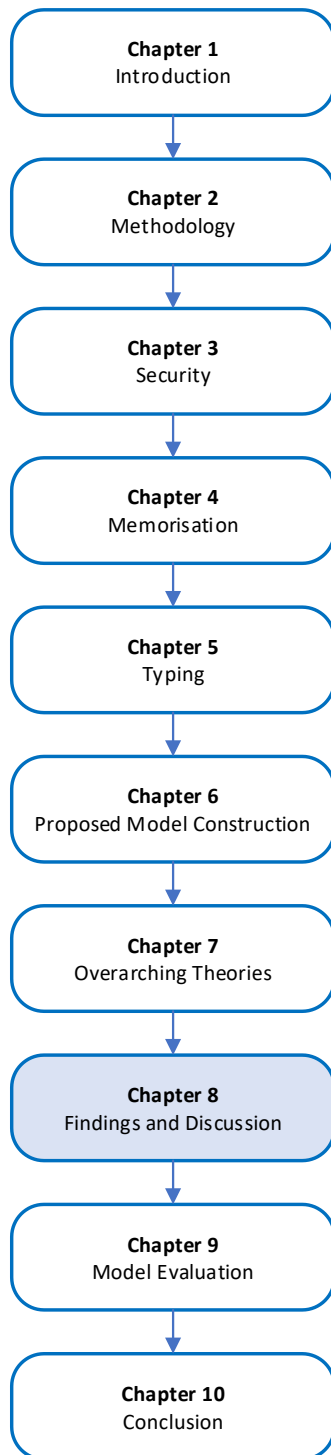
7.9 Conclusion

This chapter focused on explaining the theories used to guide the findings of this research study. Consequently, the findings gathered in this study using the supporting

theories (Shannon Entropy model, Chunking theory and Keystroke-level model) aligned to the aspects of the main theories (the ISO 9241 usability standard and the AAA framework). These two theories were the main theories used to assess whether the research goal was achieved; i.e. to simultaneously increase security and usability. Since the findings could improve certain aspects of the two main theories, it may be concluded with a certain level of confidence that the research artefact in the form of a model is able to address the research problem and the overall goal of this study.

In order to further strengthen the claim that the proposed solution can address the research problem and the overall goal of this study, primary data collection and analysis was focused primarily on assessing the impact of the proposed solution on the research problem and the overall goal. This formed the basis of the next chapter.

Chapter 8 – FINDINGS AND DISCUSSION



Chapter 8	
8.1	Introduction
8.2	Login Assessment Data Collection Approach
8.3	Login Assessment Data Analysis Approach
8.3.1	Data Analysis Process
8.3.2	Data Analysis Scoping
8.3.3	Constructs Evaluation
8.4	Data Collection Results and Discussion
8.4.1	Security Evaluation
8.4.2	Memory Evaluation
8.4.3	Typing Evaluation
8.4.4	General Observations
8.4.5	Summary of Results
8.5	Model Updates
8.5.1	Updated Security Sub-model
8.5.2	Updated Memory Sub-model
8.5.3	Updated Typing Sub-model
8.5.4	Updated Proposed Model
8.6	Conclusion

*Only heading 1 and heading 2 is displayed to avoid clutter

8.1 Introduction

This chapter reports on the data collected from the login assessment experiment and describes the empirical work carried out. The chapter provides the results of the login assessment and explains how the data collected was analysed. Trends, patterns and key observations were also noted, as well as the way they influenced the proposed model. The results of the login assessment were assessed in order to confirm components and flows/associations of the model, add components and flows/associations to the model and/or contradict components and flows/associations of the model.

Since the proposed model was constructed from secondary data, the primary data that contradicts the proposed model components and flows were confirmed with additional primary data from the expert review before any changes were made to the model. The expert review is discussed in more detail in the next chapter.

This chapter begins by explaining the data collection approach for the experiment. This is followed by the data analysis approach. After these two sections, the data collection results and discussion are provided. A section that summarises the results from the previous section is presented. This is followed by an indication of the changes to the sub-models and proposed model. Lastly, a conclusion section is provided.

8.2 Login Assessment Data Collection Approach

This section explains the login assessment experiment that was undertaken. The aim of this login assessment experiment was to collect data on user authentication interaction using passwords and passphrases. It should be noted that keystroke dynamics were not assessed through the login assessment. Accordingly, a website (www.loginassessment18.co.za) was developed to record user interaction with a user authentication process. Appendix A provides the screenshots and a screen flow diagram of the website. The website interface replicated the layout and functionality of the common user authentication interfaces currently in use. This ensured that relevant and up-to-date data was collected.

The aim of the website was to confirm and clarify findings and propositions identified from the literature relating to this research area. In the experiment, an adequate sample size was deemed to be a minimum of 65–100 users interacting with the website at least ten times (Chiasson et al., 2009; Yang et al., 2014), thus allowing for an adequate volume of data to be collected. Each user was asked to create a password and a passphrase,

and then to login with the password and the passphrase at least once a day for a minimum of ten days.

If the password was typed incorrectly, the user had the option to try again, reset their password or continue to the passphrase login screen. If the user then typed the passphrase incorrectly, they also had the option to try again or reset the passphrase. Any failed logins (password or passphrase) required the user to complete a short survey to understand why the password/passphrase had been inserted incorrectly into the website. Figure 8-1 depicts the screen flow for the website.

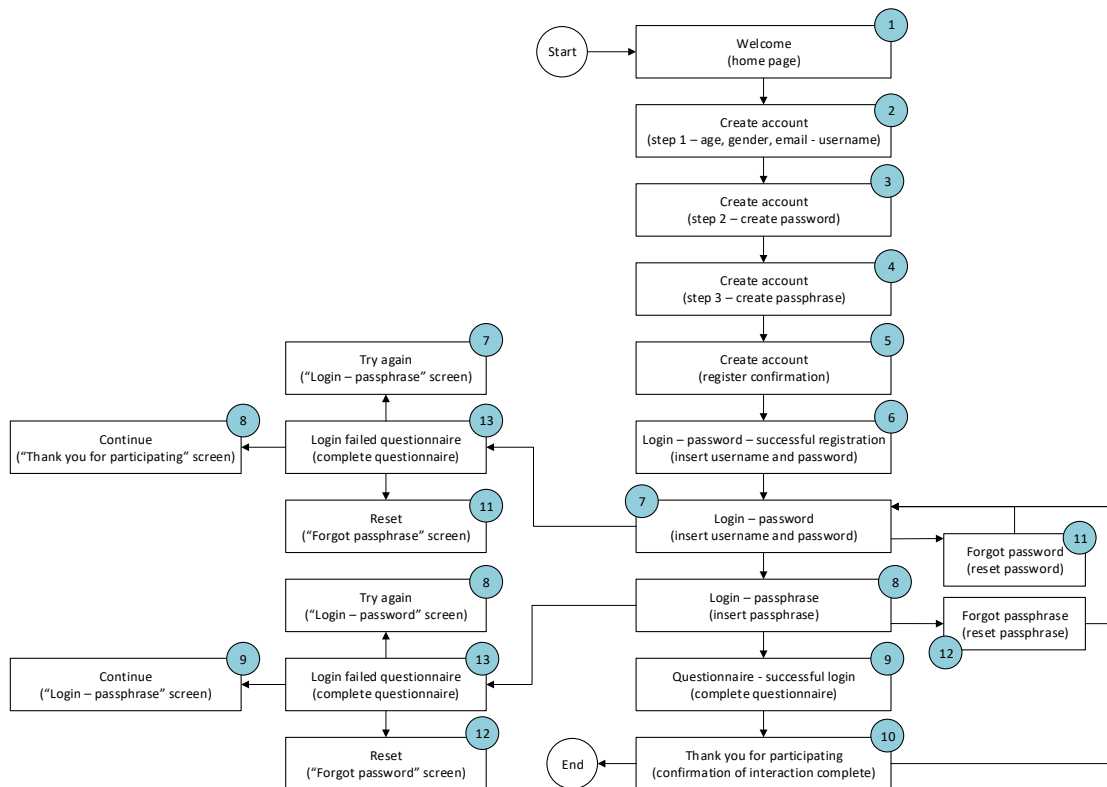


Figure 8-1: Login Assessment Website Screen Flow

Following this explanation of the data collection approach, the next section provides the measures for the login assessment.

8.3 Login Assessment Data Analysis Approach

The data analysis section firstly explains the data analysis process, followed by an explanation on scoping. The last section provides an indication of what was assessed.

8.3.1 Data Analysis Process

The login assessment produced a large quantity of data which was analysed using several different methods and at different levels of detail. It was important not to report any irrelevant data for this study; so as to ensure that the data reported aligned with the intentions of this study, a method was required to guide the analysis. The aim of the login assessment was to evaluate the findings from the three literature review chapters. For this reason, the research sub-models constructed in these chapters were used to guide the data analysis process and create the proposed model. Since some of the components in the proposed model were at a too high level to guide the login assessment data analysis, the sub-models were also used. The sub-models and proposed model were subsequently updated according to the results of the login assessment. The proposed model was then sent to the experts for review (discussed in more detail in the next chapter).

Following this explanation of the data analysis approach, the next section discusses the constructs from the sub-models that can be evaluated together with the data collected from the experiment.

8.3.2 Data Analysis Scoping

The data analysis process was initiated by firstly identifying what needed to be evaluated. Note that it was not possible to evaluate some items owing to the nature and/or limitations of the login assessment. The login assessment website (tool used to collect the data) had to be developed in such a way that participant drop-outs were kept to a minimum while still collecting as much relevant data as possible. The first step in the analysis process was to identify what exactly had to be evaluated.

The sub-models were used as a starting point for this step in the analysis. The first sub-model constructed from the literature chapters was the security sub-model, which is shown in Figure 8-2.

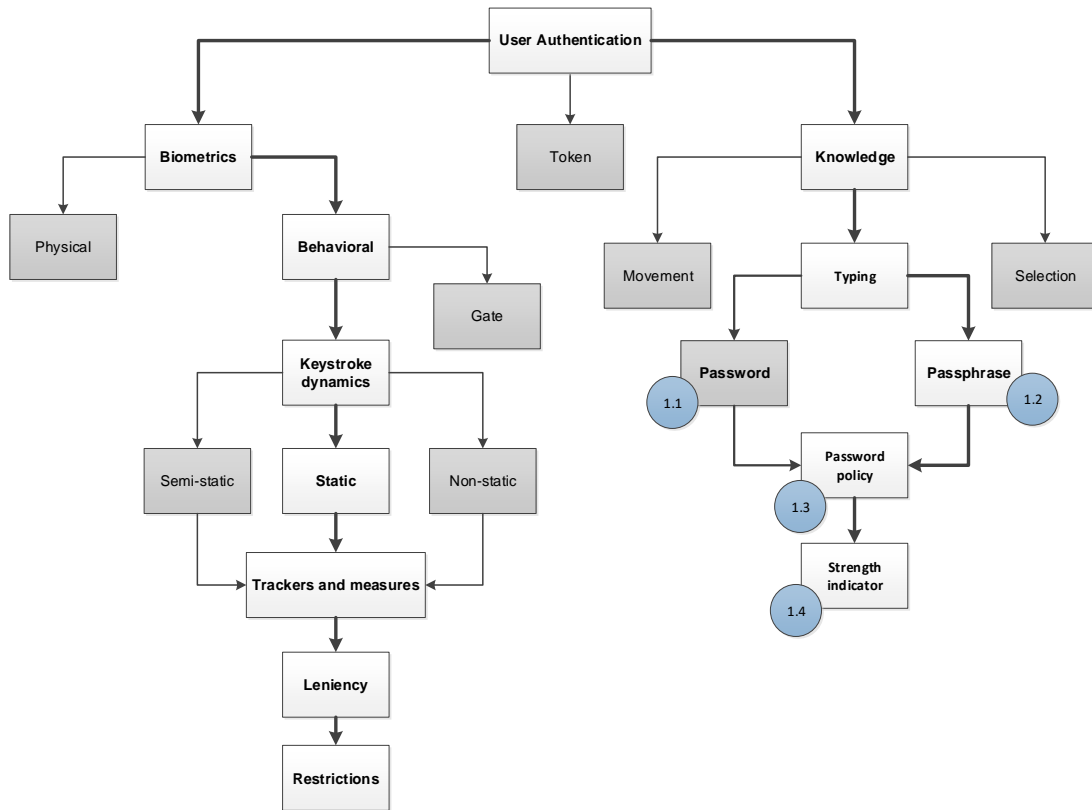


Figure 8-2: Security Sub-model

The blue numbering in Figure 8-2 indicates the constructs that were assessed by the login assessment experiment. The lowest level of detail was used as these detailed constructs make up the entire construct. In this way, it was easier to identify whether a detailed construct aligned or did not align with the login assessment results as opposed to a more high-level construct. The same approach was applied to the second sub-model focused on the memorisation of passwords and passphrases. The memory sub-model is illustrated in Figure 8-3.

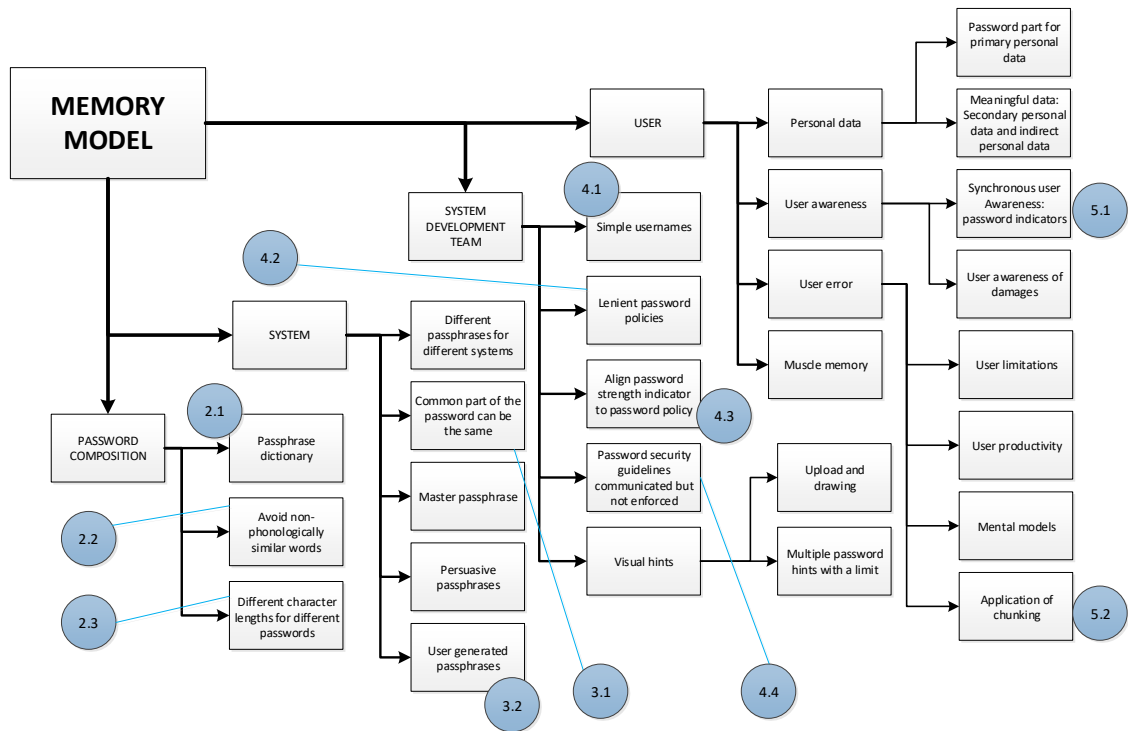


Figure 8-3: Memory Sub-model

The blue numbering in Figure 8-3 indicates the memory-focused constructs that were evaluated by the login assessment experiment. The last sub-model created from the literature chapters focused on typing impact on user authentication as a key usability aspect. The typing sub-model is depicted in Figure 8-4 below.

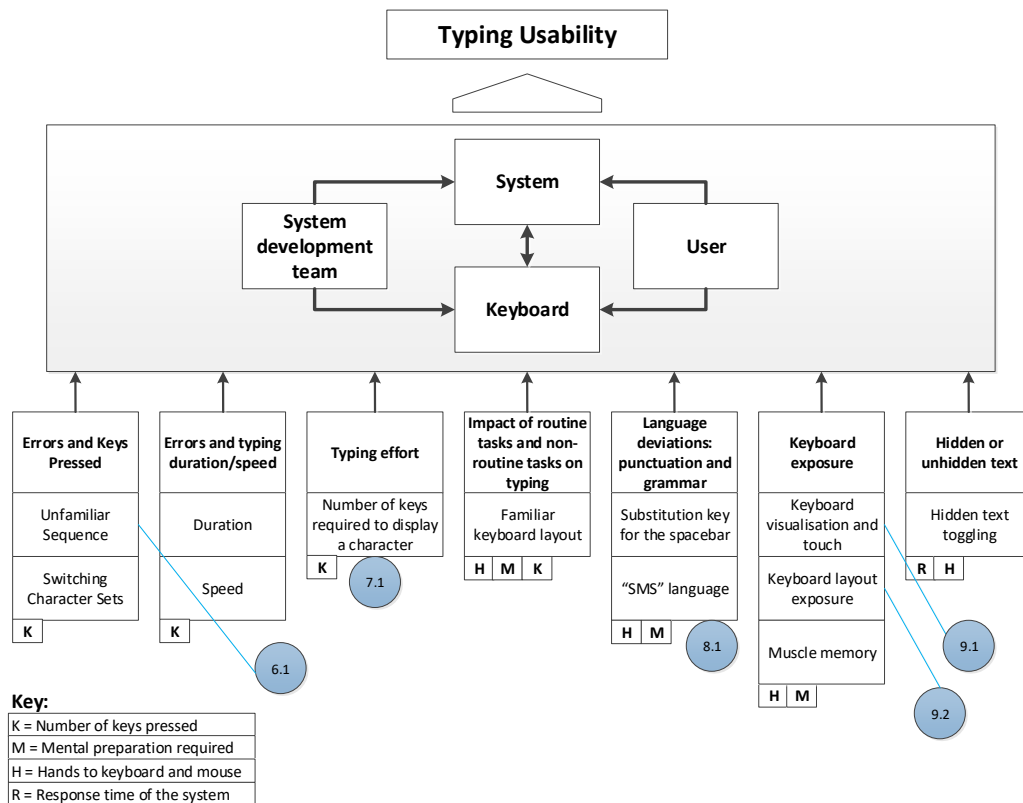


Figure 8-4: Typing Sub-model

The blue numbering in Figure 8-4 illustrates the typing constructs that were evaluated based on the data obtained from the login assessment experiment.

Now that the main constructs have been identified, the next section maps the sub-model constructs to the section that assessed the sub-model constructs.

8.3.3 Constructs Evaluation

Based on the sub-models depicted above, there are several evaluations that needed to be discussed to evaluate certain findings from the literature chapters. These evaluations are provided in Table 8-1 below. They were then organised into meaningful categories such as security, memory, typing and general. The numbering in the sub-models above is aligned to the numbers depicted in Table 8-1. Note, that there are additional “construct/research validations” in Table 8-1 that are not illustrated in the sub-models. These are aligned to similar studies’ findings, other researcher’s statements and propositions made throughout this study and are indicated in the “general” section in Table 8-1.

Table 8-1: Construct and Proposition Evaluation

Model numbering	Construct/Proposition Evaluation	Section
Security		
1.1 – 1.4	Strength indicators	8.4.1.1
Memory		
2.1	Passphrase dictionary	8.4.2.1
2.2	Phonologically similar words	8.4.2.2
2.3	Password and passphrase length and login failures	8.4.2.3
3.1	Common segments across passwords and passphrases	8.4.2.4
3.2	User-generated passphrases	8.4.2.5
4.1	Simple usernames	8.4.2.6
4.2	Lenient password policies	8.4.2.7
4.3, 4.4, 5.1	Strength indicator and password policy alignment	8.4.2.8
5.2	Application of chunking	8.4.2.9
Typing		
6.1	Unfamiliar sequence	8.4.3.1
7.1	Number of keystrokes	8.4.3.2
8.1	SMS language, acronyms and abbreviations	8.4.3.3
9.1	Keyboard visualisation and touch	8.4.3.4
9.2	Keyboard layout exposure	8.4.3.5
General		
General 1	Password and passphrase entropy	8.4.4.1
General 2	Complicated passwords, passphrases and memory issues	8.4.4.2
General 3	Similar passwords and passphrases created by users	8.4.4.3
General 4	Password and passphrase memory improvement over time	8.4.4.4
General 5	Login duration	8.4.4.5
General 6	Password and passphrase resets	8.4.4.6
General 7	User age and gender	8.4.4.7
General 8	Password and passphrase failures	8.4.4.8
General 9	Language in passwords and passphrase	8.4.4.9
General 10	User preference	8.4.4.10

Table 8-1 above provides the items that were evaluated using data collected from the login assessment.

8.4 Login Assessment Results and Discussion

This section forms the basis of this chapter. Each of the constructs/research evaluations mentioned in Table 8-1 was discussed separately. The sections below are aligned to the sections provided in Table 8-1. The results of the constructs/research evaluations based on the login assessment data are presented before a discussion is provided. At the end of the discussion an explanation is provided on any changes to the sub-models and proposed model. The login assessment experiment aimed to collect login data from 66–100 participants. A link was sent to all potential participants (this included a combination of groups: other students, work colleagues, friends and professionals) via email requesting that the email be forwarded to any other potential participants. For this reason, it is unknown how many participants were contacted. However, in terms of known participants, 123 participants interacted with the login assessment website. Of the 123 participants, 112 participants managed to complete the login assessment, i.e. 112 participants interacted with the website ten or more times. With regard to dropout, 11 participants interacted with the website one to four times before dropping out. This meant that there was a 9% dropout rate for this experiment, which was lower than expected. No reasons were given for such dropouts.

The sections included in the data collection results and discussion section are divided into four subsections. Security is the first section, followed by a section on memory. After the memory section, a typing section is provided. Lastly, general observations from the login assessment experiment are discussed.

8.4.1 Security Evaluation

The security section includes one subsection that covered all four constructs in the security sub-model.

8.4.1.1 Strength Indicators

In Chapter 3, Section 3.7, Strength Indicator, it was found that strength indicators have a positive influence on the creation of stronger passwords/passphrases by users. This was tested in the login assessment. Two forms of interfaces were used to assess the impact of two different strength indicators: 1) a traditional strength indicator and 2) an uncommon strength indicator. To compare the effectiveness of the strength meters, a

count was made of how many users created passwords and passphrases beyond the minimum requirement (password policy) and by how many characters they exceeded the minimum requirements. It should be borne in mind that, in the literature chapters, a new type of strength indicator was mentioned, the running bunny. The running bunny strength indicator introduces a moving image of a bunny running to the user; as the strength of the password increases the bunny moves faster. This section compares this strength indicator with the traditional password strength bar. The data to support this assessment has been plotted in Figure 8-5 below. The x-axis indicates the number of characters included in passwords for the respective character set and the y-axis shows the number of passwords. A total of 158 passwords were created by participants.

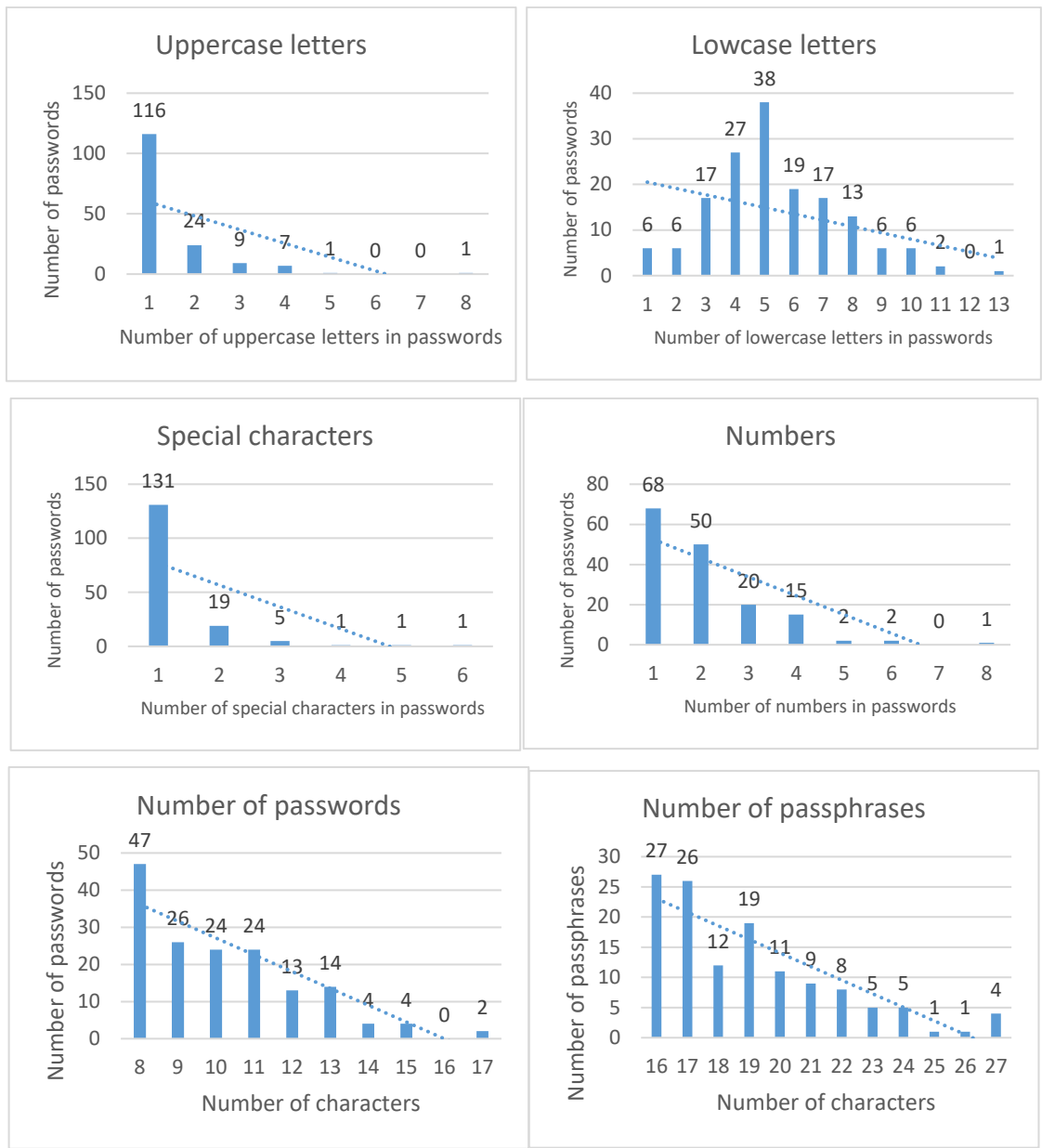


Figure 8-5: Strength Indicators

When analysing password creation patterns in Figure 8-5 (in the following charts: uppercase letters, lowercase letters, special characters and numbers) the data indicates that the majority of users include only one special character and one uppercase letter. Lowercase letters are the most common character set used in passwords, followed by numbers.

In terms of passwords and passphrases created that comply strictly to the password policies, passwords were significantly higher than passphrases (47 to 27). When comparing the number of passwords and passphrases that slightly exceed the password policy, it would seem that the different strength indicators have no impact on the creation

of stronger passwords or passphrases. When assessing the number of passwords and passphrases that significantly exceed the password policies, passphrases would be more of a culprit than passwords. This emphasises that the running bunny encourages users' curiosity to find out how fast the bunny can run. Therefore, the running bunny is an effective strength indicator than should be used on user authentication screens.

Impact on sub-models and proposed model: *In the security sub-model in Figure 8-2, the “strength indicator” construct now has an extended construct (“moving image”) to indicate that the running bunny can be used as a strength indicator.*

8.4.2 Memory Evaluation

This section is made up of nine subsections, all of which relate to login instances (login failures and successes). A passphrase dictionary is firstly discussed, followed by non-phonologically similar words. The next section discusses password length and then a section is provided on common segments included in passwords. User-generated passwords are then discussed, followed by an explanation on simple usernames and then on lenient password policies. The next section includes an explanation of strength indicators and password policy alignment. Lastly, the application of chunking is discussed.

8.4.2.1 Passphrase Dictionary

All passphrases were assessed to identify common words or classification of words. For example; common words may be Toyota and Ford, and the classification would be “cars”. Table 8-2 displays these results based on the login assessment data collected.

Table 8-2: Passphrase Dictionary

Category	Count
Miscellaneous	33
Movies	30
Sport	13
Music	12
Inappropriate	9
Location	7
Food	5
Repetitive letters (pattern)	4
Car	3
Family	3
Gaming	3
Random letters	2
Slang	2
Language	1
Religion	1
TOTAL	128

A total of 128 passphrases were created by participants throughout the login assessment experiment. There were 33 passphrases that had no common classification or were unknown in terms of determining common words. The next three classifications of passphrases related to entertainment, such as movie lines and movie titles; sports names, teams and players; and music titles and artists. Based on the data collected, it would seem that the passphrases created across users can be categorised into meaningful classifications. This emphasises that a passphrase dictionary should be created as passphrase usage proliferates. When users create passphrases that match something in the passphrase dictionary, users should be warned that the passphrase created is a common passphrase that might be susceptible to hacking but should not be restricted from using it. Restricting use may negatively affect usability, especially if a user ends up being restricted to a limited number of attempts when creating a passphrase.

Impact on sub-models and proposed model: The “passphrase dictionary” construct in the memory sub-model (in Figure 8-3) has been renamed to “passphrase dictionary warning message” to assist security and minimise usability.

8.4.2.2 Phonologically Similar Words

Phonologically similar words are words that are pronounced the same but spelt differently. For example, “bass” and “base”. Figure 8-6 provides the percentage failure results for all phonologically similar words identified in passwords and passphrases. The x-axis separates the type of failures and the y-axis indicates the percentage of login instances. Overall, 21 phonologically similar passwords were identified and 23 phonologically similar passphrases. A total of 25/157 login failures were identified for phonologically similar words in passwords. Alternatively, 11/145 login failures were identified for phonologically similar words in passphrases. Note that only passwords and passphrases that had phonologically similar words were taken into consideration for Figure 8-6.

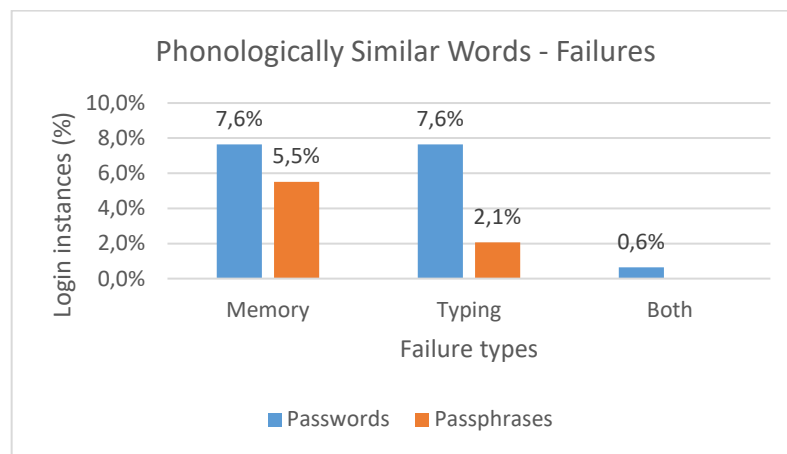


Figure 8-6: Phonologically Similar Words

In terms of failure, it can be seen in Figure 8-6 that phonologically similar words in passwords create memory and typing issues, while those in passphrases are more susceptible to memory errors and typing errors. Although the data emphasises that phonologically similar words in passphrases have a higher probability of success than those in passwords, it is recommended that these words should still be avoided whether using a password or a passphrase.

Impact on sub-models and proposed model: The “avoid phonologically similar words” construct which is part of the memory sub-model in Figure 8-3 has been confirmed as correct. Therefore, no change is required to the memory sub-model based on this section.

8.4.2.3 Password and Passphrase Length and Login Failures

Zhang and McDowell (2009) found that, in 1999, 53% of users had a password character length of below six. In 2006, this had decreased from 53% to 82% (Zhang & McDowell, 2009). This indicates that password length has increased as security has had to become more stringent. Consequently, this percentage is expected to increase in the future as hacking becomes more proliferated. Alomari and Thorpe (2019) and Zhang and McDowell (2009) also explain that the introduction of password policies has a lot to do with the increase in password length. This section aims to determine whether there is any correlation between password/passphrase length and login failures; and, if there is, is it the same for passwords and passphrases. It is expected that the longer the password length, the higher the failure rate will be. Figure 8-7 shows password and passphrase lengths and the percentage of login failures. The x-axis shows the password/passphrase length and the y-axis indicates the percentage of login instances. A total of 236/1335 password login failures were made and 83/1233 passphrase failures were made by participants.

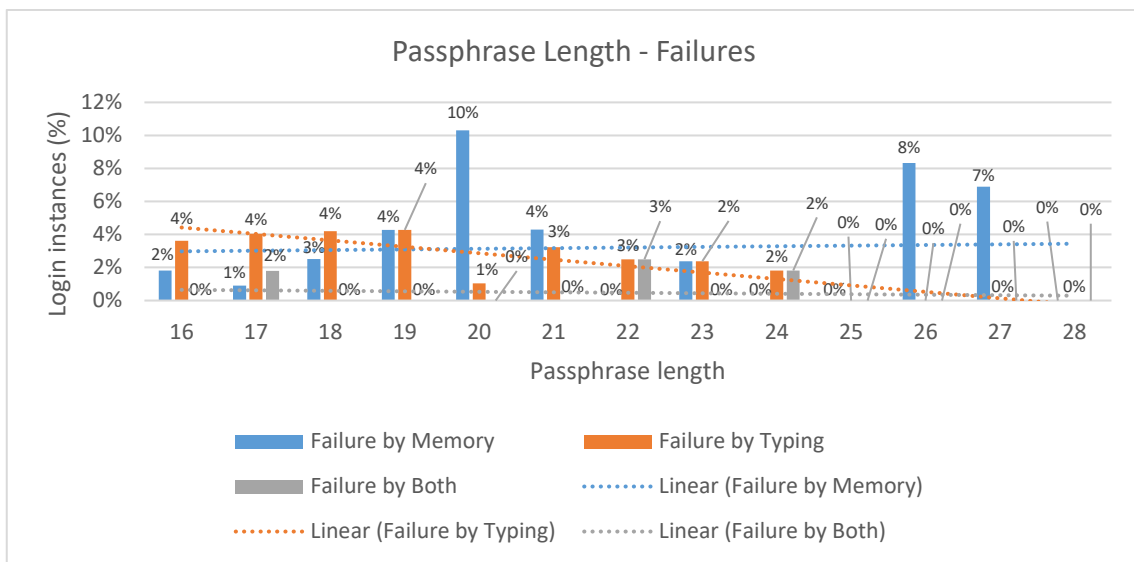
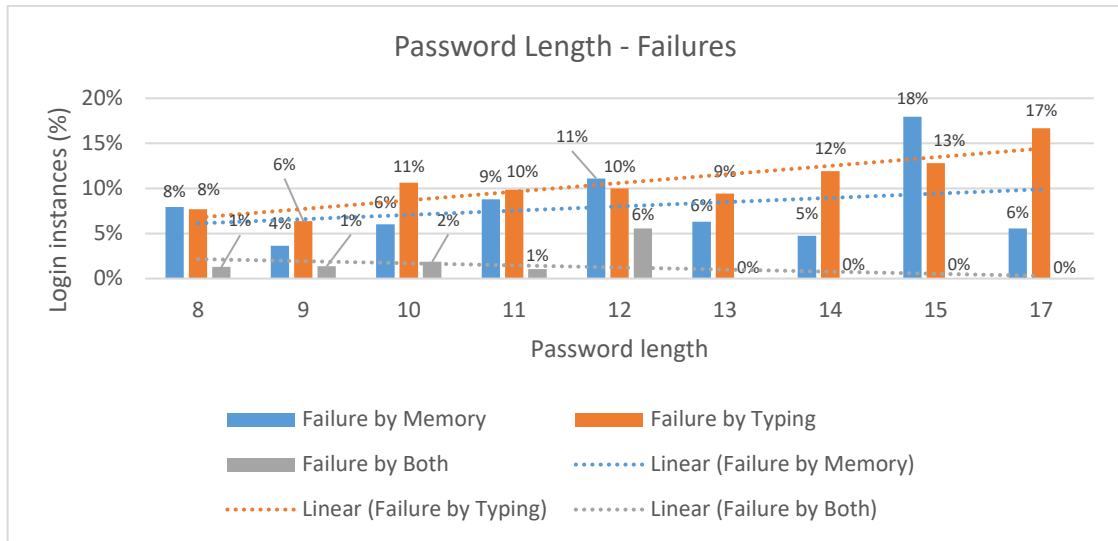


Figure 8-7: Password and Passphrase Length and Login Failures

The password graph in Figure 8-7 shows that all failure type logins indicate an upward trend. This data supports the expectation that the longer the password the greater the risk of failure.

When looking at the passphrase chart, memory failures also support the expectation of a correlation between passphrase length and risk of failure. On the other hand, passphrase typing failures indicate that failures reduce as the passphrase becomes longer. One explanation for this inconsistency could be that a long phrase may be used frequently by a user and thus he/she becomes comfortable typing that phrase. This would reduce the risk of typographical failures.

Impact on sub-models and proposed model: The “different character length for different passwords” construct in the memory model suggests that to increase security

through unpredictability, different passwords used for different systems should have different character lengths. Although this may yield true, users should be cautious as the more characters used in a password, the higher the risk of login failure. For this reason, the construct was maintained but renamed from “different character length for different passwords” to “slightly different character length for different passwords” in the memory sub-model.

8.4.2.4 Common Segments Across Passwords and Passphrases

In Chapter 4 it was recommended that users use a common segment across all passwords to assist memorisation. For example, a password format such as the first letter should always be uppercase and the last character should always be a number. Since the login assessment included only one system, this finding needed to be confirmed by assessing any common segments across passwords and passphrases. Table 8-3 provides the results. A total of 158 passwords and 128 passphrases were assessed.

Table 8-3: Common Segments Across Passwords and Passphrases

Character type	Count
Password	
Special at start	3
Special at end	33
Capital at start	59
Capital at end	5
Number at start	2
Number at end	46
Lower at start	4
Lower at end	6
Passphrase	
Words	123
Random sequence	5

Based on the data collected from the login assessment, most users already use a password format where the first character is an uppercase letter and the last character is a number. In terms of passphrases, words seem to be used more than a random sequence of lowercase letters.

Although this strategy reduces security as it makes a user’s password collection more predictable when it comes to guessing the password, it improves usability by providing assistance in memorising the password. The keystroke dynamics algorithm can cover

the loss in terms of security if this approach is used by a user to assist password memorisation.

Impact on sub-models and proposed model: As a result of the keystroke dynamics algorithm, it is suggested that users be allowed to maintain common segments across passwords. Therefore, the construct, “common parts of the password can be the same” was maintained in the memory sub-model.

8.4.2.5 User-generated Passphrases

Since the login assessment used user-generated passwords, this section assesses whether user-generated passphrases are more user friendly than user-generated passwords. Figure 8-8 shows the number of password percentage failures and the number of passphrase percentage failures. The x-axis indicates the login failure types and the y-axis displays the percentage of login instances. A total of 236/1335 password login failures occurred, and 83/1233 passphrase failures were made by participants.

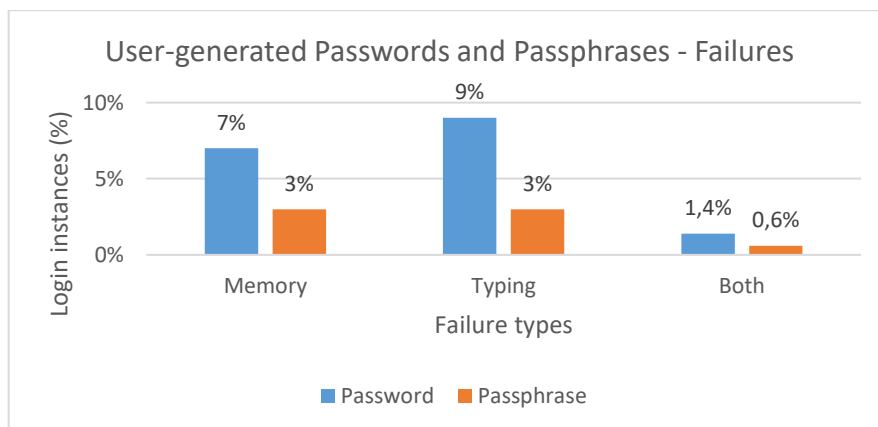


Figure 8-8: User-generated Passwords and Passphrases

Typing seems to be more of an issue with passwords than memorisation. This is probably due to all the character switching required to enter a password. In terms of passphrases, there is little difference between typing and memory failures. When comparing the failures of passwords with passphrases, passwords have a significantly higher failure rate in all three failure type categories (memory, typing and both). The data illustrates that a user has a higher probability of success using a user-generated passphrase than a user-generated password.

Impact on sub-models and proposed model: Although the login assessment experiment could not identify whether user-generated passwords are more successful than system-generated passwords, the data did show that the former has a higher

probability of login success than the latter. Therefore, there is no change to the “user-generated passphrase” construct in the memory sub-model.

8.4.2.6 Simple Usernames

Usernames have an impact on memorisation as users tend to use memory capacity for the username as well as for the password (Adams et al., 1997; Khodadadi, Islam, Baharun, & Komaki, 2016). The username should aim to place the least amount of strain on user memory. Therefore, a username should be something common that all users should have at their fingertips, for example their email address. For this reason, the username requested for the login assessment was the participants’ email address. This ensured that less memory capacity was required to recall the username. The username should not merely use the username as the email address but should also indicate this fact explicitly; this makes it as easy as possible for the user to remember that the username is their email address. Figure 8-9 illustrates an example of this approach using a screenshot from the login assessment screen.

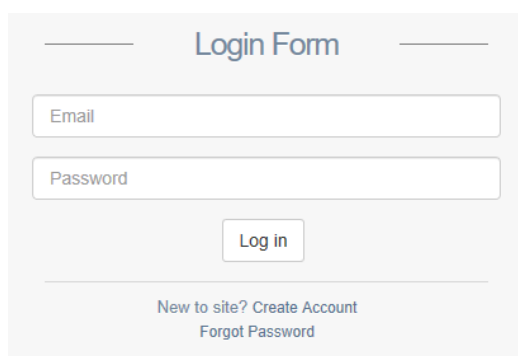
A screenshot of a login form titled "Login Form". The form is centered on a light gray background. It contains two input fields: "Email" and "Password". Below the "Password" field is a "Log in" button. At the bottom of the form, there are two links: "New to site? Create Account" and "Forgot Password".

Figure 8-9: Simple Username

Figure 8-9 is a screenshot of the login page presented to participants in the login assessment experiment. As depicted in Figure 8-9, the username was presented as an email address field. This ensured that any memory login failures related only to the password or passphrase and not to the username. Note, if a user has more than one email address, user memory capacity may be slightly affected because the user has to remember which email address they used for the particular website/system and this may also affect usability. Nevertheless, indicating to a user that the username is their email address is better for reducing user memory capacity strain than simply presenting the user with a username field.

Impact on sub-models and proposed model: *No change is made to the “simple username” construct in the memory sub-model. This section provided an example of a*

username that reduces the need for users to use additional memory capacity to recall their username.

8.4.2.7 Lenient Password Policy

It was recommended in Chapter 4 that password policies be made more lenient by allowing passphrases to be used. However, before creating leniency by proposing the use of passphrases, it had to be ascertained that passphrases would improve usability. Figure 8-10 shows all passwords and passphrases that exceed the password policy and indicates the number of percentage failures associated with these instances. The x-axis indicates the login failure types and the y-axis shows the percentage of login instances. A total of 236/1335 password login failures were made and 83/1233 passphrase failures were made by participants.

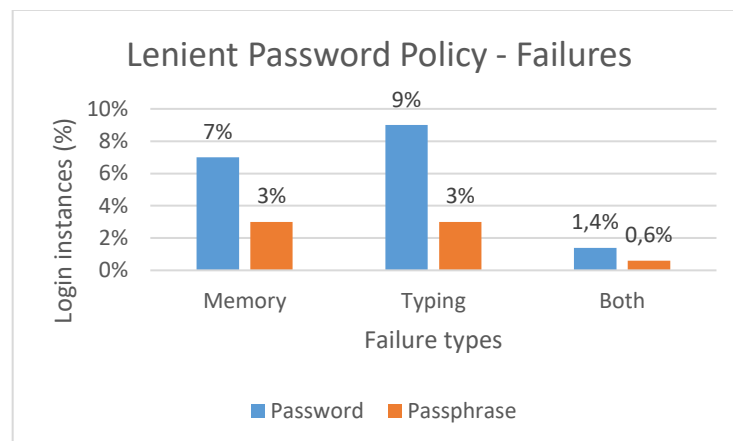


Figure 8-10: Lenient Password Policy

Figure 8-10 indicates that compared to passphrases, passwords had significantly more failures of all types: memory, typing and both (memory and typing). From a general perspective, passphrases have a lower probability of failure than passwords, regardless of the type of failure. Based on this interpretation, it is recommended that users be allowed to use passphrases for text-based authentication.

Impact on sub-models and proposed model: The “lenient password policy” construct is maintained in the memory sub-model to allow for passphrases to be created by users to improve usability.

8.4.2.8 Strength Indicator and Password Policy Alignment

Carstens et al. (2014), Furnell (2007) and Golla and Dürmuth (2018) found that systems have inconsistent strength indicators for passwords. For example, one system may indicate that the password is strong while another indicates that the same password is

weak. Althubaiti (2017) and Bhivgade et al. (2014) found that users are interested in the rules behind the strength indicators and if the strength indicator is not aligned to the password policy, users may spend time trying to understand how it works. Adams and Sasse (1999) and Woo and Mirkovic (2018) found that users are not well informed on how to create strong passwords, so it is important to explain how the strength meter works. Figure 8-11 gives an example on the login assessment where users were given an explanation of how strength meters' work.

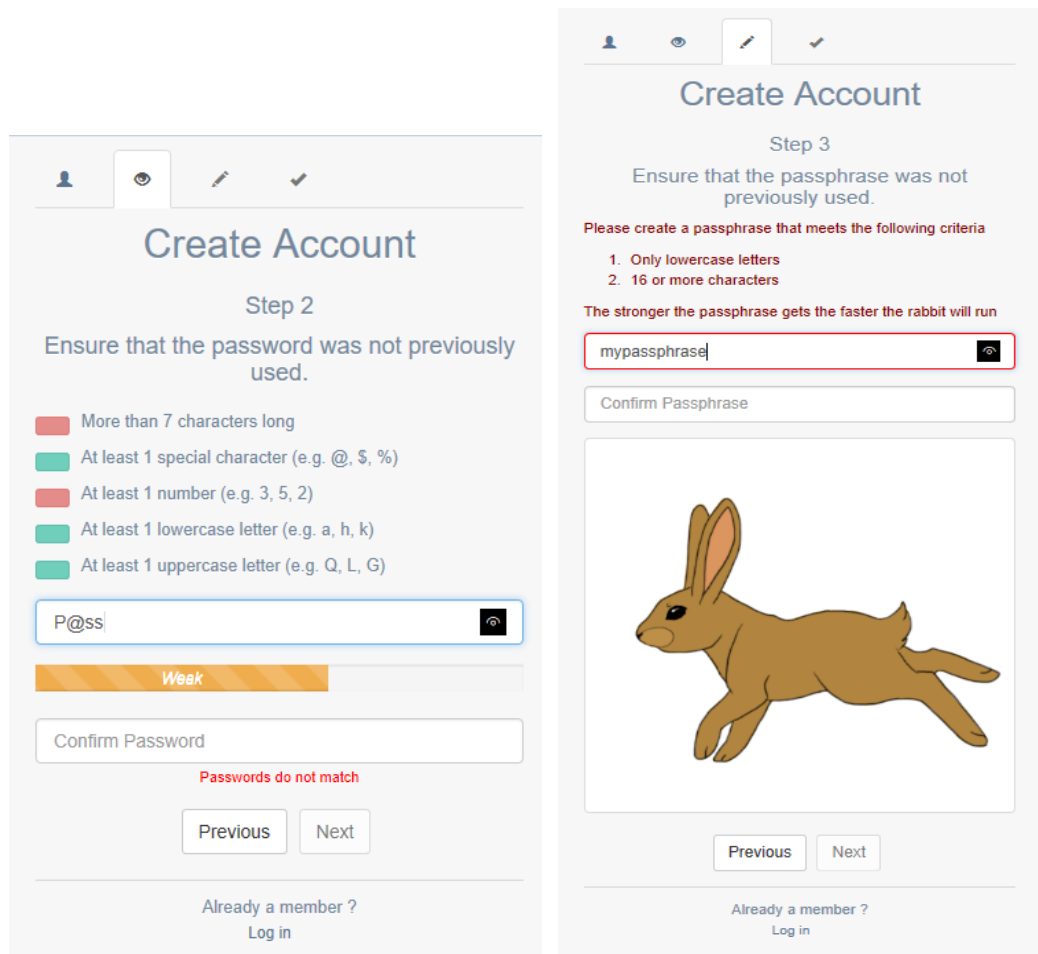


Figure 8-11: Strength Indicator and Password Policy Alignment

Figure 8-11 indicates how the password policy aligns to the strength indicators. The image on the left is a screenshot of the login assessment screen for password creation and the screenshot on the right is the passphrase creation screen from the login assessment. When looking at the password screen in Figure 8-11, the password policy rules are located above the password fields and the strength indicator is located between the password fields. As the user creates a password the password policy simultaneously indicates what rules have been met, while at the same time the strength indicator

increases as the password policy rules are complied with. Regarding the passphrase screenshot on the right, the passphrase policy is located above the passphrase fields. An explanation of the passphrase strength indicator is located below the passphrase policy, so the user understands the logic of the strength indicator.

Impact on sub-models and proposed model: *No change is made to the “align strength indicator with password policy” construct in the memory sub-model. This section provided an example of how the strength indicators are aligned to the password policy and provided to the user, so they understand what makes their password strong.*

8.4.2.9 Application of Chunking

Wright et al. (2012) found that the participants of their study did not apply the chunking concept correctly, as reasonable human limitations in terms of memory need to be considered (Sasse, 2003). This section assesses all passwords and passphrases collected from the login assessment and calculates the maximum number of chunks used for each. This assessment was performed by looking at the number of special characters, the number of digits, the number of uppercase letters, as well as the character length of both passwords and passphrases. Figure 8-12 below provides the results. The x-axis provides the chunk count for passwords/passphrase and the y-axis indicates the percentage of login instances. In total 236/1335 password login failures were made by participants. A total of 83/1233 passphrase failures were made by participants.

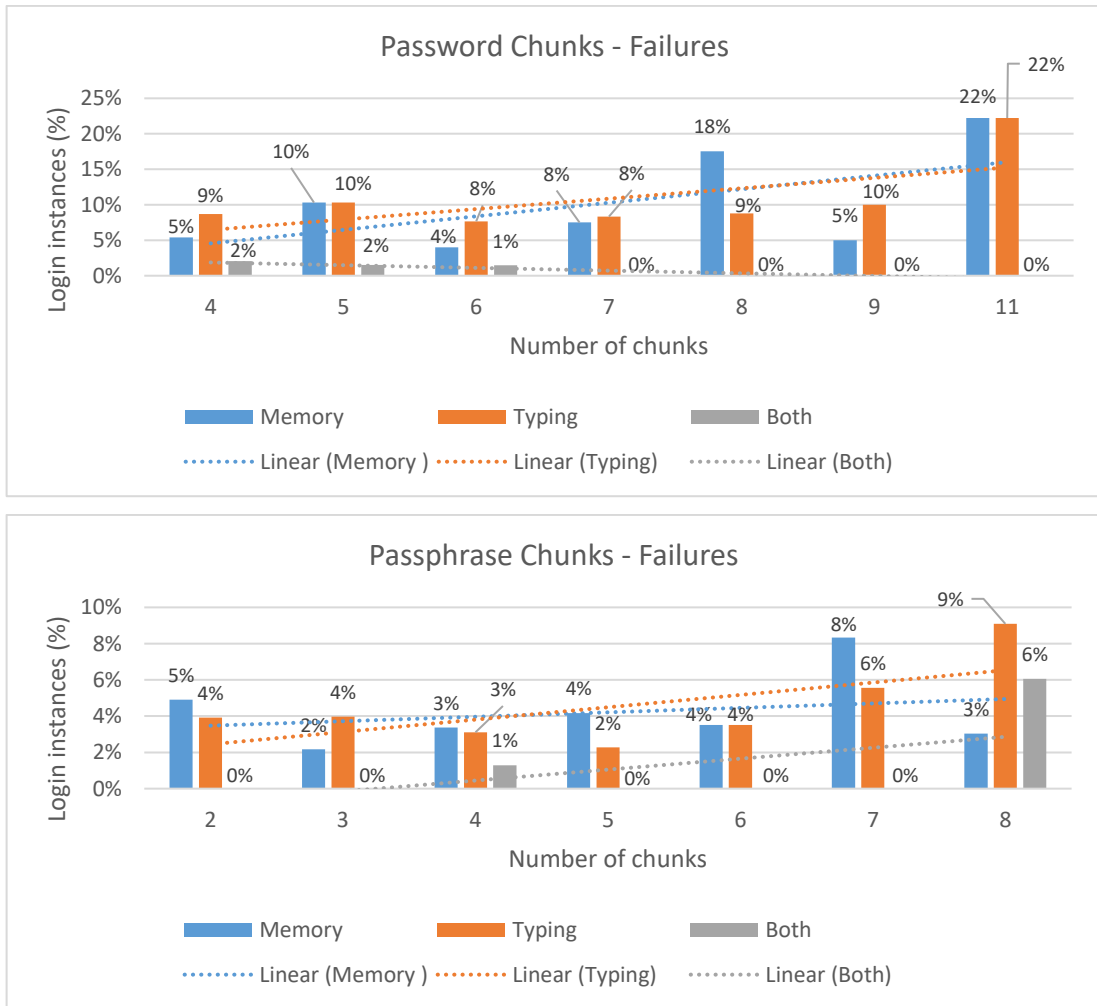


Figure 8-12: Application of Chunking

In Figure 8-12, the password failure chart and the passphrase failure chart indicates that as the number of chunks increase, the risk of memorability failure increases. This aligns with the Chunking theory (Goorha & Potts, 2019; Miller, 1956). Any passphrase over six chunks will increase the risk of memory failure. Therefore, it is suggested that passphrases created do not exceed six chunks.

It should be noted that a convenience sample was used to collect the data to assess chunking and therefore, the results cannot be generalised. However, these results are still relevant and accurate. It is suggested that further chunking research on passwords and passphrases use a more generalisable sample to verify if the results remain consistent to the findings in this research.

Impact on sub-models and proposed model: The “application of chunking” construct in the memory sub-model was maintained as users seem to subconsciously apply the Chunking theory to a sufficient level when creating passwords and passphrases.

8.4.3 Typing Evaluation

This section includes six subsections. The first section provides the results and a discussion on unfamiliar typing sequences and their impact on login success. This is followed by a discussion on the number of keystrokes and its relation to login failures. The third section analyses the impact of using SMS language, acronyms and abbreviations in passwords and passphrases has on login failures. A discussion on keyboard visualisation and touch is then provided which is followed by a discussion of keyboard layout exposure and login failure.

8.4.3.1 Unfamiliar Sequence

Bošnjak and Brumen (2019); Rieger (2004) and Lin and Wu (2011) explain that if a user types an unfamiliar sequence of characters it requires more effort and increases the risk of typing errors. An example of a familiar sequence would be a user typing common words they are familiar with such as “tree” or “apple”. Staying with the same example, an unfamiliar sequence of characters would include typing “tree” as “Tr33” or “apple” as “@ppl3”. This section assesses unfamiliar sequences by identifying passwords in which characters were replaced with special characters or numbers and then checking the number of login failures that occurred because of this. A similar concept was applied to passphrases; however, since passphrases only include one character set, those that included non-word-type patterns were identified as unfamiliar sequences. For example; “kdjenwmsajdiwkea” or “aaaaabbbbbccccddd” The results are displayed in Figure 8-13 below. The x-axis separates the type of failures and the y-axis indicates the percentage of login instances. A total of 69/301 login failures were identified for passwords with unfamiliar sequences and 10/62 login failures for passphrases with unfamiliar sequences. Note that only passwords and passphrases that had unfamiliar sequences were taken into consideration for Figure 8-13.

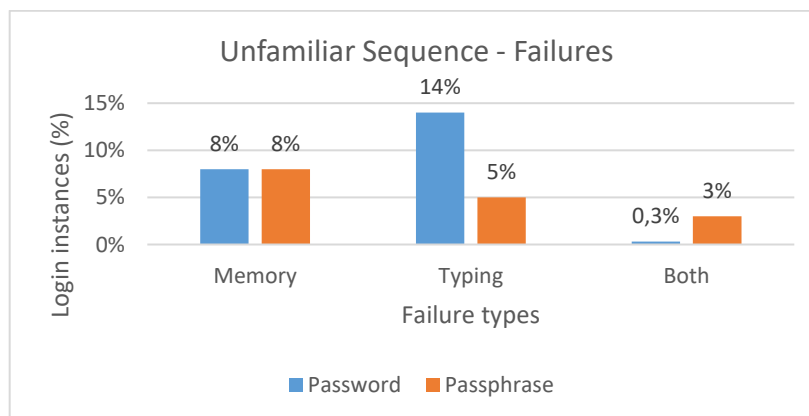


Figure 8-13: Unfamiliar Sequence

Figure 8-13 indicates that unfamiliar sequences for passwords does increase the risk of typographical errors, whereas unfamiliar sequences for passphrases does not have much of an impact in this regard. Accordingly, passphrase failures may be because an easy-to-type sequence has not been used; for example, a passphrase sequence such as “kdjenwmsajdiwkea” would give rise to more typographical errors than a passphrase sequence such as “qqqqqqqqqqqqqqqqq”.

Impact on sub-models and proposed model: The data presented in Figure 8-13 indicates that unfamiliar sequences should be avoided as they increase the risk of typographical failure. For this reason, the “unfamiliar sequence” construct in the typing sub-model was renamed to “avoid unfamiliar sequences”.

8.4.3.2 Number of Keystrokes

Keith et al. (2009) explain that the more keys that are pressed by the user, the higher the probability of an error occurring. Shay et al. (2016) explain that switching character sets increases the risk of typing a password incorrectly. The more character switching that occurs, the higher the risk of typing errors. Character switching was assessed by showing all passwords collected from the login assessment and indicating the number of character switches that are required to insert the password. Findlater et al. (2013) and Yazdi et al. (2019) add that typing errors also occur as a result of users switching between analogue devices and digital devices.

The correlation between the number of keystrokes and login failure was verified by looking at all the passwords and passphrases collected from the login assessment and counting the number of keystrokes required to insert the password and the passphrase. The number of typing login failures were mapped to these passwords and passphrases in order to see whether more keystrokes actually increased the risk of typographical errors. Note that character switching keys also had to be counted. For example, in order to type a special character, the shift key has to be held down on a typical desktop computer keyboard. Although a digital keyboard layout is different from an analogue keyboard, it is assumed that the same number of keys would have to be pressed to get a specific character. The findings are displayed in Figure 8-14 below. The x-axis indicates the number of keystrokes required to insert the password/passphrase and the y-axis shows the percentage of login instances. A total of 236/1335 password login failures were made and 83/1233 passphrase failures were made by participants.

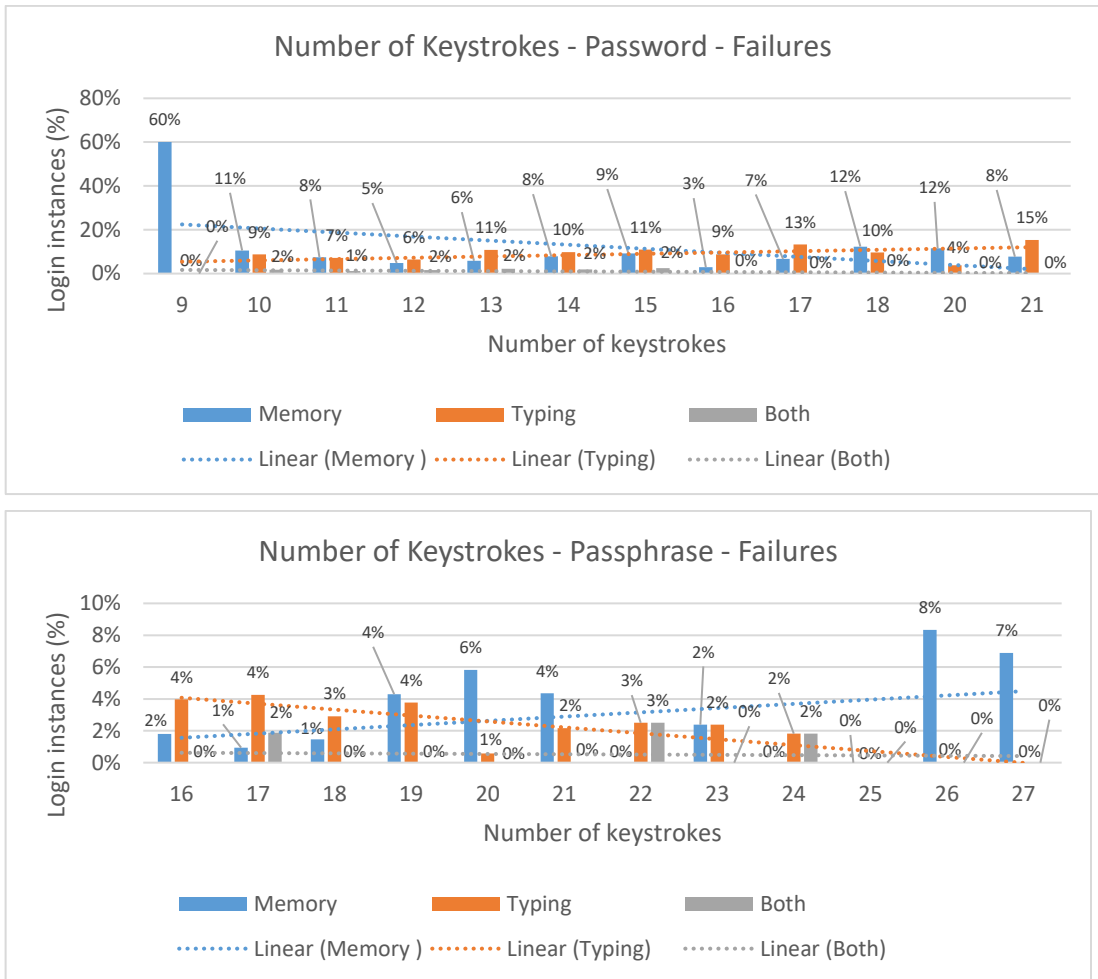


Figure 8-14: Number of Keystrokes

Figure 8-14 shows that the failure rate linked to typographical errors when entering passwords increases as the number of keystrokes increase. This aligns to the findings in Chapter 5 which explain that the more keystrokes made by the user, the greater the risk of typographical failures (Keith et al., 2009; Pidel & Neuhaus, 2019). The passphrase graph in Figure 8-14 illustrates a downward typographical trend line. Users are unlikely to exceed the passphrase policy significantly unless they choose to use a long phrase that is easy for them to remember that exceeds the passphrase policy. In this case, the user’s familiarity with the phrase they are typing may reduce the risk of typographical failure even though the phrase has a high total number of keystrokes.

Since the number of keystrokes does not affect memory, as user memory has more of an impact on character sets and character switching than merely the number of keystrokes, memory errors were not discussed in this section. The more user–keyboard interaction a user has, the more accurate the keystroke dynamic algorithm will become. However, if a user uses a password or passphrase that requires a higher than usual

number of keystrokes to be inserted, then usability may be affected as the risk of typographical errors increases.

Impact on sub-models and proposed model: *The data presented in Figure 8-14 indicates that there is a correlation between the number of keystrokes and login typographical failures. Therefore, the “number of keystrokes required to display a character” was maintained in the typing sub-model.*

8.4.3.3 SMS Language, Acronyms and Abbreviations

Forget and Biddle (2008) suggest that passphrases should be used however, they should be abbreviated to increase security as the phrase then becomes less common. However, these findings contradict the Chunking theory. If the Chunking theory is applied to this recommendation, then additional chunks are required to remember how the passphrase was abbreviated. Bošnjak, Sreš, and Brumen (2018) and Mahapatra and Magesh (2015) add that common acronyms should be avoided in passwords as the majority of them are included in dictionary attacks. This includes common “SMS” language (e.g. luv, sori, u, and lol) which can be monitored on public forums, blogs and social networks and then added to a dictionary attack list (Bošnjak, Sreš, & Brumen, 2018; Saevanee et al., 2011). That being said, uncommon acronyms and SMS language can be advantageous to the user as they save time typing and, if used frequently, may lead to less effort and fewer errors. This then has a positive impact on security and usability.

This section aims to identify all abbreviated passphrases, number of common and uncommon acronyms and SMS language used. These are then cross referenced to login failures to determine whether these instances have an impact on usability. The results are provided in Figure 8-15 below. The x-axis displays the login failure types and the y-axis shows the percentage of login instances. A total of 11/115 password login failures were made and 11/85 passphrase failures were made by participants. Note that only passwords and passphrases that incorporated SMS language, acronyms and abbreviations were taken into consideration for Figure 8-15.

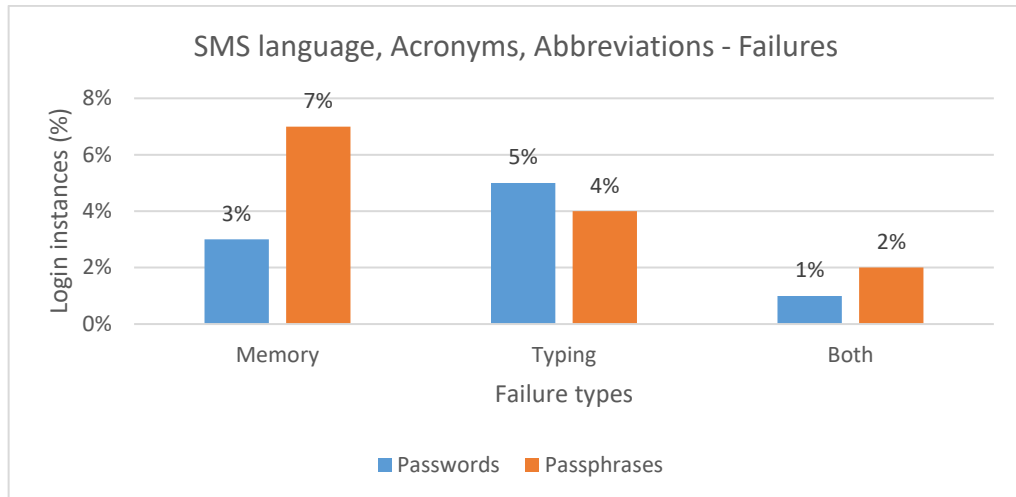


Figure 8-15: SMS Language, Acronyms and Abbreviations

Based on the findings presented in Figure 8-15, it is evident that abbreviated passphrases have a negative impact on usability, specifically memorability. In terms of abbreviated passwords, typographical errors are more likely to occur than memory failures. Although abbreviated passphrases may increase security, it is suggested that abbreviated passphrases be avoided as they have a negative impact on usability owing to the risk of memorability failures occurring. The reduction in security can be compensated for by the keystroke dynamics algorithm.

Impact on sub-models and proposed model: Based on the above data in Figure 8-15, the “SMS language” construct in the typing sub-model was renamed to “avoid SMS language, acronyms and abbreviations”.

8.4.3.4 Keyboard Visualisation and Touch

Kim et al. (2019) and Weiss et al. (2010) explain that in addition to being able to see the keys of a keyboard, a user's lack of touch can impair typing ability which may result in typing errors. Kim et al. (2014), Park and Han (2010) and Yazdi et al. (2019) supports Weiss et al. (2010), explaining that the risk of typographical login errors are higher on digital keyboards (common on cellphones and tablets) as opposed to analogue keyboards. Bi et al. (2014), Findlater and Wobbrock (2012) and Vertanen et al. (2015) argue that users who type on a cellphone touchscreen have less risk of errors as only two fingers (two thumbs) are usually used to interact with the digital keyboard. This was verified by counting the number of typographical login failures grouped by device used to login. This was applied to passwords and passphrases to determine what type of password supports different devices during the login process. Table 8-4 provides the number of login instances grouped by login device.

Table 8-4: Keyboard Visualisation and Touch

Device	Instances	Successful Logins	Typographical Errors
Computer			
Password	616	526 (85%)	90 (15%)
Passphrase	585	535 (92%)	50 (9%)
TOTAL	1201	1061 (88%)	140 (13%)
Phone			
Password	365	307 (84%)	58 (16%)
Passphrase	341	326 (96%)	15 (4%)
TOTAL	706	633 (90%)	73 (12%)
Tablet			
Password	354	266 (75%)	88 (25%)
Passphrase	307	289 (94%)	18 (6%)
TOTAL	661	555 (84%)	106 (19%)

When comparing the total typographical errors by device, Table 8-4 indicates that tablets have a higher probability of typographical login errors than computers and phones.

It was expected that computers would have the lowest typographical failures as this is the easiest to type on, out of the three devices. From the login assessment data collected, phones had a slightly lower failure rate than computers. It was also expected because phones have smaller screens than tablets, even though both are touch screens, phones will result in more login typographical errors than tablets. The proliferation of mobile phones in the last few years may be the reason why phones were found to have a lower typographical error rate than other devices. This may be because users interact the same amount or even more on their phones than they do on computers or tablets.

When comparing passwords and passphrases in terms of login devices, fewer failures with passphrases were indicated across all three devices. Tablets have the largest difference in login failure rates between passwords and passphrases, followed by phones and then computers. When assessing login device failures for passphrases, computers have a higher failure rate than phones and tablets, whereas with password login failure, tablets have a higher failure rate than phones and computers.

Impact on sub-models and proposed model: Although login failure is present on all three login devices, the data depicted in Table 8-4 indicates that passphrases have less login failures than passwords across all three device types due to less character switching. Therefore, the “keyboard visualisation and touch” construct in the typing sub-

model was maintained as the login device needs to be considered when deciding to use a password or passphrase.

8.4.3.5 Keyboard Layout Exposure

Typing behaviour may change due to lack of short-term exposure to a familiar keyboard layout (Giot et al., 2011; Kasiani & Yusuf, 2019). Switching between an analogue keyboard and a touchscreen keyboard may also create typing issues, for example the user may have used a computer in the previous login and then used a phone or tablet to login the next time. This section aims to verify whether this is correct. Any device switching was cross-referenced with login failures resulting from typing. The results are displayed in Table 8-5. Note that the results in Table 8-5 only consider device-switching login interactions. Therefore, a total of 54/283 password login failures were recorded from device switching and 14/254 login failures for passphrases.

Table 8-5: Keyboard Layout Exposure

Login device	Password				Passphrase				
	Inst-ances	Pass	Fail	% Fail	Inst-ances	Pass	Fail	% Fail	Inst-ances
Computer to phone	47	39	8	17%	40	37	3	8%	40
Computer to tablet	48	33	15	31%	37	35	2	5%	37
Phone to tablet	55	46	9	16%	53	48	5	9%	53
Phone to computer	43	34	9	21%	37	34	3	8%	37
Tablet to computer	48	45	3	6%	49	48	1	2%	49
Tablet to phone	42	32	10	24%	38	38	0	0%	38

Table 8-5 shows that the highest percentage of typing failures for passwords occurs when users switch from a computer to a tablet. This aligns to the expected result that switching between analogue and digital keyboards may increase the risk of login failure.

For passphrases, the highest typographical failure rate occurs when users switch from phone (digital) to tablet (digital). However, computer to phone and phone to computer are close to the highest login failure rate for passphrases. This covers three types of device switching; analogue to digital, digital to digital and digital to analogue. This inconsistency indicates that there is little risk of login failure resulting from device switching when using passphrases.

Login device switching does have a greater influence on passwords than passphrases. This needs to be considered when developing a keystroke dynamics algorithm.

***Impact on sub-models and proposed model:** Switching between analogue and digital keyboards does not seem to have much of an impact on passphrases. However, it was highlighted that even switching between one digital keyboard layout and another digital keyboard layout does affect the risk of failure. For this reason, the “keyboard layout exposure” construct was maintained, as it needs to be considered when developing the keystroke dynamics algorithm.*

8.4.4 General Observations

This section provides some interesting data collected by the login assessment experiment that does not have a direct influence on one or even several constructs in the sub-models or the proposed model. However, the data identified is interesting enough to be worth discussing and may have a significant influence on the sub-models or the proposed model.

This section is made up of ten subsections. The first section provides the average entropy of passwords and passphrases collected from the login assessment which is followed by an assessment on complicated passwords and subsequently an assessment of the use of similar passwords by users. This is followed by a discussion on password memory improvement over time and then login duration. Then an assessment on password and passphrase resets is provided followed by a discussion of the correlation between login failure, and user age and gender. The next section provides the general percentage of all failures for password and passphrases, broken down by failure type. The next section provides the results of the influence of login success when different languages are used in passwords and passphrases. Lastly, data on whether users prefer passwords or passphrases is provided.

8.4.4.1 Password and Passphrase Entropy

The passwords and passphrases created by the participants in the login assessment experiment were captured for assessment. The Shannon Entropy formula (Aguiar & Guedes, 2015; Arora et al., 2015; Shannon, 1948) was subsequently applied to the list of passwords and passphrases to determine their strength. The average entropy (measured in bits) of passwords and passphrases is provided in Figure 8-16 below. The x-axis displays the authentication type (passwords and passphrases) and the y-axis

shows the amount of entropy. A total of 158 passwords and 128 passphrases were used to calculate the average entropy for passwords and passphrases.

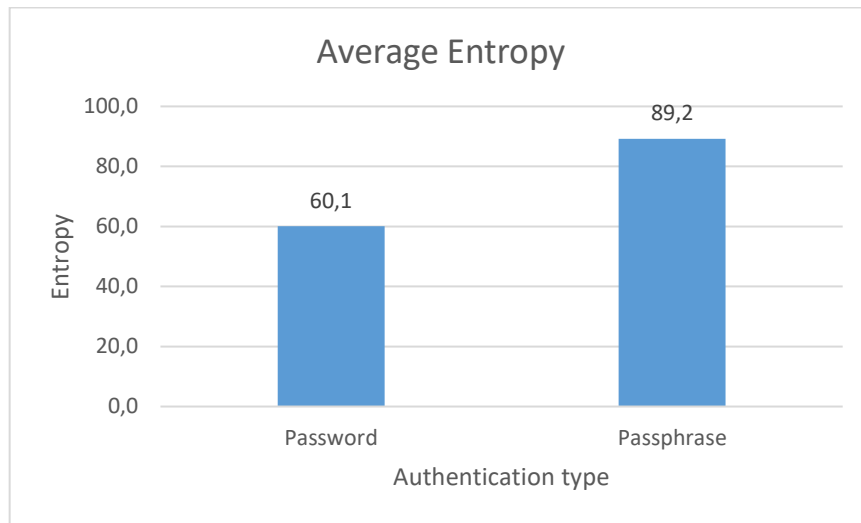


Figure 8-16: Password and Passphrase Entropy

As discussed in Chapter 3, the higher the entropy the more difficult it is for the password to be cracked. However, it should be noted that this is a rough estimate, as the Shannon Entropy formula was found not to consider certain anomalies that could limit one's ability to crack a password. Figure 8-16 indicates that the entropy of the average passphrases is higher than the average passwords. Therefore, it is suggested that passphrases be used by users as they are more secure than passwords.

Impact on sub-models and proposed model: *A recommended path for the implementation of user authentication security encourages using passphrases rather than passwords. Since this is the basis of the sub-models, no changes are required; however, the data in this section does confirm that user-generated passphrases are more secure than user-generated passwords.*

8.4.4.2 Complicated Passwords, Passphrases and Memory Issues

Complex passwords and passphrases increase the risk of a user failing to recall the password/passphrase from memory. For the purpose of this study a complicated password was deemed to be a password that involved more than the average amount of character switching, a password or passphrase that significantly exceeded the required length of a password and/or passwords/passphrases with a random character sequence (e.g. "hkwjcs" instead of "mypassword"). This list was then aligned to the number of failures arising from memory, typing or both. Figure 8-17 shows the findings from the assessment. The x-axis displays the login failure types and the y-axis shows

the percentage of login instances. Note only complicated passwords and passphrases were taken into consideration for Figure 8-15. A total of 43 complicated passwords and 46 complicated passphrases were identified. A total of 43/369 password login failures were created by participants and 28/417 passphrase login failures were created.

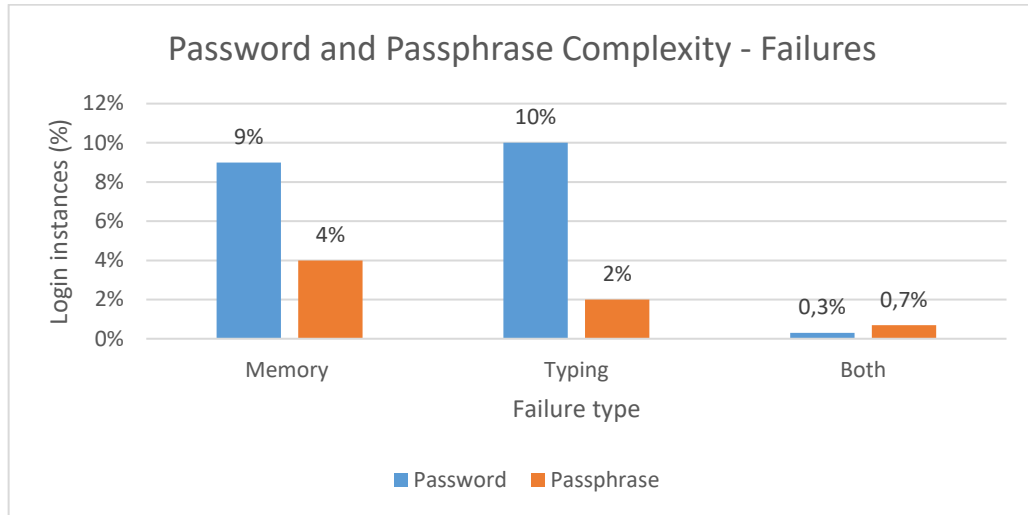


Figure 8-17: Password and Passphrase Complexity

Figure 8-17 shows that memory and typing errors were higher for passwords than passphrases. If users want to create a complicated password for the sake of having a password that is difficult to crack, it is suggested that they use a passphrase rather than a password. This assists usability.

Impact on sub-models and proposed model: No update is required to the sub-models. The findings on complicated passwords and passphrases indicate that complicated passphrases have a lower risk of errors than complicated passwords.

8.4.4.3 Similar Passwords and Passphrases Created by Users

Keith et al. (2009) and Renaud (2019) found that many users who do comply with the password changing rules, created different but similar passwords. This behaviour creates a security risk although it might assist usability. This section aims to identify passwords and passphrases that were created similar to other passwords or passphrases. These findings are displayed in Figure 8-18. The x-axis displays the login failure types and the y-axis shows the percentage of login instances. Note only similar passwords and passphrases were taken into consideration for Figure 8-18. For similar passwords and passwords (per user), there were 32/131 login failures. For similar passphrases and passphrases (per user), there were 3/23 login failures. When looking

at similar passwords and passphrases (per user), 20/234 login failures were made by participants.

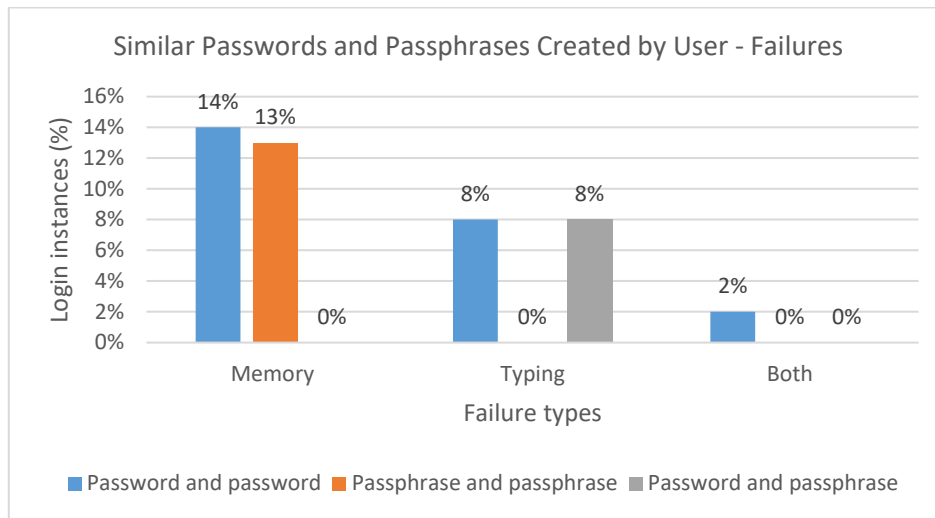


Figure 8-18: Similar Passwords and Passphrases Created by Users

When looking at similar passwords, it can be seen that these assist typing more than memory (i.e. recalling the password from memory). The same outcome is indicated for similar passphrases. However, when looking at similar passwords and passphrases, typing is affected more than memory. It was expected that a user's use of similar passwords and similar passphrases, would assist memorisation, however, the data collected indicated the opposite. This may be because it confuses users rather than helping them to remember the passwords and passphrases. For example, if passwords and passphrases are similar, users may have to use additional memory capacity to determine what parts of the passwords and passphrases are different. This may also explain why common password-passphrase combinations have fewer associated memory errors than typing errors, as there are no similarities to create user memory confusion. Since similar passwords and passphrases negatively affect both usability and security, it is suggested that similarity across passwords and passphrases should be avoided.

Impact on sub-models and proposed model: Since similar passwords and passphrases created by users have more of an impact on memory than typing, the memory sub-model was updated. A new construct was added under "password composition". The construct was labelled "avoid similar passwords/passphrases".

8.4.4.4 Password Memory Improvement Over Time

Forget and Biddle (2008), Carstens et al. (2014) and Woods and Siponen (2019) explain that, in light of the Chunking theory, the more frequently a password is used the more unlikely it is that the password will be forgotten. This is also applicable to passphrases. This section verifies this statement by counting the number of login failures for each user's login phase. For example; the total number of failures when user's logged into the system for the first time. Figure 8-19 provides the results. The x-axis shows the login instance and the y-axis shows the percentage of login failures.

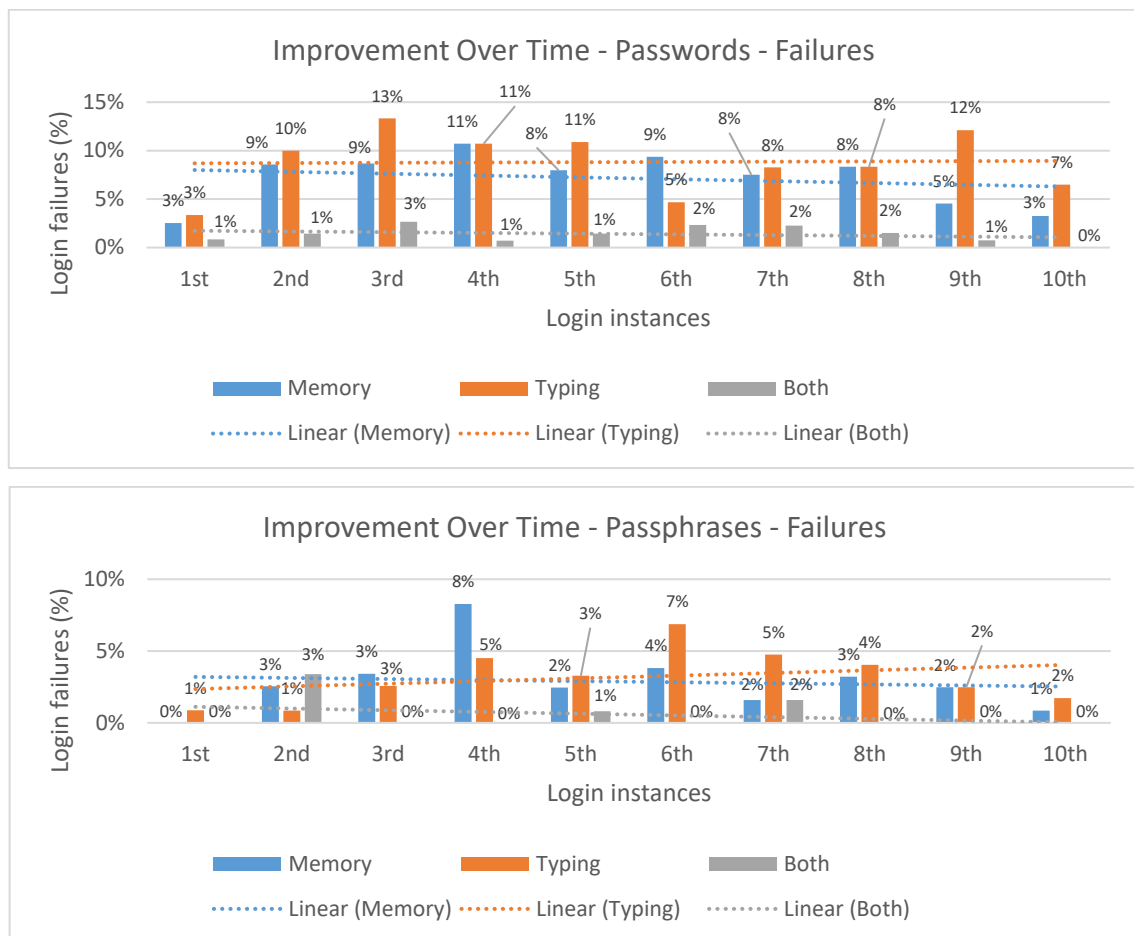


Figure 8-19: Improvement Over Time

When assessing the failure charts in Figure 8-19 for passwords, the downward trend line for memory indicates that login failure does in fact reduce, the more frequently the password is used. The same results are revealed in the passphrase graph with a downward memory trend line. The results for passwords and passphrases in terms of login frequency support the Chunking theory. The number of chunks required to memorise the password/passphrase reduces as the user gains more exposure to the password and the passphrase.

It is recommended that the length of time passwords and passphrases are used by users should be extended. The security risk of using the same password or passphrase for a long period of time can be overcome by using a keystroke dynamics algorithm. The keystroke dynamics algorithm also has an opportunity to become more accurate as the same password or passphrase will be used more times than expected.

Impact on sub-models and proposed model: *The suggestion of extending the lifespan of a passphrase is recommended to support memorability. Therefore, the memory sub-model was updated with a new construct under “password composition” which was labelled “extend passphrase lifespan”.*

8.4.4.5 Login Duration

This section has been separated into two subsections; days between logins and login time period.

8.4.4.5.1 Days between Logins

It is expected that less exposure to using a password or a passphrase will result in a higher rate of errors (Morimoto, Leyva, & Tula, 2018; Pereira, Taylor, & Jones, 2009). This risk increases the longer the period between login instances. These login instances (i.e. duration between logins) were used to determine whether short-term login breaks affect the memorisation and typing of passwords and passphrases. Choong (2014) and Strock et al. (2019) explains that this could have an impact on usability and needs to be considered for user authentication. Long-term memory is difficult to assess as it differs from person to person. Guo et al. (2013) and Strock et al. (2019) explain that the transition between short-term memory and long-term memory is based on an individual’s ability to create meaningful associations and complete the processes of rehearsal. For this reason, long-term memory assessment was avoided while different degrees of short-term memory were assessed. The results from data collected from the login assessment are provided in Figure 8-20 below.

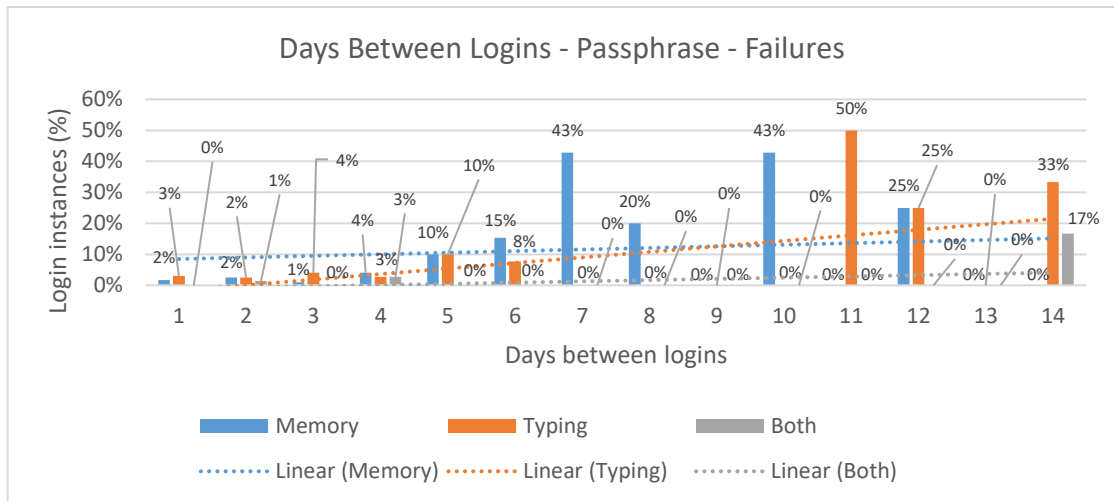
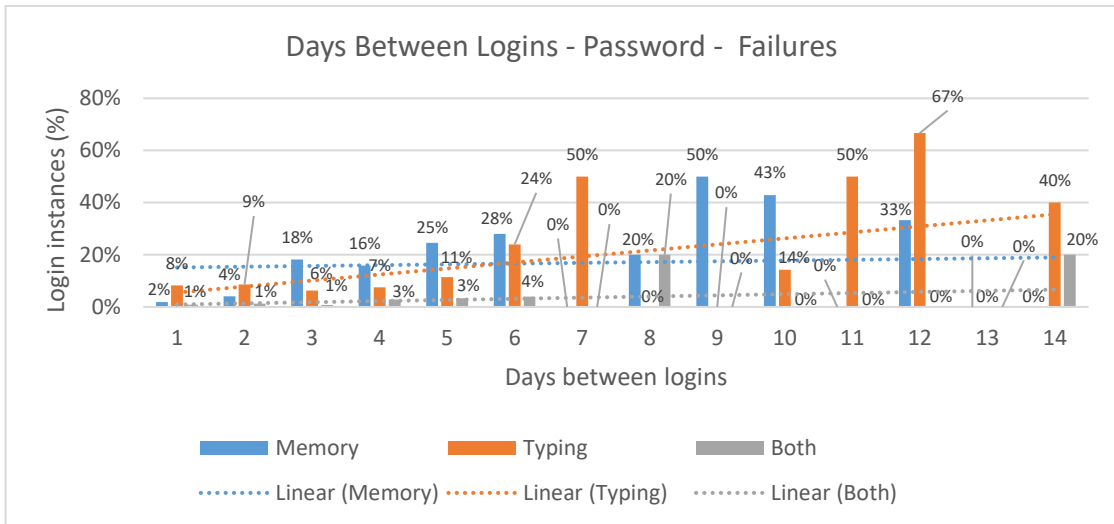


Figure 8-20: Days Between Logins

When analysing the failure charts in Figure 8-20, it is interesting to note that there is a higher risk of memory failure than typing failure as login duration increases. This is consistent for both passwords and passphrases. When looking at the memory and typing trend lines on the password and passphrase chart, it can be seen that passphrases have a higher risk of failure as opposed to passwords the longer the login duration becomes. However, when looking at login duration of less than seven days, passwords have a higher risk of failure than passphrases. In addition, typing becomes more of a risk for passwords than memory over time while passphrases have a consistently greater risk of user memory failure over typing failure.

Impact on sub-models and proposed model: *Passphrases support login failure caused by duration between logins more than passwords. Especially if user logins are more than a week apart, it is suggested that passphrases be used as opposed to passwords. Since the sub-models are considerations for implementing passphrases and*

the data in this section simply indicates that passphrases better support duration between logins than passwords; no update is necessary on the sub-models.

8.4.4.5.2 Login Time Period

Carstens et al. (2014) explain that fatigue needs to be considered with user authentication. Fatigue causes memory issues and may even cause the user to type the password or passphrase incorrectly. However, the impact of fatigue on memory and typing might not be strong enough to create login failures. This section aims to determine the magnitude of login failures arising from fatigue (see Figure 8-21). Login failures were organised into day time and night time periods. Day time was defined as the period from 06:00:00 to 17:59:59 and night time as 18:00:00 to 05:59:59. The x-axis indicates the login time period and the y-axis provides the percentage of login instances. There were 76/455 day login password failures and 160/880 night login password failures that occurred. For passphrases, 28/424 day login passphrase failures and 66/809 night login passphrase failures occurred.

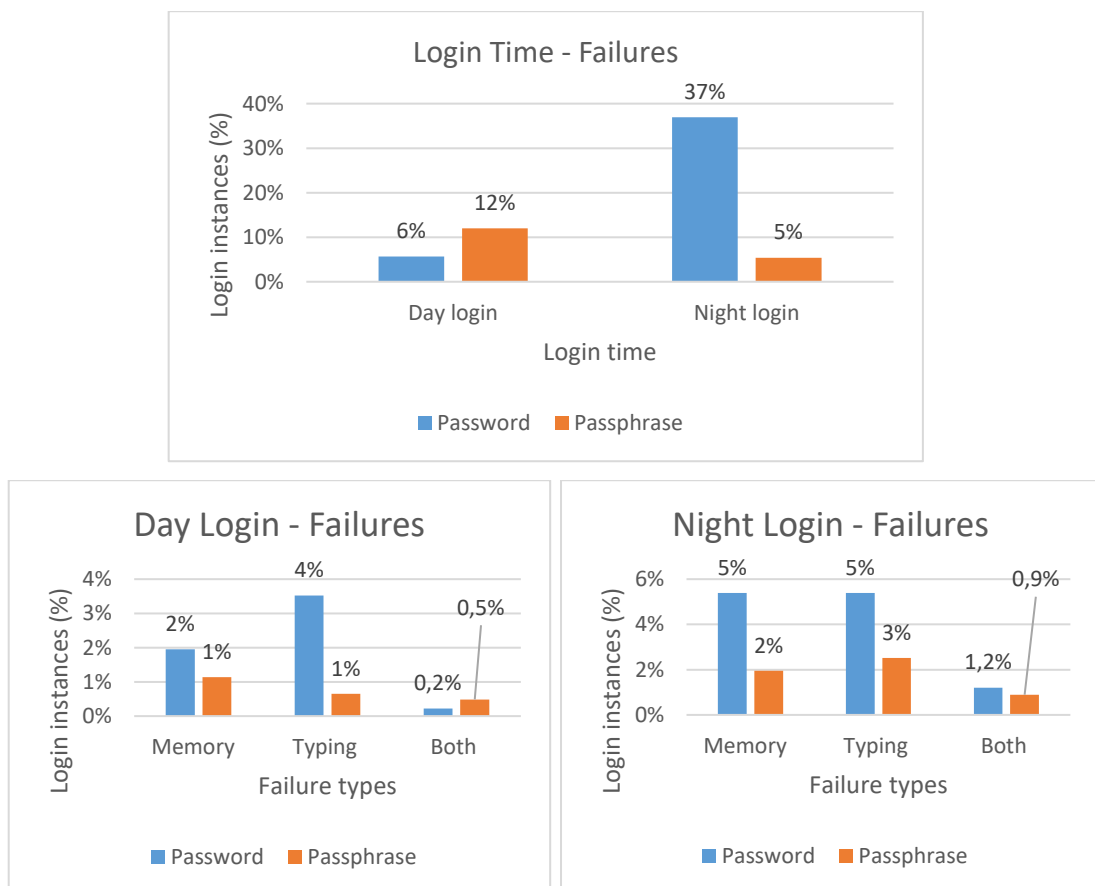


Figure 8-21: Login Time Period

In Figure 8-21 it can be seen that password and passphrase failure is higher during the night time period than the day. In addition, passwords have higher failure rates at night than passphrases. When looking at the types of failure that occur at night, password failures seem to result equally from memory and typing errors. Results for passphrases are similar but typing failures seem to be slightly higher than memory failures at night. These findings emphasise that fatigue may be responsible for higher failure rates during the night as opposed to the day.

The keystroke dynamics algorithm needs to accommodate login times by creating more lenient access during the evenings. This assumes that users have normal sleep patterns, for example this may be the opposite for users with night shift jobs. Country of login and daytime savings also have to be considered in the keystroke dynamics algorithm. *This is covered in the “leniency” attribute under the “keystroke dynamics” construct.*

There are some interesting points to consider when comparing the day time and night time failure: During the day, typing failures are more prominent in passwords while memory failures are higher for passphrases. Looking at both (memory and typing) failures, it can be seen that passphrase failure is higher than password failure during the day, while password failure is significantly higher at night than passphrase failure.

Impact on sub-models and proposed model: *Since user fatigue could cause memory or typing failures, both the memory sub-model and the typing sub-model needs to be updated. The memory sub-model is updated by adding a new construct under “user” called “user fatigue”. The same construct was added to the typing sub-model “errors and typing”.*

8.4.4.6 Password and Passphrase Resets

The literature chapters showed that the two most common login errors are caused by a user mistyping a password/passphrase or forgetting the password/passphrase. Usually, when a user mistypes the password/passphrase, they simply try to login again by attempting to retype the password/passphrase correctly. However, if a user forgets the password/passphrase, they might try to login once or twice, before resetting the password/passphrase. It was found in Chapter 5 that passphrases are easier to type than passwords and, therefore, a user may attempt to retype a passphrase more times than a password before selecting the password/passphrase reset option. This section aims to identify the number of login attempts a user will make before selecting the forget

password option. The intention was to find out whether users will attempt to retype passphrases more than passwords. The total number of password resets and passphrase resets is also provided. The data is graphically presented in Figure 8-22. The x-axis shows the authentication type and the y-axis provides the percentage of login instances. There were 48 password resets and 236 password failures and 17 passphrase resets and 83 passphrase failures.

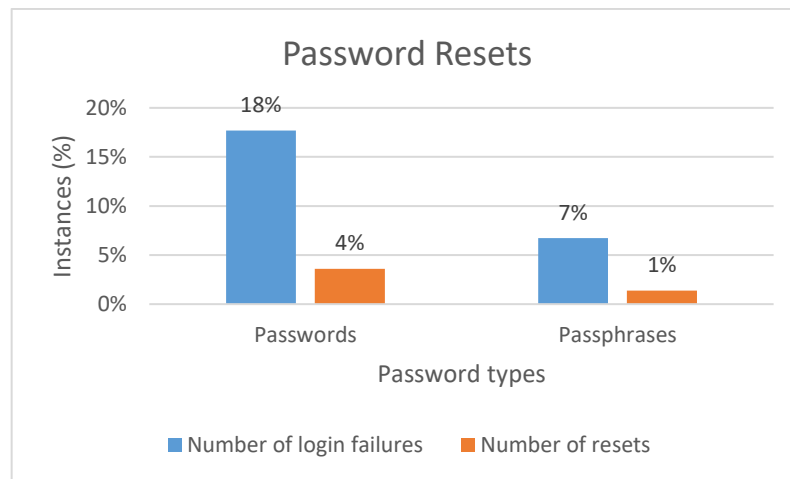


Figure 8-22: Password and Passphrase Rests

It was expected that since passphrases are easier to type than passwords, there would be more failed attempts before passphrases are reset. In other words, a failed passphrase would be retried more than a failed password before the user selects the reset option. However, when assessing the data from the login assessment experiment, there is a 1:4.9 (236 (18%) divided by 48 (4%)) possibility of a password reset and a 1:4.8 (83 (7%) divided by 17 (1%)) possibility for passphrases. In other words, a user will try an average of four to five times before resetting their password or passphrase. This emphasises that on average, users retry a password or passphrase the same number of time before selecting the reset option. Note that some users may select the reset option after two tries while other users may try three or four times.

It was expected that passphrase retries would be higher than password retries before a reset occurs. If this were true, it would have been recommended that the limit on failed login attempts on a system be increased to cater for passphrase login failure. The rationale behind this recommendation was that since passphrases are easier to type, users should be allowed to retype the passphrase more times than conventional passwords. Based on the login assessment findings discussed in the paragraph above, it is not necessary to increase the number of login attempts to cater for passphrases.

This does not affect usability and maintains the same level of security, as users have a limited number of attempts to re-enter their password or passphrase.

Impact on sub-models and proposed model: No updates required to any of the sub-models.

8.4.4.7 User Age and Gender

In this section, user age in relation to login instances is discussed. The next section relates to user age, login device and login instances. The last section displays the results of gender in relation to login instances.

8.4.4.7.1 User Age and Failure Type

Participants were organised into two main age groups. Participants in the age groups 18–25 and 26–33 years were referred to in this study as the young age group. Participants in the age groups 34–50 and 51+ years were referred to as the older age group. With the advances in technology over the past years, users have been more exposed to alternative forms of user authentication (Gao et al., 2018; Payne & Edwards, 2008); for example, pattern recognition, fingerprint unlock, and image area selection. This has resulted in users having less exposure to text-based authentication as this type of authentication was more prolific in the past. It was therefore assumed that young participants might struggle to recall and type the text-based authentications more than the older group due to less exposure. The results from the login are displayed in Table 8-6.

Table 8-6: User Age and Failure Types

Age	Success	Failures				Login Instances
		Memory	Typing	Both	Total	
18 - 25						
Password	193	18 (7%)	25 (10%)	5 (2%)	48 (20%)	241
Passphrase	204	10 (5%)	7 (3%)	0 (0%)	17 (8%)	221
26 - 33						
Password	229	24 (8%)	29 (10%)	4 (1%)	57 (20%)	286
Passphrase	246	6 (2%)	9 (3%)	4 (2%)	19 (7%)	265
34 - 50						
Password	569	36 (5%)	55 (8%)	8 (1%)	99 (15%)	668
Passphrase	578	15 (2%)	19 (3%)	3 (0%)	37 (6%)	615
50+						
Password	108	19 (14%)	11 (8%)	2 (1%)	32 (23%)	140
Passphrase	111	5 (4%)	5 (4%)	0 (0%)	10 (8%)	121
TOTAL	2238	133 (5%)	160 (6%)	26 (1%)	319 (12%)	2238

Age groups 18–25 and 26–33 show that younger users struggled more with typing passwords than any other failure instance when considering passwords and passphrases interaction. When looking at the older user groups (age 34–50 and 51+), typing failures seem to be an issue which gradually becomes more of a memory failure issue when moving more toward the over 50 age group. This is consistent across passwords and passphrases. Typing failure may be high due to the age group’s interaction with the website through a touch screen device. This aligns to the data in login device failures presented in section 8.4.4.7.2, User Age and Login Device.

It was expected that older users would struggle with the adoption of passphrases due to comfortability/familiarity with passwords. However, the data collected from the login assessment indicates that all user groups had a higher failure rate with passwords than passphrases.

Impact on sub-models and proposed model: Since age does not have an impact on passphrase failure, no updates to the sub-models are necessary.

8.4.4.7.2 User Age and Login Device

Findlater et al. (2013) and Rocha, Carneiro, and Novais (2019) conducted a study on touch screen usability and found that older users had more usability issues than younger users. They continued to explain that older users grew up in an era where only analogue keyboards were available. Therefore, younger users had a lower level of errors on

touchscreen keyboards than older users. This finding remained true even though overall exposure to keyboards was higher with older user's than younger users who have been exposed to touchscreen keyboards. The results are shown in Table 8-7.

Table 8-7: User Age and Login Device Failures

Age	Success	Failure				Login Instances
		Computer	Tablet	Phone	TOTAL	
18 - 25						
Password	193 (87%)	12 (5%)	5 (2%)	13 (6%)	30 (13%)	223
Passphrase	204 (97%)	2 (1%)	1 (0%)	4 (2%)	7 (3%)	211
26 - 33						
Password	229 (87%)	16 (6%)	5 (2%)	12 (5%)	33 (13%)	262
Passphrase	246 (95%)	5 (2%)	1 (0%)	7 (3%)	13 (5%)	259
34 - 50						
Password	569 (90%)	16 (3%)	15 (2%)	32 (5%)	63 (10%)	632
Passphrase	578 (96%)	10 (2%)	4 (1%)	8 (1%)	22 (4%)	600
50+						
Password	108 (89%)	8 (7%)	2 (2%)	3 (2%)	13 (11%)	121
Passphrase	111 (96%)	1 (1%)	1 (1%)	3 (3%)	5 (4%)	116
TOTAL	2238 (92%)	70 (3%)	34 (1%)	82 (3%)	186 (8%)	2424

When assessing login failures by device and age group, it can be seen that all users, regardless of age, have more login success with passphrases than passwords. Note, this data is specific to login device and typographical login failures. There does not seem to be any variation between age groups and login devices. In terms of user age and login device, the highest typographical failures by user age groups came from password computer logins and password phone logins.

When looking at total password login failures, it seems that older users (age 34 and older) have a higher rate of success than younger users (age 33 and younger), while passphrase login failures seem to be balanced between older and younger user groups. Based on the data in Table 8-4, user age does not have an impact on passphrases and login devices.

Impact on sub-models and proposed model: *Since passphrases and login devices do not have an impact on login failures, no updates are required to the sub-models as this aspect does not need to be considered from a usability perspective.*

8.4.4.7.3 User Gender

It is expected that gender would not influence memory or typing abilities. See the results of this assessment in Figure 8-23 below. The x-axis provides the login failure types and the y-axis indicates the percentage of login instances. A total of 135/1335 password failures and 43/1233 passphrase failures occurred for males and 102/1335 password failures and 39/1233 passphrase failures occurred for females.

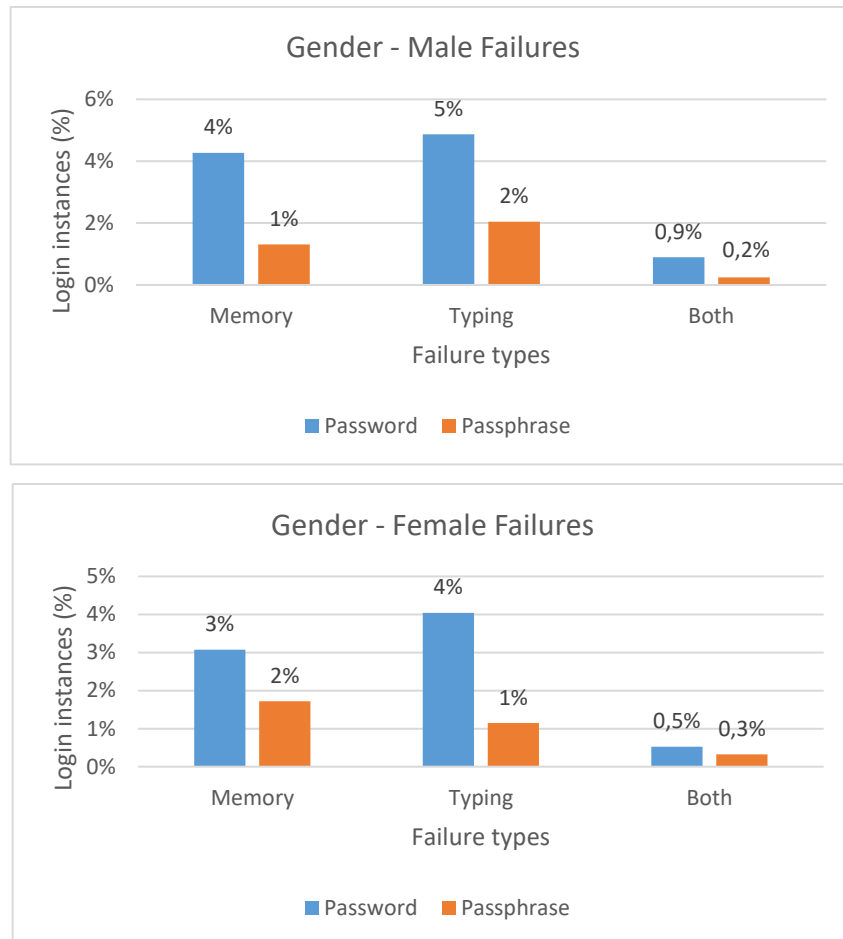


Figure 8-23: User Gender

The correlation between password failures and passphrase failures between male and female users seems to be roughly the same. Male and female users have more password failures than passphrase failures. To elaborate more on password failures, both male and female users have more password failures from typing than memory or both (memory and typing). Based on the data, there seems to be one difference in terms of gender. Males seem to have more passphrase typing failures than passphrase memory failures, while females have more passphrase memory failures than passphrase typing failures. The data collected is limited and so cannot explain the reason for the

difference; however, this does illustrate that both memory and typing issues need to be addressed as user groups usually include both male and female users.

Impact on sub-models and proposed model: No update is required to the sub-models as user gender cannot be avoided in terms of the difference in type of login failure (memory or typing). However, the data presented in Figure 8-23 indicates that passphrases have a higher rate of login success than passwords, regardless of gender.

8.4.4.8 Password and Passphrase Failures

This section simply provides the number of login failures broken down by memory, typing and both, for passwords and passphrases. The intention of this section is to discover, in general, whether passphrases are more user friendly than passwords. The possible reasons explaining the results are absent from this report. See the results in Figure 8-24 below. The x-axis shows the failure types and the y-axis indicates the percentage of login failures. A total of 236/1335 password login failures were made and 83/1233 passphrase failures were made by participants.

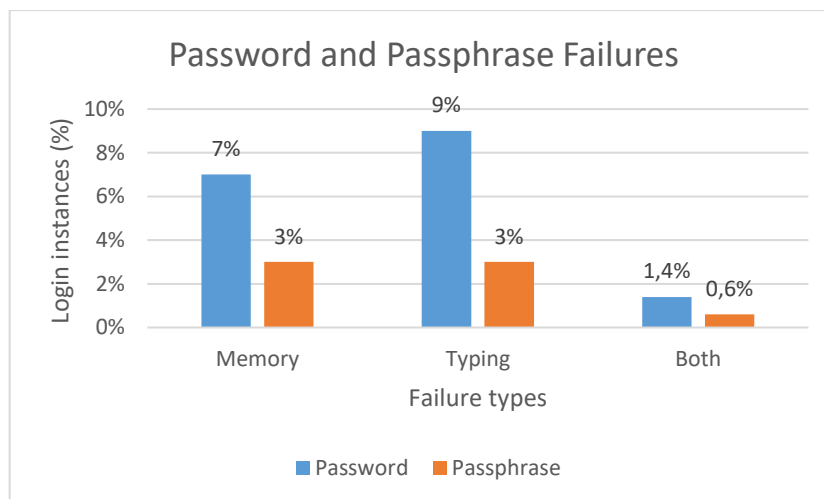


Figure 8-24: Password and Passphrase Failures

Figure 8-24 illustrates that passphrases have fewer failures than passwords in all three failure-type categories. Passphrase failures also seem to be balanced between memory failures and typing failures, while passwords have slightly more typing failures than memory errors in total. Nonetheless, Figure 8-24 depicts that memory and typing needs to be considered for passphrase usage.

Impact on sub-models and proposed model: No updates are required to the sub-models. This confirms the importance of considering the both the memory sub-model and the typing sub-model to address usability.

8.4.4.9 Language in Passwords and Passphrases

Bošnjak, Sreš, and Brumen (2018) and Clark and Arakia (2011) warn that dictionary attacks do not only include English words but also common acronyms and SMS language words (e.g. “gr8”). In Chapter 5, it was found that multiple languages can be mixed with different acronyms to further increase security. This section aims to identify all passwords and passphrases where other languages besides English were used. These passwords and passphrases were then matched to the login failures due to typing and memory to determine whether different languages improve user authentication usability. A total of 10 passwords and 8 passphrases were identified as having different languages beside English. There were 25/156 password login failures and 10/152 passphrase login failures. The results are provided in Figure 8-25.

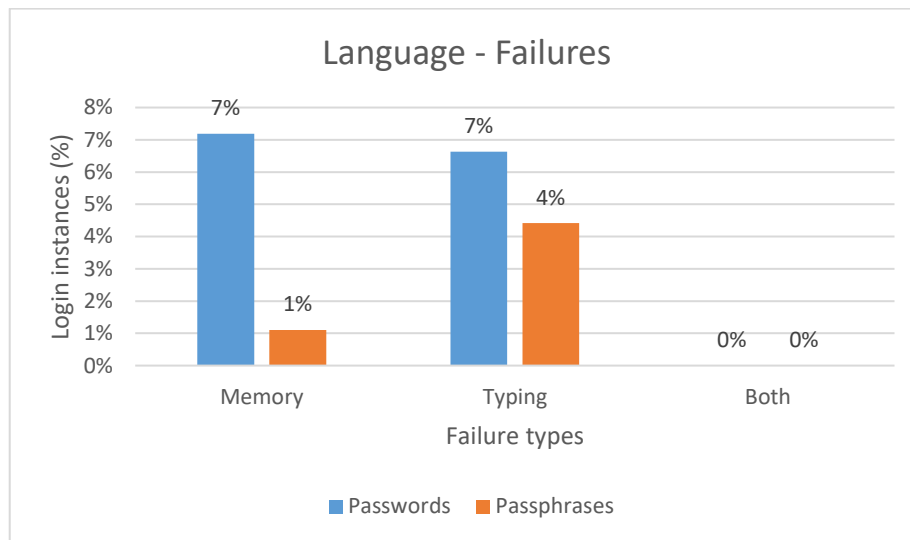


Figure 8-25: Language in Passwords and Passphrases

The data collected from the login assessment did not indicate many non-English passphrases as opposed to passwords. This may be because passphrases are a new concept and most participants decided to stick to their preferred language, in this case English. Figure 8-25 shows that password failures are equally spread between memory and typing failures for passwords. Language in passphrases does seem to affect memorisation as much as typing. Based on the data collected on using different languages besides English in passwords and passphrases would seem to indicate that passphrases are better equipped to avoid login failures than passwords when different languages are involved.

Impact on sub-models and proposed model: A new construct, “different languages” was added to the memory sub-model, under “password composition”.

8.4.4.10 User Preference

After each login, participants had to complete a short questionnaire so that the researcher could get a better understanding of the user–interface interaction. One of the questions presented to participants asked the user if they preferred using passwords, passphrases or had no preference. The results of this question are presented in Figure 8-26. A change in preference as the login assessment progressed is also indicated in Figure 8-26. A total of 170 preference changes were made by users throughout the course of the login assessment.

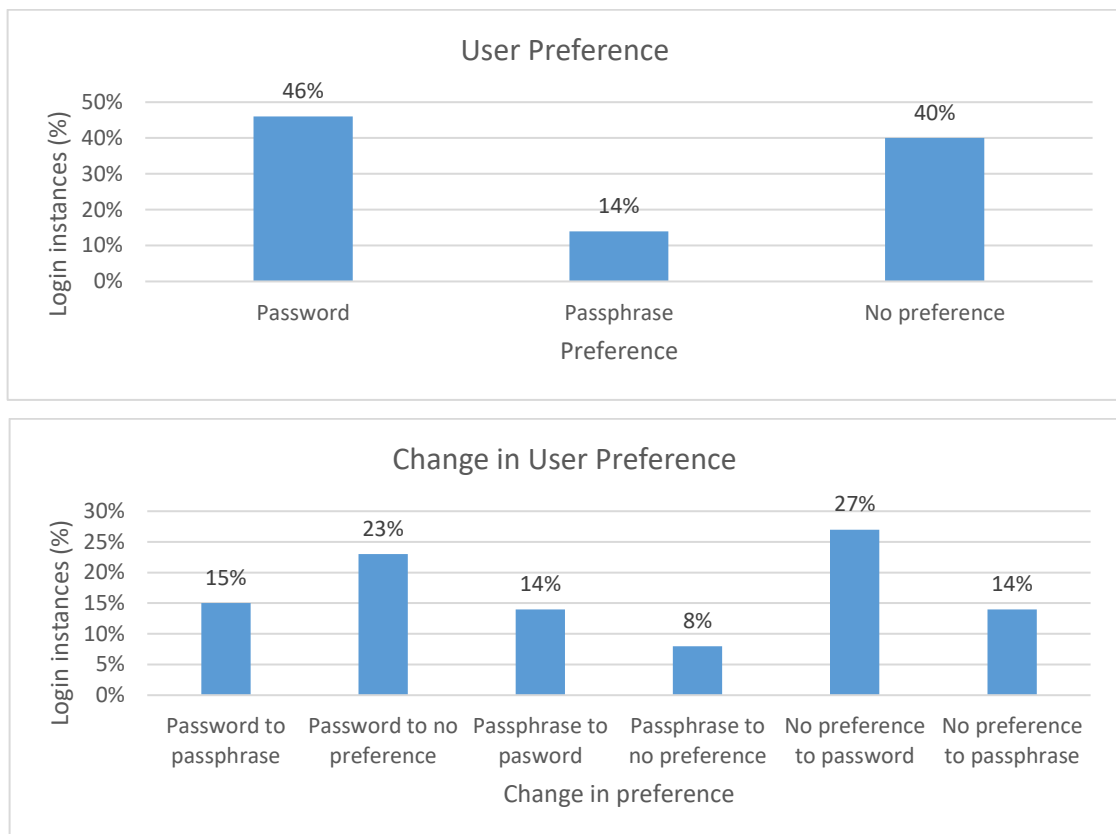


Figure 8-26: User Preference

The data from the login assessment displayed in one of the preceding sections shows that passwords have a higher rate of failure than passphrases. This is a significant finding since most users do not use passphrases in the normal course of their lives. In terms of user preference, even though the majority of users that participated in the login assessment experiment had more login success with passphrases than passwords, Figure 8-26 indicates that user preference is low for passphrases. Users seem to either have no preference or prefer passwords. The graph that depicts the number of changes in preference throughout the login assessment experiment shows that the majority of users switched from no preference to passwords or passwords to no preference. The

reasoning behind this may be because users perceive passphrases to be less secure or need some time (through frequency of passphrase use) to adjust to a move away from the use of conventional passwords.

Impact on sub-models and proposed model: *It is suggested that users should be educated on the benefits of passphrases and that adoption is encouraged. A “change control” construct was added which included two attributes; “education” and “adoption” attributes.*

8.4.5 Summary of Results

The findings and discussion in Section 8.4.1 to 8.4.4 above are summarised in Table 8-8 below. The “change” column provides the update to the proposed model based on respective data from the login assessment.

Table 8-8: Summary of Results

Section	Findings description	Change
Security		
8.4.1.1	Strength indicators	The “moving image” construct was added to the security sub-model.
Memory		
8.4.2.1	Passphrase dictionary	The “passphrase dictionary” construct was renamed to “passphrase dictionary warning message” in the memory sub-model.
8.4.2.2	Phonologically similar words	No change required
8.4.2.3	Password and passphrase length and login failures	Construct renamed from “different character length for different passwords” to “slightly different character length for different passwords/passphrases” in the memory sub-model.
8.4.2.4	Common segments across passwords and passphrases	No change required
8.4.2.5	User-generated passphrases	No change required
8.4.2.6	Simple usernames	No change required
8.4.2.7	Lenient password policies	No change required

8.4.2.8	Strength indicator and password policy alignment	No change required
8.4.2.9	Application of chunking	No change required
Typing		
8.4.3.1	Unfamiliar sequence	The “unfamiliar sequence” construct in the typing sub-model was renamed to “avoid unfamiliar sequences”.
8.4.3.2	Number of keystrokes	No change required
8.4.3.3	SMS language, acronyms and abbreviations	The “SMS language” construct in the typing sub-model was renamed to “avoid SMS language, acronyms and abbreviations”.
8.4.3.4	Keyboard visualisation and touch	No change required
8.4.3.5	Keyboard layout exposure	No change required
General Observations		
8.4.4.1	Password and passphrase entropy	No change required
8.4.4.2	Complicated passwords, passphrases and memory issues	No change required
8.4.4.3	Similar passwords and passphrases created by users	A new construct was added under “password composition”, labelled “avoid similar passwords/passphrases” to the memory sub-model.
8.4.4.4	Password and passphrase memory improvement over time	A new construct labelled “extend passphrase lifespan” was added to the memory sub-model under the “password composition” construct.
8.4.4.5.1	Days between logins	No change
8.4.4.5.2	Login time period	A new construct under “user” called, “user fatigue” was added to the memory sub-model.
8.4.4.6	Password and passphrase resets	No change required
8.4.4.7.1	User age and failure type	No change required
8.4.4.7.2	User age and login device	No change required
8.4.4.7.3	User gender	No change required

8.4.4.8	Password and passphrase failures	No change required
8.4.4.9	Language in passwords and passphrase	A new construct, "different languages" was added to the memory sub-model, under "password composition".
8.4.4.10	User preference	A "change control" construct was added which includes two attributes; "education" and "adoption" attributes.

The next section depicts the changes to the proposed model for this study.

8.5 Model Updates

The model updates section displays the updates made to the sub-models and proposed models based on the data collected from the login assessment experiment. The updated security model is firstly presented followed by the updated memory model and then the updated typing model. Lastly, the updates to the proposed model are presented.

The constructs and attributes are highlighted in green, blue and black text in Figure 8-27. Green text items (constructs and attributes) indicate those items that were already included in the proposed model and were confirmed as correct after analysing the login assessment results. The blue text items show the items that were not originally part of the proposed model based on the login assessment results. Lastly, the black text items display the items that could not be evaluated using the login assessment.

8.5.1 Updated Security Sub-model

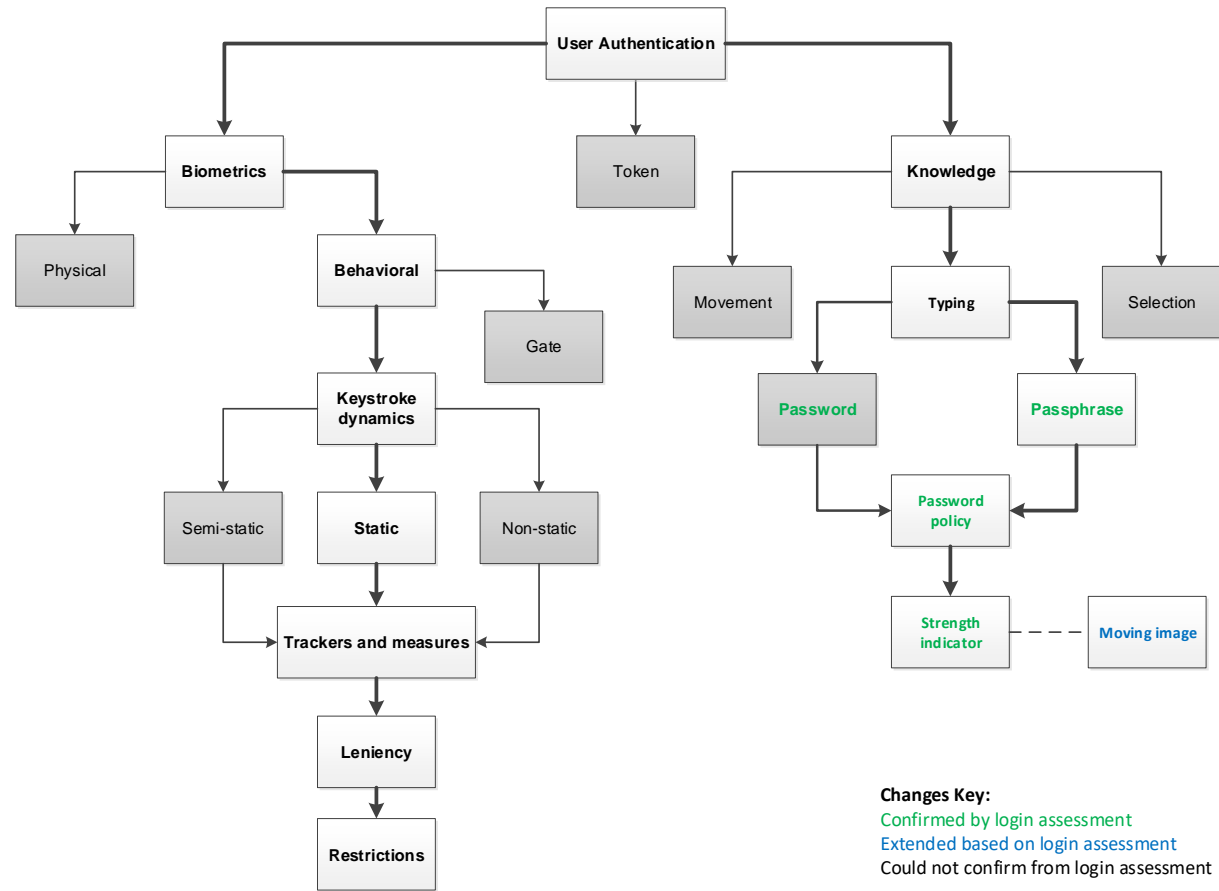


Figure 8-27: Updated Security Sub-model

The constructs in green in Figure 8-27 have been verified. The login assessment also confirmed the effectiveness of the running bunny and therefore, a new construct (“moving image”) has been associated with the “strength indicator” construct. The next section indicates the changes made to the memory sub-model.

8.5.2 Updated Memory Sub-model

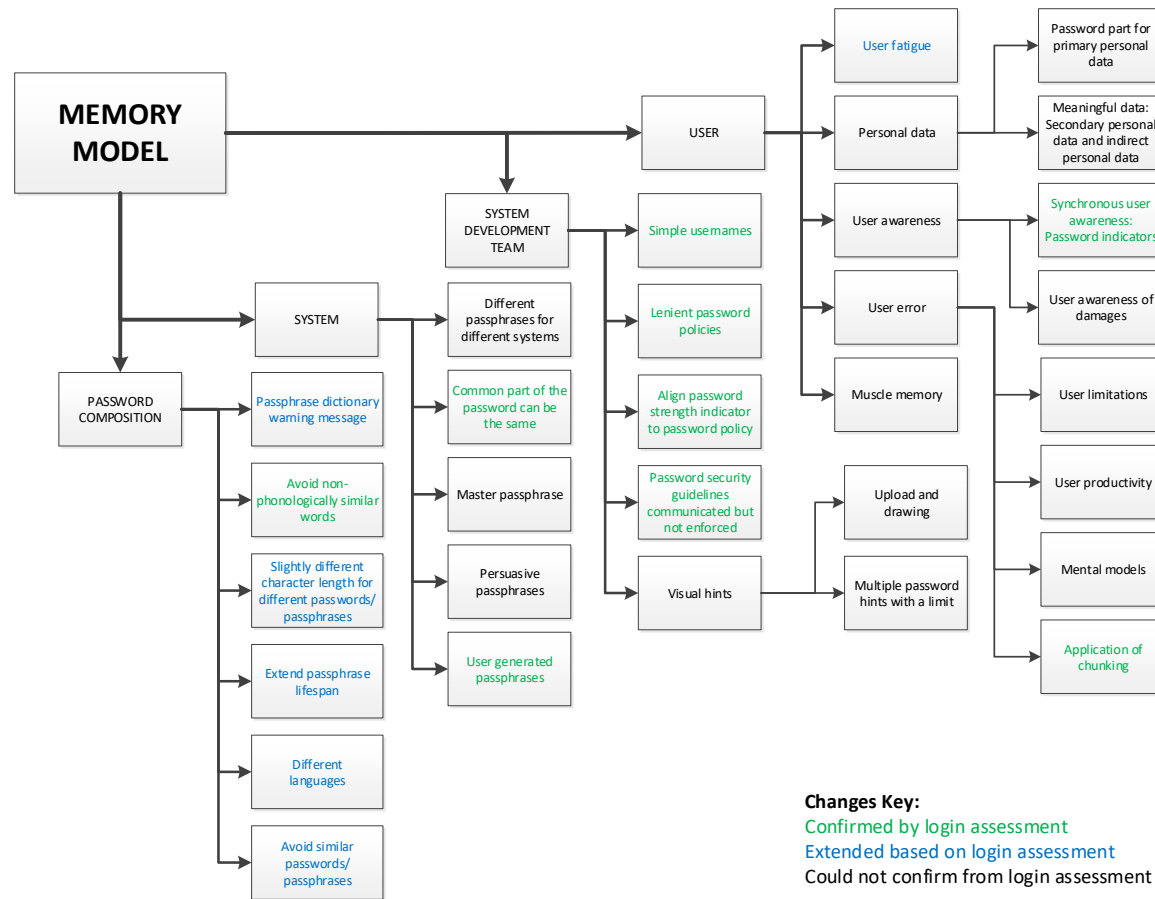


Figure 8-28: Updated Memory Sub-model

Several confirmations and updates have been made to the memory sub-model. Four constructs were added and two constructs were renamed to depict a more accurate consideration. These are indicated in blue text in Figure 8-28. The green text shows the constructs that were verified by the login assessment data.

8.5.3 Updated Typing Sub-model

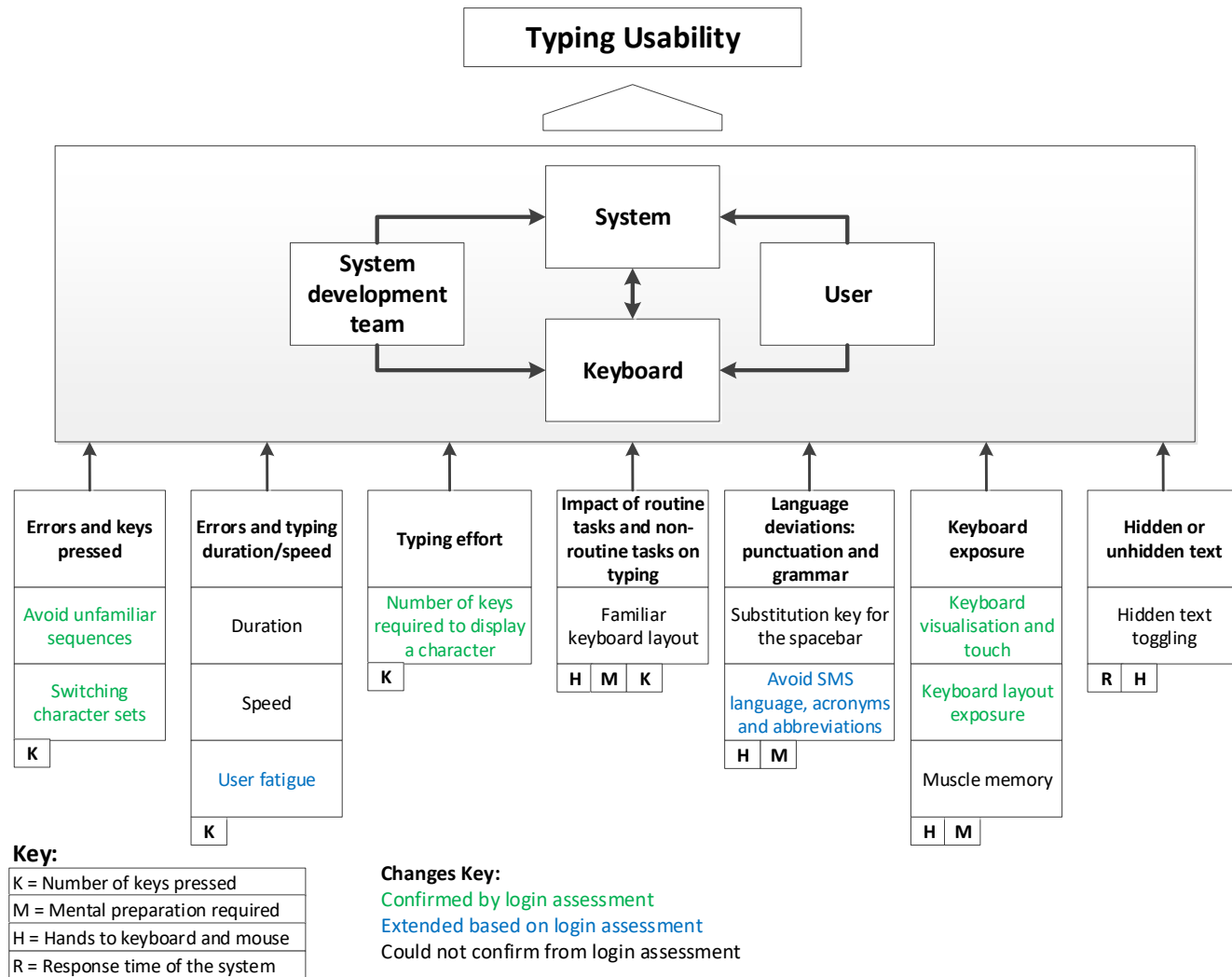


Figure 8-29: Updated Typing Sub-model

The green text presented in Figure 8-29 indicates the constructs that could be confirmed using the data collected in the login assessment. One blue text construct was added as a new construct (“user fatigue”) and placed under “errors and typing”. The other blue text construct was renamed to cater for acronyms and abbreviations. The next section discusses the updates to the proposed model.

8.5.4 Updated Proposed Model

The proposed model was created based on previous literature and was evaluated based on the results of the login assessment experiment (first phase of primary data). Figure 8-30 below depicts the changes to the proposed model in green and blue text.

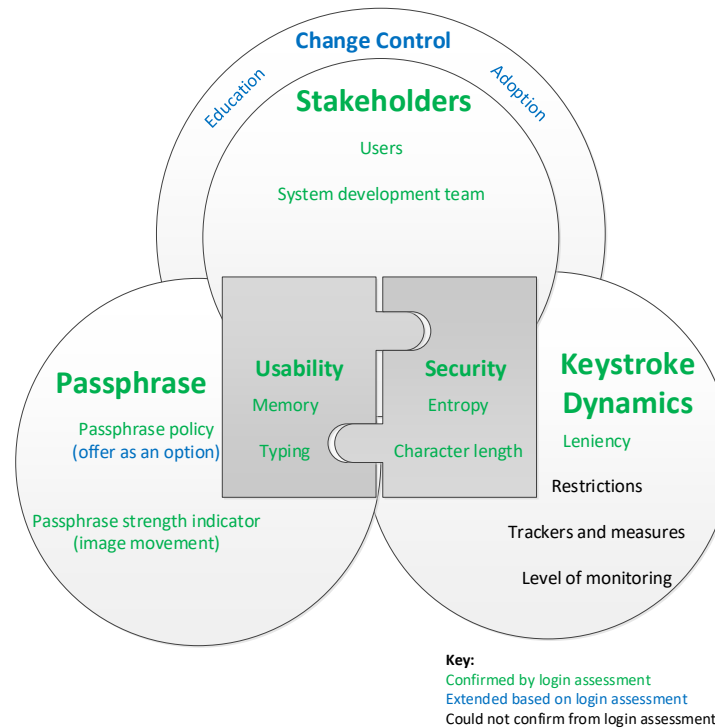


Figure 8-30: Updated Proposed Model

Based on the total number of constructs and attributes that could be assessed, it was possible to evaluate 83% (15/18 constructs and attributes) of the proposed model. Based on findings from the login assessment, it is suggested that users create a passphrase around 16 to 18 characters long. Login assessment data found that typing issues increase when passphrase length is around 22 to 25 characters. The outstanding items, in addition to the items that have already been evaluated through the login assessment, were assessed in the next phase by means of the expert review.

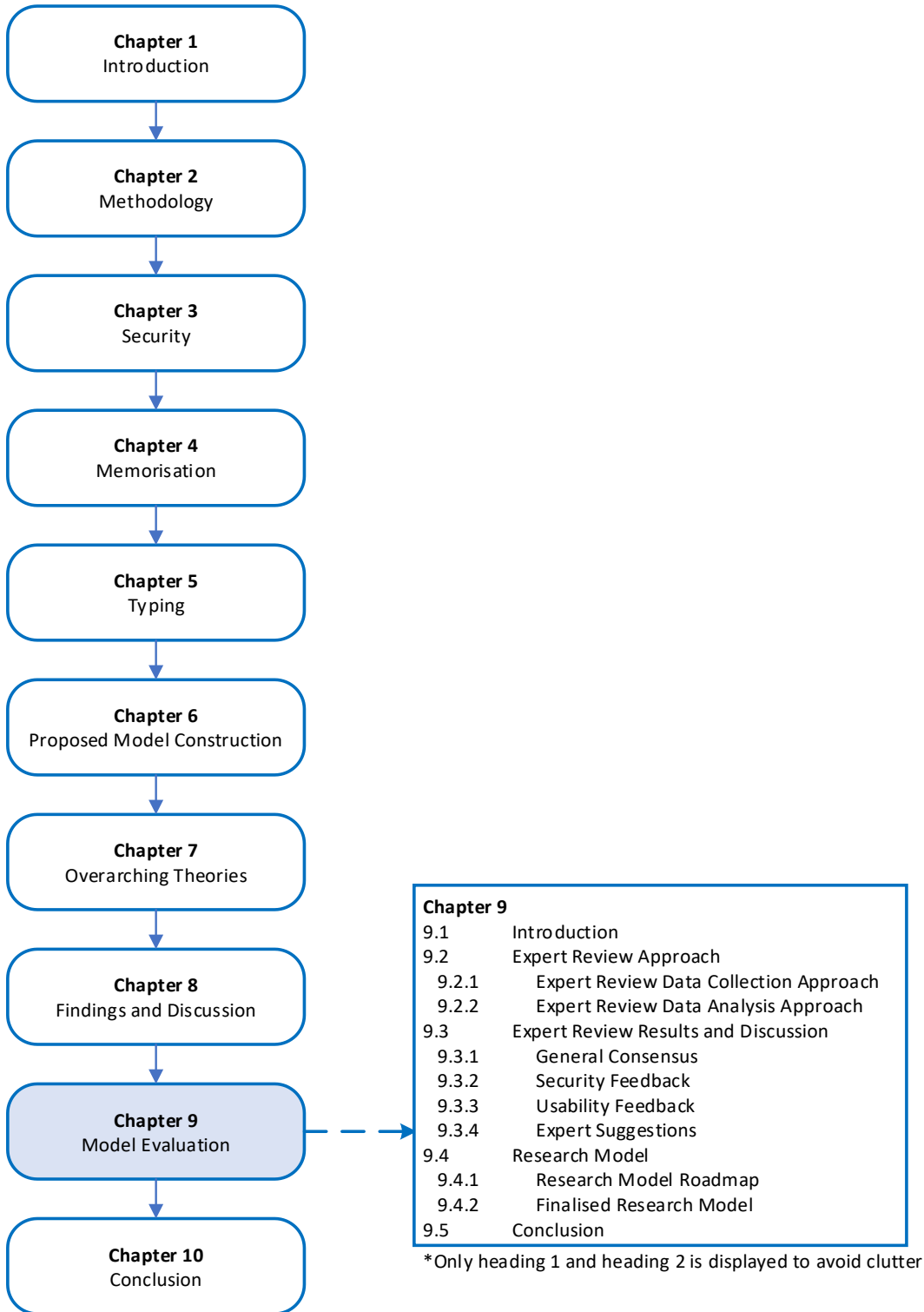
The “change control” construct was added to the proposed model. As depicted in Figure 8-30, the “change control” construct arches over the “stakeholder” construct to indicate that it influences both stakeholder groups. The system development team is required to educate users and encourage the adoption of passphrases and keystroke dynamics. It is suggested that the system development team updates the information security policy to include this solution and a user education plan.

Note that although the proposed model has been updated it is still referred to as the proposed model until it has been updated by the expert review feedback.

8.6 Conclusion

This chapter allowed for the evaluation of certain findings from the literature chapters using primary data collected from the login assessment experiment. The sub-models created in the literature chapters were used as a guideline to analyse the data collected from the login assessment. The data collected either agreed or disagreed with the literature findings. Any data collected that disagreed with the findings resulted in the sub-models being updated, as well as the proposed research model. Most of the components of the proposed model were evaluated by means of the data collected from the login assessment experiment. This increases the level of confidence in the proposed model. To further increase this confidence, the updated proposed model was reviewed by experts in the field of security and usability. The feedback from this review was subsequently used to validate and update the model. This is discussed in more detail in the next chapter.

Chapter 9 – MODEL EVALUATION



9.1 Introduction

The literature chapters created the sub-models for security, memory and typing. These sub-models were then used to construct the proposed model for this study. The login assessment, discussed in the previous chapter, was then used to evaluate both the sub-models and the proposed model. Consequently, in this chapter the proposed model is validated by two rounds of expert reviews. The results of these either confirm constructs and the flow of the proposed model, add constructs and flows to the proposed model and/or contradict constructs and flows of the proposed model.

Since this is the last phase of validation, the intention is to ensure that the areas reviewed by the experts are thoroughly critiqued, thus allowing a certain level of confidence regarding the artefact of the study. The next section explains the expert research approach.

9.2 Expert Review Approach

This section firstly defines who is deemed to be an appropriate expert. This is followed by a section explaining the number of experts required. The next section then explains the approach taken to collect the data from experts which is followed by a discussion on how the data collected from the experts was analysed.

9.2.1 Expert Review Data Collection Approach

An expert review is a data collection method used to gain insight on specific areas from experts in the field (Zhang et al., 2016). Accordingly, important aspects of the study were presented to the experts in relevant fields to gain their opinions. The experts were given a brief explanation of the research study to assist their understanding of the context before answering the questions. It was important to ensure that this explanation was kept as brief as possible to avoid any disinterest which might result in no or little feedback. A statement included in the message sent to the experts mentioned that they could contact the researcher if further detail or any clarification were required. The primary communication with experts was via email but face-to-face communication and/or a video conference was arranged where necessary. The questions were constructed in a way that encouraged different perspectives and conciseness and also ensured that all feedback collected was relevant to the study. Appendix B provides the questions and context provided to all expert reviewers.

9.2.1.1 Expert Reviewers

Barber et al. (2015) explain that the selection of experts for an expert review is influenced by the nature of the research study. Two types of experts were required for this study – security experts and usability experts. In this research experts are defined as follows:

- Security experts are considered to be anyone with a minimum of five years' industry experience in computer security.
- Usability experts are considered to be anyone with a minimum of five years' industry experience in user experience design.

After defining and identifying the experts required for this review, the next section provides the number of expert reviewers required.

9.2.1.2 Number of Expert Reviewers

When attempting to ascertain the acceptable size for an expert group it was found that there is no fixed rule for identifying the number of participants. However, based on similar research it was found that there are a number of guidelines (Edwards et al., 2015). These guidelines have been grouped into three categories:

1. The structure of questions posed to the expert group.
2. The number of questions in each round and volume of expected feedback. Questions were kept short and concise to ensure participation.
3. The amount of analysis expected to be conducted on the feedback received.

Two rounds of reviews were needed for this study. An expert review should not exceed 45 days as a longer duration may increase the risk of experts not responding or losing interest, or decrease the quality of feedback (John et al., 2017; Ludwig, 1997). The same participants were used for both rounds, which limited the number of contradictions and ensured all feedback was obtained within an acceptable timeframe. As discussed, two rounds were required for this expert review, namely:

1. Validation of the research problem and the effectiveness of the proposed solution.
2. The review of the proposed model and its impact on the research problem.

Figure 9-1 graphically depicts how the expert review was applied to this study. The following definitions of terms are provided for clarity on the actions indicated in Figure 9-1.

- **Round** – The process which begins when the researcher requests information from the experts. The experts then provide feedback which the researcher then

analyses (i.e. the data analysed). Consolidated feedback is subsequently given to experts to confirm that their feedback has been interpreted correctly. The process ends once the experts are happy with the interpretation of the feedback.

- **Questions** – The researcher commenced each round by posing questions to the expert reviewers. The questions were used to ensure relevancy and conciseness of feedback from experts.
- **Consolidate feedback** – This refers to the process of analysing the feedback received from the experts. See Section 9.2.2, Expert Review Data Analysis Approach for more detail on how the reviewers’ feedback was analysed.



Figure 9-1: Expert Review Process

Figure 9-1 shows that the completion of round 1 is dependent on the commencement of round 2. The expert review for this study ended once the experts agreed on the proposed model. Once agreement was received, the proposed model was finalised and considered to be the research model for this study.

Based on the above considerations, feedback from a minimum of ten experts (five security experts and five usability experts) was required to be collected. Twelve experts were contacted however only ten experts (five security experts and five usability experts) provided feedback. The next section provides the data analysis approach for this expert review.

9.2.2 Expert Review Data Analysis Approach

Once feedback had been received from the experts and the round completed, analysis commenced. Analysis only began once feedback was received from a minimum of ten experts:

1. Similar feedback across experts was grouped together to eliminate duplication.

2. Once the feedback was grouped, it was summarised to eliminate “waste” (irrelevant feedback).
3. The cleaned feedback, after step 1 and step 2 above, was then sent to the experts to confirm that nothing had been misinterpreted.

Once the proposed artefact had been updated, it was sent to the experts for approval. Following this approval, the proposed model was renamed the “research model”. This is presented later in the chapter.

Now that the analysis approach for the expert review has been explained, the next section provides the results and a discussion.

9.3 Expert Review Results and Discussion

This section aims to discuss the feedback received from the expert review. The expert review comprised of two rounds: round 1 focused on the impact of the proposed solution on security and usability, and the challenges that may arise during and after the implementation of this solution, while the second round collected feedback on the proposed model. Because the feedback from round 1 and round 2 overlapped, it was consolidated and organised into specific categories. This eliminated the need to report duplicate feedback obtained from the two rounds of expert reviews. All feedback was arranged into four categories – general consensus, security feedback, usability feedback and suggested considerations. The first section discusses the general consensus from the expert review feedback.

9.3.1 General Consensus

After assessing the results, it was found that eight out of ten experts believed that this solution would increase security and usability. However, they did explain that some challenges needed to be addressed before implementation as they might cripple the use and success of this solution. One of the two experts who did not believe this solution would increase security and usability opined that passwords are stronger than passphrases. This disagrees with the findings of Chapter 3 that passphrases are stronger than passwords. However, the expert’s disagreement indicated that user education on the strength of passwords and passphrases is important for adoption. The other expert who disagreed explained that the keystroke dynamics algorithm needs to be lenient at the beginning to gauge the user’s typing pattern. During this period the user is vulnerable to an attack thus making them temporarily unsecure. This is a high-level

summary of the feedback received from the experts. The following sections provide more detailed feedback with feedback on security being discussed first.

9.3.2 Security Feedback

The following security feedback was received by the experts. The main bullet points and first sub-bullet points provide a summary of the expert feedback. The bullet point in italics indicate the update made to the proposed model based on the feedback.

- One expert explained that a password management tool would still be more effective than this solution.
 - He suggested that keystroke dynamics be added to the password management tools. These may be seen as a system on its own. Therefore, it is possible to apply this solution to a password management tool.
 - *The “type of system” was added as a construct to the proposed model.*
- An expert reviewer warned that this does not address the problem of database intrusions and would do more damage to the user as the hacker would now also have access to the users’ password/passphrase patterns.
 - It was suggested that users’ passwords/passphrases be stored on one database server while the users’ keystroke patterns could be stored on a different one. Both servers would then have to be hacked to gain unauthorised access to the user’s account. It is also suggested that the password/passphrase and keystroke pattern data be encrypted to further increase security.
 - *A construct labelled “database server” was added to the proposed model.*
- The lengthiness of a passphrase renders it more secure than a short password with multiple character sets.
 - This aligns to the findings of this study. However, users should be educated on this as many still believe that passwords are stronger than passphrases.
 - *No update required to the proposed model as this is covered under the “change control” construct.*
- Phishing is still a risk with passphrases.
 - Keystroke dynamics makes phishing attacks more difficult for unauthorised parties as the attacker now also needs to observe how the user typed the password/passphrase.

- *No update required to the proposed model as keystroke dynamics reduces the risk of phishing attacks.*
- Users often create passwords that just meet the minimum password policy requirements. Users may only create a passphrase which is a portion of a phrase if the phrase is much longer than 16 characters. This has an impact on usability as users need to recall the phrase and also the start and end point of the phrase.
 - It is expected that users will use the full phrase. That being said, a portion of a common phrase may be more secure than the entire phrase, although it should be noted that this may have a negative impact on usability.
 - *No update to the proposed model is required. Unlike conventional passwords, the user is given the flexibility to create a passphrase using a full phrase or part of a phrase.*

9.3.3 Usability Feedback

This section focuses on feedback collected from experts that relates to usability. Like the previous section, the main bullet points and first sub-bullet points provide a summary of the expert feedback. The italicised bullet points indicate and explain the update made to the proposed model based on the feedback.

- Adoption will be a challenge, especially since the majority of users have become accustomed to passwords. If users have not experienced any damage from a security attack, they may stick to what's currently working for them; in this case, passwords.
 - User education/training may be vital to encourage users to adopt this method of authentication.
 - *No update is required to the proposed model as this is covered under the "change control" in the proposed model.*
- User distraction may affect usability as it might not match the keystroke pattern and thus the login process would need to be repeated. Furthermore, if a user needs to login urgently and thus ends up logging in faster than usual it may delay the login process even further.
 - Keystroke dynamics leniency needs to be less stringent to cater for these situations.
 - *No update to the proposed model is required as this forms part of the "leniency" attribute.*

- Mobile devices with different keyboard layouts and size may also change a user's typing patterns.
 - Passphrases support typing on mobile devices better than passwords. While applications can identify whether a user is logging in with a mobile device or desktop computer, they cannot differentiate between a cellphone and a tablet. Keystroke dynamics should record a keyboard pattern for a desktop computer and a separate keyboard pattern for mobile devices (cellphones and tablets). If keystroke dynamics needs to be more accurate by differentiating between cellphones and tablets, then the user can indicate this prior to login. It should be noted that this negatively affects usability as the user needs to complete an additional step. It is suggested that users be presented with both options and then they can decide on the level of stringency.
 - *“Login device” has been added to the proposed model as a layer around the “keystroke dynamics” construct to indicate that separate keystroke patterns need to be created for different login devices.*
- One expert explained that the solution does not support browser password management tools that automatically paste the password into the password field once the login page loads. Since this is done by the system, no user–keyboard interaction takes place during the login process.
 - If this two-tier user authentication solution is implemented correctly, a password management tool is not required. In addition, password management tools have a lower level of usability than passphrases and keystroke dynamics combined, as they require the user to interact with multiple devices or multiple systems. It is suggested that if this study's proposed solution is implemented on a website, a message should be added on the user registration page that the keystroke dynamics algorithm will not work if a password management tool is used to automatically insert the password in the respective field.
 - *No update is required to the proposed model.*
- Typing behaviour may change throughout the day. It may also change randomly. For example, workload may be temporarily high causing fatigue for a random week. Additionally, a hand or arm injury that affects typing and takes a user three to six months to recover from will change the user's typing pattern temporarily.

- It is suggested that users be given the option to reset or disable the keystroke dynamics data collected on them to accommodate any injuries that may impair normal login interaction. This process should be similar to a password reset process to avoid usability issues.
 - *The attribute “reset/disable” has been added under the “keystroke dynamics” construct in the proposed model.*
- Privacy concerns may be questioned if user–keyboard interaction is recorded without the user’s consent.
 - The user needs to be informed of and acknowledge understanding of keystroke dynamics before it can be used.
 - *No update required to the proposed model as this forms part of the “education” attribute.*
- Some devices such as old cellphones do not have a full keyboard (example button "1" on the keyboard is used to type letters "a", "b", "c" and "d").
 - Although this issue is unlikely to occur, it is more significant in developing countries with low levels of information and communication technology usage. A separate keystroke pattern would need to be recorded for these types of device.
 - *This is covered under the recently added “login device” construct in the proposed model.*
- Some users with little IT knowledge may not comprehend the workings of keystroke dynamics when a login failure arising from a false positive appears. Again, this is likely to occur in developing countries.
 - Users should be provided with a short instructional video and a diagram to explain this solution. A contact number or support forum should also be provided for answering any questions a user may have after seeing the video. Once the solution becomes more widespread, this won’t be necessary.
 - *An explanation of keystroke dynamics to relevant parties was included in the “change control” construct.*
- No influence on usability as keystroke dynamics is more of a backend solution. It will be accepted by users as it increases security and results in no change to the process from a user perspective.
 - Although true, false positives will negatively affect usability. Keystroke dynamics needs to be lenient enough to reduce the risk of false positives.

- *No update is needed to the proposed model as this relates to the “leniency” construct under the “keystroke dynamics” construct.*
- Password/passphrase resets will cause difficulties in verifying typing patterns when a password/passphrase is reset.
 - Users should be asked whether they would like to reset their recorded typing pattern when they reset their password/passphrase. However, a user’s typing pattern is not expected to change drastically when the user creates another passphrase as opposed to a password. This process should be similar to a typical password reset process to avoid usability issues.
 - *No update is needed to the proposed model as this is covered under the “reset/disable” attribute under the “keystroke dynamics” construct.*
- Adoption of a new approach, especially since it relates to security, will be difficult.
 - If users have not experienced any personal damage arising from lack of security in the past, they may just stick to what is working for them. Trust has to be earned from users before acceptance is possible. Older generation users may be reluctant to adopt this solution as it changes the way in which they have previously authenticated themselves. User education is vital for the adoption of this solution.
 - *This forms part of the “change control” construct in the proposed model. Therefore, no updates are required.*
- It may not work well for certain systems. For example, ATMs and telephone banking. These will need to be classified as additional devices as typing patterns will probably differ between ATMs and desktop or mobile devices, for example.
 - ATMs, for example, need to be classified as a different login device. Users should be able to create different typing pattern groups based on device login, thus further increasing security.
 - *This is covered under the “login device” construct in the proposed model.*
- The keystroke dynamics algorithm must be constantly learning and adapting to the user's changes in typing patterns.
 - Keystroke dynamics becomes more accurate as the number of times the user logs into the system increases. The algorithm can also be run on text-based fields post login, either permanently or temporarily, to speed up the learning process involved in keystroke dynamics.

- *No update is required to the proposed model as this is covered under the “level of monitoring” attribute.*
- Users like to know that their data is secure. However, they are often reluctant to adopt a solution that will collect data on them regardless of the type of data.
 - Users should be informed that the interaction data collected will not be shared with anyone else or used for anything else but to validate logins. It is also important to communicate to the user that the data collected is secured.
 - *Change control should be set up in a manner that can confidently assure the user that interaction data will not be used to their detriment in any way. No update was required as this activity forms part of the “change control” construct.*
- This does not resolve the problem of having different passwords for different applications.
 - Since passphrases are easier to recall than passwords and keystroke dynamics monitor typing patterns, it may be acceptable to have the same passphrase for multiple systems.
 - *This can be explained during the “change control” process. Therefore, no update was required to the proposed model.*

9.3.4 Expert Suggestions

The last section regarding the feedback from experts refers to suggestions for improving the proposed solution. The main bullet points in this section provide the suggestions made by the experts. The sub-bullet points in italics indicate the updates made to the proposed model.

- Consider the intention of the system before determining the leniency of the keystroke dynamics algorithm. For example, banking systems will have a more stringent keystroke dynamics algorithm than a social media system such as Twitter.
 - *This is considered under the “type of system” construct that was recently added to the proposed model.*
- The keystroke dynamics algorithm should be tested first by recording user–keyboard interaction and not restricting the user from accessing the system. After the first five or ten logins for example, keystroke dynamics should be enforced. At the time the keystroke dynamics algorithm is enforced, the system should have

sufficient data to reduce the likelihood of a false positive login. Perhaps allow the user to enter the system if the passphrase is not typed correctly but also send an SMS or email to the user if the keystroke pattern is incorrect. If the user eventually stops receiving notifications via SMS or email, then it is reasonable to assume that the keystroke dynamics algorithm is authenticating correctly and then the user has the option to enable the keystroke dynamics feature.

- *This is covered under the “reset/disable” attribute in the proposed model.*
- Keystroke dynamics should not be used in isolation from another authentication method.
 - *The solution incorporates passphrases with keystroke dynamics. No update was required to the proposed model.*
- The passphrase strength indicator should be aligned to the password policy.
 - *A double-ended arrow has been added to the proposed model to indicate that the passphrase policy needs to be aligned to the passphrase strength indicator.*
- Passwords have created usability issues stemming from the user being forced to memorise difficult passwords. For this reason, usability improves as users have more flexibility in creating a password or passphrase. Introducing passphrases gives users the option to access a vast range of passwords (i.e. passphrases). However, it is recommended that passphrases be introduced as an option in addition to passwords. Users then have the option to avoid character switching.
 - *Text has already been added under the “passphrase policy” attribute in the proposed model which states “offer as an option”.*
- Perhaps provide the user with the option to use keystroke dynamics instead of enforcing it without user approval. This should be asked on user registration and is easily activated at any time after login.
 - *This is covered under the “reset/disable” attribute in the proposed model.*
- Add an example or description to the keystroke dynamics constructs to avoid assumptions being made.
 - *The following descriptions were provided under the keystroke dynamics construct:*
 - *“Login time range” has been added under “leniency”.*
 - *“Login with x hand/s, x fingers” has been added under “restrictions”.*

- *“Time between key up, key down, between keys” has been added under “trackers”.*
 - *“Keyboard interaction time: min, max, sum, average” has been added under “measures”.*
 - *“Option provided to users” has been added under “reset/disable”.*
- If a moving image is used as a strength indicator and the speed change is not noticeable when a character is added or removed, it might not be very effective. It is suggested that a speedometer-type animation be added to the moving image which more accurately indicates a speed change.
 - *“Image movement” under the “passphrase” construct in the proposed model has been renamed to “image movement and speed indicator”.*
- Switch the “security” and “usability” puzzle pieces around as passphrases relate more closely to usability and keystroke dynamics relates more closely to security.
 - *The “security” and “usability” puzzle pieces have been switched.*

This section completes the feedback received from the experts. All the feedback collected either required the proposed model to be updated, or it required no update as the feedback was addressed by an existing construct. The italicised bullet points in this section explained what update was needed to the proposed model. The next section introduces the final research model for this study.

9.4 Research Model

This section firstly discusses the way the research model evolved over the course of this research study before presenting the finalised research model.

9.4.1 Research Model Roadmap

The research artefact for this study took the form of a model. Before presenting the finalised research model, a roadmap is presented which shows when the research model was drafted and how the model was adapted before it was finalised.

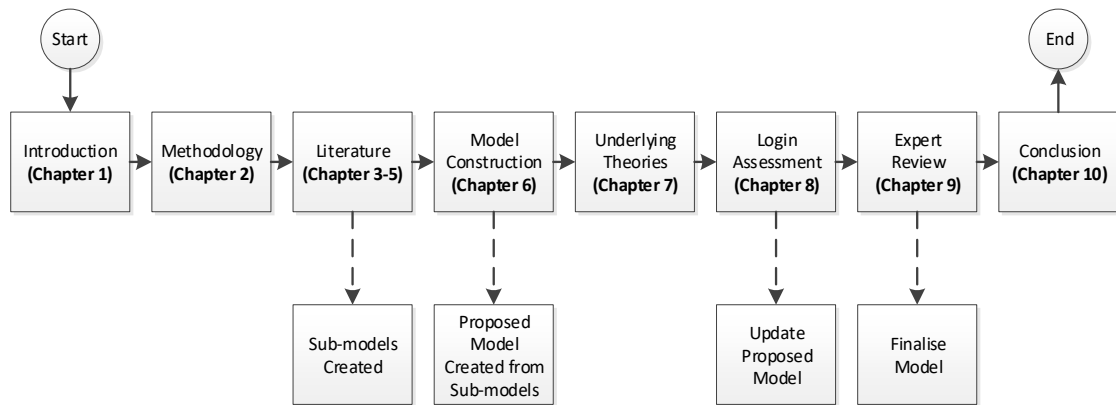


Figure 9-2: Research Model Roadmap

Figure 9-2 illustrates the research journey taken to construct, evaluate, validate, update and finalise the research artefact for this study. The literature chapters (Chapters 3 to 5) were used to create sub-models relating to security, memorisation of passwords/passphrases and typing passwords/passphrases. These three models were then used to create a single research model (referred to as the proposed model as it was not yet evaluated and validated) which included key constructs relating to security, memorisation and typing. This was accomplished in Chapter 6. After data from the login assessment experiment was collected and analysed, the proposed model was updated based on the findings of the assessment. This was done in Chapter 8 and was considered to be an evaluation of the proposed research model. Chapter 9 discussed the review feedback received from ten experts in the field on the updated proposed model. This feedback was in turn used to validate the model with the experts and subsequently to update the proposed model. The completion of the process of updating the proposed model with the expert review feedback allowed the proposed model to be finalised.

The research model has now been finalised and no further updates are required. The next section introduces the finalised research model for this study.

9.4.2 Finalised Research Model

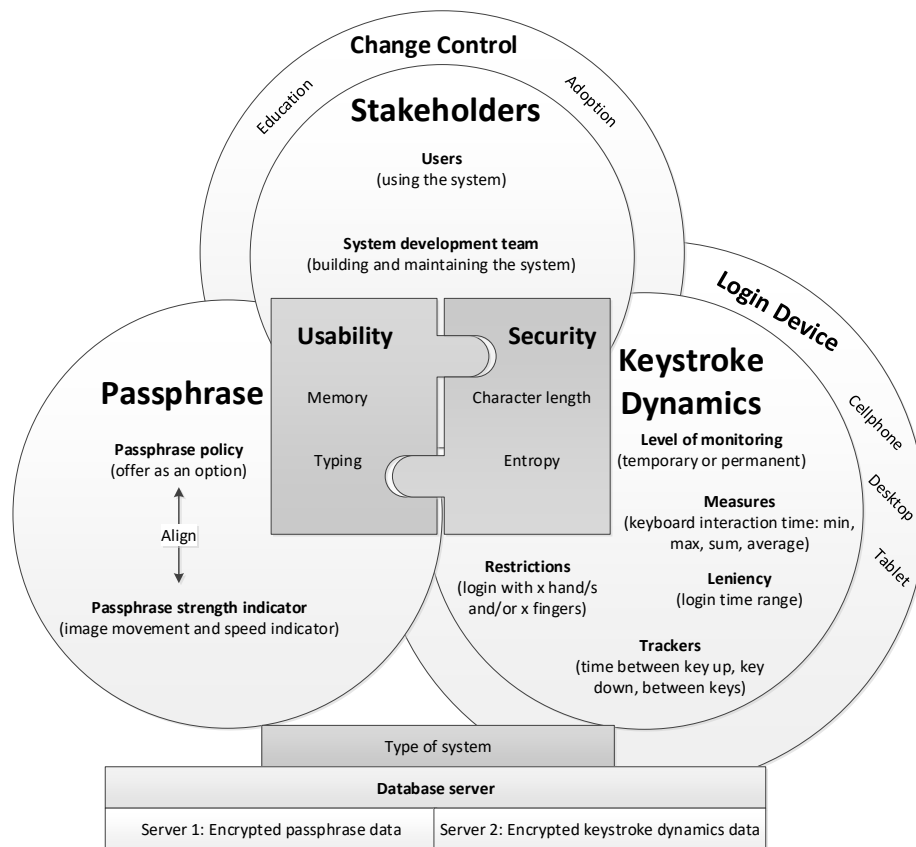


Figure 9-3: Research Model

The detailed rationale for each update to the proposed model was provided in Section 9.3, Expert Review Results and Discussion. This section briefly summarises the changes made to the proposed model, which has resulted in the finalisation of the research model for this study.

- Descriptions have been added to all attributes under the stakeholders, passphrase and keystroke dynamics constructs to clarify their meaning and avoid ambiguity.
- In the passphrase construct, an alignment arrow was added between the two attributes to indicate that they had to be in alignment.
- A login device has been added as a layer around the keystroke dynamics construct. This shows that users can indicate to the system what login device is being used so the correct keystroke dynamics pattern can be cross verified with the user's keyboard interaction. Alternatively, the system can determine the login device. However, the system is limited to identifying only logins that occur on desktops or mobile devices (cellphone and tablet). Users may be given both

options or a system-identified device may be a default option which the user can change at any time.

- The type of system construct has been added below the passphrase and keystroke dynamics constructs to indicate that the nature and intention of the system determines how the two-tier user authentication approach is set up to best support the user.
- Every system needs a database to store the encrypted passwords/passphrases. A user's keystroke patterns are also stored in the database. In order to spread the risk of a database attack, it was suggested that the user's encrypted passwords/passphrases be stored on a database server that is separate from the encrypted keystroke patterns.
- Entropy has been added to the security construct to indicate that it can be used to assess the level of security for passphrases and keystroke dynamics.

The research model can be considered finalised for this study as it has been updated with feedback from the expert review. This research model can be used to assist the implementation of a two-tier user authentication method which involves passphrases and keystroke dynamics to improve security and usability in the user authentication process. The last section in this chapter provides a summary.

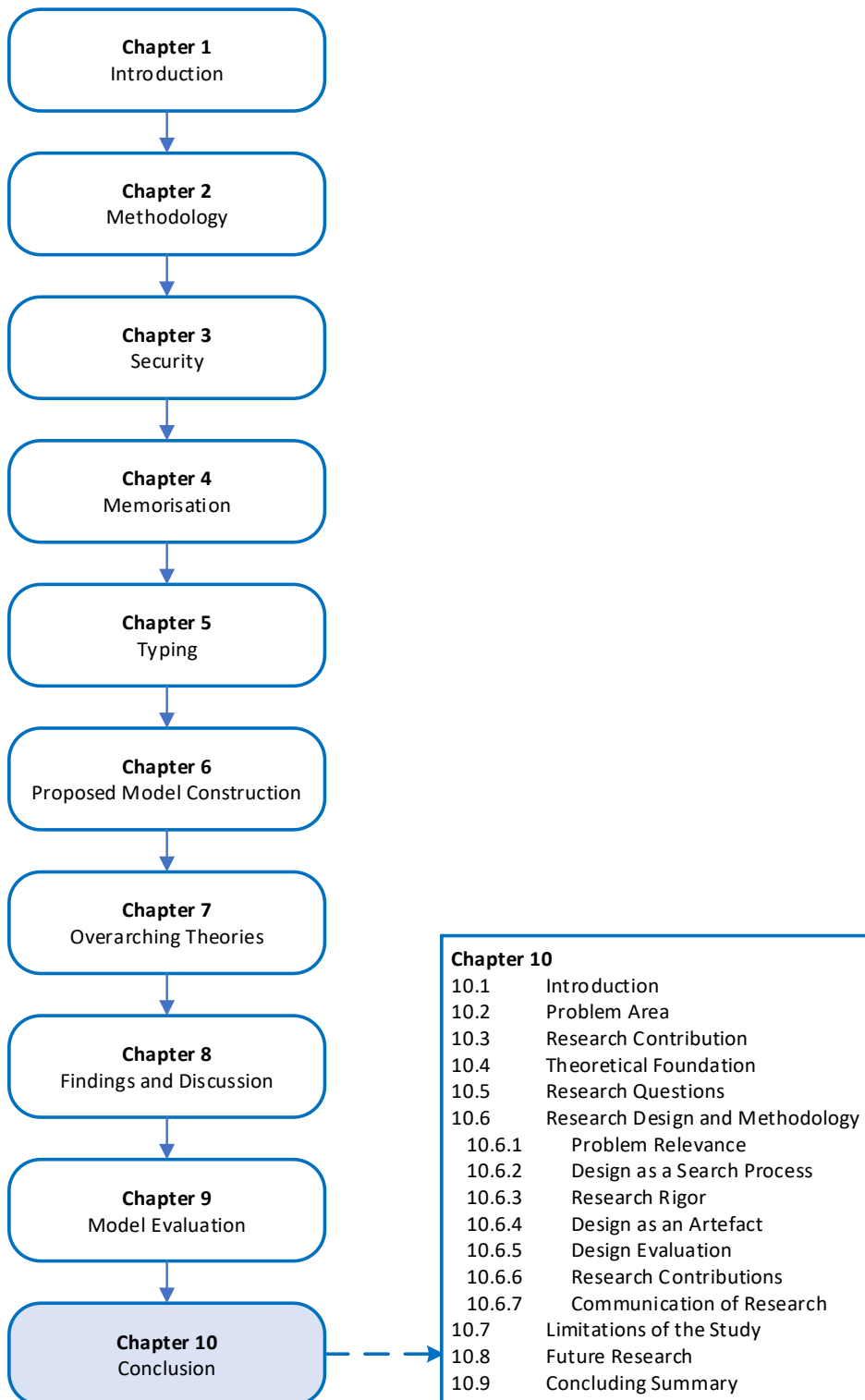
9.5 Conclusion

This chapter focused on two rounds of expert reviews that were conducted for this study and in which ten experts were involved. The feedback received from the experts was consolidated, duplicate comments were removed and then feedback was grouped into categories to easily determine the updates that needed to be made to the proposed research model.

The experts provided valuable inputs to update the proposed model. Once the updates to the proposed model had been made and approved by the experts, the model was finalised. This means that with a certain degree of confidence the research model can now answer the main research question and therefore address the research problem.

The main research question for this study sought to ascertain: What two-tier user authentication solution will improve system usability without compromising system security? The research model provided in Figure 9-3 indicates what constructs need to be considered in order to successfully implement a two-tier user authentication solution which simultaneously improves security and usability.

Chapter 10 – CONCLUSION



*Only heading 1 and heading 2 is displayed to avoid clutter

10.1 Introduction

Hofstee (2006) and Poth (2018) suggests that a conclusion chapter should focus on a brief summary of what was introduced in Chapter 1, while also including the findings of the study. For example, while Chapter 1 presented the research problem, this chapter summarises the problem, focusing more on the solution, which was based on the research findings.

This chapter begins with the problem area and then discusses the research contribution. The next section contains the theoretical foundation for the study and then the research questions are reiterated. The research design and methodology are then provided followed by a discussion on how certain aspects of the study were evaluated and validated. The scope of the research is then defined after which future research aligned to this study is suggested. The last section provides a concluding summary. A discussion of the research problem is discussed in the next section.

10.2 Problem Area

As time passes, hackers are finding it easier to crack passwords. This has caused development teams to force users to create stronger passwords (Shay et al., 2014; Yıldırım & Mackie, 2019). As a result, users have found it difficult to stay current with the user authentication process. This has created what is termed “usability issues” in the user authentication process. It was found that usability issues have two primary sources – memorisation of the password and typing the password.

If a user’s memorisation of the password and/or the ability to correctly type the password fails, then the user either decides to try again or resets their password. Both decisions negatively impact the user as this is additional time spent on authenticating themselves before using the system (Hussain et al., 2018; Keith et al., 2009; Shay et al., 2014). This is supported by a number of findings from the login assessment (see Sections 8.4.4.6 and 8.4.4.8) and further confirmation was received from the expert review feedback (see Sections 9.3.1 and 9.3.3).

10.3 Research Contribution

This research contributes to the body of knowledge in the information systems discipline on user authentication that focuses on security and usability. Using the Shannon Entropy formula, the research found that passphrases are more secure than passwords. The

formula also revealed that keystroke dynamics largely increases user authentication security. Accordingly, this study also contributes to usability in user authentication.

The study found that two primary issues influence user authentication usability: memorisation and typing of the password. It was found that memorisation issues can be reduced by the user making use of passphrases instead of passwords. These findings were found to be consistent when assessing typing issues.

Lastly, a two-tier user authentication solution involving passphrases and keystroke dynamics was suggested. Based on the research findings, this suggested user authentication method improves security and usability when compared to conventional text-based user authentication solutions. The required stakeholders that should be involved in order to implement the suggested two-tier solution successfully were also identified, as well as the roles and responsibilities required of each stakeholder. The components required to successfully implement a keystroke dynamics algorithm were also identified, and their manipulation to increase security was also discussed.

10.4 Theoretical Foundation

Three primary theories were used to drive this study, one relating to security and two focused on usability. The Shannon Entropy formula was used to assess security, while the Chunking theory and the Keystroke-level model were used to assist usability assessment.

The Shannon Entropy theory is a formula used to calculate the number of guesses required before finding an answer, assuming certain variables are known (Aguiar & Guedes, 2015; Arora et al., 2015; Shannon, 1948). This formula was applied to passwords and passphrases to determine which types of passwords are stronger from a security point of view. It was found that passphrases are more difficult to guess than passwords even though passphrases have one character set. It was also found that keystroke dynamics further increased user authentication security. After looking at user authentication security, user authentication usability was assessed.

Two usability theories were applied in this study; one focused on memorisation usability and the other referred to typing usability. Memorisation and typing issues were found to be the primary issues affecting user authentication usability. Subsequently, the Chunking theory was used to assess memorisation usability, as the theory explains that the likelihood of a person memorising an item is higher if that person can associate a personal experience to that item (Bošnjak & Brumen, 2016; España, 2016; Miller, 1956).

This then frees up memory capacity to memorise more items. Applying this theory to passwords, it was found that passphrases are easier to memorise than passwords.

The second usability theory applied in this study was the Keystroke-level model. This theory focused on assessing the typing issues related to user authentication. The Keystroke-level model provides six considerations in the form of user actions which need to be executed when a user interacts with a keyboard (John & Kieras, 1994; Jorritsma et al., 2015; Lee et al., 2015). These actions were applied to typing passwords to assess whether password or passphrases would better support these actions. It was found that passphrases are easier to type than passwords, mainly due to the character switching that occurs in passwords.

These theories assist in answering the research sub-questions. The next section discusses these questions.

10.5 Research Questions

This study identified three sub-questions that were answered in order to assist in the answering of the main research question. The answering of the three sub-questions allowed for the construction of the proposed model. The main research question was then answered by finalising the proposed model.

What two-tier passphrase model for user authentication will improve system usability without compromising system security?

1. What needs to be considered when ensuring the security of passphrases and keystroke dynamics algorithm as a method of user authentication?

When using keystroke dynamics in conjunction with passphrases, it was found that these two types of user authentication complement each other, especially from a security perspective. For passphrases, it is important to consider the password policy and the password strength indicator and ensure these are aligned to each other. In terms of passphrases, it was also found that there is a positive correlation between passphrase character length and passphrase strength, i.e. the longer the passphrase, the stronger the passphrase. In terms of the keystroke dynamics algorithm, there are a few things to consider when configuring it. This includes monitoring (static, non-static or semi-static), trackers and measures, leniency and restrictions. An appropriate level of each needs to be assessed while taking into consideration that all these components should evolve over time depending on the amount of user interaction. This solution will

significantly increase security in current user authentication methods. The security sub-model provided further detail.

2. *What factors in terms of system usability influence the memorisation of passphrases and may impact a keystroke dynamics algorithm?*

Usability is negatively impacted due to users having to remember complicated passwords. To answer this sub-question, factors that influence the memorisation of passwords were identified and then the proposed two-tier user authentication solution was assessed in terms of whether it was able to address these issues. It was subsequently found that the proposed solution could address these issues; however, a number of factors must be considered to reduce the risk of user memory failures in user authentication. These can be grouped into the following categories: user, development team, system and password composition. The memorisation sub-model provides all the factors which should be considered.

3. *What system input factors influence the use of passphrases and may impact the keystroke dynamics algorithm?*

If a user remembers their password, the next step is to insert the password into the system as effortlessly as possible. Similarly, to memorisation factors, typing factors were identified and organised into seven categories: errors and keys pressed, errors and typing duration/speed, typing effort, impact of routine tasks and non-routine tasks on typing, language deviations: punctuation and grammar, keyboard exposure and hidden or unhidden text. Considering these factors reduces the risk of user typing errors when implementing the proposed user authentication solution. The typing sub-model provides more details on user typing.

The three sub-models answered the three sub-questions. The sub-models were then used to construct the proposed model which assisted in answering the main question. The main research question was answered once the proposed model had been updated by means of the login assessment experiment and expert review. The above discussion briefly summarised the way in which the research sub-questions were answered. Table 10-1 includes a more detailed presentation of the discussion sections which assisted in answering the main question and the sub-questions. In order to avoid Table 10-1 from drowning in too much detail, only the main chapter headings are provided.

Table 10-1: Chapter Sections that Addressed the Respective Research Questions

Research Number	Research Question	Sections	
Main Research Question	What two-tier passphrase model for user authentication will improve system usability without compromising system security?	6.3	Literature Chapter Models
		6.4	Development of the Proposed Model
		6.5	Part Model Considerations
		6.6	Whole Model Considerations
		6.7	Proposed Model
		6.8	Constructing the Proposed Model
		6.9	Conclusion
		8.4	Login Assessment Results and Discussion
		8.5	Model Updates
9.3	Expert Review Results and Discussion		
9.4	Research Model		
Note that three research questions had to be addressed to answer the main research question. These three questions were considered to be research sub-questions and are indicated below, along with their contribution areas.			
Research Sub-question 1	What needs to be considered when ensuring the security of passphrases and keystroke dynamics algorithm as a method of user authentication?	3.1	Introduction
		3.2	User Authentication
		3.3	How Passwords are Cracked
		3.4	Forms of Authentication
		3.5	Passwords and Passphrases Defined
		3.6	Password Policies
		3.7	Strength Indicators
		3.8	Summary of Password and Passphrase Components
		3.9	Biometrics
		3.10	What is Keystroke Dynamics
		3.11	Summary of Keystroke Dynamics Components
		3.12	Suggestion for Keystroke Dynamics Solution
		3.13	Perceived Keystroke Dynamics Limitations Resolved

		3.14 3.15 3.16 6.3 8.4 8.5 9.3	Entropy of the Proposed Solution Summary of the Proposed Security Model Conclusion Literature Chapter Models Login Assessment Results and Discussion Model Updates Expert Review Results and Discussion
Research Sub-question 2	What factors in terms of system usability influence the memorisation of passphrases and may impact a keystroke dynamics algorithm?	4.1 4.2 4.3 4.4 4.5 6.3 8.4 8.5 9.3	Introduction Definition of Chunking Memory Components Passphrase and Keystroke Dynamics Memorisation Usability Model Conclusion Literature Chapter Models Login Assessment Results and Discussion Model Updates Expert Review Results and Discussion
Research Sub-question 3	What system input factors influence the use of passphrases and may impact the keystroke dynamics algorithm?	5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 5.10 6.3	Introduction Keystroke-level Model Keystroke-level Model Application Keystroke-level Model Scope Usability Impact on Typing Different Password Types Types of Typographical Errors User Authentication Process Typing Issues Affecting Usability Passphrase and Keystroke Dynamics Typing Usability Model Conclusion Literature Chapter Models

		8.4	Login Assessment Results and Discussion
		8.5	Model Updates
		9.3	Expert Review Results and Discussion

The above table explains how the research questions were answered and in which chapter sections the related discussions can be found. The next section summarises the research design and methodology.

10.6 Research Design and Methodology

This study applied a mixed method approach to data collection and analysis, which involved a combination of qualitative and quantitative research. The qualitative research approaches used in this study included research material on user authentication security and usability, and two sets of expert reviews. The quantitative research approaches included the data collected from the login assessment experiment.

The design science methodology was followed throughout this study. This methodology was found to be the most appropriate methodology for this study to ensure that quality results were achieved. Adherence to the methodology requires that seven guidelines be considered to ensure adequate research was conducted. More detail on this research design and methodology can be found in Chapter 2. The following subsections below explain the main findings of each design science guideline considered.

10.6.1 Problem Relevance

The research problem identified relates to current user authentication security protocols creating usability issues. The problem was identified through secondary data and then confirmed using certain data collected from the login assessment experiment as well as feedback from the expert review. It was also confirmed that memorisation and typing issues were the root cause of text-based user authentication security.

10.6.2 Design as a Search Process

Literature relating to user authentication security and usability was mostly used to understand the research problem and to-date efforts to address the problem, while the login assessment was used to confirm certain findings. In addition to confirming these findings, the expert review was also used to check whether there were any gaps that might not have been considered.

10.6.3 Research Rigor

Theories, methods and related research were used to validate assumptions and identify certain findings. These findings were then evaluated through the login assessment to determine whether the findings would remain consistent in reality. The majority of the login assessment data aligned to the literature findings, however a few findings contradicted the literature. The data collected from the login assessment was used to obtain feedback from users, the main stakeholder for the proposed solution. The expert review was then used to obtain feedback from stakeholders who would be developing the solution. This ensured that all the main stakeholders that would be involved in the proposed solution provided input to the research artefact which takes the form of a model.

10.6.4 Design as an Artefact

This research study began by proposing a two-tier user authentication model that incorporated passphrases and keystroke dynamics to address the research problem. The model is capable of increasing system security and usability in the user authentication process. In addition, the model can maintain an acceptable balance between system security and usability. The model was developed, evaluated and validated with relevant literature, users (through the login assessment) and stakeholders who had the knowledge to develop such a solution (through the expert review).

10.6.5 Design Evaluation

A combination of primary data and secondary data was used in the evaluation and validation processes. This is discussed in more detail below.

Findings from secondary data were collected from journals, conference proceedings, books, websites, theories, methodologies, frameworks, models, case studies and reports. These findings allowed for the construction of three sub-models to answer the majority of each research sub-question. The three sub-models were then used to construct the proposed research model for this study. The research model was termed “proposed” as it had not yet been evaluated and validated on the basis of primary data.

The sub-models and proposed model were then updated based on the findings of the login assessment experiment. The login assessment experiment focused on the collection and analysis of data on user and login interaction. After the sub-model and proposed model had been updated based on the login assessment, the proposed model was sent to the experts for review.

The second phase of primary data took the form of an expert review. An expert review is a method used to evaluate or validate certain research findings using professionals in the respective discipline (John et al., 2017; Molich & Jeffries, 2003). Experts used for the expert review were professionals in computer security and usability and the feedback collected was used to update to proposed model, which was then referred to as the “final” model for this research study.

The majority of the secondary data used in this research can be found in the literature chapters (Chapters 3, 4 and 5). The primary data in this study was used for evaluation and validation in terms of updating the proposed model.

According to the design science methodology three aspects have to be addressed to confidently state that thorough evaluation was conducted (Cronholm & Göbel, 2015). These three aspects are discussed further below.

- **Utility** – Can the target audience use it?
 - The issue was raised that adoption may be problematic owing to a lack of education. However, the solution is simple as users are concerned with creating a passphrase as opposed to a password. From a system development team perspective, more effort is required to implement the solution. However, once the information security policy and the system authentication is updated/developed, minor maintenance is required. In summary, for both stakeholder groups, the amount of effort does not outweigh the value provided thus fulfilling the design science utility requirement.
- **Efficacy** – Does it fulfil the intention it was created for?
 - Through research rigour (literature, login assessment experiment and expert review), it was found that the research solution does address the research problem. Less login failures occur with passphrases which also results in less passphrase resets due to memory or typing issues as opposed to conventional passwords.
- **Quality** – Does it address the requirements for successful use?
 - In terms of the research problem, the solution needs to address three main requirements:
 1. Increase user authentication security
 2. Increase user authentication usability

3. If security increases then usability must not decrease and if usability increases, security must not decrease.
- Throughout the research efforts, the three points mentioned in the previous bullet points were considered. These also formed part of the research sub-questions. The research solution does indeed fulfil all three requirements.

This next section continues with the next design science guideline.

10.6.6 Research Contributions

This research study developed a model that provides the system development team with an alternative user authentication solution that satisfies both security and usability. The model provides considerations when implementing a two-tier user authentication solution that includes the use of passphrases as the first tier of authentication and keystroke dynamics as the second tier. The information security policy in organisations should also be updated to be less stringent while increasing security and usability.

10.6.7 Communication of Research

The findings will be published in academic journals and/or conferences (two journal papers have been drafted and currently under review). The thesis will also be accessible at the Rhodes University library.

The seven sub-sections above explained how each guideline was considered and contributed to the research study. This also served as confirmation that all seven design science guidelines were considered. The next section discusses the scope of the study.

10.7 Limitations of the Study

This study focused specifically on passphrases and keystroke dynamics as types of user authentication methods. Other forms of authentication were not assessed.

The Shannon Entropy theory was used to assess the strength for the proposed user authentication solution. Although there are many other methods of assessing the strength of user authentication methods, this theory was found to be the most effective and consistent as well as convenient, even though it has been criticised by some researchers for being too general (Becker, Parkin, & Sasse, 2018; Guo & Zhang, 2018; Khan, Pečarić, & Pečarić, 2017; Rass & König, 2018). Only the short-term memorisation of passwords and passphrases was assessed in this study.

The login assessment experiment, used to confirm certain findings, was web-based which could be accessed at any time to assist convenient participation. The participants in the login assessment were asked to avoid the use of any password recall functionality and to refrain from recording the password and passphrase anywhere. The experiment assumed that all participants adhered to this instruction throughout the experiment. For convenience, the login assessment participants were South African residents (Etikan & Bala, 2017). It should be noted that this did not affect the sample representation of the population. It is also important to note that keystroke dynamics was not assessed using the login assessment experiment due to resource and time constraints. This will need to form part of future research within this research area.

A keystroke dynamics algorithm was also not applied to the login assessment as this would have resulted in a large amount of data to be collected and reporting on such data would have resulted in too large of a scope for this research study. Data collected through the application of a keystroke dynamics algorithm was recommended as a future research item in section 10.8, Future Research.

The scope of this study has been emphasised through its delimitations. The next section provides future research suggestions.

10.8 Future Research

The aim of this section is to discuss further research which may emanate from this study. Below are suggested areas of research which should be considered.

- Assess the usability of the keystroke dynamics algorithm and its rate of false positives during the user login process. A keystroke dynamics algorithm can be developed in numerous ways. This study provided the keystroke dynamics components that can be manipulated to assist usability. However, the study did not identify how best to manipulate these components to achieve the best level of usability.
- This study focused on a two tier authentication solution involving passphrases and keystroke dynamics with slightly more emphasis on passphrase security and usability. Further research can be conducted that provides more focus on keystroke dynamics.
- In terms of memorisation, only short-term memory was assessed. Further research should be conducted to determine whether this solution better supports long-term memorisation of text-based authentication.

- Section 8.4 (Login Assessment Results and Discussion) provided a statistical analysis across certain aspects of the proposed research model. These results can be further analysed using the t-test or ANOVA (Analysis of Variance) (Harms, 2019) to check if there are some statistical differences.

Research in relation to the above suggestions will be helpful for adoption and implementation and will contribute significantly to this research area. The last section provides a concluding summary.

10.9 Information Security Policy Recommendation

Data collected from the login assessment experiment showed that passphrases are stronger than conventional passwords created by the login assessment participants. The login assessment also confirmed that participants find it easier to type and memorise passphrases than passwords. The login assessment forced participants to create passwords that comply with password policies applied to common websites with the following requirements.

The password created must have:

- At least eight characters
- At least one upper character
- At least one special character
- At least one number

The above password policy requirements were found to provide less support to security and usability than passphrases. It is suggested that passphrases be added to the password policy but should not replace the currently popular password policy. This is necessary as the login assessment, in conjunction with feedback from the expert review, found that change management is required. Allowing conventional password policies to offer passphrases may begin to change users mind set on passphrases. Conventional password policies should be updated as follows:

- The password created must have:
 - At least eight characters
 - At least one upper character
 - At least one special character
 - At least one number
- Or a passphrase created must have:

- At least sixteen characters
- All lowercase letters

After analysing the login assessment data, it was found that the best passphrase length that reduces the risk of memory and typing errors is 16 to 18 characters long. Typing issues are likely to occur when passphrase length starts exceeding 18 characters. Likewise, the risk of login failure due to memory occurs when passphrases exceed 25 characters. However, purely from a security perspective, the longer the passphrase, the stronger the passphrase. Therefore, it is not suggested that a character limit be imposed on passphrases in the password policy. That being said, the login assessment found that majority of participants try to create a password and passphrase that closely meets the minimum requirements. Thus, it is likely that users will naturally create a passphrase in the range of 16 to 18 characters long.

In conjunction with the above recommendation, it is suggested that a lenient keystroke dynamics algorithm be implemented which can evolve (become more stringent) as more user-keyboard interaction data is collected on a specific user. The stringency level of the keystroke dynamics algorithm must be separate per user based on the number of times the user interacts with the system. From a usability perspective, it is also suggested that the user be notified that the keystroke dynamics algorithm should become more stringent before it is applied. The user should have the option to accept or decline this change. If the user accepts the change in stringency, the system must allow a user to revert back to the less stringent keystroke dynamics algorithm if they are not satisfied with the change.

10.10 Concluding Summary

This chapter provided a summary of the research problem, how this research was conducted and what the main findings were that addressed the research problem. The intention of this chapter is to highlight the originality and significance of this study. It was recommended that users adopt passphrases as opposed to passwords, as the former are more secure than the latter and more user friendly from a memorisation and typing perspective. The system development team should not allow users to create a passphrase with fewer than 16 characters (through enforcement of the password policy) as security may be compromised. That being said, it is recommended that users create a passphrase around 16 to 18 characters long. Data collected from the login assessment found that the risk of memory and/or typing issues increased when passphrase length

exceeded 20 characters. This recommended passphrase length should be communicated to the user on the registration screen.

Users should not be concerned about keystroke dynamics as this is a back-end solution to improve security. From a system development team perspective, it is vital that the keystroke dynamics algorithm is of high quality otherwise it may have a negative impact on usability. It is suggested that the keystroke dynamics algorithm be developed with basic user-keyboard monitoring until sufficient data is collected on a user. This reduces the risk of false-positive logins, thus reducing the risk of usability issues created by keystroke dynamics. In summary, the more user-keyboard interaction data is collected on a user, the stronger the security will become.

References

- Abinaya, R., & Sigappi, A. N. (2018). A Keystroke Dynamic Based Biometric for Person Authentication. *International Journal of Pure and Applied Mathematics*, 118(5), 769-783.
- Adams, A., & Sasse, M. A. (1999). Users are Not the Enemy. *Communications of the ACM*, 42(12), 40-46.
- Adams, A., Sasse, M. A., & Lunt, P. (1997). Making Passwords Secure and Usable. *People and Computers XII* (pp. 1-15). London: Springer.
- Adanali, Y. K. (2017). Rational Choice Theory: Its Merits and Limits in Explaining and Predicting Cultural Behaviour. *Erasmus Journal for Philosophy and Economics*, 10(1), 137-141.
- Adham, M., Azodi, A., Desmedt, Y., & Karaolis, I. (2013). How to Attack Two-factor Authentication Internet Banking. *International Conference on Financial Cryptography and Data Security* (pp. 322-328). Berlin: Springer.
- Aguiar, V., & Guedes, I. (2015). Shannon Entropy, Fisher Information and Uncertainty Relations for Log-periodic Oscillators. *Physica A: Statistical Mechanics and its Applications*, 423(1), 72-79.
- Alguliyev, R., Aliguliyev, R., & Yusifov, F. (2018). Role of Social Networks in E-government: Risks and Security Threats. *Journal of Communication and Media Technologies*, 8(4), 363-376.
- Alhassan, M. M., & Quaye, A. A. (2017). Information Security in an Organization. *International Journal of Computer*, 24(1), 100-116.
- Allen, C. G., & Komandur, S. (2019). The Relationship between Usability and Biometric Authentication in Mobile Phones. *Human-computer Interaction* (pp. 183-189). Cham: Springer.
- Almalki, S., Chatterjee, P., & Roy, K. (2019). Continuous Authentication Using Mouse Clickstream Data Analysis. *International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage* (pp. 76-85). Cham: Springer.
- Alomari, R., & Thorpe, J. (2019). On Password Behaviours and Attitudes in Different Populations. *Journal of Information Security and Applications*, 45(1), 79-89.

- Alshehri, A., Coenen, F., & Bollegala, D. (2018). Iterative Keystroke Continuous Authentication: A Time Series Based Approach. *Künstl Intell*, 32(4), 231-243.
- Althubaiti, S. (2017). Improving the Design and Usability of Password Creation Systems. *Abstracts on Human Factors in Computing Systems* (pp. 244-247). Denver: ACM.
- Anderson, J. D., Wagovich, S. A., & Brown, B. T. (2019). Phonological and Semantic Contributions to Verbal Short-term Memory in Young Children with Developmental Stuttering. *Journal of Speech Language and Hearing Research*, 62(3), 644-667.
- Archibald, J., & Ferguson, R. I. (2017). Assessing the Impact of Affective Feedback on End-user Security Awareness. *International Conference on Human Aspects of Information Security, Privacy, and Trust* (pp. 143-159). Cham: Springer.
- Arifin, S. R. (2018). Ethical Considerations in Qualitative Study. *International Journal of Care Scholars*, 1(2), 30-33.
- Arora, P., Hanmandlu, M., & Srivastava, S. (2015). Gait Based Authentication Using Gait Information Image Features. *Pattern Recognition Letters*, 68(2), 336-342.
- Arthana, I. K., Pradnyana, I. M., & Dantes, G. R. (2019). Usability testing on website wadaya based on ISO 9241-11. *Journal of Physics Conference Series*, 1165(1), 1-9.
- Arthur, N., & McMahon, M. (2018). *Contemporary Career Development Theories: International Perspectives*. Abingdon: Routledge.
- Ayyagari, R., Lim, J., & Hoxha, O. (2019). Why Do Not We Use Password Managers? A Study on the Intention to Use Password Managers. *Contemporary Management Research*, 15(4), 227-245.
- Babaeizadeh, M., Bakhtiari, M., & Maarof, M. A. (2014). Keystroke Dynamic Authentication in Mobile Cloud Computing. *International Journal of Computer Applications*, 90(1), 29-36.
- Babbie, E. (2005). *The Basics of Social Research*. Toronto: Thomson Wadsworth.
- Backus, D., Ferriere, A., & Zinab, S. (2015). Risk and Ambiguity in Models of Business Cycles. *Journal of Monetary Economics*, 69(1), 42-63.

- Banerjee, S. P., & Woodard, D. L. (2012). Biometric Authentication and Identification Using Keystroke. *Journal of Pattern Recognition Research*, 7(1), 116-139.
- Barber, C., Amberg, A., Custer, L., Dobod, K. L., Glowienke, S., Van Gompelf, J., Wichardm, J. (2015). Establishing Best Practise in the Application of Expert Review of Mutagenicity Under ICH M7. *Regulatory Toxicology and Pharmacology*, 73(1), 367-377.
- Becker, I., Parkin, S., & Sasse, M. A. (2018). The Rewards and Costs of Stronger Passwords in a University: Linking Password Lifetime to Strength. *Proceedings of the 27th USENIX Security Symposium* (pp. 239-253). Baltimore: USENIX.
- Bellamy, R., John, B., & Kogan, S. (2011). Deploying CogTool: Integrating Quantitative Usability Assessment into Real-world Software Development. *33rd International Conference on Software Engineering* (pp. 691-700). Waikiki: ACM.
- Bergadano, F., Gunetti, D., & Picardi, C. (2002). User Authentication Through Keystroke Dynamics. *ACM Transactions on Information and System Security*, 5(4), 367-397.
- Bhanbhro, H., Hassan, S. R., Nizamani, S. Z., Bakhsh, S. T., & Alassafi, M. O. (2018). Enhanced Textual Password Scheme for Better. *International Journal of Advanced Computer Science and Applications*, 9(7), 209-215.
- Bhivgade, T., Bhusari, M., Kuthe, A., Jiddewar, B., & Dubey, P. (2014). Multi-factor Authentication in Banking Sector. *International Journal of Computer Science and Information Technologies*, 5(2), 1185-1189.
- Bi, X., Ouyang, T., & Zhai, S. (2014). Both Complete and Correct?: Multi-objective Optimization of Touchscreen Keyboard. *SIGCHI Conference on Human Factors in Computing Systems* (pp. 2297-2306). Toronto: ACM.
- Blanchard, N. K. (2019). Secure and Efficient Password Typo Tolerance. *ACM Conference* (pp. 1-14). New York: ACM.
- Blanchard, N. K., Malaingre, C., & Selker, T. (2018). Improving Security and Usability of Passphrases with Guided Word Choice. *Proceedings of the 34th Annual Computer Security Applications Conference* (pp. 723-732). San Juan: ACM.
- Boonkrong, S. (2012). Security of Passwords. *Information Technology Journal*, 8(2), 112-117.

- Bošnjak, L., & Brumen, B. (2016). What Do Students Do With Their Assigned Default Passwords? *39th International Convention on Information and Communication Technology, Electronics and Microelectronics* (pp. 1430-1435). Opatija: IEEE.
- Bošnjak, L., & Brumen, B. (2019). Rejecting the Death of Passwords: Advice for the Future. *Computer Science Information Systems*, 16(1), 313–332.
- Bošnjak, L., Sreš, J., & Brumen, B. (2018). Brute-force and Dictionary Attack on Hashed Real-world Passwords. *41st International Convention on Information and Communication Technology, Electronics and Microelectronics* (pp. 1161-1166). Opatija: IEEE.
- Botha, R. A., Furnell, S. M., & Clarke, N. L. (2009). From Desktop to Mobile: Examining the Security Experience. *Computers and Security*, 28(4), 130-137.
- Braunstein, P. (2015). Making Secure Easy-to-remember Passwords. *COMP*, 1(1), 1-15.
- Braz, C., & Robert, J. M. (2006). Security and Usability: The Case of the User Authentication Methods. *Proceedings of the 18th International Conference of the Association Francophone d'Interaction Homme-Machine* (pp. 199-203). Montreal: ACM.
- Brill, C., & Olmsted, A. (2016). Password Reinforcement Leveraging Social Media. *International Conference on Information Society (i-Society)* (pp. 135-136). Dublin: IEEE.
- Brostoff, S., & Sasse, A. M. (2000). Are Passfaces More Usable than Passwords? A Field Trial Investigation. *People and Computers XIV - Usability or Else!* (pp. 405–424). London: ACM Press.
- Brumen, B. (2019). Security Analysis of Game Changer Password System. *International Journal of Human-Computer Studies*, 126(1), 44-52.
- Bryant, K., & Campbell, J. (2006). User Behaviors Associated with Password Security and Management. *Australian Journal of Information Systems*, 14(1), 81-100.
- Buhrmester, M., Kwang, T., & Gosling, S. D. (2011). Amazon's Mechanical Turk: A New Source of Inexpensive, Yet High-quality, Data? *Perspectives on Psychological Science*, 6(1), 3-5.

- Burkhard, M., & Koch, M. (2012). Evaluating Touchscreen Interfaces of Tablet Computers for Elderly People. *Workshopband Mensch & Computer* (pp. 53-59). Munich: Oldenbourg Verlag.
- Burr, W. E., Dodson, D. F., Newton, E. M., Perlner, R. A., Polk, W. T., Gupta, S., & Nabbus, E. A. (2017). *Electronic Authentication Guideline (800-63-2)*. Gaithersburg: National Institute of Standards and Technology.
- Carstens, D. S., Malone, L. C., & Mccauley-Bell, P. R. (2007). Applying Chunking Theory in Organizational Password Guidelines. *Journal of Information, Information Technology & Organizations*, 2(1), 97-113.
- Carstens, D. S., Mccauley-bell, P. R., Malone, L. C., & Demara, R. F. (2014). Evaluation of the Human Impact of Password Authentication Practices on Information Security. *Informing Science Journal*, 7(1), 67-85.
- Chakraborty, J., & Nguyen, N. (2018). The Effect of Simulation in Large-Scale Data Collection - An Example of Password Policy Development. *Cambridge Workshop on Universal Access and Assistive Technology* (pp. 263-273). Cham: Springer.
- Charoen, D., Raman, M., & Olfman, L. (2008). Improving End User Behavior in Password Utilization. *Systemic Practice and Action Research*, 21(1), 55-72.
- Cheng, J., Yang, Y., Shao, J., & Liao, J. (2015). Mechanism on Computer Access Permission Management Based on the Proposed Dynamic Password Algorithm. *International Journal of Security and Its Applications*, 9(11), 403-418.
- Chethan, B. K., Siddappa, M., & Jayanna, H. S. (2020). Novel Framework using Dynamic Passphrase Towards Secure and Energy-efficient Communication in Manet. *Electrical and Computer Engineering*, 10(2), 1552-1560.
- Chiasson, S., Forget, A., Stobert, E., Van Oorschot, P. C., & Biddle, R. (2009). Multiple Password Interference in Text Passwords and Click-Based Graphical Passwords. *Proceedings of the 16th ACM Conference on Computer and Communications Security* (pp. 1-12). Chicago: ACM.
- Choi, H., Jeong, J., Woo, S. S., Kang, K., & Hur, J. (2019). Password Typographical Error Resilience in Honey Encryption. *Computers & Security*, 87(1), 1-8.

- Choong, Y. Y. (2014). A Cognitive-Behavioral Framework of User Password Management Lifecycle. *International Conference on Human Aspects of Information Security, Privacy, and Trust* (pp. 127-137). Cham: Springer.
- Choong, Y. Y., Theofanos, M., & Liu, H. K. (2014). *United States Federal Employees' Password Management Behaviors – a Department of Commerce Case Study*. Gaithersburg: National Institute of Standards and Technology.
- Clark, E., & Arakia, K. (2011). Text Normalization in Social Media: Progress, Problems and Applications for a Pre-processing System of Casual English. *Procedia - Social and Behavioral Sciences* (pp. 2-11). Sapporo: Elsevier.
- Colnago, J., Devlin, S., Oates, M., Swoopes, C., Bauer, L., Cranor, L., & Christin, N. (2018). "It's not actually that horrible": Exploring Adoption of Two-Factor Authentication at a University. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (pp. 1-12). New York: ACM.
- Cowan, N. (2010). The Magical Mystery Four: How is Working Memory Capacity Limited, and Why? *Current Direction of Psychological Science*, 19(1), 51-57.
- Cox, D., Cox, J. G., & Cox, A. D. (2017). To Err is Human? How Typographical and Orthographical Errors Affect Perceptions of Online Reviewers. *Computers in Human Behavior*, 75(1), 245-253.
- Crawford, H., & Ahmadzadeh, E. (2017). Authentication on the Go: Assessing the Effect of Movement on Mobile Device Keystroke Dynamics. *13th Symposium on Usable Privacy and Security* (pp. 163-173). Santa Clara: USENIX.
- Cresswell, J. W., & Clark, V. L. (2011). *Designing and Conducting Mixed Method Research*. Thousand Oaks: Sage Publications.
- Cronholm, S., & Göbel, H. (2015). Empirical Grounding of Design Science Research Methodology. *International Conference on Design Science Research in Information Systems* (pp. 471-478). Cham: Springer.
- Daribay, A., Obaidat, M. S., & Krishna, P. V. (2019). Analysis of Authentication System Based on Keystroke Dynamics. *International Conference on Computer, Information and Telecommunication Systems* (pp. 1-6). Beijing: IEEE.

- Davis, D., Monrose, F., & Reiter, M. K. (2004). On User Choice in Graphical Password Schemes. *13th Conference on USENIX Security Symposium* (pp. 1-13). San Diego: USENIX.
- Davis, F. D., Bagozzi, R. P., & Warshaw, P. R. (1989). User Acceptance of Computer Technology: A Comparison of Two Theoretical Models. *Management Science*, 35(1), 982–1003.
- De Luca, A., von Zezschwitz, E., Pichler, L., & Hussmann, H. (2013). Using Fake Cursors to Secure On-screen Password Entry. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 2399-2402). Paris: ACM.
- De Munari, E., Cozzutti, G., & Romero-Naranjo, F. J. (2016). Music and Movement: A Comparative Study between the BAPNE and Suzuki Methods. *ERPA International Congresses on Education*, 26(1), 1-5.
- De Ru, W. G., & Eloff, J. H. (1997). Enhanced Password Authentication through Fuzzy Logic. *IEEE Expert: Intelligent Systems and Their Applications*, 12(6), 38-45.
- Deese, J. (1959). Influence of Inter-Item Associative Strength upon Immediate Free Recall. *Psychological Reports*, 5(3), 305-312.
- Demchenko, Y., Ngo, C., de Laat, C., Wlodarczyk, T. W., Rong, C., & Ziegler, W. (2011). Security Infrastructure for On-demand Provisioned Cloud Infrastructure Services. *3rd International Conference on Cloud Computing Technology and Science* (pp. 255-263). Athens: IEEE.
- Dib, A., & Ghazi, S. (2019). Anti-Shoulder Surfing Login Based on Multi-Entry Models on Onscreen Keyboard. *International Conference on Networking and Advanced Systems* (pp. 1-5). Annaba: IEEE.
- Dmitrienko, A., Liebchen, C., Rossow, C., & Sadeghi, A.-r. (2014). On the (In) Security of Mobile Two-Factor Authentication. *International Conference on Financial Cryptography and Data Security* (pp. 1-18). Berlin: Springer.
- Dooley, J. F. (2018). The Machines Take Over: Computer Cryptography. In *A Brief History of Cryptology and Cryptographic Algorithms* (pp. 167-184). Cham: Springer.
- Doumont, J. (2002). Magical Numbers: The Seven-Plus-or-Minus-Two Myth. *IEEE Transactions on Professional Communication*, 45(2), 123-127.

- Duran, V., & Şentürk, Ş. (2019). Development of Reasoning Styles Scale. *Universal Journal of Educational Research*, 7(5), 1205-1215.
- Dutta, S., Madnick, S., & Joyce, G. (2016). SecureUse: Balancing Security and Usability Within System Design. *International Conference on Human-Computer Interaction* (pp. 471-475). Cham: Springer.
- Edwards, H., Dunlop, M., Mallick, A., & O'Callaghan, F. (2015). Outcomes Following Childhood Arterial Ischaemic Stroke: A Delphi Consensus on What Parents Want from Future Research. *European Journal of Paediatric Neurology*, 19(2), 181–187.
- Ehteshami, A., Sadoughi, F., Saeedbakhsh, S., & Isfahani, M. K. (2013). Assessment of Medical Records Module of Health Information System According to ISO 9241-10. *Acta Informatica Medica*, 21(1), 36–41.
- Epp, C., Lippold, M., & Mandryk, R. L. (2011). Identifying Emotional States Using Keystroke Dynamics. *SIGCHI Conference on Human Factors in Computing Systems* (pp. 715-724). Vancouver: ACM.
- Eslami, S. P., & Ghasemaghahi, M. (2018). Effects of Online Review Positiveness and Review Score Inconsistency on Sales: A Comparison by Product Involvement. *Journal of Retailing and Consumer Services*, 45(1), 74-80.
- España, L. Y. (2016). Effects of Password Type and Memory Techniques on User Password Memory. *Psi Chi Journal of Psychological Research*, 21(4), 269-275.
- Etikan, I., & Bala, K. (2017). Sampling and Sampling Methods. *Biometrics & Biostatistics International Journal*, 5(6), 215–217.
- Fatima, R., Siddiqui, N., Umar, M. S., & Khan, M. H. (2019). A Novel Text-Based User Authentication Scheme Using Pseudo-dynamic Password. *Information and Communication Technology for Competitive Strategies* (pp. 177-186). Singapore: Springer.
- Fellows, R., & Liu, A. (2015). *Research Methods for Construction*. Chennai: John Wiley and Sons.
- Findlater, L., & Wobbrock, J. (2012). Personalized Input: Improving Ten-finger Touchscreen Typing Through Automatic Adaptation. *SIGCHI Conference on Human Factors in Computing Systems* (pp. 815-824). Austin: ACM.

- Findlater, L., Froehlich, J. E., Fattal, K., Wobbrock, J. O., & Dastyar, T. (2013). Age-Related Differences in Performance with Touchscreens Compared to Traditional Mouse Input. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 343-346). Paris: ACM.
- Florencio, D., & Herley, C. (2007). A Large-Scale Study of Web Password Habits. *Proceedings of the 16th international conference on World Wide Web* (pp. 657-666). Banff: ACM.
- Forget, A., & Biddle, R. (2008). Memorability of Persuasive Passwords. *CHI '08 Extended Abstracts on Human Factors in Computing Systems* (pp. 3759-3764). Florence: ACM.
- French, A. M. (2012). Too Sophisticated for the User to Access Their Information. *Journal of Internet Banking and Commerce*, 17(2), 1-14.
- Fujita, Y., Inomata, A., & Kashiwazaki, H. (2019). Implementation and Evaluation of a Multi-factor Web Authentication System with Individual Number Card and WebUSB. *20th Asia-Pacific Network Operations and Management Symposium* (pp. 1-4). Matsue: IEEE.
- Furnell, S. (2007). An Assessment of Website Password Practices. *Computers and Security*, 26(7), 445-451.
- Gafni, R., Pavel, T., Margolin, R., & Weiss, B. (2017). Strong Password? Not with Your Social Network. *Journal of Applied Knowledge Management*, 5(1), 27-41.
- Gagneja, K., & Jaimes, L. G. (2017). Computational Security and the Economics of Password Hacking. *International Conference on Future Network Systems and Security* (pp. 30-40). Cham: Springer.
- Gahlot, A., & Gupta, U. (2016). Gaze-based Authentication in Cloud Computing. *International Journal of Computer Applications*, 1(1), 14-20.
- Gao, B., Kim, H., & Udayan, J. D. (2018). A Study on Usability and Security of Mid-Air Gesture-Based Locking System. *Advances in Big Data and Cloud Computing* (pp. 313-325). Singapore: Springer.
- Garcia, S. N., Zarca, A. M., Hernández-Ramos, J. L., Bernabe, J. B., & Gómez, A. S. (2019). Enforcing Behavioral Profiles through Software-Defined Networks in the Industrial Internet of Things. *Applied Sciences*, 9(21), 1-21.

- Geerts, G. L. (2011). A Design Science Research Methodology and its Application to Accounting Information Systems Research. *International Journal of Accounting Information Systems*, 12(2), 142-151.
- Gilbert, J. E., & Swanier, C. A. (2008). Learning Styles: How Do They Fluctuate? *Institute for Learning Styles Journal*, 1(1), 29-40.
- Giot, R., El-Abed, M., & Rosenberger, C. (2011). *Keystroke Dynamics Authentication*. Croatia: InTech.
- Golla, M., & Dürmuth, M. (2018). On the Accuracy of Password Strength Meters. *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (pp. 1567-1582). Toronto: ACM.
- Gordon, D. (2010). Forty Years of Movie Hacking: Considering the Potential Implications of the Popular Media Representation of Computer. *International Journal of Internet Technology and Secured Transactions*, 2(1), 59-87.
- Greene, K. K., Kelsey, J., & Franklin, J. M. (2016). *Measuring the Usability and Security of Permuted Passwords on Mobile Platforms*. Gaithersburg: National Institute of Standards and Technology.
- Guo, L., Wan, Q., Wan, C., Zhu, L., & Shi, Y. (2013). Short-Term Memory to Long-term Memory Transition Mimicked in IZO Homo Junction Synaptic Transistors. *IEEE Electron Device Letters*, 34(12), 1581-1583.
- Guo, Y., & Zhang, Z. (2018). LPSE: Lightweight Password-strength Estimation for Password Meters. *Computers and Security*, 73(1), 507-518.
- Hammersley, M., & Traianou, A. (2012). *Ethics in Qualitative Research: Controversies and Contexts*. Los Angeles: Sage Publications.
- Hammond, M., & Wellington, J. (2013). *Research Methods: The Key Concepts*. New York: Routledge.
- Harms, C. (2019). A Bayes Factor for Replications of ANOVA Results. *The American Statistician*, 73(4), 327-339.
- Harte, D., & Law, L. (2019). Incorporating Different Learning Styles Into a Home Exercise Program. *Journal of Hand Therapy*, 32(1), 128-132.

- Haskett, J. A. (1984). Pass-algorithms: A User Validation. *Communications of the ACM*, 27(8), 777-781.
- Hayes, J. (2016). Identifying Fatigue through Keystroke Dynamics. *Journal of Undergraduate Engineering Research*, 9(2), 1-10.
- He, J., Tang, Z., Wu, D., Wu, L., Lin, C., & Luo, F. (2019). Multidimensional Data Security Exchange Modeling and its Optimization. *Energy Science and Power Engineering*, 252(1), 1-7.
- Heartfield, R., & Loukas, G. (2016). A Taxonomy of Attacks and a Survey of Defence Mechanisms for Semantic Social Engineering Attacks. *ACM Computing Surveys*, 48(3), 1-38.
- Heuer, H., & Sanders, A. F. (1989). Reviewed Work: Perspectives on Perception and Action. *American Journal of Psychology*, 102(3), 424-428.
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75-105.
- Hingmire, A., & Saliya, S. (2017). A Multimodal Metric for Password Strength Estimation. *International Journal of Recent Trends in Engineering and Research*, 3(12), 21-30.
- Hofstee, E. (2006). *Constructing a Good Dissertation: A Practical Guide to Finishing a Master's, MBA or PhD on Schedule*. Johannesburg: EPE.
- Holloway, I., & Wheeler, S. (1995). Ethical Issues in Qualitative Nursing Research. *Nursing Ethics*, 2(3), 223-232.
- Holstein, D. K. (2006). Wi-Fi Protected Access for Protection and Automation a work in progress by CIGRE Working Group B5.22. *PES Power Systems Conference and Exposition* (pp. 2004-2011). Atlanta: IEEE.
- Horcher, A.-M. (2018). One Size does not Fit Mobile: Designing Usable Security. *Symposium on Usable Privacy and Security* (pp. 1-5). Baltimore: USENIX.
- Hornbæk, K., & Law, E. L.-C. (2007). Meta-analysis of Correlations Among Usability Measures. *SIGCHI Conference on Human Factors in Computing Systems* (pp. 617-626). San Jose: ACM.

- Hou, L., Wei, L., Wang, C., Wang, A., & Xu, J. (2018). Research on Two-Factor Identity Authentication System Based on Smart Phone and User Password. *International Conference on Cloud Computing and Security* (pp. 712-723). Cham: Springer.
- Houshmand, S., & Aggarwal, S. (2012). Building Better Passwords Using Probabilistic Techniques. *Proceedings of the 28th Annual Computer Security Applications Conference* (pp. 109-118). Orlando: ACM.
- Hsu, C.-C., & Sandford, B. A. (2007). The Delphi Technique: Making Sense Of Consensus. *Practical Assessment, Research & Evaluation*, 12(10), 1-8.
- Hsu, R.-H., Lee, J., Quek, T. Q., & Chen, J.-C. (2018). Reconfigurable Security: Edge-computing-based Framework for IoT. *IEEE Network*, 32(5), 92-99.
- Huh, J. H., Kim, H., Bobba, R. B., Bashir, M. N., & Beznosov, K. (2015). On the Memorability of System-generated PINs: Can Chunking Help? *Symposium On Usable Privacy and Security* (pp. 197-209). Ottawa: USENIX.
- Hussain, T., Atta, K., Bawany, N. Z., & Qamar, T. (2018). Passwords and User Behavior. *Journal of Computers*, 13(6), 692-704.
- Ibrahim, M. H. (2015). AATCT: Anonymously Authenticated Transmission. *International Journal of Advanced Computer Science and Applications*, 6(9), 251-259.
- Indulska, M., & Recker, J. (2008). Design Science in IS Research: A Literature Analysis. *Biennial ANU Workshop on Information Systems Foundations* (pp. 285-303). Canberra: Gregor, Shirley and Ho, Susanna.
- Islam, S., & Atwood, J. W. (2006). A Framework to Add AAA Functionalities in IP Multicast. *Advanced International Conference on Telecommunications/Internet and Web Applications and Services* (pp. 58-64). Guadeloupe: IEEE.
- Jabareen, Y. R. (2009). Building a Conceptual Framework: Philosophy, Definitions, and Procedure. *The International Journal of Qualitative Methods*, 8(4), 49-62.
- Jablon, D. P. (1996). Strong Password-only Authenticated Key Exchange. *SIGCOMM Computer Communication Review*, 26(5), 5-26.
- Jadhav, C., Kulkarni, S., Shelar, S., Shinde, K., & Dharwadkar, N. V. (2017). Biometric Authentication Using Keystroke Dynamics. *International Conference on IoT in Social, Mobile, Analytics and Cloud* (pp. 870-875). Palladam: IEEE.

- Jamaluddin, T. S., & Revett, K. (2012). Automated Generation of Benchmark Data for Keystroke Dynamics Based Authentication Schemes. *Egyptian Computer Science Journal*, 36(1), 79-88.
- Jansen, W. (2004). *Authenticating Mobile Device Users Through Image Selection*. Gaithersburg: National Institute of Standards and Technology.
- Jermyn, I., Mayer, A., Monroe, F., Reiter, M. K., & Rubin, A. D. (1999). The Design and Analysis of Graphical Passwords. *Proceedings of the 8th Conference on USENIX Security Symposium* (pp. 1-14). Washington DC: USENIX.
- Jiang, S. (2018). Network Security in RWNs. *Wireless Networking Principles: From Terrestrial to Underwater Acoustic* (pp. 203-229). Singapore: Springer.
- Jilani, A. A., Usman, M., Nadeem, A., Malik, Z. I., & Halim, Z. (2011). Comparative Study on DFD to UML Diagrams Transformations. *World of Computer Science and Information Technology Journal*, 1(1), 10-16.
- John, B. E., & Kieras, D. E. (1994). *The GOMS Family of Analysis Techniques: Tools for Design and Evaluation*. Pittsburgh: Carnegie Mellon University.
- John, B., Kadavevaramath, R. S., & Edinbarough, I. A. (2017). Designing Software Development Processes to Optimize Multiple Output Performance Characteristics. *Software Quality Professional*, 19(1), 16-24.
- Johnson, J. (2014). Recognition is Easy; Recall is Hard. In J. Johnson, *Designing with the Mind in Mind* (pp. 109-117). Burlington: Morgan Kaufmann Publishers .
- Jorritsma, W., Haga, P. J., Cnossen, F., Dierckx, R. A., Oudkerk, M., & van Ooijen, P. M. (2015). Predicting Human Performance Differences on Multiple Interface Alternatives: KLM, GOMS and CogTool are Unreliable. *6th International Conference on Applied Human Factors and Ergonomics*, 3(1), 3725-3731.
- Kaiser, J., & Reichenbach, M. (2002). Evaluating Security Tools towards Usable Security: A Usability Taxonomy for the Evaluation of Security Tools Based on a Categorization of User Errors. *Proceedings of the IFIP 17th World Computer Congress*, 1(1), 247-256.
- Karwowski, W., Eberts, R., Salvendy, G., & Noland, S. (2007). The Effects of Computer Interface Design on Human Postural Dynamics. *Ergonomics*, 37(4), 703-724.

- Kasiani, , & Yusuf, M. (2019). Developing Ergonomics-based Practice System to Improve Students' Typing Skills. *International Research Journal of Engineering, IT & Scientific Research*, 5(4), 28-37.
- Kävrestad, J., Eriksson, F., & Nohlberg, M. (2019). Understanding Passwords - A Taxonomy of Password Creation Strategies. *Information and Computer Security*, 27(3), 453-467.
- Keith, M., Shao, B., & Steinbart, P. (2007). The Usability of Passphrases for Authentication: An Empirical. *International Journal of Human-Computer Studies*, 65(1), 17-28.
- Keith, M., Shao, B., & Steinbart, P. (2009). A Behavioral Analysis of Passphrase Design and Effectiveness. *Journal of the Association for Information Systems*, 10(2), 63-89.
- Kelley, P. G., Komanduri, S., Mazurek, M. L., Shay, R., Vidas, T., Bauer, L., . . . Lopez, J. (2012). Guess Again (and Again and Again): Measuring Password Strength by Simulating Password-Cracking Algorithms. *Security and Privacy* (pp. 1-15). San Francisco: IEEE.
- Khan, M. A., Pečarić, Đ., & Pečarić, J. (2017). Bounds for Shannon and Zipf-mandelbrot Entropies. *Mathematical Methods in the Applied Sciences*, 40(18), 7316-7322.
- Khodadadi, T., Islam, M. A., Baharun, S., & Komaki, S. (2016). Evaluation of Recognition-based Graphical Password Schemes in Terms of Usability and Security Attributes. *International Journal of Electrical and Computer Engineering*, 6(6), 2939-2948.
- Kim, H., Yib, S., & Yoon, S.-Y. (2019). Exploring Touch Feedback Display of Virtual Keyboards for Reduced Eye Movements. *Displays*, 56(1), 38-48.
- Kim, J. H., Aulck, L., Bartha, M. C., Harper, C., & Johnson, P. W. (2012). Are There Differences in Force Exposures and Typing Productivity between Touchscreen and Conventional Keyboard? *International Annual Meeting of the Human Factors and Ergonomics Society* (pp. 1104-1108). Boston: Sage Publications.
- Kim, J. H., Aulck, L., Thamsuwan, O., Bartha, M. C., & Johnson, P. W. (2014). The Effect of Key Size of Touch Screen Virtual Keyboards on Productivity, Usability, and

- Typing Biomechanics. *Journal of the Human Factors and Ergonomics Society*, 56(7), 1235-1248.
- Kim, J., Kim, H., & Kang, P. (2018). Keystroke Dynamics-based User Authentication Using Freely Typed Text Based on User-adaptive Feature Extraction and Novelty Detection. *Applied Soft Computing*, 62(1), 1077-1087.
- Knight, S. A., & Cross, D. (2012). Using Contextual Constructs Model to Frame Doctoral Research Methodology. *International Journal of Doctoral Studies*, 7(1), 39-62.
- Koester, H. H., & Mankowski, J. (2015). Automatic Adjustment of Keyboard Settings Can Enhance Typing. *Assistive Technology: The Official Journal of RESNA*, 27(3), 136-146.
- Komogortsev, O. V., Mueller, C. J., Tamir, D., & Feldman, L. (2009). An Effort Based Model of Software Usability. *International Conference on Software Engineering Theory and Practice* (pp. 1-9). Orlando: International Society for Research in Science and Technology.
- König, C., Hofmann, T., & Bruder, R. (2012). Application of the User-centred Design Process According ISO 9241-210 in Air Traffic Control. *Work*, 41(1), 167-174.
- Kothe, E. J., Ling, M., North, M., Klas, A., Mullan, B., & Novoradovskaya, L. (2019). Protection Motivation Theory and Pro-environmental Behaviour: A Systematic Mapping Review. *Australian Journal of Psychology*, 1(1), 1–22.
- Kozak, M., Krzanowski, W., Cichocka, I., & Hartley, J. (2015). The Effects of Data Input Errors on Subsequent Statistical Inference. *Journal of Applied Statistics*, 42(1), 2030-2037.
- Kuka, E., & Bahiti, R. (2018). Information Security Management: Password Security Issues. *Interdisciplinary Studies*, 7(2), 43-47.
- Kuo, C., Romanosky, S., & Cranor, L. F. (2016). Human Selection of Mnemonic Phrase-based Passwords. *Proceedings of the 2nd Symposium on Usable Privacy and Security* (pp. 67-78). Pittsburgh: ACM.
- Lasrado, L., Vatrapu, R. K., & Andersen, K. N. (2015). Maturity Models Development in IS Research: A Literature Review. *Information Systems Research Seminar* (pp. 1-12). Oulu: IRIS.

- Lee, A., Song, K., Ryu, H. B., Kim, J., & Kwon, G. (2015). Fingerstroke Time Estimates for Touchscreen-based Mobile Gaming Interaction. *Human Movement Science, 44*(1), 211-224.
- Leino, K., Todi, K., Oulasvirta, A., & Kurimo, M. (2019). Computer-supported Form Design Using Keystroke-level Modeling with Reinforcement Learning. *Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion* (pp. 85-86). Marina del Ray: ACM.
- Lewis, J. (2003). Cyber Terror: Missing in Action. *Knowledge, Technology and Policy, 16*(2), 34-41.
- Li, Y., Wang, H., & Sun, K. (2018). Email as a Master Key: Analyzing Account Recovery in the Wild. *Conference on Computer Communications* (pp. 1646-1654). Honolulu: IEEE.
- Lin, C.-H., Liu, J.-C., & Lee, K.-Y. (2018). On Neural Networks for Biometric Authentication Based on Keystroke Dynamics. *Sensors and Materials, 30*(3), 385-396.
- Lin, C.-J., & Wu, C. (2011). Factors Affecting Numerical Typing Performance of Young Adults in a Hear-and-type Task. *Ergonomics, 54*(12), 1159-1174.
- Liu, C.-T. (2013). *Taiwan Patent No. US8441377B2*.
- Liu, L., Dillon, E., & Zhang, J. (2017). Finding a Holistic Design for Elderly People to Type on Smartphones. *Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments* (pp. 91-95). Island of Rhodes: ACM.
- Loos, L. A., & Crosby, M. E. (2018). Cognition and Predictors of Password Selection and Usability. *International Conference on Augmented Cognition* (pp. 117-132). Cham: Springer.
- Loos, L. A., Ogawa, M. B., & Crosby, M. E. (2019). Impedances of Memorable Passphrase Design on Augmented Cognition. *Human-Computer Interaction* (pp. 84-92). Cham: Springer.
- Ludwig, B. (1997). Predicting the Future: Have You Considered Using the Delphi Methodology? *Journal of Extension, 35*(5), 1-4.

- Lupu, C., & Valeriu, L. (2014). Biometrics Used for Authentication in Internet-banking Applications. *Constantin Brâncuși*, 3(1), 57-63.
- Ma, X., Gong, Y., Gao, X., & Xiang, Y. (2017). The Teaching of Chinese as a Second or Foreign Language: A Systematic Review of the Literature 2005–2015. *Journal of Multilingual and Multicultural Development*, 38(9), 1-16.
- Mahapatra, K. C., & Magesh, S. (2015). Analysis of Vulnerabilities in the Protocols used in SCADA Systems. *International Journal of Advanced Research in Computer Engineering and Technology*, 4(3), 1014-1019.
- Maini, R., Kimmet, T., Cunningham, H., & Vaughan, W. (2013). Passwords: Analyzing Different Kinds of Security and Authentication Methods. *Human Computer Interaction* (pp. 1-6). Clemson: Spring.
- Maoneke, P. B., & Flowerday, S. (2019). Password Policies Adopted by South African Organizations: Influential Factors and Weaknesses. *International Information Security Conference* (pp. 30-43). Cham: Springer.
- Marechal, S. (2008). Advances in Password Cracking. *Journal in Computer Virology*, 4(1), 73-81.
- Marquardson, J. (2012). Password Policy Effects on Entropy and Recall: Research in Progress. *Americas Conference on Information Systems* (pp. 1-9). Seattle: AIS.
- Matta, P., & Pant, B. (2018). TCpC: A Graphical Password Scheme Ensuring Authentication for IoT Resources. *International Journal of Information Technology*, 1(1), 1-11.
- McCarney, D., Barrera, D., Clark, J., Chiasson, S., & van Oorschot, P. C. (2012). Tapas: Design, Implementation, and Usability Evaluation of a Password Manager. *28th Annual Computer Security Applications Conference* (pp. 89-98). Orlando: ACM.
- Melicher, W., Ur, B., Segreti, S. M., Komanduri, S., Bauer, L., Christin, N., & Cranor, L. F. (2016). Fast, Lean, and Accurate: Modeling Password Guessability Using Neural Networks. *Proceedings of the 25th USENIX Security Symposium* (pp. 175-191). Austin: USENIX.
- Meng, W., Zhub, L., Liac, W., Hand, J., & Lie, Y. (2019). Enhancing the Security of FinTech Applications with Map-based Graphical Password Authentication. *Future Generation Computer Systems*, 101(1), 1018-1027.

- Miller, G. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review*, 101(2), 343-352.
- Milton, M. J., Ramakrishnan, B., & Das, R. (2016). Designing a Novel Two-tier Authentication Algorithm for Web Service Architecture. *Telecommunication, Electronic and Computer Engineering*, 8(9), 67-75.
- Mittal, P., & Singh, N. (2016). Speech Based Command and Control System for Mobile Phones: Issues and Challenges. *2nd International Conference on Computational Intelligence and Communication Technology* (pp. 729-732). Ghaziabad: IEEE.
- Mogire, N., Ogawa, M.-B., Minas, R. K., Auernheimer, B., & Crosby, M. E. (2018). Forget the Password: Password Memory and Security Applications of Augmented Cognition. *International Conference on Augmented Cognition* (pp. 133-142). Cham: Springer.
- Molich, R., & Jeffries, R. (2003). Comparative Expert Reviews. *Extended Abstracts on Human Factors in Computing Systems* (pp. 1060-1061). Florida: ACM.
- Monrose, F., & Rubin, A. D. (2000). Keystroke Dynamics as a Biometric for Authentication. *Future Generation Computer Systems*, 16(1), 351–359.
- Monrose, F., Reiter, M. K., & Wetzel, S. (2002). Password Hardening Based on Keystroke Dynamics. *International Journal of Information Security*, 1(1), 69–83.
- Montesdioca, G. P., & Maçada, A. C. (2015). Measuring User Satisfaction with Information Security Practices. *Computers and Security*, 48(1), 267-280.
- Morimoto, C. H., Leyva, J. A., & Tula, A. D. (2018). Context Switching Eye Typing Using Dynamic Expanding Targets. *Proceedings of the Workshop on Communication by Gaze Interaction* (pp. 1-9). Warsaw: ACM.
- Morse, J. M., & Niehaus, L. (2016). *Mixed Methods Design: Principles and Procedures*. New York: Routledge.
- Moumane, K., Idri, A., & Abran, A. (2016). Usability Evaluation of Mobile Applications Using ISO 9241 and ISO 25062 Standards. *SpringerPlus*, 5(1), 548-560.
- Muhamad, W. N., Razali, N. A., Ishak, K. K., Hasbullah, N. A., Zainudin, N. M., Ramli, S., . . . Msaad, N. J. (2019). Enhance Multi-factor Authentication Model for Intelligence Community Access to Critical Surveillance Data. *International Visual Informatics Conference* (pp. 560-569). Cham: Springer.

- Muthiya, P., Padv, S., Patil, D., & Patil, D. (2017). Achieving Flatness: Selecting Honeywords From Existing User Passwords. *International Journal for Research in Engineering Application and Management*, 2(10), 25-27.
- Mwagwabi, F., McGill, T., & Dixon, M. (2018). Short-term and Long-term Effects of Fear Appeals in Improving Compliance with Password Guidelines. *Communications of the Association for Information Systems*, 42(7), 147-182.
- Naor, M., Pinkas, B., & Ronen, E. (2019). How to (Not) Share a Password: Privacy Preserving Protocols for Finding Heavy Hitters with Adversarial Behavior. *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (pp. 1369-1386). London: ACM.
- Narayanan, A., & Shmatikov, V. (2005). Fast Dictionary Attacks on Passwords Using Time-space Tradeoff. *Proceedings of the 12th ACM Conference on Computer and Communications Security* (pp. 364-372). Alexandria: ACM.
- Nicholson, J., Vlachokyriakos, V., Coventry, L., Briggs, P., & Olivie, P. (2018). Simple Nudges for Better Password Creation. *Proceedings of the 32nd International BCS Human Computer Interaction Conference* (pp. 1-12). Belfast: Northumbria Research Link.
- Nikora, M. T., Hunt, T. D., & Ryan, G. (2018). CacophonyViz : Visualisation of Birdsong Derived Ecological Health Indicators. *Journal of Applied Computing and Information Technology*, 22(1), 1-14.
- Novoselov, S., Kudashev, O., Shchemelinin, V., Kremnev, I., & Lavrentyeva, G. (2018). Deep CNN Based Feature Extractor for Text-Prompted Speaker Recognition. *International Conference on Acoustics, Speech and Signal Processing* (pp. 5334-5338). Calgary: IEEE.
- Oates, B. J. (2006). *Researching Information Systems and Computing*. London: Sage Publications.
- Obeidallah, R., Ahmad, A. A., Farouq, F., & Awad, S. (2015). Students Authentication in E-assessment Sessions: A Theoretical Biometric Model for Smartphone Devices. *International Journal of Business Information Systems*, 19(4), 450-464.
- Pansa, D., & Chomsiri, T. (2018). Integrating the Dynamic Password Authentication with Possession Factor and CAPTCHA. *Joint 10th International Conference on Soft*

Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS) (pp. 530-535). Toyama: IEEE.

- Park, Y. S., & Han, S. H. (2010). Touch Key Design for One-handed Thumb Interaction with a Mobile Phone: Effects of Touch Key Size and Touch Key Location. *International Journal of Industrial Ergonomics*, 40(1), 68-76.
- Parsons, K., McCormac, A., Butavicius, M., & Ferguson, L. (2010). *Human Factors and Information Security: Individual, Culture and Security Environment*. Edinburgh: Command, Control, Communications and Intelligence Division.
- Patas, J., Milicevic, D., & Goeken, M. (2011). Enhancing Design Science Through Empirical Knowledge: Framework and Application. *International Conference on Design Science Research in Information Systems* (pp. 32-46). Berlin: Springer.
- Patel, H., & Jha, J. (2015). Securing Data in Cloud Using Homomorphic Encryption. *International Journal of Science and Research*, 4(6), 1892-1895.
- Payne, B. D., & Edwards, K. W. (2008). A Brief Introduction to Usable Security. *IEEE Computer Society*, 12(3), 13-20.
- Paz, F., & Granollers, T. (2019). Redesign of a Questionnaire to Assess the Usability of Websites. *International Conference on Human Systems Engineering and Design: Future Trends and Applications* (pp. 423-428). Cham: Springer.
- Pearman, S., Zhang, S. A., Bauer, L., Christin, N., & Cranor, L. F. (2019). Why People (Don't) Use Password Managers Effectively. *Proceedings of the Fifteenth USENIX Conference on Usable Privacy and Security* (pp. 319-338). Santa Clara: USENIX.
- Pereira, C., Taylor, J., & Jones, M. (2009). Less Learning, More Often: The Impact of Spacing Effect in an Adult E-Learning Environment. *Journal of Adult and Continuing Education*, 15(1), 17-28.
- Perera, C., Zaslavsky, A., Christen, P., Salehi, A., & Georgakopoulos, D. (2012). Capturing Sensor Data from Mobile Phones Using Global Sensor Network Middleware. *23rd International Symposium on Personal, Indoor and Mobile Radio Communications* (pp. 24-29). Sydney: IEEE.
- Pidel, C., & Neuhaus, S. (2019). Breaking: Password Entry is Fine. *International Conference on Human-computer Interaction* (pp. 67-80). Cham: Springer.

- Pilgrim, J., Vasinda, S., Bledsoe, C., & Martinez, E. (2019). Critical Thinking is Critical: Octopuses, Online Sources, and Reliability Reasoning. *The Reading Teacher Journal*, 73(1), 85-93.
- Pleva, M., Bours, P., Ondáš, S., & Juhár, J. (2017). Improving Static Audio Keystroke Analysis by Score Fusion of Acoustic and Timing Data. *Multimedia Tools and Applications*, 76(24), 25749-25766.
- Ponkshe, R. V., & Chole, V. (2015). Keystroke and Mouse Dynamics: A Review on Behavioral Biometrics. *International Journal of Computer Science and Mobile Computing*, 4(2), 341-345.
- Poth, C. N. (2018). *Innovation in Mixed Methods Research: A Practical Guide to Integrative Thinking with Complexity*. London: Sage Publications.
- Priva, U. C. (2010). Constructing Typing-time Corpora: A New Way to Answer Old. *Proceedings of the Annual Meeting of the Cognitive Science Society*. 32, pp. 43-48. Quebec: Curran Associates.
- Proctor, R. W., Lien, M.-C., Vu, K.-P., Schultz, E., & Salvendy, G. (2002). Improving Computer Security for Authentication of Users: Influence of Proactive Password Restrictions. *Behavior Research Methods, Instruments, and Computers*, 34(2), 163-169.
- Punch, K. F. (2006). *Developing Effective Research Proposals*. London: Sage Publications.
- Quezada, A., Ramírez, R. J., Jiménez, S., Noriega, A. R., Inzunza, S., & Garza, A. A. (2017). Usability Operations on Touch Mobile Devices for Users with Autism. *Journal of Medical Systems*, 41(11), 1-11.
- Raghavan, K., Desai, M. S., & Rajkumar, P. V. (2017). Managing Cybersecurity and E-commerce Risks in Small Businesses. *Journal of Management Science and Business Intelligence*, 2(1), 9-15.
- Rajanen, M., & Rajanen, D. (2017). Usability Benefits in Gamification. *GamiFIN Conference* (pp. 87-95). Pori: CEUR.
- Rajkumar, N. M., Dhurka, V., & Kayathri, P. (2016). Survey a Secured Privacy Authentication with Recovery. *World Scientific News*, 37(1), 265-278.

- Rass, S., & König, S. (2018). Password Security as a Game of Entropies. *Entropy*, 20(5), 1-12.
- Raul, N., Shankarmani, R., & Joshi, P. (2019). A Comprehensive Review of Keystroke Dynamics-based Authentication Mechanism. *International Conference on Innovative Computing and Communications* (pp. 149-162). Singapore: Springer.
- Reese, K., Smith, T., Dutson, J., Armknecht, J., Cameron, J., & Seamons, K. (2019). A Usability Study of Five Two-Factor Authentication Methods. *Proceedings of the 15th USENIX Conference on Usable Privacy and Security* (pp. 357-370). Santa Clara: USENIX.
- Renaud, K. (2019). People Need an Incentive to Use Strong Passwords. We Gave Them One. *The Wall Street Journal - Europe Edition*, 1(1), 1-12.
- Renaud, K., & Zimmerman, V. (2017). Enriched Nudges Lead to Stronger Password Replacements...But Implement Mindfully. *Information Security for South Africa* (pp. 1-9). Johannesburg: IEEE.
- Rensing, C., Karsten, M., & Stiller, B. (2002). AAA: A Survey and a Policy-based Architecture and Framework. *IEEE Network*, 16(6), 22-27.
- Revett, K., & Bahaa, A. S. (2012). On the Deployment of Password Hints Using Pre-attentive Visual Priming for One-time Passwords. *International Journal of Computing Academic Research*, 1(2), 66-78.
- Riedl, R., & Fischer, T. (2018). System Response Time as a Stressor in a Digital World: Literature Review and Theoretical Model. *International Conference on HCI in Business, Government, and Organizations* (pp. 175-186). Cham: Springer.
- Rieger, M. (2004). Automatic Keypress Activation in Skilled Typing. *Journal of Experimental Psychology Human Perception and Performance*, 30(3), 555-565.
- Ritter, L. A., & Sue, V. M. (2007). The Survey Questionnaire. *New Directions for Evaluation*, 2007(115), 37-45.
- Robinson, S., Arbez, G., Birta, L. G., Tolk, A., & Wagner, G. (2015). Conceptual Modeling: Definition, Purpose and Benefits. *Winter Simulation Conference* (pp. 2812-2826). Huntington Beach: IEEE.

- Rocha, R., Carneiro, D., & Novais, P. (2019). The Influence of Age and Gender in the Interaction with Touch Screens. *Conference on Artificial Intelligence* (pp. 3-12). Cham: Springer.
- Russell, C., Hogan, L., & Kenny, M. J. (2012). *Ethics for Graduate Researchers: A Cross-disciplinary Approach*. London: Elsevier.
- Saevanee, H., Clarke, N. L., & Furnell, S. (2011). SMS Linguistic Profiling Authentication on Mobile Device. *5th International Conference on Network and System Security* (pp. 224-228). Milan: IEEE.
- Sahin, C. S., Lychev, R., & Wagner, N. (2015). General Framework for Evaluating Password Complexity and Strength. *Computer Science - Cryptography and Security* (pp. 1-11). Ithaca: ArXiv.
- Saini, B. S., Kaur, N., & Bhatia, K. S. (2018). Authenticating Mobile Phone User Using Keystroke Dynamics. *International Journal of Computer Science and Engineering*, 6(12), 372-377.
- Saleh, Z., & Mashhour, A. (2018). Using Keystroke Authentication Typing Errors Pattern as Non-Repudiation in Computing Forensics. *International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies* (pp. 1-6). Sakhier: IEEE.
- Sannicolas-Rocca, T., Schooley, B., & Spears, J. L. (2014). Designing Effective Knowledge Transfer Practices to Improve IS Security Awareness and Compliance. *47th Hawaii International Conference on System Sciences* (pp. 3432-3441). Washington DC: IEEE.
- Santuka, V., Banga, P., & Carroll, B. J. (2010). *AAA Identity Management Security*. Indianapolis: Cisco Press.
- Sasse, A. (2003). Computer Security: Anatomy of a Usability Disaster, and a Plan for Recovery. *Human-computer Interaction and Security Systems* (pp. 1-4). Fort Lauderdale: ACM.
- Sawant, M. M., Nagargoje, Y., Bora, D., Shelke, S., & Borate, V. (2013). Keystroke Dynamics: Review Paper. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(10), 4018-4020.

- Scarfone, K., & Souppaya, M. (2009). *Guide to Enterprise Password Management (Draft)*. Gaithersburg: National Institute of Standards and Technology.
- Schechter, S., Herley, C., & Mitzenmacher, M. (2010). Popularity is Everything A New Approach to Protecting Passwords from Statistical-guessing Attacks. *Proceedings of the 5th USENIX Conference on Hot Topics in Security* (pp. 1-6). Washinton DC: USENIX.
- Schoonenboom, J., Johnson, R. B., & Fröhlich, D. (2018). Combining Multiple Purposes of Mixing within a Mixed Methods Research Design. *International Journal of Multiple Research Approaches*, 10(1), 271-282.
- Schuessler, J. H. (2017). "Chunking" Semester Projects: Does it Enhance Student Learning? *Journal of Higher Education Theory and Practice*, 17(7), 115-120.
- Schulze, R. (2018). Identity and Access Management for Cloud Services Used by the Payment Card Industry. *International Conference on Cloud Computing* (pp. 206-218). Seattle: Springer.
- Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell Systems Technical Journal*, 27(3), 379-423.
- Shay, R., Cranor, L. F., Komanduri, S., Durity, A. L., Huh, P. S., Mazurek, M. L., . . . Christin, N. (2014). Can Long Passwords be Secure and Usable? *Conference on Human Factors in Computing Systems* (pp. 2927-2936). Toronto: AMC.
- Shay, R., Komanduri, S., Durity, A. L., Huh, P. S., Mazurek, M. L., Segreti, S. M., . . . Cranor, L. F. (2016). Designing Password Policies for Strength and Usability. *ACM Transactions on Information and System Security*, 18(4), 1-33.
- Siddiqui, Z., Abdullah, A. H., Khan, M. K., & Alghamdi, A. S. (2014). Smart Environment as a Service: Three Factor Cloud Based User Authentication for Telecare Medical Information System. *Journal of Medical Systems*, 38(1), 1-14.
- Šidlauskas, A. (2018). Users Electronic Data Protection Features. *Social Transformations in Contemporary Society*, 6(1), 78-88.
- Skovgaard, C., Almquist, N. W., & Bangsbo, J. (2018). The Effect of Repeated Periods of Speed Endurance Training on Performance, Running Economy, and Muscle Adaptations. *Scandinavian Journal of Medicine and Science in Sports*, 28(2), 381-390.

- Stanton, J. M., Stam, K. R., Mastrangelo, P., & Jolton, J. (2015). Analysis of End User Security Behaviors. *Computers and Security*, 24(2), 124-133.
- Strock, A., Rougier, N., & Hinaut, X. (2019). Using Conceptors to Transfer between Long-Term and Short-Term Memory. *International Conference on Artificial Neural Networks* (pp. 19-23). Cham: Springer.
- Sulaiman, S., Lokman, N., & Hussien, N. S. (2017). Usability Study on Mobile Web Pre-fetching. *Journal of Telecommunication, Electronic and Computer Engineering*, 9(3), 213-217.
- Tamil, E. M., Othman, A. H., Abidin, A. Z., Idris, M. Y., & Zakaria, O. (2007). Password Practices: A Study on Attitude Towards Password Usage Among Undergraduate Students in Klang Valley. *Journal of Advancement of Science and Arts*, 3(1), 37-42.
- Teh, P. S., Teoh, A. B., & Yue, S. (2013). A Survey of Keystroke Dynamics Biometrics. *The Scientific World Journal*, 2013(1), 1-24.
- Toapanta, M., Maflab, E., & Orizagac, J. (2018). Conceptual Model for Identity Management to Mitigate the Database Security of the Registry Civil of Ecuador. *Materials Today: Proceedings*, 5(1), 636-641.
- Trumbo, C. W. (2002). Information Processing and Risk Perception: An Adaptation of the Heuristic-systematic Model. *Journal of Communication*, 52(2), 367-382.
- Tsai, C.-J., & Shihb, K.-J. (2019). Mining a New Biometrics to Improve the Accuracy of Keystroke Dynamics-based Authentication System on Free-text. *Applied Soft Computing*, 80(1), 125-137.
- Turan, M. S., Barker, E. B., Burr, W. E., & Chen, L. (2010). *Recommendation for Password-based Key Derivation: Part 1: Storage Applications*. Gaithersburg: National Institute of Standards & Technology.
- Ur, B., Noma, F., Bees, J., Segreti, S. M., Shay, R., Bauer, L., . . . Cranor, L. F. (2015). "I Added '!' at the End to Make It Secure": Observing Password Creation in the Lab. *Symposium on Usable Privacy and Security* (pp. 123-140). Ottawa: USENIX.
- Varalakshmi, V. (2015). A Survey on Secure PIN Authentication for ATM Transactions. *International Journal of Advanced Research in Science*, 2(10), 951-954.

- Vertanen, K., Memmi, H., Emge, J., Reyal, S., & Kristensson, P. O. (2015). VelociTap: Investigating Fast Mobile Text Entry Using Sentence-based Decoding of Touchscreen Keyboard Input. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (pp. 659-668). Seoul: ACM.
- Victor, A. N., Dogonyaro, N. M., Victor, Y. L., Meshach, B., & Ayobami, E. (2018). Password Knowledge Versus Password Management. *i-Manager's Journal on Computer Science*, 6(3), 16-24.
- Vittori, P. (2019). Ultimate Password: Is Voice the Best Biometric to Beat Hackers? *Biometric Technology Today*, 2019(9), 8-10.
- Vu, K.-P. L., Tai, B.-L., Bhargav, A., Schultz, E. E., & Proctor, R. W. (2004). Promoting Memorability and Security of Passwords Through Sentence Generation. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 48(13), 1478-1482.
- Walliman, N. (2011). *Research Methods: The Basics*. New York: Routledge.
- Wang, D., & Wang, P. (2015). On the Usability of Two-factor Authentication. *Proceedings of 10th International Conference on Security and Privacy in Communication Networks* (pp. 141-150). Beijing: Springer.
- Wang, D., He, D., Haibo, C., & Ping, W. (2016). Fuzzy PSM : A New Password Strength Meter Using Fuzzy Probabilistic Context-free Grammars Fuzzy PSM : A New Password Strength Meter Using Fuzzy Probabilistic Context-free Grammars. *46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks* (pp. 1-12). Toulouse: ResearchGate.
- Wang, H., Lymberopoulos, D., & Liu, J. (2015). Sensor-based User Authentication. *European Conference on Wireless Sensor Networks* (pp. 168-185). Cham: Springer.
- Weber, R. (2012). Evaluating and Developing Theories in the Information Systems Discipline. *Journal of Associated Information Systems*, 13(1), 1-30.
- Weir, M., Aggarwal, S., Collins, M., & Stern, H. (2010). Testing Metrics for Password Creation Policies by Attacking Large Sets of Revealed Passwords. *Proceedings of the 17th ACM Conference on Computer and Communications Security* (pp. 162-175). Chicago: CCS.

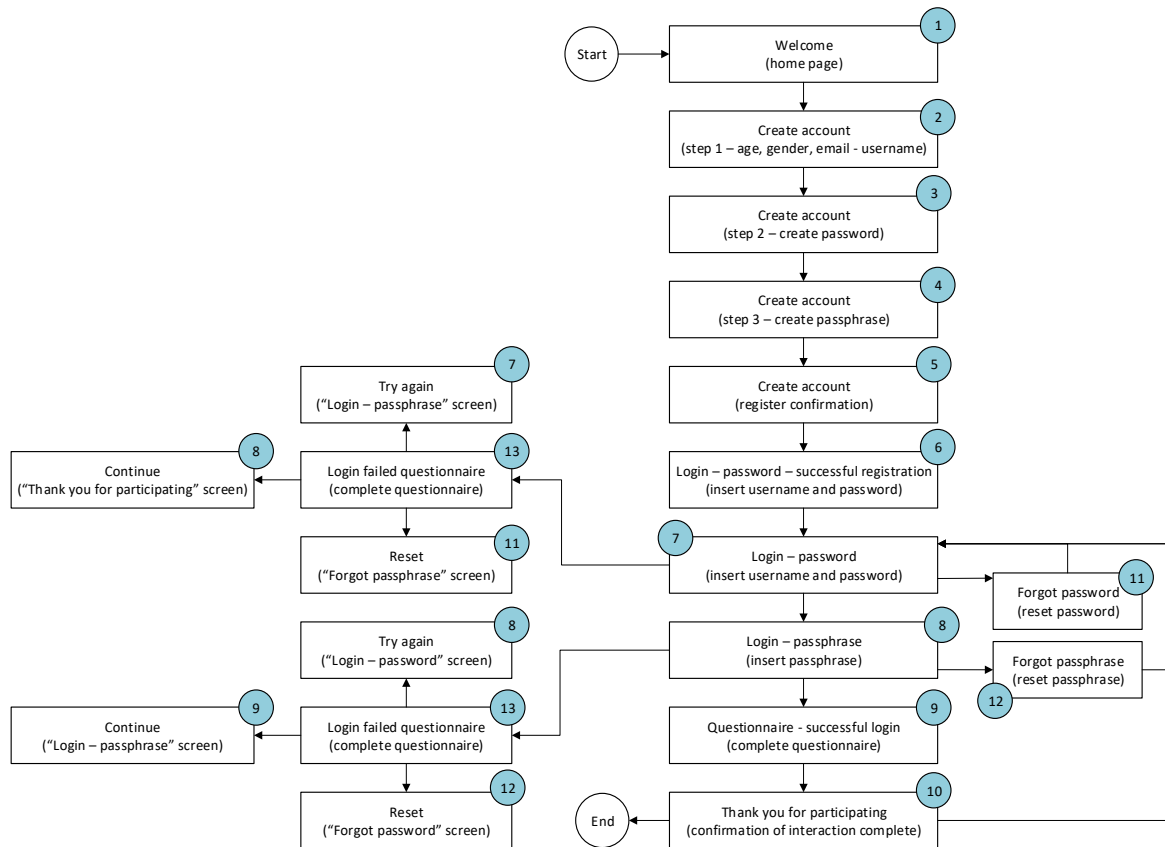
- Weiss, M., Hollan, J. D., & Borchers, J. (2010). Augmenting Interactive Tabletops with Translucent Tangible Controls. In M. C. Tomfelde, *Tabletops - Horizontal Interactive Displays* (pp. 149-170). London: Springer.
- Wiedenbeck, S., Waters, J., Birget, J. C., Brodskiy, A., & Memon, N. (2005). Authentication Using Graphical Passwords: Effects of Tolerance and Image Choice. *Proceedings of the 1st Symposium on Usable Privacy and Security* (pp. 1-12). Pittsburgh: ACM.
- Wijayarathna, C., & Arachchilage, N. A. (2019). Using Cognitive Dimensions to Evaluate the Usability of Security APIs: An Empirical Investigation. *Information and Software Technology*, 115(1), 5-19.
- Wimberly, H., & Liebrock, L. M. (2011). Using Fingerprint Authentication to Reduce System Security: An Empirical Study. *IEEE Symposium on Security and Privacy* (pp. 32-46). Washington DC: IEEE.
- Wixom, B. H., & Todd, P. A. (2005). A Theoretical Integration of User Satisfaction and Technology Acceptance. *Information Systems Research*, 16(1), 85-102.
- Woo, S. S., & Mirkovic, J. (2018). GuidedPass: Helping Users to Create Strong and Memorable Passwords. *Research in Attacks, Intrusions, and Defenses* (pp. 250-270). Cham: Springer.
- Woods, N., & Siponen, M. (2019). Improving Password Memorability, While Not Inconveniencing the User. *International Journal of Human-computer Studies*, 128(1), 61-71.
- Wright, N., Patrick, A. S., & Biddle, R. (2012). Do You See Your Password? Applying Recognition to Textual Passwords. *Proceedings of the Eighth Symposium on Usable Privacy and Security* (pp. 1-14). Washington DC: ACM.
- Yan, J., Blackwell, A., Anderson, R., & Grant, A. (2004). Password Memorability and Security: Empirical Results. *IEEE Security and Privacy*, 2(5), 25-31.
- Yang, Y., Lindqvist, J., & Oulasvirta, A. (2014). Text Entry Method Affects Password Security. *Learning from Authoritative Security Experiment Result* (pp. 11-20). Arlington: USENIX.

- Yazdi, M. A., Negahban, A., Cavuoto, L., & Megahed, F. M. (2019). Optimization of Split Keyboard Design for Touchscreen Devices. *International Journal of Human-computer Interaction*, 35(6), 468-477.
- Yıldırım, M., & Mackie, I. (2019). Encouraging Users to Improve Password Security and Memorability. *International Journal of Information Security*, 18(6), 741-759.
- Yona, Y., & Diggavi, S. (2017). The Effect of Bias on the Guesswork of Hash Functions. *International Symposium on Information Theory* (pp. 2248-2252). Aachen: IEEE.
- Yoon, E.-J., & Kee-Young, Y. (2011). Cryptanalysis of a Simple Three-party Password-based Key Exchange Protocol. *International Journal of Communication Systems*, 24(4), 532-542.
- Zhang, J., Luo, X., Akkaladevi, S., & Ziegelmayer, J. (2009). Improving Multiple-password Recall: An Empirical Study. *European Journal of Information Systems*, 18(2), 165-176.
- Zhang, Z., Zhang, Z., & Yang, Y. (2016). The Power of Expert Identity: How Website-recognized Expert Reviews. *Tourism Management*, 55(1), 15-24.
- Zheng, Z., Cheng, H., Zhang, Z., Zhao, Y., & Wang, P. (2018). An Alternative Method for Understanding User-chosen Passwords. *Security and Communication Networks*, 1(1), 1-12.
- 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(1), 227-237.

Appendix A

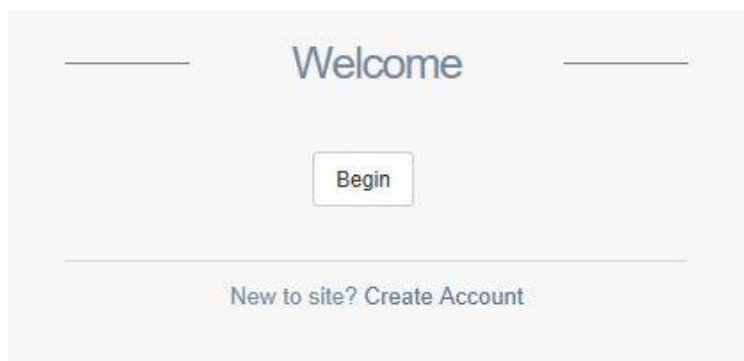
The section provides the screen flow and screenshots of the login assessment website (www.loginassessment18.co.za). Note, the website will be active until 31 December 2019.

Screen flow







Screenshots

Screen 1



Screen 2



   

Create Account

Step 1

Already a member ?
[Log in](#)

Screen 3

Create Account

Step 2

Ensure that the password was not previously used.

- More than 7 characters long
- At least 1 special character (e.g. @, \$, %)
- At least 1 number (e.g. 3, 5, 2)
- At least 1 lowercase letter (e.g. a, h, k)
- At least 1 uppercase letter (e.g. Q, L, G)

Already a member ?
[Log in](#)

Screen 4

Create Account

Step 3

Ensure that the passphrase was not previously used.

Please create a passphrase that meets the following criteria

1. Only lowercase letters
2. 16 or more characters

The stronger the passphrase gets the faster the rabbit will run



Already a member ?
[Log in](#)

Screen 5

— [User] [Eye] [Pencil] [Checkmark] —

Create Account

Complete

You have successfully completed all steps.
Please click register to continue
or go back and rectify any issues

[Previous](#) [Register](#)

Already a member ?
[Log in](#)

Screen 6

Login Form

Successfully registered.
Login with your details below

Email

Password

[Log in](#)

New to site? [Create Account](#)
[Forgot Password](#)

Screen 7

Login Form

Email

Password

[Log in](#)

New to site? [Create Account](#)
[Forgot Password](#)

Screen 8

Passphrase

16+ lower case characters only

Log in

New to site? Create Account
Forgot Passphrase

Screen 9

Logged In

You have Successfully Logged In

Did you use a previously used password/passphrase?

I have previously use the password

I have previously use the passphrase

What device was used to login?

Phone Tablet Computer

What type of password did you prefer using?

Password Passphrase No Preference

Other comments (optional)

Submit

Screen 10

Thank you for participating. Please login again tomorrow.

Back to Login

Screen 11

————— Forgot Password —————

Ensure that the password was not previously used.

- More than 7 characters long
- At least 1 special character (e.g. @, \$, %)
- At least 1 number (e.g. 3, 5, 2)
- At least 1 lowercase letter (e.g. a, h, k)
- At least 1 uppercase letter (e.g. Q, L, G)

Email

Password

Weak

Confirm Password

Screen 12

————— Forgot Passphrase —————

Ensure that the passphrase was not previously used.

Please create a passphrase that meets the following criteria

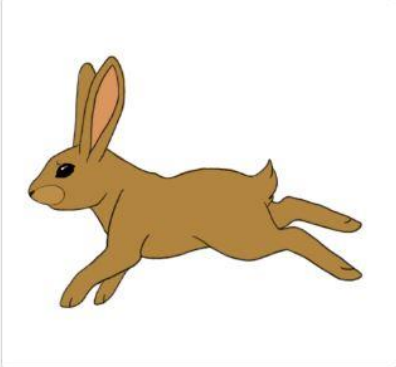
1. Only lowercase letters
2. 16 or more characters

The stronger the passphrase gets the faster the rabbit will run

Email

Enter Passphrase

Confirm Passphrase



[Go Back](#)

Screen 13

Failed Login

You entered your password or passphrase incorrectly at least once. Please complete a short survey so we can understand why.

Did you use a previously used password/passphrase?

I have previously use the password
 I have previously use the passphrase

Was this login failure due to a memory or typing error?

Memory Typing Both

What device was used to login?

Phone Tablet Computer

What type of password do you prefer using?

Password Passphrase No Preference

Other comments (optional)

Appendix B

This section includes the material that was provided to the experts for each round of the expert review.

Instructions

Please complete the questions below with as much detail as possible. There are two sets of questions. Question-set 1 has a total of two questions and question-set 2 has a total of three questions. All questions need to be answered. Once feedback is received from all experts, the researcher will consolidate all feedback and will then be sent back to experts to confirm that the feedback was interpreted correctly. Note that significant inconsistencies in feedback may require feedback from additional questions.

If any further information is required or anything needs to be clarified please contact the researcher by email: bhaveer.bhana@yahoo.com.

Question-set 1

Problem and Proposed Solution

The problem which this study aims to solve is the imbalance between system security and system usability concerning authentication. Current solutions either increase security and reduce usability or reduce security and increase usability.

A two-tier authentication method is proposed to resolve the problem mentioned above. The first tier of authentication allows the user to use a passphrase as a text-based user authentication method instead of a LUDS password. This removes the need for a user to create a password that is more difficult to type and memorise when compared to passphrases which do not include a combination of different character sets such as special characters, uppercase letters, lowercase letters and numbers. By addressing the typing and memorisation issues in the user authentication process, the usability aspect of the user authentication process is improved.

The second tier of authentication is used to further increase security. This study proposes that a keystroke dynamics algorithm be used as the second tier of authentication. Keystroke dynamics is a behavioural biometric authentication method which tracks a user's typing patterns in the backend of a system. Because keystroke dynamics is a backend solution, it has little to no impact on the usability, if implemented correctly. In addition, it provides a significant increase in security. If an unauthorised

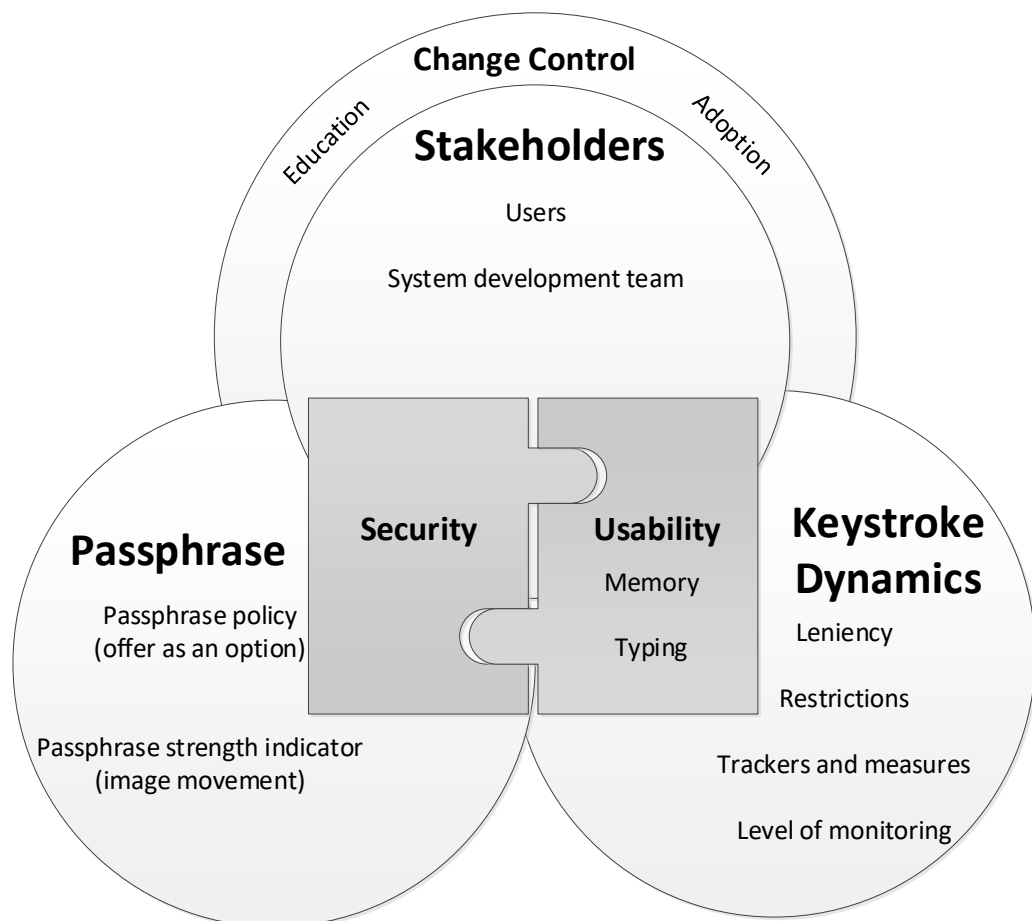
party wants access to a system, he/she will not only need to get the user's passphrase, but will also need to determine how the user enters the passphrase into the system.

Questions for set 1

1. Do you believe this solution will increase security and increase usability? If not, why?
2. What problems do you envision with this proposed solution that could jeopardise the proposed solution's capability of increasing both security and usability?

Question Set 2

Proposed Artefact



Questions for set 2

1. Why do you or don't you believe that this model will increase security?
2. Why do you or don't you believe that this model will increase usability?
3. Why do you or don't you believe that this model will be accepted by users?
4. Do you have any further comments or recommendations?

Appendix C

A journal publication based on this research is under review in the Journal of Computers and Security. The paper is titled "Passphrase and Keystroke Dynamics Authentication: Usable Security".