

**AN ENHANCEMENT ON TARGETED  
PHISHING ATTACKS IN THE STATE OF  
QATAR**

**By**

**Yousef Al-Hamar**

**A thesis submitted in partial fulfilment of the requirements of  
Liverpool John Moores University for the degree of Doctor of  
Philosophy**

**December 2019**

## **DECLARATION**

This dissertation is the result of my own work and includes nothing, which is the outcome of work done in collaboration except where specifically indicated in the text. It has not been previously submitted, in part or whole, to any university or institution for any degree, diploma, or other qualification.

Yousef Khalid Al-Hamar

Word count (Excluding acknowledgement, appendices and references): .... Words

## **ACKNOWLEDGEMENTS**

Firstly, I would like to thank Allah for giving me this opportunity; undertaking this PhD has been a truly life-changing experience for me and it would not have been possible to do without the support and guidance that I received from many people. I would like to extend thanks to the many people, in many countries, who so generously contributed to the work presented in this thesis.

Special mention goes to my enthusiastic supervisor, Dr Hoshang Kolivand. My PhD has been an amazing experience, not only for his tremendous academic support, but also for giving me so many wonderful opportunities especially for sharing his taxonomic expertise so willingly, and for being so dedicated to his role as my supervisor.

Last, but by no means least, thanks go to my mother, father and my siblings for almost unbelievable support. They are the most important people in my world, and I dedicate this thesis to them.

## **ABSTRACT**

The latest report by Kaspersky on Spam and Phishing, listed Qatar as one of the top 10 countries by percentage of email phishing and targeted phishing attacks. Since the Qatari economy has grown exponentially and become increasingly global in nature, email phishing and targeted phishing attacks have the capacity to be devastating to the Qatari economy, yet there are no adequate measures put in place such as awareness training programmes to minimise these threats to the state of Qatar. Therefore, this research aims to explore targeted attacks in specific organisations in the state of Qatar by presenting a new technique to prevent targeted attacks. This novel enterprise-wide email phishing detection system has been used by organisations and individuals not only in the state of Qatar but also in organisations in the UK. This detection system is based on domain names by which attackers carefully register domain names which victims trust. The results show that this detection system has proven its ability to reduce email phishing attacks. Moreover, it aims to develop email phishing awareness training techniques specifically designed for the state of Qatar to complement the presented technique in order to increase email phishing awareness, focused on targeted attacks and the content, and reduce the impact of phishing email attacks. This research was carried out by developing an interactive email phishing awareness training website that has been tested by organisations in the state of Qatar. The results of this training programme proved to get effective results by training users on how to spot email phishing and targeted attacks.

# CONTENTS

<b>CHAPTER 1 INTRODUCTION</b>	<b>13</b>
<b>1.1 INTRODUCTION</b>	<b>13</b>
<b>1.2 PROBLEM BACKGROUND</b>	<b>15</b>
<b>1.3 PROBLEM STATEMENT</b>	<b>18</b>
<b>1.4 AIM</b>	<b>19</b>
<b>1.5 OBJECTIVES</b>	<b>19</b>
<b>1.6 METHODOLOGY</b>	<b>20</b>
1.7 PHASE 1: LITERATURE AND DATA COLLECTION	21
1.8 PHASE 2: ENTERPRISE CREDENTIAL SPEARPHISHING ATTACK DETECTION FRAMEWORK	21
1.9 PHASE 3: REQUIRED AWARENESS	22
1.10 PHASE 4: INTERACTIVE AWARENESS TRAINING PROGRAMME	23
<b>1.11 RESEARCH SCOPE</b>	<b>23</b>
<b>1.12 RATIONALE FOR THIS STUDY</b>	<b>24</b>
<b>1.13 WHO IS THE RESEARCH FOCUSED ON?</b>	<b>25</b>
<b>1.14 RESEARCH CONTRIBUTIONS</b>	<b>27</b>
<b>CHAPTER 2 BACKGROUND AND TERMINOLOGY</b>	<b>29</b>
<b>2.1 INTRODUCTION</b>	<b>29</b>
<b>2.2 PHISHING STRATEGIES</b>	<b>32</b>
<b>2.2.1 MIMICKING ATTACK</b>	<b>32</b>
2.2.2 FORWARD ATTACK AND POPUP ATTACK	32
2.2.3 ROD-AND-REEL METHOD	32
2.2.4 DRAGNET METHOD	32
2.2.5 LOBSTERPOT METHOD	32
2.2.6 GILLNET PHISHING	33
2.2.7 EMAIL PHISHING	33
2.2.8 eFAX	33
2.2.9 INSTANT MESSAGING	33
2.2.10 SOCIAL NETWORK	34
<b>2.3 TECHNIQUE DESCRIPTION</b>	<b>34</b>

2.3.1 BROWSER VULNERABILITIES	34
2.3.2 CLICKJACKING	35
2.3.3 CLOUD COMPUTING	36
2.3.4 CROSS-SITE SCRIPTING ATTACK	38
2.3.5 DRIVE-BY-DOWNLOAD	40
2.3.6 TROJAN ATTACK	41
2.3.7 JAVASCRIPT OBFUSCATION	43
2.3.8 MALWARETISMENT	43
2.3.9 MAN IN THE MIDDLE ATTACK	43
2.3.10 MOBILE PHONE	48
2.3.11 PHISHING KIT	50
2.3.12 SPEARPHISHING	51
2.3.13 SQL INJECTION	52
2.3.14 TABNAPPING	53
2.3.15 TYPO SQUATTING	54
2.3.16 WHALING	55
2.3.17 WIPHISHING AND EVIL TWIN	55
<b>2.4 PHISHING ATTACKS</b>	<b>56</b>
<b>2.5 CONCLUSION</b>	<b>59</b>
<b>CHAPTER 3 LITERATURE REVIEW</b>	<b>60</b>
<b>3.1 INTRODUCTION</b>	<b>60</b>
3.2 TECHNICAL ANTI-PHISHING TECHNIQUE GLOSSARY	62
3.2.1 BLACKLISTS	62
3.2.2 HEURISTICS	62
3.2.3 VISUAL SIMILARITY	63
3.2.4 DATA MINING	63
3.3 TECHNICAL ANTI-PHISHING STRATEGY CASE STUDIES	63
3.3.1 ML-BASED METHODS	64
<b>3.3.2 CLOUD-THREAT INSPECTION APPLIANCE</b>	<b>67</b>
<b>3.3.3 LEARNING-BASED MECHANISMS</b>	<b>68</b>
<b>3.3.4 DATABASE-MATCH WHITELISTING</b>	<b>69</b>
<b>3.3.5 FEATURES ANALYSIS WHITELISTING</b>	<b>69</b>
<b>3.3.6 HEURISTIC APPROACHES</b>	<b>71</b>
<b>3.3.7 STRING-MATCHING</b>	<b>72</b>

<b>3.3.8 URL TOKEN-SYSTEM</b>	<b>73</b>
<b>3.3.9 LEXICAL METHODS</b>	<b>73</b>
<b>3.3.10 INCLUSIVE ANTI-PHISHING TECHNOLOGY</b>	<b>74</b>
<b>3.3.11 PHISHWHO</b>	<b>74</b>
<b>3.3.12 CASE-BASED REASONING</b>	<b>75</b>
<b>3.4 SPEARPHISHING</b>	<b>75</b>
<b>3.5 SPEARPHISHING DEFENCE CASE STUDIES</b>	<b>76</b>
3.5.1 MACHINE LEARNING	76
3.5.2 TEXT MINING	76
3.5.3 EXTRACTING EMAIL FEATURES	77
3.5.4 USER AWARENESS PROGRAMS	79
3.6 SMISHING	80
3.6.1 SMISHING INTRODUCTION	80
3.6.2 ANTI-SMISHING CASE STUDIES AND END-USER BEHAVIOUR	80
3.6.3 ATTACK CLASSIFICATION	81
3.6.4 SMS SPAM FILTERING	81
3.6.5 MALICIOUS FEATURE ANALYSIS	82
3.6.6 USER AWARENESS	82
<b>3.6.7 USER AWARENESS DEFENCE APPROACHES INTRODUCTION</b>	<b>83</b>
3.6.8 THE ROLE OF USER TRAITS AND RELATED STUDIES	86
3.6.9 LEARNING TEMPLATES	88
3.6.10 INTERACTIVE TRAINING GAME	89
3.6.11 EMPIRICAL STUDIES AND COMBINED APPROACHES	89
3.7 EXISTING PROBLEMS AND DISCUSSION	91
<b>3.8 CONCLUSION</b>	<b>93</b>
<b>CHAPTER 4 PROPOSED METHOD</b>	<b>95</b>
<hr/>	
<b>4.1 INTRODUCTION</b>	<b>95</b>
4.2 SPEARPHISHING	95
4.3 ATTACK TAXONOMY	96
<b>4.4 THREAT MODEL</b>	<b>97</b>
<b>4.5 PROPOSED SOLUTION</b>	<b>102</b>
4.6 FEATURE EXTRACTION	103
4.7 DKIM AND SPF	117
<b>4.8.3 COMPLEMENTARY FILTERING AND CHECKS</b>	<b>120</b>

<b>4.9 CONCLUSION</b>	<b>122</b>
<b>CHAPTER 5 EMAIL PHISHING TRAINING WEBSITE</b>	<b>124</b>
<b>5.1 INTRODUCTION</b>	<b>124</b>
<b>5.2 GOALS</b>	<b>124</b>
<b>5.3 METHODOLOGY/DEVELOPMENT PROCESS</b>	<b>125</b>
5.3.1 WHY USE ASP.NET CORE INSTEAD OF OTHER LANGUAGES FOR SERVER SIDE:	125
5.3.2 WHY USE ANGULAR 7 FOR FRONT END:	126
5.3.3 ANGULAR IS SCALED	126
5.3.4 ANGULAR IS TRUSTWORTHY	127
5.3.5 ANGULAR IS FAMILIAR	127
5.3.6 ANGULAR HAS A STRONG ECOSYSTEM	127
5.3.7 WHY USE MYSQL AS DATABASE	127
5.3.8 WHY USE NODE.JS AS MAIL SENDER SERVICE	129
<b>5.4 DATA COLLECTION</b>	<b>129</b>
<b>5.4.1 INTERVIEW GUIDE AND FOCUS GROUP THEMES</b>	<b>130</b>
<b>5.5 RESULTS AND DISCUSSION</b>	<b>130</b>
5.5.1 DATABASE DESIGN AND USER EXPERIENCE	130
5.5.2 INTERACTIVE QUIZ	131
5.5.3 PHISHING AWARENESS TRAINING	131
5.5.4 DATABASE DESIGN	132
5.6 USE CASE DIAGRAM	133
5.7 PROJECT LAYOUT DIAGRAM	133
5.8 BUSINESS PROJECT LAYOUT	133
<b>CONCLUSION</b>	<b>134</b>
<b>CHAPTER 6 TEST AND RESULTS</b>	<b>135</b>
<b>6.1 INTRODUCTION</b>	<b>135</b>
<b>6.2 ENTERPRISE CREDENTIAL SPEARPHISHING ATTACK DETECTION (ECSPAD)</b>	<b>135</b>
6.2.1 LJMU.AC.UK	135
6.2.2 INSTAGRAM.COM	143
6.2.3 ALPINA.QA	151
6.2.4 MOTC.GOV.QA	152
<b>6.3 COMPARISON</b>	<b>155</b>

6.3.1 TRENDMICRO EMAIL PHISHING DETECTION	155
6.3.2 TEST RESULTS	158
6.4.1 INTERVIEW RESULTS	163
<b>6.4.2 DISCUSSION</b>	<b>166</b>
<b>6.5 RECOMMENDATIONS AND STUDY IMPACTS</b>	<b>166</b>
6.5.1 GOVERNMENT	166
6.5.2 ICT SECTOR	167
6.5.3 INSTITUTIONS	167
6.5.4 RESEARCH AND ACADEMIC INSTITUTIONS	167
<b>6.6 CONCLUSION</b>	<b>168</b>
<b><u>CHAPTER 7 CONCLUSION</u></b>	<b><u>169</u></b>
<b>7.1 OVERALL CONCLUSION</b>	<b>169</b>
<b>7.2 RESEARCH CONTRIBUTIONS</b>	<b>172</b>
<b>7.3 DIFFICULTIES AND SOLUTIONS</b>	<b>174</b>
<b>7.4 FUTURE WORK</b>	<b>175</b>
<b><u>REFERENCES</u></b>	<b><u>177</u></b>
<b><u>APPENDIX A DATABASE DESIGN AND USE CASES</u></b>	<b><u>201</u></b>

## List of Figures

FIGURE 1.1 RESEARCH FRAMEWORK	20
FIGURE 2.1 THE RELATION OF THE THREE COMPONENTS IN CLOUD COMPUTING (GRUSCHKA AND IACONO, 2009)	37
FIGURE 2.2 THE CAPTCHA ATTACK (GELERNTER AND HERZBERG 2016)	40
FIGURE 2.3 BOTNET LIFECYCLE (SCHILLER AND BINKLEY, 2011)	42
FIGURE 2.4 MITM ATTACK	44
FIGURE 2.5 THE MITM ATTACK USING A TRANSPARENT PROXY	45
FIGURE 2.6 THE MITM ATTACK USING A PROXY PROGRAM THAT ACTS AS A RELAY TO THE LEGITIMATE WEBSITE	47
FIGURE 2.7 THE DNS POISONING ATTACK	48
FIGURE 2.8 THE FOUR WAYS OF CONTROL TRANSFER IN A MOBILE PHONE (FELT AND WAGNER, 2011)	49
FIGURE 2.9 THE TABNAPPING ATTACK (RASKIN, 2010)	54
FIGURE 2.10 THE WIPHISHING/EVIL TWINS ATTACK	56
FIGURE 3.1 DATA MINING METHODOLOGY (ALEROUD & ZHOU, 2017)	63
<b>FIGURE 3.2 THE PROPOSED PHISHING DETECTION SYSTEM (ZOUINA AND OUTTAJ, 2017)</b>	<b>70</b>
FIGURE 3.3 THE PROPOSED PHISHING TAXONOMY (ALEROUD & ZHOU, 2017)	84
FIGURE 3.4 THE FRAMEWORK OF CROWD EVALUATION AND MEASUREMENT FOR TRUSTWORTHINESS AND SECURITY OF SOCIAL MEDIA PLATFORM BASED ON FEATURED SIGNALS (ZHANG & GUPTA, 2018)	85
FIGURE 3.5 THE ARCHITECTURE OF THE XDROID SYSTEM. (RASHIDI, FUNG, & BERTINO, 2017)	87
FIGURE 4.1 REGISTERED DOMAIN NAME	98
FIGURE 4.2 SENT EMAIL TO USER	99
FIGURE 4.3 LEGITIMATE WEBSITE	99
FIGURE 4.4 CLONE WEBSITE	99
FIGURE 4.5 SNIFFED USERNAME AND PASSWORD	100
FIGURE 4.6 LJMU IT DEPARTMENT RESPONSE	101
FIGURE 4.7 OVERVIEW OF PROPOSED PROTECTION SYSTEM	102
FIGURE 4.8 OVERVIEW OF PROPOSED PROTECTION SYSTEM	104
FIGURE 4.9 SEND IP ADDRESS	106
FIGURE 4.10 CHECK DOMAIN SIMILARITY PROCESS	112
FIGURE 4.11 SCP OUTPUT	113
FIGURE 4.12 NCC OUTPUT	114
FIGURE 4.13 DKIM AND SPF PROCESS	118
FIGURE 4.14 EXAMPLE FROM RECEIVED EMAIL BY GMAIL	120
FIGURE 4.15 EXTRA FILTER CHECK PROCESS	120
FIGURE 5.1 BENCHMARK OF ASP.NET CORE AND OTHER FRAMEWORKS PERFORMANCE COMPARISON	126

FIGURE 5.2 A HOME PAGE OF THE DEVELOPED WEB-BASED TOOL	130
FIGURE 5.3 AN INTERACTIVE QUIZ DESIGN FOR THE USER	131
FIGURE 5.4 PHISHING AWARENESS TRAINING AND LEARNING OBJECTIVES	132
FIGURE 5.5 A DATABASE DESIGN FOR STORING DATA FROM USERS OF OUR WEB-BASED TOOL.	<b>ERROR!</b>
<b>BOOKMARK NOT DEFINED.</b>	
FIGURE 5.6 USE CASE DIAGRAM OF THE PROJECT	<b>ERROR! BOOKMARK NOT DEFINED.</b>
FIGURE 6.1 SCP RESULT FOR CASE STUDY 1	141
FIGURE 6.2 NCC RESULT FOR CASE STUDY 1	142
FIGURE 6.3: CALCULATE SCP RESULT	142
FIGURE 6.4: CALCULATE NCC RESULT	143
FIGURE 6.5: FRAMEWORK RESULT FOR CASE STUDY 1	143
FIGURE 6.6: SCP RESULT	148
FIGURE 6.7: NCC RESULT	149
FIGURE 6.8: CALCULATE SCP RESULT	149
FIGURE 6.9: CALCULATE NCC RESULT	150
FIGURE 6.10 FRAMEWORK RESULT FOR TEST 2	151
FIGURE 6.11: PHISHING EMAIL	156
FIGURE 6.12: PHISHING EMAIL CONTENT	157
FIGURE 6.13: CLONED WEBSITE	157
FIGURE 6.14: USER CREDENTIALS	157
FIGURE 6.15 COMPARISON BETWEEN DIFFERENT METHODS	158
FIGURE 6.16 AN OFFICE 365 EMAIL TEMPLATE AND THE OFFICE ONEDRIVE LOGIN PAGE	160
FIGURE 6.17 PERCENTAGE OF PEOPLE THAT GET FOOLED BY EMAIL PHISHING	160
FIGURE 6.18 THE NUMBER OF PEOPLE WHO WERE FOOLED USING DIFFERENT OPERATING SYSTEM	161
FIGURE 6.19 SECOND CAMPAIGN TO INVESTIGATE THE PEOPLE’S UNDERSTANDING ON PHISHING	162
FIGURE 6.20: A SNAPSHOT OF WEB-BASED TOOL STEPS FOR RAISING PHISHING AWARENESS TO ORGANIZATIONAL EMPLOYEES	162
FIGURE 6.21: CHALLENGES OF IMPLEMENTING EMAIL PHISHING AWARENESS TRAINING PROGRAMMES IN QATAR	163
FIGURE 6.22: LEVEL OF PUBLIC KNOWLEDGE REGARDING EMAIL PHISHING	164
FIGURE 6.23: RESPONSE ON EMAIL PHISHING AWARENESS TRAINING PROGRAMMES IN QATAR	165
FIGURE 6.24 :EVALUATION OF THE PROPOSED EMAIL PHISHING FRAMEWORK FOR RAINING AWARENESS	166
FIGURE 5.5 A DATABASE DESIGN FOR STORING DATA FROM USERS OF OUR WEB-BASED TOOL.	201
FIGURE 5.6 USE CASE DIAGRAM OF THE PROJECT	202
FIGURE 5.7 THE PROJECT’S LAYOUT	203
FIGURE 5.8 THE MAIN CLASSES OF BUSINESSES	204

## List of Table

TABLE 1.1: QUESTIONS THAT EXPERTS WERE ASKED IN ORDER TO GATHER USEFUL INFORMATION ABOUT EMAIL PHISHING	25
TABLE 1.2: PERCEIVED OR ANTICIPATED IMPACT OF THIS RESEARCH TO THE RELEVANT STAKEHOLDERS IN QATAR	26
TABLE 2.1: CUSTOM URL XSS ATTACKS (OLLMANN, 2004)	39
TABLE 2.2: URL OBFUSCATION	46
TABLE 2.3: TYPO SQUATTING EXAMPLE	54
TABLE 6.1: CASE STUDY 1	137
TABLE 6.2: RESULT FOR CASE STUDY 1	138
TABLE 6.3: VALID DOMAIN NAME (LJMU.AC.UK)	138
TABLE 6.4: INCOMING DOMAIN NAME (LJMUAC.UK)	139
TABLE 6.5: CASE STUDY 2	145
TABLE 6.6: RESULT FOR TEST 2	145
TABLE 6.7: VALID DOMAIN NAME (INSTAGRAM.COM)	146
TABLE 6.8: INCOMING DOMAIN NAME (INSATGARM.COM)	146
TABLE 6.9: VALID DOMAIN NAME (ALPINA.QA)	151
TABLE 6.10: INCOMING MAIL DOMAIN (ALPNIA.QA)	151
TABLE 6.11: ALPINA.QA TEST	152
TABLE 6.12: VALID DOMAIN (MOTC.GOV.QA)	153
TABLE 6.13: INCOMING MAIL DOMAIN (MOTCOGV.QA)	154
TABLE 6.14: MOTC.GOV.QA TEST CASE	154
TABLE 6.15: TEST AND RESULTS	158

## LIST OF ABBREVIATIONS AND ACRONYMS

AP- access point	Phishing Web Sites	ECC- elastic cloud computing
APT- Advanced Persistent Threat	CDS- check domain similarity	ECSPAD - Enterprise Credential spear phishing Attack Detection
APWG- anti phishing group	CEO- chive executive officer	ECSPAD - Enterprise Credential spear phishing Attack Detection
BEM- business email comprised	CFC- complementary filtering checks	ECSPAD- enterprise credential spear phishing targeted attack detection
BVM – ball vector machine	CSS- cascaded style sheet	FE- feature extraction
C&C- command and control	DKIM- domain key identification mail	FTP- file transfer protocol
CANTINA- Content-Based Approach to Detecting	DNS- domain name system	HMM- hidden markov model
HTML- hyper-text mark-up language	MAAS- malware as a service	PSO- Practical support Optimization
IM- Insta messaging	MIM- man in middle	RF- random forest
IRC- internet relay chat	ML- machine learning	SAAS- software as a service
ISP- Internet service provider	NCC- number of common character	SCP- similar character place
IT- information technology	NCSC- national cyber security center	SCP- similar character place
KNN- key nearest neighbor	NN – neural network	SMS- short message service
LAAS- infrastructure as a service	OS- operating system	SOP- same origin policy
LJMU- Liverpool	PC- Personal computer	SPF- sender policy framework
IOT- internet of things	PHP- personal home page	SQL- structure query language

SSID- service set identifier	TAD- target identification domain	locator
SSL- secure socket layer	UI- user interface	US- United States
SSO- single sign on	UK- United Kingdom	XSS- cross site script
SVM- support vector machine	URL- universal resource	

## **LIST OF APPENDICES**

**NO TABLE OF FIGURES ENTRIES FOUND.**

# CHAPTER 1

## INTRODUCTION

### 1.1 Introduction

Cybercrime is a growing problem with an ever-increasing risk due to the vast number of connected devices such as smartphones and the Internet of Things (IoT). There has been a dramatic increase in the daily use of smartphones (Heinze et al., 2016) for social Internet surfing, gaming and social networking and for transaction activities such as banking and emails. The use of emails on smartphones has increased over 180% in the last three years (Heinze et al., 2016). By 2020, banking customers will use mobile devices to manage their current accounts 2.3 billion times – more than desktop PCs, branch and telephone banking put together. However, there are security issues associated with emails, ranging from viruses to malware to other sophisticated threats such as email phishing. Email phishing can lead to data breaches and hence damage to business's reputation.

Email phishing awareness must continue evolving to address the advancement of targeted email attacks. This awareness of targeted attacks should be provided not just to users, but also to employees of any targeted businesses. It is also vital to provide awareness to law enforcement personnel responsible for investigating email phishing attacks (Davinson and Sillence, 2010). Customers ought to be aware of the advancement and sophistication used by email phishers of technical deceit in phishing emails that make them very difficult to detect. Organisations and businesses are also required to be aware of the advancement of phishing attacks in order to design effective awareness training programmes for their employees, this will reduce email phishing targeted attacks (Aldawood and Skinner, 2018) (Skopik et al., 2017)

Phishing email is among the highest ranked cybercrimes, costing organisations an average of \$3.7 million per year in the USA (Kamat and Gautam, 2018). The British government-backed cybersecurity body reported that phishing attacks during 2015 cost British consumers a total of £174.4 million (Kamat and Gautam, 2018). The economy of Qatar is now another target of phishing cybercrime. Oil and gas industries operating in Qatar are increasingly becoming the targets. It is estimated that Qatar will spend \$25bn

in increasing chemical and petrochemical production (Tabassum et al., 2018). This spending is likely to increase during the FIFA World Cup in 2022 in Qatar.

Due to the severity of email phishing attacks, the UK government through its National Cyber Security Centre (NCSC) introduced the Phishing Guidance (Skopik et al., 2017). This guidance is for organisations of all sizes and in all sectors. The guidance provides advice and recommendations on how the UK organisations can defend themselves against phishing email attacks. It proposes a multi-layered approach that if deployed, can effectively improve an organisation's resilience against email phishing, whilst mitigating risks and disruptions to personnel productivity. The US government departments such as the Federal Trade Commission and Security and Exchange Commission have also provided guidelines for the US organisations in order to mitigate the risks of email phishing attacks.

Most of the existing solutions that are deployed to detect phishing emails are technical in nature (Jakobsson, 2018; Gascon et al., 2018; McElwee et al., 2018). They are deployed on the personal computers and smartphones to monitor and filter phishing emails. The investment in personnel training on email phishing as a preventive strategy together with proper utilization of technical solutions can yields better results. Therefore, the first step to reduce email phishing attacks is to implement email phishing awareness programmes to help email users to avoid all the attacks associated with email phishing. Although awareness training programmes will not completely stop the risks of the email phishing attacks, they will effectively reduce them (Mouton et. al., 2016). Awareness training programmes are known as proactive solutions and hence, they are more effective with low investment costs (Mohammad et al., 2015).

Cyber criminals, who use targeted phishing attack (Spearphishing) techniques, divide their victims, personalize the emails, impersonate specific senders and use other techniques to bypass the detection systems. Their mission goal is to convince the targets to click on a link or open an attachment. The difference between a phishing campaign that may use an email database to send their emails and Spearphishing attacks is to choose targets within a specific organisation with a specific task (Suganya, 2016).

Since email is one of the most commonly used media of communication in either private or official use, it has been widely used by attackers for phishing. This research focuses

on email phishing threats and how to effectively reduce these threats based on two complementary strategies. The first and former is to present a new technique to prevent being attacked. In this, path, we have presented a new technique, ECSPAD (Enterprise Credential Spearphishing Attack Detection) to detect targeted Spearphishing attacks. ECSPAD has been tested based on different enterprise email security systems such as TrendMicro, Yahoo, Outlook and Gmail. Based on the results, we proved that the proposed method has a much better detection rate than most of the existing systems. The complementary strategy is to present an interactive website awareness which trains users by conducting a quiz and appropriate interactive training programme for employees and citizens in the state of Qatar, and to propose a system that can be used by consumers and business to detect phishing emails in the State of Qatar. As detailed in chapter 2, Qatar has recently been one of the targets witnessing an increase in email phishing attacks due to its economic prosperity (Varghese, J. 2017).

## **1.2 Problem Background**

The phishing emails are usually framed to ensure that people perceive them to be genuine and are deceived to see them as such. This is why they are usually framed with brands that people trust when in reality they are but fakes only out to steal the identity of people who have access to information that the attackers want (Lininger and Dean, 2005).

Email phishing attacks are increasingly becoming a big concern to people, governmental and non-governmental organisations. Web browsers and email applications have become more sophisticated in detecting email phishing, yet despite all these advances, people who use these services are not aware and there is little in terms of awareness campaigns or research that prevent people from being duped or being victims of phishing emails or websites. Therefore, it is important that people are aware of the existing email phishing threats and the most common techniques that are used by attackers, to be able to protect themselves from those types of phishing attacks and to understand how they should all be contributing to ensure security. (Flowerday and Tuyikeze, 2016). It has been proven that employees should be trained and become aware of email phishing threats. If this is not achieved, then threat levels will not be contained and any detection system that is meant to secure organisations will always be inadequate (Okeke and Shah, 2016).

The 2016 report by Kaspersky on Spam and Phishing listed Qatar as one of the top 10

countries by percentage of user attacks (Gudkova et al., 2016). The increasing number of phishing attacks in Qatar has prompted researchers in academia and industry to recommend email phishing awareness training programmes (Al-Mohannadi et al., 2018).

Spearphishing is the most prevalent delivery method for Advanced Persistent Threat (APT) attacks. By using such methods and harvesting social networks for personal information about the targets, an attacker can make emails that are extremely accurate and compelling. Once the email has bypassed the target detection system and the target clicks on a link or opens an attachment, the attacker will establish a foothold in the network, enabling them to fulfil their attacks (Ho et al., 2017).

Laszka et al. (2016) suggested using email filters to mitigate Spearphishing attacks, which work via a maliciousness score above a chosen threshold. However, an optimal choice such as threshold value, involves a trade-off between the risks from delivered malicious emails and the cost of blocking benign traffic. In addition, email scoring cannot be reliable as an attacker may configure an email server to achieve the lowest score possible, therefore such methods cannot be used on APT attacks.

In addition, another way that email phishing threats can be minimised is to make sure that users are provided with effective awareness training in order to be conversant with security mechanisms that are in place (Singh and Pandey, 2014). It must be made aware that security is everyone's responsibility. Otherwise, the potential of email phishing to be detrimental increases exponentially, which might be catastrophic to the individual and potentially to the organisation (Alotaibi et al., 2016). Furthermore, staff should be made aware how they should behave at work and how information is accessed and shared (Da Veiga and Eloff, 2010). Organisations should also ensure that there are strict guidelines on how information and data should be used to ensure that the organisation's resources are not misused (Teufel et al., 2018). Training awareness programmes should not be general, rather they should be tailored to the organisation and it is vital that every single user is made aware of these, and this should be an ongoing programme to remind staff. This is critical because the threats posed by email phishing and the tools that they use are always evolving and getting more sophisticated (Ma et al., 2008). This will ensure that staff are empowered to allow them and the organisation to effectively fight the threat from email phishing.

In fact, there is a consensus that there is a need to provide more adequate cyber information security in Qatar to provide awareness information, guidance and training for employees to and ensure that the citizens become aware of the threats from hackers, using a variety of techniques such as phishing emails. There is a notion that Qatar is a soft touch making it a target for phishing emails because of the distinct prevalent lack of security information. This is very concerning to stakeholders and the government.

Al-Hamar et al., (2011) conducted research that reviewed the threat posed by email phishing to Qatar and compared this to the threat that email phishing posed to the UK. She concluded that the threat posed by phishing is increasing and it is becoming very common in Qatar. The study concluded that one of the reasons for phishing emails attacks in Qatar being successful is the distinct gullibility or rather the tendency for Qataris to be fooled online. This is because of a lack of awareness of email phishing threats and how these can be avoided. This still is a concern, because it is an acceptable fact that hackers using phishing email methods are able to easily obtain personal information that allows them to steal the victim's identity through the internet.

According to Choi et al., (2017), there is a direct threat that email phishing has on world financial institutions. The Qatari economy has grown exponentially and become increasingly global in nature. Therefore, email phishing has the capacity to be devastating to the Qatar economy, yet there are no adequate measures put in place such as awareness training programmes to minimise this threat to the state of Qatar.

This threat posed by email phishing is emphasised by Al-Hamar and Khalid (2010), where they concluded that email phishing is a major issue in Qatar. They attributed this to lack of awareness training programmes. The study also focussed on the perception of Qatar as a soft touch for such attacks because of a lack of awareness and of the laws that make it easier for email phishing attacks. Some of the key facilitators for email phishing is the beliefs, culture and personal traits of Qataris. Therefore, email phishing is a major concern. This justifies the reason for this study.

In Qatar, email phishing is considered as a big problem and has an impact on its society and the government institutions. The economic growth in all sectors in which the Internet plays a big role has attracted email phishers to commit crimes. In most of the research that has been done so far, there is a lack of email phishing awareness programme in Qatar.

Therefore, one of the objectives of this research is to develop an email phishing awareness training framework to be used by organisations in the state of Qatar. The framework will help to effectively reduce the email phishing threats in Qatar. This framework will also help Qatar government institutions and organisations to enhance email phishing awareness for their employees through a set of recommendations.

### **1.3 Problem Statement**

In the first quarter of 2017, it was reported that Qatar had faced a total of 93,570 phishing attacks from January 1 to March 31. These figures were revealed by the senior security researcher (Varghese, J. 2017) of Kaspersky Lab. These figures were reported in Vienna, Austria at a cyber-security event organised by Kaspersky Lab.

Speaking to Gulf Times at the event, Assolini said, “In 2016 Kaspersky Lab products blocked more than 268,000 phishing attacks targeting users in Qatar. In the first quarter of this year, we have observed that such attacks were on the increase in the country and the number as of March 31 is 93,570.”

It was also reported at the same event that “targeted phishing attack is also on the increase in Qatar over the smartphones as mobile phones are a key target of the cyber hackers of late. Most people do not have any protection on their mobiles, and they are prone to more such attacks. According to the expert, cyber criminals are now using a new form of targeted phishing attack called Smishing, using SMS service. They send targeted SMS announcing some promotions or other news using a hyperlink to access the particular site. If the person clicks on the link, the cyber criminals can easily access all the information on the phone.”

Spearphishing attack still is the most preferred vector for instigating targeted attacks. This is because, there is a lack of detection systems that can detect such attacks specially when attackers use Advanced Persistent Threat (APT) and the targeted users continue to fall prey to spearphishing emails, causing substantial damage to their respective organizations. Attackers will use the available organisational information on the Internet to gather necessary information about the chosen targets and make their APT more effective (Gupta et al, 2017; Krombholz et al, 2015; Parsons et al, 2015).

## **1.4 Aim**

This research aims to explore targeted attacks in specific organisations and presents a new technique to prevent targeted attacks by filtering the domain of the sender. Moreover, it aims to develop an email phishing awareness training programme specifically designed for the state of Qatar in order to increase email phishing awareness, focused on targeted attacks and the content, and reducing the impact of phishing email attacks.

## **1.5 Objectives**

The aims will be achieved by the following objectives,

- To conduct a thorough literature review on phishing emails impacts in the state of Qatar and the world in general
- To present a new technique for spearphishing targeted attacks, taking domain similarity into account
- To design and develop an email phishing training awareness programme to be used by organisations in the state of Qatar
- To develop an enterprise email phishing detection system to be used by organisations and individuals in the state of Qatar.
- To evaluate the effectiveness of the proposed email phishing awareness programme and the enterprise wide detection system in the state of Qatar

## 1.6 Methodology

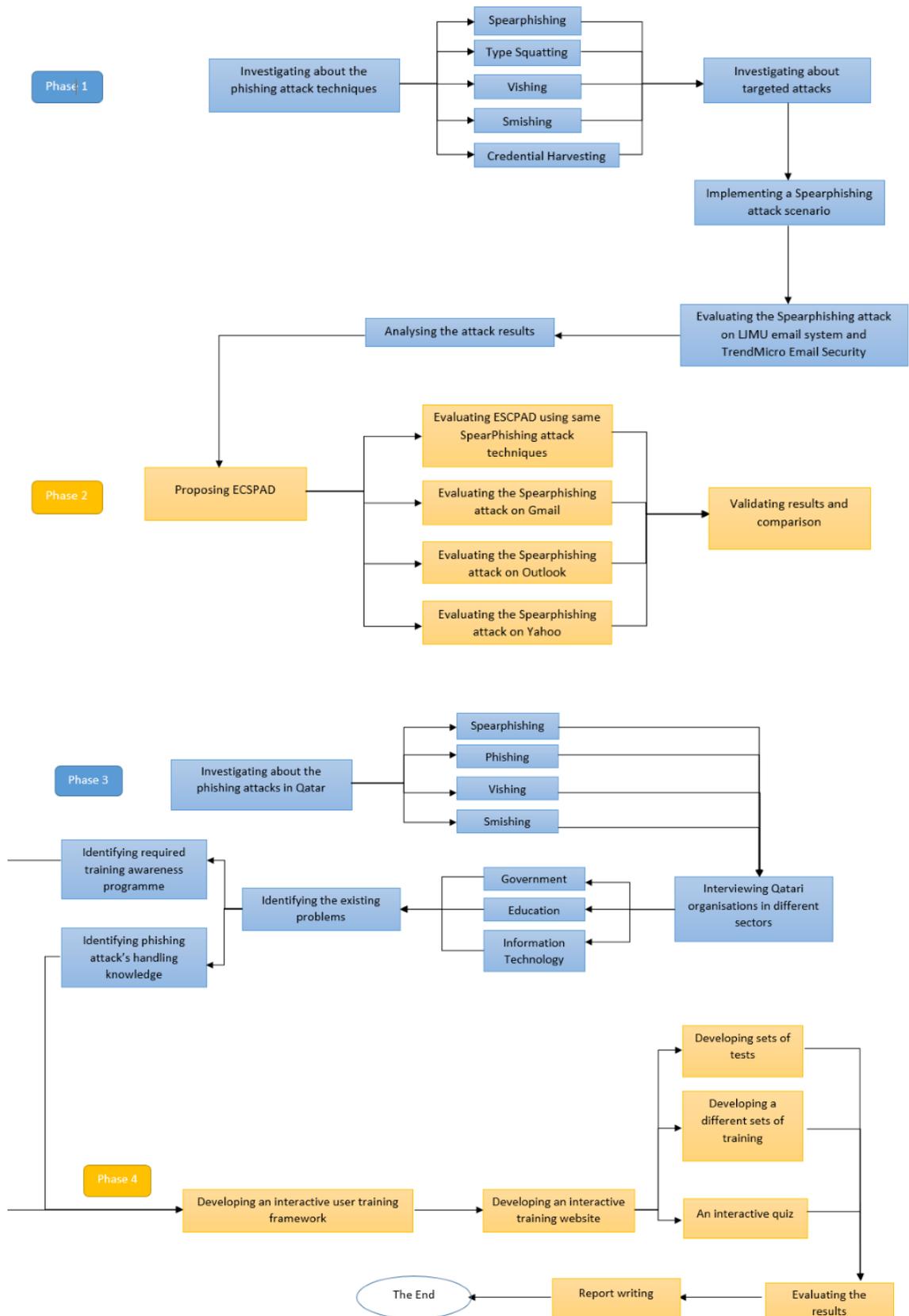


Figure 1.1 Research Framework

To achieve the aims and objectives highlighted in Chapter 1, the following methodology is introduced for a successful achievement. This chapter will discuss the process of completing each objective (Figure 1.1).

The evaluation and analysis of the phishing attack techniques and detections using the proposed framework includes the following phases:

## **1.7 Phase 1: Literature and Data Collection**

A detailed background research on Phishing attack techniques has been done to find all techniques that can be used by an attacker to perform a phishing attack. Then we did detailed research on five techniques to gather technical information of how attackers will use those techniques to perform an attack.

Afterwards, we investigate further into targeted phishing attacks, or spearphishing. In this type of attack, attackers will carefully choose their targets and will use a mixture of Spearphishing, Type Squatting and Credential Harvester techniques to launch an attack that can have high chances of success. In order to have a better understanding of the technical side, we implanted a targeted attack on Liverpool John Moores University account (my own account as an example) using Spearphishing, Type Squatting and Credential Harvester techniques.

Once the implementation step was done, we then performed an attack on our own email account using a same domain name to Liverpool John Moores University but with one “.” less (LJMU domain = ljmu.ac.uk – our domain = ljmuac.uk).

Afterward, we analysed the whole attack scenario with the results gathered from the attack.

## **1.8 Phase 2: Enterprise Credential Spearphishing Attack Detection Framework**

By reviewing Phase 1 and the results of the performed attack, we proposed an Enterprise Credential Spearphishing Attack Detection Framework that can detect a targeted attack when the attacker uses a mixture of different techniques, which leads to an advanced

phishing attack. This type of attack is very hard to detect because the attacker carefully has designed them to be successful 99% of the time. The proposed method can detect a targeted attack when the attacker uses an advanced phishing attack (Spearphishing, Type Squatting and Credential Harvester techniques). We evaluated the proposed system with different domains and different scenarios in different organisations and countries to get more results to justify our method. In addition, we tested TrendMicro Email Security Systems, Gmail, Outlook and Yahoo. Once all results were validated then we compared it to other available standard and widely used email security systems. The results showed that only the Gmail phishing email detection system could detect the attack. Therefore, we achieved one of our objectives, which is to develop an enterprise email phishing detection system that is effective, able to detect Enterprise Credential Spearphishing attacks and can be used by organisations and individuals in the state of Qatar.

### **1.9 Phase 3: Required Awareness**

The research commenced by investigating phishing attacks. Four main phishing attacks were investigated, these are spear phishing, phishing, vishing and Smishing. Smishing uses mobile systems, in particular, the technology of short message service (SMS) to gain private information or identity. A misleading SMS is sent to a victim requesting them to disclose their confidential information or refers them to a fake link to disclose their confidential information. Vishing is the fraudulent practice of making phone calls or leaving voice messages purporting to be from reputable companies in order to induce individuals to reveal personal information, such as bank details and credit card numbers. Spear phishing refers to an attack targeted specifically against a group, organisation or individual. Spear phishing has a far higher success rate because the content of the phishing email is tailored to the receiver, therefore it is less likely to arouse suspicion.

After gaining knowledge of these types of phishing, the next task was to interview personnel in Qatari organisations of different sectors. The interviews were carried out through a questionnaire and face to face meetings. The sectors interviewed were education, information technology and personnel from the Qatar main government. These sectors were chosen because they depend heavily on email communication to carry out their daily work from within and outside Qatar. One of the designed questions was to ask them if there were any problems facing their organisations in terms of phishing attacks. It was revealed that there were three main problems, 1) there was a lack of awareness training programmes for their personnel and the public in general. If these training

programmes were to be implemented, then they could reduce the phishing attacks. If they were to develop the awareness training programme, another problem they were to face is 2) how to develop an effective interactive awareness training programme. Another problem, which was noted, was 3) the knowledge on how to handle phishing attacks if they occur. The major phishing problem revealed through the interview was the email phishing.

### **1.10 Phase 4: Interactive Awareness Training Programme**

After studying the revealed problems through the interview, the next phase was to develop the awareness training programme for Qatar government institutions and the public in general. This development was based on the website, the training and testing were designed to be interactive to give users a feel and demonstration of what email phishing entails. Website users are given an opportunity to answer some basic questions through a questionnaire before starting the training. This will help to present users with appropriate training based on their level. After the interactive training, users are awarded with some points and later on are asked to answer some interactive questions to assess their knowledge after the training. Users are given more options to repeat the questions whenever they prefer until they are satisfied with their results. The effectiveness of the awareness training programme will be evaluated by how users perform after their training sessions.

### **1.11 Research Scope**

This research focuses only on one type of phishing, email phishing. This is because email phishing is widely used for phishing. The target country is the state of Qatar, this is because Qatar has been witnessing an unprecedented increase in phishing email attacks due to its success in economic development. The convenience of access to government organisations and the assumptions that Qatar lacks email phishing awareness training programmes are other factors that led the author to choose Qatar as a basis of this research.

There are several technical solutions proposed in the literature to detect and defend email phishing attacks, however, these techniques cannot detect and stop genuine looking phishing emails. Email phishing does not attack devices, but it attacks human perception which is hard to defend using technical solutions alone. Therefore, awareness training

programmes are essential to effectively reduce the success of email phishing attacks. This research develops a novel awareness training framework for Qatari citizens and organisations with a proposed technical solution to detect email phishing attacks.

In Spearphishing targeted attacks, attackers choose individual targets within a specific organisation with a specific task. A Spearphishing targeted attack, involves sending spoofed emails, which ask the victim to follow links and enter their credentials into a malicious website hosted by the attacker. This spoofed site is designed to appear indistinguishable from a recognisable trusted site. Attachment-driven Spearphishing struggles to succeed against many email providers' malware-filtration systems, which proactively check emails for malicious software. In this research we proposed a method to detect such attacks.

## **1.12 Rationale for this study**

This study was carried out to find answers to the following research questions:

- What are targeted phishing attacks and what has been done to tackle them?
- How can Spearphishing attacks be controlled?
- What is the impact of targeted email phishing in organisations in Qatar?
- What are the needs of organisations in Qatar to reduce the targeted phishing attacks?
- How can we contribute to the prevention of targeted phishing attacks in Qatar?

It is hypothesised that these are questions that can only be answered by presenting a strategy to prevent targeted attacks and improve the awareness of people in Qatar, especially those whose duty it is to maintain and secure computer systems. Hence, these experts and stakeholders were contacted in order to extract useful information and data that could help in finding answers to the research questions. The questions that were asked are shown in Table 1.1

The whole research was premised on the belief that it was possible to get all the data and information that was necessary in order to correctly answer the questions that are listed in Table 1.1.

**Table 1.1: Questions that experts were asked in order to gather useful information about email phishing**

NUMBER	QUESTION
1	Are there any email phishing awareness training programmes in Qatar?
2	What are the challenges and problems facing organisations in implementing email phishing awareness training programmes?
3	What are the contributing factors affecting email phishing in the state of Qatar?
4	What are the counter processes that are available to sectors to detect email phishing and to limit its impact?
5	What is the level of public general knowledge, especially employees of organisations about the threat of email phishing in Qatar?
6	If there are email phishing awareness training programmes in some organisations, how could these programmes be improved?

### **1.13 Who is the research focused on?**

This is a significant research program especially due to the capacity to improve the email phishing attacks both in terms of filtering the emails and enhancing the awareness to reduce the chances of attacks in the state of Qatar. This research is targeted at the public,

the businesses and relevant organisations in the state of Qatar. The relevance of this research to the relevant sectors of Qatar economy is shown in Table 1.2.

**Table 1.2: Perceived or anticipated impact of this research to the relevant stakeholders in Qatar**

STAKEHOLDER	REASON
<b>GOVERNMENT</b>	Key reason for conducting this research is based on the fact that email phishing is increasingly becoming more sophisticated and government institutions are the main target of attacks. In Qatar, the rapid increase in using email as the official means of communications in the government institutions means that the threat of email phishing should not be underestimated. Therefore, it is important to know the threat to Qatar in order to propose and recommend training awareness to fit the purposes and recommendations that can help fight the menace of email phishing. The particular importance is to ensure that the general population is aware of the threat of email phishing. In addition, it is envisaged that the outcomes of this study can eventually lead to people being more aware of email phishing threats that will eventually reduce the chances of successful attacks.
<b>INFORMATION AND COMMUNICATION TECHNOLOGY SECTOR</b>	Another key issue that this study hopes to achieve is to determine the level of compliance of institutions and stakeholders when it comes to making sure that people are secure. This is why talking to relevant people is vital to how data collection is conducted in this study. The outcomes of the study would benefit the companies in Qatar, especially the IT sectors.

<p><b>INSTITUTIONS</b></p>	<p>It is strongly believed that this study targets a very important issue to the government and its institutions in Qatar; the Qatar news agency has been hacked recently which had a big effect on Qatar. The results and recommendations will help every sector in Qatar and even the citizens to provide processes that can help in reducing the threat of email phishing which will also change the way of thinking when they receive an email and how to react. The recommendations will help in developing key email phishing awareness training programmes to ensure the reduction of phishing email attacks.</p>
<p><b>RESEARCH AND ACADEMIC INSTITUTIONS</b></p>	<p>There are several researchers investigating the impact of email phishing threats to security and how these threats can be reduced. The investigation also shows that Qatar is facing an enormous email phishing threat, which is a big concern for government and non-government and will put them under risk while their employees are not well-educated and trained in security awareness. However, the recommendations will be specific and tailored to the needs of Qatar with the peculiar email phishing problem as it relates to the country's needs. The interviews that have been conducted with government and non-government organizations have discussed the major problems in Qatar and what awareness and training is needed. It is given that if the recommendations of this research are taken into consideration and implemented it will raise the awareness to the issues and significantly reduce the chance of a successful attack.</p>

## 1.14 Research Contributions

This research has three main contributions to knowledge as outlined below,

- Develop an enterprise-wide email phishing detection system to be used by organisations and individuals in the state of Qatar. This detection system is based on domain name comparison by which attackers carefully register domain names which victims trust. This detection system is novel and will help to reduce email phishing attacks.
- The development of an email phishing awareness training framework to be used by organisations in the state of Qatar. The framework will help to effectively reduce the email phishing threats in Qatar. This framework will also help Qatar government institutions and organisations to enhance email phishing awareness for their employees through a set of recommendations.
- Develop an interactive website to enhance the awareness of users in terms of targeted email phishing attacks to be used by organisations and individuals in the state of Qatar. This interactive website helps to train users on how to spot email phishing threats and hence reduce the threats to employees and citizens in Qatar.

# CHAPTER 2

## BACKGROUND AND TERMINOLOGY

### 2.1 Introduction

Neutralising the cybersecurity threat of phishing is not easy; over the years, the attacks have reciprocally gained sophistication, adapting to the ever-more stringent parameters and new techniques applied by anti-phishing strategists (Mishra and Gupta, 2014). By using a variety of social engineering methods to hoodwink web-surfers, phishing poses a risk to the cyber-security of users, often extracting crucial, confidential information using these methods (AlMomani et al., 2013, Heartfield and Loukas, 2016). Even web-surfers who are not naive to the risk of phishing can still be vulnerable to these attacks (Sheng et al., 2007), because their ability to discern between legitimate and illegitimate pages may be confounded by a false webpage that is designed by phishers to accurately emulate the features of the legitimate site it is imitating.

Embedding a mechanism within the web browser which sends an alert when users are confronted with a potentially disingenuous phishing page is one of the most commonly employed tactics for guarding users against attacks. These mechanisms usually use lists as their foundation for synthesising solutions; that is to say, web browsers often employ the use of blacklists or whitelists. These lists will then be compared with the domains users are attempting to visit, and alerts are sent accordingly (Tewari et al., 2016, Gupta et al., 2017). Technical experts work alongside security software in order to determine whether new domains belong on these lists. Several different elements of page identity are considered by the software in order to make a decision about authenticity (Xiang et al., 2011).

In the last half of 2014, the anti-phishing working report discovered 132,972 unique phishing attacks between July and December, globally (APWG, 2014). The industries which are most likely to come under attack are e-commerce, banks, and money transfer companies, for the obvious reason that these promise the most lucrative reward for phishers. The following top-tier domains were utilised by 75% of phishing pages: .net, .cf, .pw, .tk, and .com. The report also found that during the given time period, the median uptime for phishing sites (i.e. uptime) increased to 10 hours and 6 minutes. In 2015's first three quarters, the financial service sector and banking sector ceased to be the most

vulnerable sector, falling into third and second place respectively (APWG, 2015). Evidently, attackers began to prioritise Internet Service Providers (ISPs) during this time frame, with them taking first place as the most commonly targeted industry sector (APWG, 2015). The reason for this change of tactic becomes clear when we consider the opportunities ISP accounts offer phishers for gleaning confidential information such as credit card and identification data (Saad et al., 2014, Alomari et al., 2014) Once gained, this personal information can even be utilised for further phishing endeavours; for example, attackers are able to use hacked accounts to send spam mail. The Business Email Compromise (BEC) fraud of 2015 exemplifies a serious case where a successful phishing attack cost industries large amounts of money (APWG, 2015); with the use of spearphishing methods, the phishers were able to dupe their targets into making transfers and fraudulent transactions.

Blacklisting is commonly used to guard users against phishing. Often, these mechanisms are embedded within web browsers as plug-ins which perform a check on every URL and operate on the basis of phishing identification measures which include user votes. This then alerts users of the malicious nature of pages they are trying to visit when a domain appears in the blacklist and blocks the connection to protect them. Some examples of this type of anti-phishing plug-in are as follows: Google-safe browsing for Firefox (Schneider et al., 2007), phishing filter for Internet Explorer (Microsoft, 2005). The blacklist, though, needs to be constantly updated for these measures to be effective, and the update process is often not as speedy as it needs to be, especially considering the fact that many phishing websites typically have short life-spans, with uptimes of only a few hours.

Data mining and artificial intelligence have also been employed as a method of phishing detection, as other researchers have pointed out. These methods are the most effective by far in engendering web-safety. One such system of phishing detection is CANTINA+, proposed by *Xiang et al.* (2011). This Machine Learning Framework operates on the basis of URL characteristics which are used in conjunction with query results from a search engine as well as features of HTML. CANTINA+ proved to be a fairly successful method, with a 92% rate of recognition. Another detection method proposed by *Fu et al.* (2006) uses earth mover's distance (EMD) to ascertain whether websites are visually similar to one another.

*Li et al.* (2015) put forward a hybrid system of anti-phishing which attained a recognition rate of 99% with the use of imaging techniques, namely the image detection system PSO-SVM (Particle swarm optimization support vector machine). The method operates by

comparing results after generating a query and sending it twice, to two separate DNS (domain name system) servers. Although this method yielded highly effective results, the system used to obtain it is vulnerable to tampering by attackers. By using what is called a man-in-the-middle attack, it is possible to corrupt and therefore obstruct the system from all successful detection, leaving data insecure. Another real-time phishing and spam detection approach was developed by *Thomas et al.* (2011), who used traits such as redirect frequency, DNS and geo-location data, and elements of HTML, URL and JavaScript in their system. However, this method relies on gathering certain data, like HTML and JavaScript, which can become irretrievable if phishing attacks block the Crawler or IP from gathering it. Much like the system proposed by Jeeva and Rajsingh. (2016), two phases constitute the anti-phishing discernment process: firstly, the potentially threatening URL is checked against a whitelist, or repository. If it does not appear there, further enquiries are made using an association rule mining algorithm, which comprises the second part of the recognition process.

*Ramesh et al.* (2014) proposed approach begins by inputting suspect key-words into a search engine and compares these links with those of the target website. After inputting only existing links into a target identification (TID) algorithm, a DNS lookup uses the suspicious website's IP address to check its domain name. Therein is contained the disadvantageous nature of this defence mechanism; despite successful recognition of 99.62% of cases, as with *Li et al.* (2015), this method can be stopped in its tracks with a man-in-the-middle attack.

There are counter-approaches in the literature which shift their focus away from discerning between legitimate and illegitimate sites, instead preferring to offer methods of authentication for users which do away with this necessity. A study by Huang et al. (2011) proposes one such solution amongst others: a third-party-generated one-time passcode which the user receives by message each time, replacing the usual mechanism of permanent passwords. The fact that the third party is totally crucial to this transaction poses a potential flaw; if the third party was to be targeted, the entire security system would instantly collapse. Dependence on a third party is also the downfall of the system designed by Yue and Wang (2010), which proposes a plugin which sends intentionally incorrect passwords and logins when confronted by potential phishing sites and tests the response. The fact that this plugin forms the cornerstone of the detection method means that users are easily left vulnerable if it fails due to technical issues or attack.

## **2.2 Phishing strategies**

The following section outlines certain general categorisations which are used to describe specific kinds of phishing attack, pinpointing their main features in a few words. Four of these methods are specific to literature on phishing targeted at the banking sector: Gillnet phishing and the dragnet, lobsterpot and rod-and-reel methods.

### **2.2.1 Mimicking attack**

This describes phishing emails and websites which imitate the appearance and features of legitimate and official websites and entities, in order to hoodwink users into interacting with the fraudulent site as they would the legitimate one, and therefore reveal personal information or make illegitimate transactions.

### **2.2.2 Forward attack and popup attack**

These types of attack are where phishers obtain confidential user information through man-in-the-middle techniques like pop-up windows (popup attack) and proxy websites (forward-attack).

### **2.2.3 Rod-and-reel method**

This method requires the attacker to seek out targets directly, identifying them through an initial interaction or contact which then enables them to contrive situations in which the target parts with his or her confidential information.

### **2.2.4 Dragnet method**

This is another term which is used to describe sites, pop-ups or emails which ask the user to take some urgent action which will reveal confidential information, attempting to convince users of the illegitimate site's legitimacy using a generally trusted corporation's identity-elements, like names or logos.

### **2.2.5 Lobsterpot method**

This is another technique which achieves its aims of obtaining secret information by generating fraudulent, counterfeit websites which imitate trusted, official websites.

## **2.2.6 Gillnet phishing**

This describes the practice of encoding malicious code into emails and websites which are designed to infect users' computers, change computer settings and observe a user's activities in order to catch any confidential data which may be profitable.

## **2.2.7 Email Phishing**

Email is a well-established form of communication in our modern age, because it allows virtual messages to be sent instantly across vast distances. Email phishing is one of the most popular forms of data-mining cyber-attack. Using various techniques, some of which have already been explicated, phishers dupe targets into surrendering their personal information by the mimicry of certain features from trusted websites.

## **2.2.8 eFax**

eFax, alternately known as internet fax or online fax, is distinct from the archaic means of faxing because it does not use a phone network, but rather IP. Efax.com (j2 Global, 2017) is one example of an efaxing service which enables people to manage their faxes through email. eFax is preferable to manual faxes because it dispenses with the necessity for a fax machine. However, gaining the facility of online use also exposes faxes to phishing practices much in the same way as for email, and this puts confidential data at risk.

## **2.2.9 Instant Messaging**

The first time AOL fell victim to phishing, it happened through instant messaging (IM) (James, 2005). By impersonating AOL's administrators, phishers were able to hoodwink instant messaging users by sending instant messages which pretended to be alerting them to problems with billing and asking for their login and credit card data in order to rectify the issue. IM was first conceived of in the '90s. In the beginning, clients such as Yahoo!, MSN, AIM, Pingin and ICQ dominated the sector. Now, this method of communication is accessible for users through multiple devices and applications, both portable and stationary - these days, all smartphones have the capability to host instant messaging services. Social networks like Facebook, Instagram, Twitter and even dating applications like Tinder and Bumble contain an embedded instant messaging facility. Over the years, the capacities of instant messaging have expanded significantly, granting users the ability to go beyond text-based communication. There are now features for interaction including

sending photos, emojis, locations, links, videos, to name a few. Many social networks enable the user to make calls and share live video through their instant messaging plugin or application. Due to these services replacing text messaging as a medium of communication for many users (Marxism, 2016), IM is an attractive platform for phishers looking to target as many users as possible.

### **2.2.10 Social Network**

On this side of the millennium, social media, otherwise known as social networking, has dominated amongst methods of communication and networking (Silic and Back, 2016). There are a wide range of platforms offering different services and niches to users: Twitter, Facebook, Google+ and Instagram are some of the most popular forms of generic social networking, with platforms like Foursquare, Spotify, Tumblr and Pinterest catering to different interests and enabling connections between like-minded people. Although all of these platforms multiply opportunities for sharing and connectivity, they also multiply opportunities for phishers who are looking to take advantage of personal data.

## **2.3 Technique description**

This section outlines the different techniques phishers use to take advantage of various vulnerabilities and weaknesses in browsers, such as cross-domain vulnerability. These attacks include the creation of popup windows, taking advantage of auto-fill function, clickjacking and so on. This section provides an overview of the literature's findings in this area, providing a good knowledge base for anyone seeking to learn about the latest in phishing attack methods, in order to advance defence technology and be aware of suspicious entities.

### **2.3.1 Browser Vulnerabilities**

Phishers are able to take advantage of several weak points in browsers in order to deploy attacks on victims. What is more, the fact that browsers are continually updated with new functions and available plug-ins and add-ons means that they are continually offering phishers new opportunities to exploit weaknesses, as each of these new features will contain untested novelties (Ollmann, 2004). Phishing operations carried out through the medium of browsers, add-ons and plugins are difficult to anticipate and protect against. The *window.createPopup* method (Dormann and Manion, 2004, Milletary, 2013) as it is known, is a key example of this type of browser vulnerability. The technique, which

impacted Microsoft's Internet Explorer users in 2004, operated as follows (Microsoft, 2005): phishers engineered a program which generated borderless pop-up windows which can appear anywhere over any screen or window, and do not contain any of the familiar window management features which users are accustomed to interacting with. This allows phishers to keep their URL hidden, and also to fraudulently display a HTTPS padlock item which tricks users into believing they are interacting with a legitimate interface.

Another vulnerability common to Microsoft Internet Explorer in 2005 is cross-domain vulnerability (Dormann, 2005, Mitnick and Simon, 2011). With the employment of DHTML Edit ActiveX control, phishing attackers can obtain control of another web domain when their target clicks on a malicious page. Using these means, it becomes possible for the phisher to infiltrate legitimate sites and bury malicious content in them. Plugins are appealing targets for phishers, because they often offer loopholes in their security systems. In 2015, plug-in vulnerability doubled compared to the previous year following more widespread usage of Adobe plug-ins, chiefly Flash Player (Symantec, 2016). People have employed several different means of combatting this issue. Constantly updating security measures and introducing patches is one method, whilst others have opted to discontinue their endorsement of problematic plugins; Chrome, for instance, chose to eliminate its native hosting of Flash Player (LaForge, 2016).

The auto-fill function is a common pitfall for users; available in most browsers, it allows phishing to easily and automatically gain access to users' information by successfully tricking the user into believing they are entering their information into a legitimate site (Kuosmanen, 2017). This function, which usefully supplies a user's information without them having to manually type it in, often has access to users' login information and even sensitive financial information across several platforms. One tactic utilised in phishing attacks involves disguising forms which are compatible with the auto-fill function as innocuous information requests for data such as email or name. By inserting hidden fields, however, the phisher is able to collect more confidential data such as password, telephone number, or address via the auto-fill function. The request for information and responds to it without the knowledge of the user.

### **2.3.2 Clickjacking**

User interface (UI) redressing attacks, otherwise referred to as clickjacking (Patil and

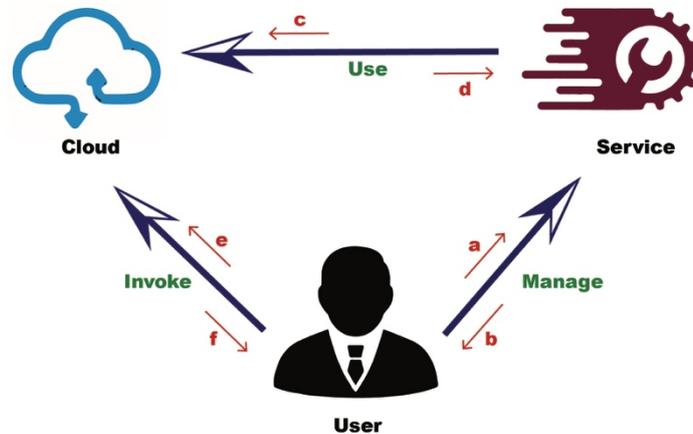
Patil, 2015), is a phishing technique which manipulates a user's UI when interacting with a site, and forces them to perform an action, for example triggering a link, without meaning to. This technique enables phishers to extract certain interactions from users, for example making a PayPal transaction, consenting to share their image and voice through their device's microphone and webcam capabilities, allowing personal data to be shared (Huang et al., 2012), and performing actions on social media platforms like Facebook and Leech Video, for example liking/following pages (Yadav and Nagpal, 2016), or making posts (Mahemoff, 2009). This is done by covering the target page with a total overlay (Akhawe et al., 2014, Huang et al., 2012) for instance a div container. Resultantly, if users attempt to click on any part of the site they are trying to access, they will be redirected to a destination chosen by the phisher through the false, overlaid image.

Alternatively, a partial overlay can be deployed to leave certain features of the website exposed to victims whilst obscuring chosen real information with illegitimate copy to achieve certain goals. Another approach to the use of decoy overlays is the employment of invisible, or hidden frames which are placed over particular elements, for example buttons (Patil and Patil, 2015). This means that phishers can dupe users into unknowing interactions with hyperlinks of their choice, which are invisibly overlaid over what look like legitimate points of action on the site. Compromising temporal integrity (Akhawe et al., 2014, Huang et al., 2012) is another click-jacking technique, which exploits delayed clicking and pauses in action on the user's behalf. When users double-click on or hover the mouse pointer over a targeted site feature, phishers are able to take advantage of the small temporal gap left by these actions, inserting elements (e.g. the PayPal 'Pay' button) over the feature which the user meant to click in the instant of delay before the user clicks.

### **2.3.3 Cloud Computing**

Cloud computing pertains to three online service models: Platform-as-a-Service (Paas), Infrastructure-as-a-Service (IaaS), and Software-as-a-Service (SaaS). Cloud Services are booming in popularity with the introduction of new services by providers (Weins, 2016). Phishers are able to target cloud providers as a method of personal data collection, and then go on to use the obtained information to carry out further attacks (a technique known as a password reuse attack). This is because these cloud services are dependent on email addresses for login authentication. Six different attack surfaces were indicated by Gruschka and Jensen (2010). The service, the users and the cloud provider comprise the three elements of cloud computing, the relations between which are illustrated in Figure

2.1. Using these service instances, cloud service providers are able to offer their customers a bespoke interface which enables access to the cloud, whether this be using the model type SaaS, IaaS or PaaS. Moreover, direct cloud controls such as adding and deleting are made available to users.



**Figure 2.1 The relation of the three components in cloud computing (Gruschka and Iacono, 2009)**

As Figure 2.1 displays, attack surfaces present themselves as follows: a) service-to-user, b) user-to-service, c) cloud-to-service, d) service-to-cloud, e) cloud-to-user and f) user-to-cloud.

The Dropbox phishing attack of 2012 (Gibbs, 2016) is a telling case of the vulnerability of cloud computing. As a result of a Dropbox employee reusing their LinkedIn password, phishers were able to access an administrative account, which in turn compromised the entire company network and user database, which included passwords. The only safeguard that stood in the way of a potentially catastrophic breach of user security was the encryption and “salting” of the database’s passwords. Using Figure 2.1, we can identify the vulnerability which enabled this attack, or in other words attack surface, as e) cloud-to-user.

Another case of a cloud provider security breach is that which happened in 2008 with Amazon Elastic Cloud Computing (EC2) (Gruschka and Iacono, 2009). Employing a SOAP interface, this IaaS provides customers with virtual access to the server, with abilities to take actions with the controlled machine such as terminating or starting new

instances. Although the message is signed digitally, the interface contains a flaw which enables phishers to make modifications to a message which has been intercepted (McIntosh and Austel, 2005). This gives phishers the ability to masquerade as legitimate users, creating a botnet and performing malicious commands. This can also be described as e), a cloud-to-user attack (Figure 2.1).

As what happened with DocuSign showed (Henderson, 2017), once phishers have accessed user information by infiltrating a cloud server's data, they can then use this information for further phishing attacks. In this example, the attackers were able to use the gathered email-addresses to send users malware through attaching an infected Word doc to emails masquerading to be from DocuSign's official account.

### **2.3.4 Cross-site Scripting attack**

XSS, or cross-site scripting, is where an attacker takes advantage of weaknesses in a website in order to use custom URLs or insert hostile code which misdirects viewers (Ollmann, 2004, Digicert, 2009). These weaknesses result from errors in page construction leading to unfiltered external content entering the system, without the necessary obstacles in place. Inevitably, some of this will be malware or malicious script (Emigh, 2005, Rader and Rahman, 2015). Same-Origin Policy (SOP) exists in order to prohibit scripts which are attempting to access or interact with information which originates elsewhere. Usually, this stops phishers from being able to follow users and glean personal information when a target uses a different website. However, the XSS attack is able to evade the SOP (Ruderman, 2016).

As Table 2.1 evidences, in the case of full substitution of HTML, targets are tricked into interacting with fraudulent elements of the page by the phishers, who code features like login forms redirecting the user to another website into pages where they appear in the place that the legitimate feature would usually be found. Another approach attackers use is to embed scripted content inline, by embedding malicious links from an external source within the URL. JavaScript is often employed in this sort of XSS attack (Kals et al., 2006, Symantec, 2016). Using this script, it is possible to redirect targets to false forms which ask for confidential data such as password or username, whilst maintaining their belief that they are interacting with a legitimate site (Elledge, 2007). It is also possible to redirect targets to false pages in the same manner (Milletary, 2013). The following URL is one possible format that can be used to redirect a victim without his/her knowledge:

```
<script>window.onload=function().varlink=document.getElementsbytagname("a");link[0].href="https://false.com/";</script>
```

Although the target believes they are visiting index.php, by failing to double-check the URL they are clicking, they will land on <https://false.com> instead.

**Table 2.1: Custom URL XSS attacks (Ollmann, 2004)**

<b>CUSTOM URLS</b>	<b>EXAMPLE</b>
<b>FULL HTML SUBSTITUTION</b>	http://legitimate.com/login?URL=http://phisher.com/login/false.html
<b>INLINE EMBEDDED SCRIPTING CONTENT</b>	http://legitimate.com/login?page=1&client=<SCRIPT>phishcode
<b>LOAD EXTERNAL SCRIPTING CODE</b>	http://legitimate.com/login?page=1&response=phisher.com%21phishcode.js

Stored XSS and reflected XSS were identified as the two main forms of XSS attack by Kals et al. (2006). Patil and Patil (2015) made the same distinction, also indicating two classes of attack: persistent and non-persistent. Persistent XSS, or in other words stored XSS, refers to malware which has been deeply embedded into a machine or application, coded and stored there. Non-persistent XSS, also known as reflected XSS, refers to attacks which rely on users erroneously clicking on coded URLs which lead to false and phishing sites. The fundamental difference between the two is that stored XSS resides in data until it is detected and deleted; until that time, it continues to damage the system it has infiltrated. For instance, if a phisher uses the means of a message board post to store

scripts, and there is a lack of filtration and input validation in the message board's moderating system, then the malware which is stored there will be executed each time the page is visited.

The SOP can be circumvented by any method which makes the user share confidential information without their knowledge. As found by Gelernter and Herzberg (2016), CAPTCHA has been used by phishers to this end. As Figure 2.2 demonstrates, a CAPTCHA attack can trick targets into revealing personal information because they are also submitting to a legitimate site alongside the phishing site (1). The phisher extracts the information by employing a cascaded style sheet (CSS) and inline frame (iframe) with a target legitimate page which users are trying to access, in order to create a form which reveals confidential info, disguised as a CAPTCHA form (2). Then, unbeknownst to the user, the CAPTCHA form that they submit also contains hidden features which extract confidential data such as login information, and feed it back to the phisher (3). In order for this sort of attack to be successful, the page must be presented in a way where the tampering is undetectable.

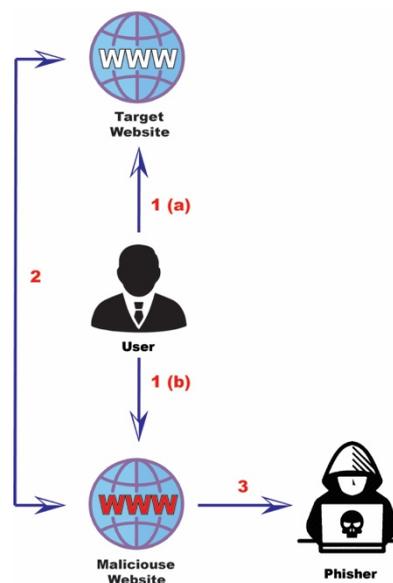


Figure 2.2 The CAPTCHA attack (Gelernter and Herzberg 2016)

### 2.3.5 Drive-by-Download

This term refers to the strategy of writing shell code, viruses or malware into webpages

and emails so that devices are infected by them just by opening an HTML email (Akhawe et al., 2014) or website (Patil and Patil, 2015, Cova et al., 2010). This technique can also be deployed through Internet Relay Chat (IRC) service providers (Elledge, 2007). Usually using JavaScript in order to take advantage of browser or plug-in vulnerabilities, malicious are written into websites or HTML emails, or hosted in servers.

### **2.3.6 Trojan attack**

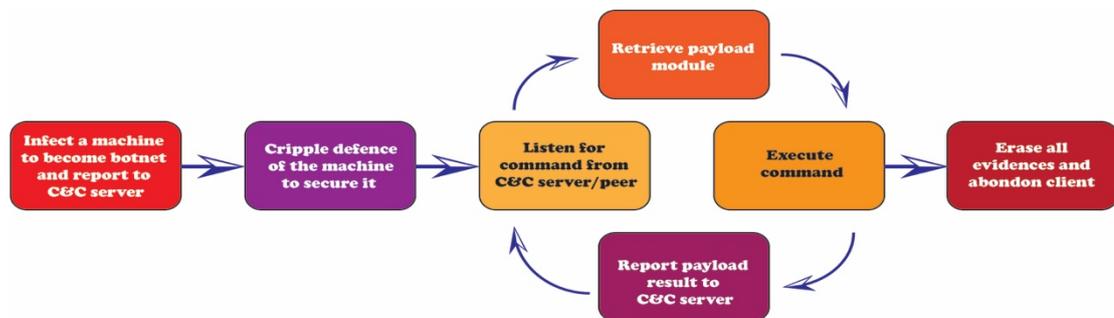
This type of attack, termed a Trojan, refers to a sort of bot or spyware (Milletary, 2013; Symantec, 2016; Digicert, 2009; Emigh, 2005; Rader and Rahman, 2015; Nagunwa, 2014; Elledge, 2007; Cova et al., 2010; Banday and Qadri, 2011; Chaudhry et al., 2016; Suganya, 2016; Schiller and Binkley, 2011), named after the famous story of the Trojan victory over the ancient Greeks. By hiding their troops in a wooden horse sent in as a gift, the military was able to get inside the city and attack from inside. Trojan spyware works in the same way, with software that appears to be legitimate acting as a vessel for malware. The appearance of legitimacy convinces the target to run or install the malicious software and gives the phisher access to the device (Landwehr et al., 1994). Once this is done, the phisher can employ keystroke loggers and screen capture to obtain the target's confidential data as they enter it (Elledge, 2007). The *TrojanSpy:MSIL/Omaneat* (Spring, 2017), for instance, is a keystroke logger; it records everything a user types, as well as a Windows user's browsing history and application launch history. *user.Linux.Ekoms.l* (Kovacs, 2016), conversely, is able to execute the screen capture function with a Linux device. Web Trojans are another form of this type of attack (Henderson, 2017; Banday and Qadri, 2011; Chaudhry et al., 2016; Suganya, 2016). By opening a pop-up window imitating a legitimate site when a user opens a legitimate webpage's login portal, the phisher fools the user into entering their login details into the pop-up instead of a legitimate webpage, therefore submitting data to a false site which the phisher controls.

Bot malware allows phishers to access, command and control (C&C) target machines remotely (Milletary, 2013; Schiller and Binkley, 2011). Devices under C&C are defined as botnets. This sort of phishing relies on creating a network of botnets, acquiring control of as many devices as possible, in order to perform any of the following actions (Milletary, 2013):

- Operating as web servers for further distribution of malware and hosting illegitimate sites

- Sending phishing and spam emails
- Operating as a proxy service provider network
- Operating as redirectors to phishing pages

Figure 2.3. describes typical botnet lifecycles (Schiller and Binkley, 2011). After infecting a device, the new botnet alerts the C&C server to its presence. Then, defence mechanisms like anti-virus and malware detection systems are disabled in order to protect the malware from being discovered. Once this is in place, the botnet awaits the prompt from peers or other C&C servers, and upon receiving it proceeds to download the payload module which tells it its function. After executing this function, it reports its success to the C&C server. The C&C server is then able to remotely direct the botnet to terminate after destroying all traces of its presence.



**Figure 2.3 Botnet Lifecycle (Schiller and Binkley, 2011)**

Botnets can also be used for fast-flux attacks (RSA, 2016; Zhou, 2015). Fast-flux is a term which also describes a legitimate practice used by organisations, of creating several IP addresses for a single domain name in order to share the burden of processing between servers and ensure a consistent service. Phishers use fast-flux to sustain longer-duration attacks and avoid detection by decentralising attacks. Employing botnets to create a network of proxies, phishers can execute attacks remotely for a long time whilst maintaining anonymity, because attacks cannot be traced back to the C&C server which is controlling them. Phishers reap the same rewards as legitimate organisations also in terms of uptime; even if one botnet is identified and destroyed, the others in this network of botnets can continue to sustain the attack, and botnets can be continually added to it. This proxy-based tactic makes it much harder to disable a phishing domain name, because it is not reliant on a single machine.

### **2.3.7 JavaScript obfuscation**

Phishers utilise JavaScript to obscure or, in other words, obfuscate web browser windows' chrome areas (Elledge, 2007), which comprises status bar, menu area, toolbar and address bar. JavaScript allows phishers to fabricate a legitimate URL in a superimposed address bar, complete with padlock icon and 'https' details to convince the viewer that they are visiting a secure page. Visually, the false site appears indistinguishable from the real. There are several different ways to use JavaScript to trick users; phishers can use a virus which redirects users to phishing sites when they type in a page's address (Millettary, 2013), or through embedded JavaScript on a phishing page which runs when the page is visited.

### **2.3.8 Malwaretishment**

This describes the use of spoof advertisements containing embedded malware to spread it into victims' machines (Nagunwa, 2014). When targets click on the advert, dynamic malware designed to exploit software vulnerabilities and obtain confidential data is injected into the device. This is an appealing method for phishers as a result of the abundance of online ad-hosting services, many of which require minimal information from their subscribers in their highly straightforward application forms (Symantec, 2016). What is more, this allows phishers access to legitimate hosts, who will be distributing phishing pages without the phishers needing to attack or gain control from them (Symantec, 2013). This makes the phishing ad more believable, since it is embedded in a page that users already trust, and sites often fail to make it clear when their ads are sponsored by ad networks and not curated by the organisation itself. Users may not know that these ads come from elsewhere (Xing et al., 2015), and that they are not checked by the site (Sood and Enbody, 2011).

### **2.3.9 Man in the Middle attack**

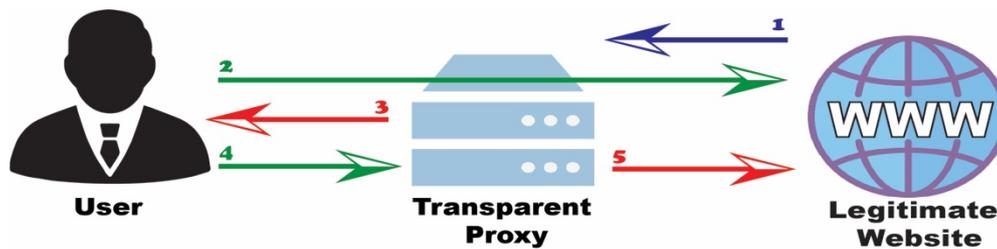
Man-in-Middle (MITM) attacks refer to when phishers intercept communication between a web application and a user (Ollmann, 2004; Millettary, 2013; Banday and Qadri, 2011; Suganya, 2016; Digicert, 2009). As Figure 2.4 demonstrates, in this manner the phisher is able to 'overhear' and glean confidential data which is being submitted by victims to web-apps.



**Figure 2.4 MitM attack**

The lack of external indicators, alerting either party to the attack, means that these attacks often go undetected; the web application appears to function normally to the user despite the phisher intercepting in the authentication process, and the information they requested from the site appears as expected. This approach is often employed to sidestep two-factor authentication defence mechanisms by recording the data entered by the user during the process (Milletary, 2013; Emigh, 2005). HTTPs or SSL web traffic is insufficient to protect against this type of phishing. The attacker can either utilise two distinct SSL connections, between itself and the real server, and itself and the phishing target (Ollmann, 2004), or use only one SSL connection between itself and the server, rendering the user's SSL checks useless (Emigh, 2005). DNS cache poisoning, pop-up attacks and proxies can all be used to carry out MITM attacks. Within the realm of proxies, several approaches can be delineated: URL obfuscation, browser proxy configuration, relay attacks, and PHP cURL module and transparent proxy.

By contrast with traditional proxies, transparent proxies do not need to be configured by the client, because all the data is intercepted before it reaches its destination. The unwitting target has no way to perceive the proxy, because they reach their chosen destination, data loads correctly, and everything operates with no apparent differences. Transparent proxies are used by legitimate organisations for actions like redirection, caching and authentication (Gibb, 2015), but they also offer a vulnerability for phishers to take advantage of in a MITM attack (Ollmann, 2004). By designing a false site based on the real page's source code, phishers are able to set up proxy caches which are difficult for users to detect or defend against (Rader and Rahman, 2015). The proxy server intervenes when users open the page, and instead shows them the counterfeited website, made of the legitimate website's source code. This counterfeit page then mediates communication between the legitimate organisation and the client, and passes data between the parties whilst simultaneously recording any useful confidential information that goes through it for the phisher. This process is illustrated in Figure 2.5:



**Figure 2.5 The MITM attack using a transparent proxy**

Using URL obfuscation, phishers are able to hoodwink users into using spoofed websites; this is the process of either designing a phishing URL which closely resembles that of the real site, or to find a way to obscure suspicious-looking URLs from the viewer. Approaches to obfuscation include third-party shortened URLs, hostname obfuscation, and bad domain names. The latter method entails the use of characters or numbers to closely imitate legitimate domain names, as exemplified in Table 2.2. Phishers have learned to take advantage of the visual similarity between certain letters in different languages' character sets, for instance the 'o' in the Cyrillic keyboard is indistinguishable from the English 'o'. By substituting letters for these lookalike letters, it is possible to register counterfeit domain names which appear almost or completely identical to the original (Ollmann, 2004; Rader and Rahman, 2015).

Aside from transparent proxy, phishers have other methods of conducting MITM attacks, one such being browser proxy configuration (Ollmann, 2004; Rader and Rahman, 2015). This entails infecting victims with a malware which allows the phisher to change proxy settings from the client side, so that all traffic is forced to go through a phishing proxy controlled by the attacker. The first stage of the attack can be executed by drive-by-download, which causes machines to become infected when users open HTML mail or phishing sites (Patil and Patil, 2015; Cova et al., 2010). Once the malware has been activated, browser proxy settings are configured to channel all data through a proxy address controlled by the phisher, allowing him/her to intercept the data being sent and received between users and the legitimate web servers, which they are interacting with (Table 2.2).

**Table 2.2: URL Obfuscation**

TYPE OF URL OBFUSCATION	EXAMPLE
<b>LEGITIMATE DOMAIN NAME</b>	<a href="https://true.truesite.com">https://true.truesite.com</a>
<b>OBFUSCATION BY INTERCHANGING THE DOMAIN AND SUBDOMAIN NAME</b>	<a href="https://truesite.true.com">https://truesite.true.com</a>
<b>OBFUSCATION USING COUNTRY CODE TOP-LEVEL DOMAIN</b>	<a href="https://true.truesite.com.my">https://true.truesite.com.my</a>
<b>OBFUSCATION BY USING PART OF THE LEGITIMATE URL AS SUBDOMAIN</b>	<a href="https://true.truesite.site.com">https://true.truesite.site.com</a>

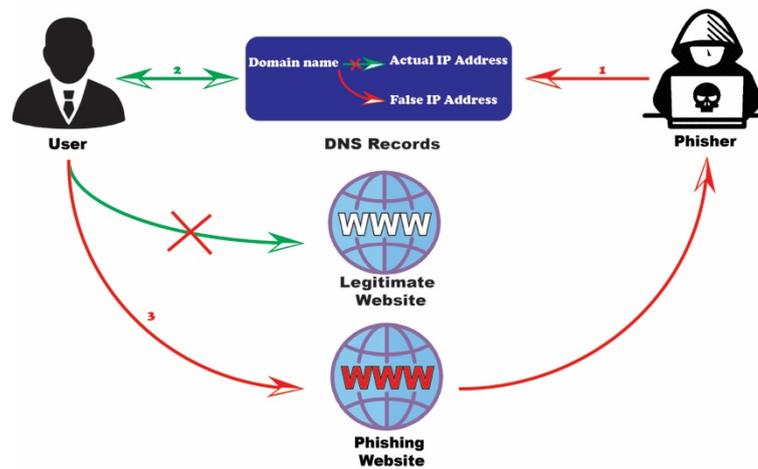
Proxy programs allow phishers to relay the contents of real sites to users without copying and reproducing their source code, as is the case with transparent proxies. This method intercepts confidential information by showing users a false, imitated version of the page they are trying to visit. Figure 2.6 delineates the attack flow of this method. An example of this technique being used in the real world is Operation Huyao (Hayashi, 2014). The phisher employed the SEO technique in order to present users with a link to a phishing website in place of a popular shopping page as one of the top-ranking links brought up by the search engine. The unsuspecting victims believed they were interacting with the real site while visiting the counterfeit one, because the legitimate web page's contents were being relayed directly to those viewing the fake page. Everything on the website behaved as it should, so that when customers wanted to pay for their shopping carts, they were unaware that it was a phishing page asking for their bank account details.



**Figure 2.6 The MITM attack using a proxy program that acts as a relay to the legitimate website**

The Hypertext Pre-processor client URL (PHP cURL) offers another avenue for phishers wishing to conduct MITM attacks (RSA, 2016). This command line tool utilises URL syntax in order to facilitate data transfer through multiple protocols including HTTP, FTP, and HTTPS. By reverse engineering data requests sent to the real organisation's website, phishers can generate scripts which direct certain actions, for example sending funds from a target account to the phisher's while making the transaction appear legitimate. In this way, phishers can financially exploit victims and also collect confidential data such as phone number and home address.

Domain Name Server (DNS) poisoning, also known as pharming, involves altering DNS records so that the entry inside points domain names towards false IP addresses. This can be done directly using the DNS server (Rader and Rahman, 2015), from the network side via the DNS Cache (Ollmann, 2004; Milletary, 2013) or from the client's end through a host file (Milletary, 2013; Emigh, 2005; Banday and Qadri, 2011). Using malware, the phisher is able to modify the DNS records (Milletary, 2013). The term DNS cache describes a copy of DNS records retrieved from a DNS server. These records are utilised to resolve URL domain names with their corresponding IP addresses, that being the address of that page. By modifying DNS records, phishers can make domain names redirect to phishing sites instead of the legitimate one that users are trying to access. Entering the URL manually is ineffective in preventing this kind of attack. The attack flow of pharming is shown in Figure 2.7.



**Figure 2.7 The DNS poisoning attack**

### 2.3.10 Mobile phone

The rapid, exponential growth of mobile phones as a market and a popular commodity has made mobile phone phishing an increasingly attractive and lucrative avenue for phishers (Felt and Wagner, 2011; Vural and Venter, 2011). Today's huge number of mobile phone users means that phishers have an ever-increasing pool of victims to choose from when seeking profitable confidential information. Mobile phone interfaces force organisations to produce simplified versions of their web-pages, with fewer authenticity indicators in the user interface (Felt and Wagner, 2011). This makes it hard for users to know whether they are interacting with real pages or fake ones, which is useful for the purposes of phishers.

Four means of control transfer exploited by phishers which are common to mobile phone functionality were outlined by Felt and Wagner (2011), as outlined in Figure 2.8. The transfer of control from a mobile application to different mobile application is referred to as mobile sender to mobile target. For instance, if an application provides options for users to share content on other social media platforms like Twitter or Facebook, or if an application forwards users to related apps or external app stores, this provides a vulnerability for phishers to take advantage of. If phishers can convince users to download phishing applications, the unaware victim will input their confidential data into the phishing application, believing that they are interacting with a legitimate entity.

Mobile sender to web target describes transfer control from mobile app to web browser, for example if an app redirects users to a web-based login page. Just as in the previous

example, for a phisher to exploit this vulnerability it is necessary for the victim to have downloaded and installed a phishing application on their phone. At the point when the victim expects and believes they are being redirected to a legitimate web-based login page, they are actually being shown a phishing site, and asked to input their confidential login info therein. By hiding the browser's URL bar, the phisher is able to make the attack unnoticeable.

Transfer of control from web browser to mobile application is referred to as web sender to mobile target. This type of control transfer describes, for instance, when users click links which open via an application they have installed on their phone. By imitating real application screens, phishers can hoodwink targets into thinking they are interacting with their genuine application when they are in fact dealing with a phisher's simulation in the web browser. Believing their eyes, users will enter their login details at the spoofed page's request. In this way, the phisher gains access to the user's login details.

Web sender to web target is used to describe a common function we often see, where a hyperlink on a web page redirects you to another web page. Phishers are prone to obscuring or obfuscating the real URL bar and showing a spoofed URL bar, even inhibiting victims from scrolling up and viewing the real URL bar by making the view snap back to the false URL bar when the victim scrolls (Niu et al., 2008; Rydstedt et al., 2010). In the same way as the previous examples, this allows phishers to create features which trick users into entering their confidential information to the phisher's database.



**Figure 2.8 The four ways of control transfer in a mobile phone (Felt and Wagner, 2011)**

It is also possible for phishing attacks to be launched using mobile notification services (Xu and Zhu, 2012). Operating systems (OS) like Blackberry OS and Android offer users the choice to customise their notifications. Through third-party Trojan apps, which the

user has installed on their phone, phishers are able to generate fake notifications which appear to be from well-known applications like Facebook or Instagram, and redirect victims to phishing login interfaces.

Aside from misusing victims' mobile notification services, applications infected with malware are able to monitor, record and relay to the phishing server any information that the user submits to websites through the malicious application (TrendLabs, 2012). In this manner, phishers may be able to collect phone numbers in order to text the victim directly through the app or launch an SMS phishing attack (SMiShing) (Tufts, 2012).

As mentioned, many mobile phone phishing methods require the victim to have downloaded a malware-infected application onto their phone. SMiShing and telephone fishing (vishing) are both means of achieving this first necessary step (PandaLabs, 2015). Another method, crossover threats, allows phishers to install malware onto victims' phones without their knowledge or consent (Symantec, 2016). When users are installing apps directly onto their phones through web browsers, phishers are able to collect secret information by stealing browser cookies and remotely install phishing apps onto the device.

### **2.3.11 Phishing kit**

This tool allows phishers with no advanced knowledge and skill in programming to easily create phishing emails, pages and scripts (Millettary, 2013). These kits are sometimes available for free in underground kit developer distribution circles (Cova et al., 2010), and can also be purchased in the cybercriminal marketplace (Symantec, 2016). The incentive for developers to share their software can be understood when we consider the widespread inclusion of 'back doors' in these free software packages, that leaks back any phished information to the developer (Cova et al., 2010; Chaudhry et al., 2016). Although these kits have no direct part in harvesting personal and secret data from users, they are useful tools for those wishing to harvest this data. They make phishing a practice accessible to anyone, regardless of expertise. The "Universal Man-in-the-Middle Phishing Kit" is one such example of these tool kits (Elledge, 2007; Singh, 1970). This tool aids in carrying out MITM attacks, which intercept data as users interact with real websites. Facilitating other phishers has developed into a full-on industry; beyond selling kits on the dark web, advanced phishers have developed a business model known as Malware-as-a-Service (MaaS) (VeriSign, 2012), which describes the practice of being

paid to manage other people's phishing endeavours for them. MaaS comprises three components as a service: developing toolkits, deploying them, and taking care of the overheads and financial cost of the endeavour (Moreno, 2016). The first step is a software development task. The second concerns the active providing of a service: i) deploying the software, ii) distribution of the kit, iii) hosting the phishing website. This final aspect, hosting, is the financially burdensome aspect of this business; it is necessary to spend money on finance-related services such as financial data providers and money mules. From development to bookkeeping, this model describes the entire process of MaaS enterprises. Often, these cybercriminal service providers offer packages which include all of these different services, and also offer the alternative to purchase elements individually.

### **2.3.12 Spearphishing**

This refers to an attack targeted specifically against a group, organisation or individual (Ollmann, 2004; Symantec, 2016; Elledge, 2007; Singh, 1970; Nikiforakis et al., 2014; Rietta, 2006). This method has grown in popularity (Rietta, 2006), superseding that of more conventional techniques like random and mass email phishing. The reason for this is that spear phishing has a far higher success rate than these other, more generalised methods (Krombholz et al., 2015). Because the content of the phishing email is tailored to the receiver, it is less likely to arouse suspicion.

Spear phishing works because people generally trust communications which come from entities with whom they already hold an account or are familiar (Downs et al., 2006). Phishing sites which replicate organisations that users have previously interacted with in their legitimate forms, are less likely to arouse suspicion and cause them to check the authenticity closely. Some phishers even impersonate specific users' friends (Jagatic et al., 2007) or colleagues (Halevi et al., 2015) to ensure a higher success rate. Phishers can, for instance, contact a staff member in an organisation whilst pretending to be a colleague from another department, who for legitimate-seeming reasons asks the victim to respond with important login details or open malicious attachments.

This technique can yield great success and lead to entire data networks being compromised in an institution (Elledge, 2007). This is the preferred method for phishers carrying out what is described as an Advanced Persistent Threat (APT) attack (Hayashi, 2014) which is an attack targeted at a specific organisation with specific goals. The

personalised nature of spear phishing makes it an ideal means of attaining this goal. APT attacks are typically carried out over a long time, and care is taken to avoid drawing any attention to the infiltration before the set objectives are achieved. Making use of malware or zero-day vulnerability exploits, phishers launch APT attacks in order to achieve goals such as sabotage or espionage (Symantec, 2011).

In order to create personalised spear phishing emails, it is first necessary to obtain some data about the target. One means of achieving this is browser sniffing (Jagatic et al., 2007), which is a technique of “sniffing” out the websites that a target has visited by viewing access time for certain cache cookies, DNS caching and URL (Felten and Schneider, 2000). If access time for a certain DNS lookup or URL is brief, this is evidence that the user has accessed the website before, since a DNS cache already exists for the DNS entry, or the browser has created a cache for quick access to the site. Cache cookies also allow phishers to monitor which sites are frequently accessed by their victims. All of these means allow for the development of a personally targeted attack which draws on what the phisher knows to be the victim’s established network of interests and affiliations. This sniffing technique can be deployed by embedding JavaScript containing malware into websites, web-ads, HTML email or search engine optimisation, and sending links to these in emails (Felten and Schneider, 2000). Once installed, the malware will report back to the phisher with the victim’s access times, allowing a personalised attack to be devised.

### **2.3.13 SQL Injection**

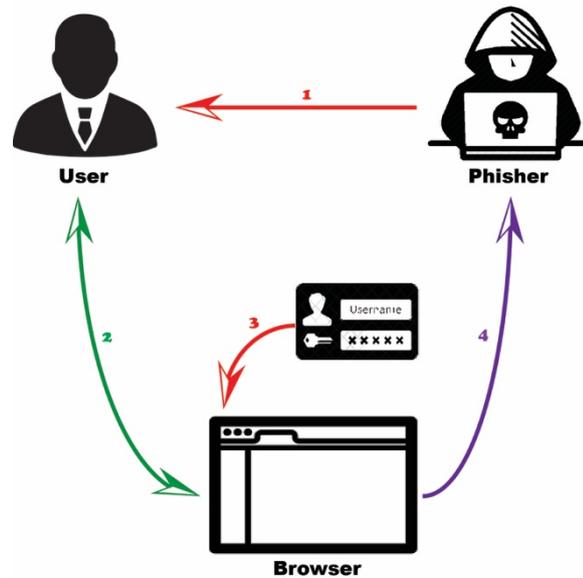
The Structured Query Language (SQL) Injection method takes advantage of database vulnerabilities, where there is no proper filtration mechanism in place, meaning database commands can be injected and executed in order to extract information (Emigh, 2005). By injecting SQL commands into SQL queries and statements on webpages, it is possible for phishers to modify a statement’s original purpose (Patil and Patil, 2015; Kals et al., 2006; Rietta, 2006). The SQL commands are concatenated with the injected code through user-input variables, so that the dynamic SQL command is enacted and executed. This can be done by tampering with existing SQL queries, or adding multiple queries (Nagunwa, 2014).

The following case demonstrates how SQL operates (Kals et al., 2006; Nagunwa, 2014): an SQL query is designed to obtain data from a table comprised of user login data. The query is concatenated by inserting ‘OR 1 = 1 –’ into the user field along with a blank

password field. The user authentication aspect of the query is entered using an always true statement, that being  $1 = 1$ , and an OR. As a result, every user record in the table is caused to be true. The comment command ‘–’ causes the SQL database engine to not recognise the remaining string after the command, making it register as a comment instead. Once this query is set into motion, the database obtains the required data and leaks it back to the server which launched the attack.

### **2.3.14 Tabnapping**

Tabnapping was introduced to the world by Firefox’s creative leader, Aza Raskin, in 2010. This term, which describes a kind of online ‘kidnapping’ using a web browser’s tab system, is a portmanteau of the words ‘kidnap’ and ‘tab’. As delineated in Figure 2.9, the attack flow for this technique is as follows: a link which redirects to a phishing website is emailed to a victim. After they have clicked the link, a browser tab will load what appears to be an unexceptional, normal webpage. However, embedded JavaScript monitors page activity and keeps track of web browsing. When the user changes their view to another tab in the browser, the phishing mechanism takes the opportunity to load a spoofed authentication mechanism, imitating legitimate hosts like PayPal or Gmail. Then, the phishing mechanism waits for the user to spot the spoofed page, and hopes for them to enter their login details, believing that their session has timed out with an entity they often use and have established trust with. In this way, victims send their confidential login information to phishers without their knowledge. This attack is explicated in detail on Aza Raskin’s website (Raskin, 2010). Because victims are unlikely to notice or be aware that JavaScript can be employed to transform pages which have already loaded, phishers are able to have success with this method compared to more traditional pop-up methods, which give themselves away more obviously. However, this method does rely on the targeted user being in the habit of browsing with several tabs, rather than focusing on just one or a few, in order for the reload to evade their attention amongst multiple tabs.



**Figure 2.9 The Tabnapping attack (Raskin, 2010)**

### 2.3.15 Typo squatting

This is a domain squatting technique, which describes when phishers register domain names which are based on likely typos for real, popular sites (Banerjee et al., 2008; Wang et al., 2006; Waziri, 2015). This attack only affects those who manually type a URL into their address bar. As shown in Table 2.3, there are five main possible types of domain name typo which phishers can take advantage of. When the URL targeted by the phisher is misspelled, with missing characters and nearby keyboard letters, the faulty URL can be set to arrive at a phishing website which injects malware into the victim's machine or imitates the legitimate page they were trying to access (Wang et al., 2006).

**Table 2.3: Typo Squatting Example**

TYPO TYPE	EXAMPLE
MISSING DOT TYPOS	www.yourbank2us.com
CHARACTER OMISSION TYPOS	www.yourbankus.com

<b>CHARACTER PERMUTATION TYPOS</b>	www.yourbank2su.com
<b>CHARACTER REPLACEMENT TYPOS</b>	www.yourbanl2us.com
<b>CHARACTER INSERTION TYPOS</b>	www.yourbank2uss.com

### 2.3.16 Whaling

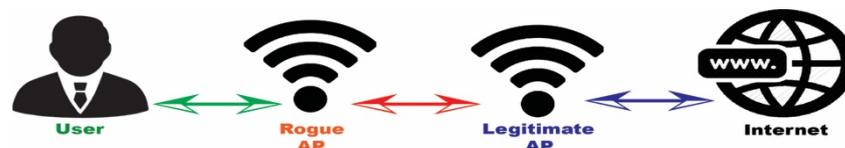
Whaling is a targeted attack like spear phishing, but as suggested by the name, it is an attack of this kind which is levelled specifically at high-up executives (Ollmann, 2004) who are able to access an organisation's highly privileged resources and data. By deploying malware in the machines of particular targets, phishers are able to utilise keyloggers as well as gain backdoor access to an organisation's data and networks. Since this attack is personally targeted, more time will be spent carefully creating personalised attack vectors which increase the likelihood of the target engaging with the infected link or software which is sent via eFax or email. It is a pre-emptive measure which precedes a prolonged attack termed business email compromise (BEM) (FBI, 2017). Using a high-level executive's credentials, phishers are able to send instructions on behalf of the CEO to employees asking them to make payments via unauthorised wire-transfer.

### 2.3.17 WiPhishing and Evil Twin

Evil Twin, or WiPhishing (Sinha et al., 2006; Song et al., 2010), uses wireless networks as the basis for its method. As Figure 2.10 shows, phishers place themselves between legitimate wireless access points (AP) and clients with the use of rogue AP. This rogue AP has the radio frequency and network name or Service Set Identifier (SSID) as a legitimate AP which is operating in the area. There is software that allows laptops to be turned into wireless network access point within a network (TrueSoftware, 2017); with the use of this, the phisher puts the rogue AP within proximity to the victim with the hope that they will choose the rogue AP because it has the strongest signal out of several identical alternatives. Once the victim connects to this network, it becomes possible for

phishers to intercept information that is sent and received from the victim's device. By carrying this attack out in public areas like airports, hotels and cafés, where people will be on the lookout for free wifi hotspots, phishers are able to lure more victims into their trap (Song et al., 2010).

Software that enables a laptop to be the access point in a wireless network is available (TrueSoftware, 2017). Then, the phisher will place this rogue AP closer to the user which may cause the user to connect to the AP with the strongest signal or having the computer to automatically connect to the rogue AP as it chooses the AP with the strongest signal from a group of APs with the identical name. Finally, the phisher is able to eavesdrop on the information that the user submitted and received through the rogue AP that the user's machine is connected to. The phisher normally will target public places with free hotspot such as cafes, airports, hotels, etc. (Song et al., 2010).



**Figure 2.10 The WiPhishing/Evil Twins attack**

## **2.4 Phishing Attacks**

Researchers have put forward several blacklist- whitelist- and heuristics- based defence mechanisms to protect against phishing (Section 6). Entities such as PhishTank (PhishTank) and Anti-PhishingWorking Group have compiled reports on authentication services and phishing (Jain, A.K., Gupta, B.B., 2016). Over the years, many tools have been developed which are designed to safeguard against the most common phishing attacks: browsing services like Microsoft SmartScreen Filter, Norton Safe Web, McAfee SiteAdvisor and Google Safe Browsing are just some of the tools developed to this end. Nonetheless, phishing practices have evolved in sophistication in tandem with defence systems, often staying one step ahead in the game of avoiding notice and bypassing safeguards (Yue and Wang, 2010). The struggle between phishers and anti-phishers is an

enduring one.

Spoof emails are the most popular form of drawing victims towards phishing pages. By creating a primary layer of context, for instance a request for login authentication or a security-based information update, phishers are able to trick victims into clicking on phishing URLs. The next layer of context is comprised of the appearance and interface of the false webpage; by closely imitating specific legitimate sites and their identity markers, phishers scam victims into entering their login details into phishing facades (Yue, 2013).

The aforementioned primary layer of context, or first-layer context, is subject to two obstacles or constraints (Yue, 2013). Firstly, many users may be savvy to the appearance and format of suspicious emails; if an email appears disingenuous, users will not click on the first link, and the attack is stopped in its early stage because the URLs remain unclicked, and phishing sites unvisited (Downs et al., 2006; Jakobsson and Ratkiewicz, 2006). Secondly, spam filters are frequently updated and become better and better at filtering out phishing emails (Whittaker et al., 2010), which prevents the first stage of attack from even being shown to potential victims.

The second layer of context also faces two safeguarding strategies that threaten to disarm the attack (Whittaker et al., 2010). Firstly, browsers often have mechanisms in place which are designed to detect and alert users of suspicious websites. Users with a basic awareness of the threat of phishing will be deterred from visiting the page or entering any information, thwarting the attempts of the phishing attacker to gather confidential data (Akhawe and Felt, 2013). Secondly, if the visual appearance and identity markers of the spoofed page are off-kilter and somehow incorrect, again savvy users will turn back from the site and refuse to supply the personal information that is sought by phishers (Downs et al., 2006; Dhamija et al., 2006; Hong, 2012; Jackson et al., 2007; Sheng et al., 2010).

The second layer of context is comprised of several different elements, which must be executed to a certain standard of imitation in order to trick users into compromising their information. Appearance, which includes features such as text, images, styles and page layouts, makes up the victim's all-important first impression. Low visual correlation between a spoof site and the original site makes it more likely for users to become aware of the phishing site's illegitimate nature (Dhamija et al., 2006). Simple phishing websites

often only achieve a low visual similarity. Advanced phishers are capable of producing much higher quality, almost identical imitations of their corresponding sites, while expert-level phishers are able to generate phishing sites which are visually absolutely identical to the legitimate site.

When we talk about a phishing site's *page depth*, we are referring to how many webpage levels are linked and organised together to form the phishing webpage. Often, users click several links when they visit a webpage. Phishers can quickly lose trust if they continually redirect users to invalid or missing links, meaning that users will refuse to submit their private information. What is more, unmodified links which take the victim to legitimate webpages leave several avenues through which to lose access to victims as they are redirected to pages outside of the phisher's domain of control. Simpler spoof websites, which as mentioned, tend not to attain a good visual correlation, have a page depth of one, wherein website links are only altered in part. Advanced phishing sites have a limited page depth; advanced phishers tend to alter a limited number of elements, attaining a partial level of visual similarity. Advanced or extreme phishing webpages have unlimited page depth, often with every visible element modified on a page to resemble the legitimate site in order to hoodwink the maximum number of victims.

Phishing websites are able to support many forms of dynamic user interaction, for instance form submission, search functions, clicking and JavaScript-based interactions like dynamic URL creation, or other kinds of DOM element creation. Developing dynamic user interaction capabilities is useful to phishers, because these features are more likely to trick victims into compromising their information. Dynamic user interaction is often not supported by simple and even advanced phishing websites; extreme phishing pages, however, have been known to support this kind of functionality.

There are many different phishing page types. Besides traditional phishing pages, there is Web Single Sign-On (SSO) phishing. Whereas more traditional phishing pages target user data through generating pages tailored to imitate banking, shopping, and other commonly used websites, SSO phishing works in a more focused way to attain user authentication credentials for particular services, for example email or social media accounts. This allows phishers to login as the user in other dependent third-party applications and websites. Simple phishing pages tend not to support SSO phishing, whilst advanced pages are more likely to support traditional phishing as well as lower

quality SSO phishing features. Conversely, extreme phishing pages support higher quality SSO phishing as well as traditional phishing practices.

## **2.5 Conclusion**

The glossary of phishing attack types provided above, as well as the literature summarised on memorable or momentous phishing attack cases, serve to create a knowledge base for building defences which are up-to-date with the sophistication of phishing attacks. Our overall aim in this research is to create two different methods for reducing phishing attack success. The research above helps us to achieve this objective because it gives a sense of the immense complexity of phishing attacks, and how it is necessary to focus on every small detail that phishers use to get away with attacks. Also, by summarising important information, this section provides an overview and a database for creating an awareness program or materials. As such, it has been crucial in achieving the objectives of the thesis.

# CHAPTER 3

## LITERATURE REVIEW

### 3.1 Introduction

Due to the fact that Qatar will host FIFA World Cup 2022 there have been escalating cases of phishing attacks with the involvement of the labour force for the construction work of various stadiums and infrastructure (Whitaker, 2017). Safeen Malik is an example of this case; she was behind the phishing attacks regarding the issue of migrants' rights in Qatar. She created a fake profile and contacted different individuals via email and social media platforms since 2016 (Kovacs, 2017). Her aim was destined to target journalists and activists amongst other entities in Nepal and Qatar interested in migrants' rights and hack their private data intended for malicious purposes. The culmination of the phishing attack is due to the realization that the Qatar government has been exploiting migrant workers especially from Nepal. Safeena Malik used subjects of concern like human trafficking and ISIS funding and fake Google invitations to capture the attention of targeted users and lure them to phishing pages. (Kovacs, 2017).

According to experts, the phishing attacks might be linked to a state-sponsored actor although the Qatari government distanced itself from the attacks and showed concern on how to arrest the vice. Also, the attacks could be a motive to damage the reputation of Qatar (Kovacs, E. 2017).

Phishing attacks are evident and measures need to be put in place to arrest them. Due to their rise, government agencies especially in ICT and Internal Security should initiate civil education to the citizens as a way of creating awareness and the way to navigate through if they fall victims (Al-Hamar, Dawson and Al-Hamar, 2011). Also, individuals must be very careful while receiving emails from unidentified sources and as a way of avoiding being victims, they should avoid opening such emails or clicking on malware-infused links. Therefore, people should embrace enabling the two-factor authentication option to enhance credible security on their accounts.

Qatar as a developing nation is facing an increase in phishing attacks towards business and individuals themselves which poses a great compromise on the security and privacy

of information.

Phishing is defined as the deceitful action by cyber criminals whereby they send emails using fake profiles or accounts to the target group of individuals so that they can hack their accounts and gain access to their private information regarding usernames and passwords among other sensitive information. The act is intended for malicious purposes or theft once the targeted individual information has been hacked by the cyber criminals. Phishing mainly consists of copying the layout of common online sites' features like Google, Facebook, LinkedIn and Twitter and replicating their login pages such that individuals would mistakenly authenticate through them. Once they navigate through them their personal information is obtained (Kovacs, 2017).

According to Assolini, a senior security researcher "Qatar has faced a total of 93,570 of phishing attacks in the first quarter of 2017" targeting Qatari users' sensitive information such as credit card numbers (Varghese, 2017). The professional also clarified that "In 2016, about 47.48% of the phishing attacks targeted the financial sector while over 24% targeted global internet portals" with the consistent use of unsecured WiFi the users face the risk of cyber criminals gaining access to their private details to plan attacks accordingly (Varghese, 2017).

Today, the key target of malicious cyber hackers is on the rise over the Smartphones due to users being reluctant to have protection on the mobiles to mitigate those attacks. A new form of phishing attack known as Smishing which use Sms service is currently being preferred by cyber criminals. Smishing entails sending Sms to individuals to notify them of promotions or other critical news using a hyper link to access the specific site. Additionally, Assolini notes that protecting network devices to reduce chances of changing the DNS (Domain Name Servers) is the only way to lessen cybercrimes (Varghese, 2017).

Phishing attack has been also increasing in business ventures worldwide. Cyber criminals have been hacking individuals' businesses accounts consequently denying the chances of expanding the business and meeting management expenses. The phishing cases attached on businesses include receiving links with a virus rendering operating systems unusable until the encryption is unlocked. During the period the business account is hacked, the owners lose revenues due to the clients not accessing information on products they require

(Whitehead,G.P. 2017).

Qatar is known as a country of immigrants because the population of non-residents is 88.4%, however, the country is still relatively conservative where Islam is the main religion at 67.7% of all other religions (World Population Review, 2019). The Qatari culture is therefore influenced by the Islamic culture. This has a major influence on its citizens' in their daily lives. The Islamic culture has made the Qatari people to be generally trustful and generous. These characteristics has made Qatari people to be potential and easy target for phishing attacks. Al-Hamar et. al., (2010) found that one of the factors that make Qataris vulnerable to email phishing is culture. One of the respondent in their research mentioned that “Qataris are affected by the manner of incentives and disincentives which can sometimes lead to circulating and exchanging phishing e-mails and sites among people, which gives the phishing e-mails credibility.” They further mentioned that “They trust communications which appear to come from official sources or trustworthy institutes in the society, such as a bank.”

## **3.2 Technical Anti-phishing Technique Glossary**

The common goal of anti-phishing techniques is to detect and filter phishing attacks automatically. There are several phishing detection techniques.

### **3.2.1 Blacklists**

Previously detected phishing URLs, phone numbers, IP addresses and other keywords are frequently blacklisted. According to (Sheng et al, 2009), the blacklist technique is not effective against zero-hour phishing attacks. This is because a phishing site has to be previously listed in order to be detected. According to Sheng et al, (2009), 47% to 83% of phishing URLs were blacklisted after 12 hours.

### **3.2.2 Heuristics**

There are some common characteristics that are found in phishing emails or SMS. Therefore, they can be detected by heuristic algorithms in zero-hour phishing attacks. Heuristic algorithms have an advantage over blacklists in zero-hour phishing attacks

(Sharifi and Siadati, 2008). However, heuristics algorithms can sometimes misclassify legitimate content (e.g. legitimate emails or SMS).

### 3.2.3 Visual similarity

This detection technique uses content presentation instead of content code (Medvet, Kirda and Kruegel, 2008). A snapshot of every suspected website is taken and matched against a whitelist of legitimate websites such as Amazon, eBay and PayPal. The resultant greyscale images are analysed and key features are detected.

### 3.2.4 Data mining

Data mining techniques are used to detect phishing attacks as a document classification by using clustering and ML algorithms such as k-nearest neighbours (k-NN), logistic regression (LR), CART, SVM, random forests (RF) and neural networks (NN) (Abu-Nimeh et al, 2007). The methodology used in data mining techniques is expressed in Figure 3.1. Legitimate and phishing data are used as input data samples. The dataset is obtained from (Khonji.org, 2019). Text mining is used to prepare data into an appropriate data structure. Tools like RapidMiner (RapidMiner, 2019) can be used in text mining. Keywords that are commonly found in phishing attacks are then extracted as features which can be used to classify phishing attacks. A model can then be developed and its validity tested on the test dataset.

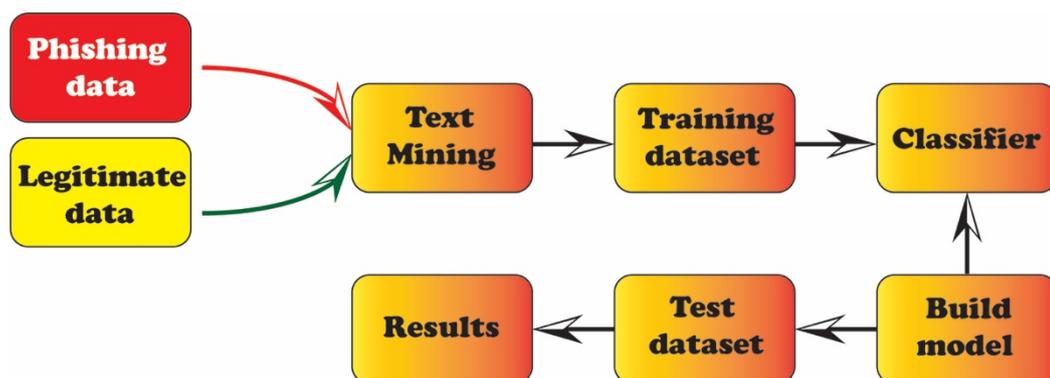


Figure 3.1 Data mining methodology (Aleroud & Zhou, 2017)

## 3.3 Technical Anti-Phishing Strategy Case Studies

### 3.3.1 ML-based methods

There are various strategies to detect phishing attacks and threats, for instance, using web topology structure, analysing various features of benign and phishing URLs and utilising a structural DOM. However, most of the detection strategies target phishing websites (Pan and Ding, 2006) (Sanglerdsinlapachai and Rungsawang, 2010). Detecting phishing pages by analysing the properties of a web page based on a structural DOM was proposed by Pan and Ding (Pan and Ding, 2006). Their approach is based on detecting abnormal behaviours caused by a phishing site which maliciously claims a false identity. This abnormal behaviour is compared with an honest site that is indicated by HTTP transactions and some web DOM objects in the page. Although this approach seems to be a good step for using ML to detect phishing sites, it has some limitations in coping with images shown on the tested websites. Moreover, the attacker can deceive the proposed scheme by reducing the number of structural features extracted. More work is needed on the identity extractor. Finally, phishing sites using Arabic cannot be detected using their approach. Li et al. (Li, Yang and Ding, 2016) proposed an approach which is based on the differences between phishing websites and the imitated target websites. They use a ball support vector machine (BVM) algorithm to detect and classify phishing sites by utilising the perspective of the web topology structure. In order to counteract the weaknesses of detection based on single web page features, they extract 12 statistic indices of websites as topological features. They used a BVM classifier to classify the feature vectors instead of utilising an SVM. Furthermore, their experimental results showed faster training speed and higher phishing detection accuracy. However, their experiment results did not show any superiority of BVM over SVM in term of detecting phishing websites with 96.6% and 96.2%, respectively. Their work was based on the English language, which cannot be used to detect Arabic phishing websites.

A rich set of web page features to detect phishing attacks was used by Xiang et al. to propose CANTINA+ which utilises the power of ML techniques and the expressiveness of these features (Xiang et al, 2011). CANTINA+ consists of three major modules. The first module uses phishing toolkits to find high similarity between phishing web pages and examines a webpage's similarity to known phishing attacks. The second module employs the feature that phishing attacks usually use login forms to request sensitive information. The third module uses 15 highly expressive features with ML algorithms to classify web pages. Furthermore, the model uses 15 features which consist of six features

that deal with the URL of the web page, four features that inspect its HTML content and five features that search the web for information about that web page. The proposed CANTINA+ compared six learning algorithms in the training of the phishing detector, including LR, SVM, Bayesian Network (BN), Adaboos, J48 decision tree and RF. In addition, the proposed system uses two filters. The first is a login form filter that classifies web pages with no identified login form as legitimate. The second is a near-duplicate phish detector which utilises hashing to catch highly similar phish. The CANTINA+ was evaluated by a randomised evaluation that achieved 92% TP and a time-based evaluation which had 92% TP on unique testing phish. However, the proposed framework cannot detect compromised legitimate domains, web pages that are made up of images or Arabic phishing web pages.

A domain top-page similarity feature in ML-based solutions for web phishing detection was proposed by Sanglerdsinlapachai and Rungsawang (Sanglerdsinlapachai and Rungsawang, 2010). They used the features introduced in the Carnegie Mellon anti-phishing and network analysis tool (CANTINA) which were utilised with a domain top-page similarity feature. These extracted features were applied to an ML-based phishing detection system. Furthermore, they used five out of eight features from CANTINA and added to them a domain top-page similarity feature. The proposed system produced a 19.50% error rate, which is quite high. The dataset used to evaluate the proposed system is too small to reflect the system's applicability for detecting phishing pages. Finally, this solution cannot detect phishing emails, pages or websites that use the Arabic language.

James et al. (2013) used a different approach; they employed ML techniques to analyse various features of benign and phishing URLs for detecting phishing websites (James, Sandhya and Thomas, 2013). They analysed four ML classifying algorithms in a Waikato environment for knowledge analysis (WEKA) workbench and MATLAB. The ML algorithms were Naive Bayes, J48 decision tree, k-NN and SVM. Moreover, they used host-based, page-based and lexical feature extraction to collect URLs and analyse their features. First, they gathered phishing and benign URLs. Second, they created a database of feature values from host-based, popularity-based and lexically based feature extractions. Finally, different ML methods were classified and evaluated by the database to select a particular classifier for implementing it in MATLAB. However, their tested sample was not big enough and could not give a clear decision about their solution and the chosen ML algorithm. In addition, even with the small sample of phishing websites

their solution had 93% detection accuracy. Like other surveyed solutions, this approach cannot be employed for detecting Arabic phishing websites. Another URL phishing detection strategy was proposed by Basnet and Doleck. They proposed a heuristic-based approach to classify phishing URLs by utilising only the available information on URLs (Basnet and Doleck, 2015). Their main aim was either alerting the user to the potential phishing URL or by masking the potential phishing threat. They used 138 features in detecting phishing URLs and these features were grouped into four different classifications, which were lexically based features, keyword-based features, reputation-based features and search engine-based features. Moreover, their approach started with running scripts to gather benign and phishing URLs in order to form datasets. Then, batches of scripts to extract a number of features were used to categorise the instances into their corresponding classes. Various ML algorithms implemented in the WEKA library (Hall et al, 2009) were employed to build models from training data. The used algorithms were SVMs with RBF kernel and linear kernel, MLP, RF, Naïve Bayes (NB), LR and C4.5 which is implemented as J48 in WEKA. Finally, they applied various sets of test data on the models in order to predict classes of data instances and compare them with the actual class of the data to calculate the accuracy of the classification models. In addition, they evaluated their solution approach with 31,000 non-phishing URLs and more than 16,000 phishing URLs. They acquired an error rate of 0.3%, 0.2% false positive and 0.5% false negative. Although this approach used 138 features, these features might affect the performance of the solution and the speed in detecting the phishing URLs. The authors did not mention in their solution how the URLs and users are masked and alerted, respectively. It also needs to be implemented or attached to a web browser to be tested in terms of accuracy and efficiency. The solution cannot deal with phishing URLs that target Arabic users.

Others have tried to detect phishing emails, for instance, Basent et al. who applied various ML approaches for detecting phishing emails by utilising known as well as new features (Basnet, Mukkamala and Sung, 2008). They employed 16 input structural features that can assist in discovering phishing attacks. These features are HTML email, IP-based URLs, age of domain name, number of domains, number of sub-domains, presence of JavaScript, presence of form tags, number of links, URL-based image sources, matching domains (From and Body) and keywords. In addition, they claimed that by using these features phishing emails can be detected with very limited a priori knowledge about the method used to launch a phishing attack or about the adversary. They used SVM (biased

SVM and leave-one-out models), NNs, self-organising maps (SOMs) and K-means. For evaluation purposes, they used 4,000 emails and achieved 97.99% as the best result. However, it is not clear how the solution was implemented or how the detection process worked. The features used by the proposed model need to be validated, ranked and take into account Arabic phishing websites. Salem et al. (Salem, Hossain and Kamala, 2010) looked at phishing threats from a different angle by investigating the level of risk awareness to phishing emails. They proposed a methodology to develop an awareness security programme to address the risk of phishing and intelligent tools to deal with a number of phishing techniques. They focused on phishing through email as it has a more direct effect on financial transactions in comparison to other methods. Their solution is a mix of various tools to avoid different types of attack such as email phishing and smishing. The proposed system has a number of phases. The first phase is a set of fuzzy logic rules to assess emails accurately as either phishing emails or not. These rules are used to implement the fuzzy logic expert system, which is another phase. However, it is not clear how many phases and fuzzy logic rules were used to build the system and they have not mentioned what type of AI algorithms were used. In addition, they claim the system marked 78% of the email set as phishing emails, yet it is not clear how this result was obtained or whether it is accurate.

### **3.3.2 Cloud-threat Inspection Appliance**

*Lin et al.* proposed a cloud-threat inspection appliance (CIA) system to defend against spear phishing threats (Lin et al, 2015). By using CIA to develop a transparent hypervisor monitor and hardware-assisted virtualisation technology such as Intel-VT and AMD-V, the authors were able to conceal the presence of the detection engine in the hypervisor kernel. In addition, the CIA can analyse the common behaviours of malware such as process information system activity calls on a system. It also made a test program and monitoring module executed at various points. Thus, malware is unable to determine whether it is in a monitoring environment or not. The CIA also utilised a document pre-filtering algorithm to filter PDF attachments and prevent them from being sent to the hypervisor monitor for deeper analysis. The authors claimed the document pre-filtering algorithm detected 77% of malware attached to PDF files. During 2014, 65 unknown samples that were not detected by commercial anti-virus software were found by the CIA (Lin et al, 2015). It inspected 780,000 emails belonging to 200 user accounts at a company. However, the big question is the overload incurred by installing the CIA on a

hypervisor. Moreover, the solution made a test program and monitoring module that executed at various points, which added more load to the host system. Finally, the solution does not cope with phishing websites that use the Arabic language.

### **3.3.3 Learning-based mechanisms**

Adebowale et al., (2019) used an Adaptive Neuro-Fuzzy Inference System (ANFIS) based robust scheme using the integrated features of the text, images and frames for web-phishing detection and protection (Adebowale, Lwin, Sánchez, & Hossain, 2019). The proposed scheme uses an intelligent ANFIS algorithm with a knowledge model and one input. The ANFIS is a network structure method that facilitates systematic computation of gradient vectors, it combines the least squares and the gradient descent methods, and it utilises a useful fusion learning technique to derive the output error. It allows data learning by using the connectionist approach for computation, and therefore the exact rules are from the fuzzy inference point of view. The method in this study uses a table in which the features or data of a valid website are stored for reference purposes. The data will include website images, text and frames features. A total of 35 features are extracted to model the ANFIS; 22 elements represent the structure of the text-based properties, 8 features represent the frame-based properties, and 5 elements constitute the image-based resources of the website. The focus of the proposed intelligent phishing detection and protection scheme is to hinder website-based phishing attacks that aim to entice victims into giving out their confidential and sensitive information. Moreover, the proposed solution achieves 98.3% accuracies.

*J. Mao et al* explored learning techniques to improve phishing detection techniques and proposed a learning-based mechanism to evaluate the similarity of web page layouts and identify phishing (Mao et al., 2018). The proposed solution is based on the aggregation analysis mechanism to automatically generate rules to determine layout similarity of web pages and then detect phishing pages. Moreover, it defines the rules to extract and create effective page layout features and develop a phishing page classifier based on two typical learning algorithms, supporting vector machine and decision tree. It first trains a similarity classifier using page layout features, then uses the classifier to detect phishing pages. The evaluation phase used 2,900 phishing web pages from phishtank.com. It showed that the approach was effective in creating classifiers and detecting phishing pages via page layout similarity.

### 3.3.4 Database-Match Whitelisting

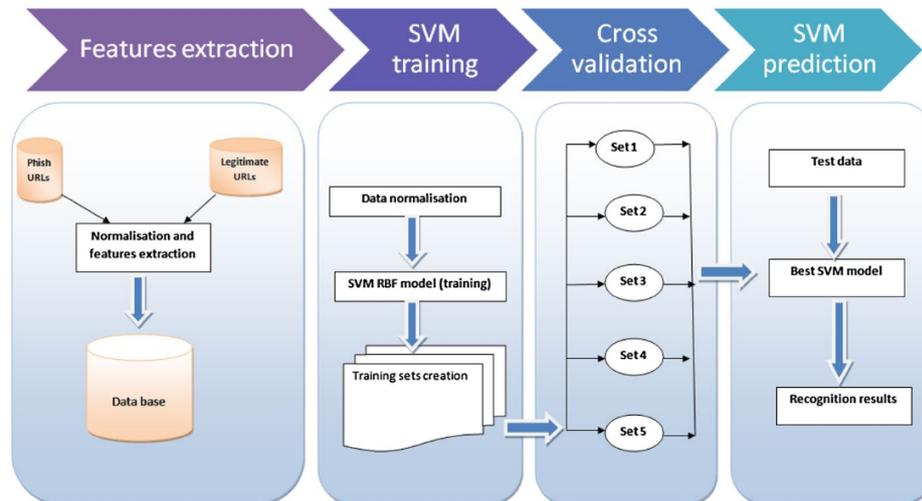
*Jain et al (Jain and Gupta 2016)* proposed a solution that has a fast access time and high detection rate. The proposed system is based on a white-list database, meaning that if a user visited a webpage that is not in the database, the system will generate an alarm and warn the user not to disclose any information with that website. The other feature that they are using is to validate a webpage based on hyperlink extraction features. However, the proposed method can be bypassed by an attacker when a “while-listed” domain is compromised and phishing webpages uploaded into the white-listed domain directory. For example, we have abc.com as a white-listed domain, but attacker will compromise the domain and create a directory abc.com/logins, and then upload the phishing webpages to that directory. In this case, the proposed method become useless. This method is one of the common methods for attackers to perform a phishing attack with a high rate of success as everything is legitimate, except the webpages in the logins directory which is the phishing webpages uploaded by attackers to fool the users.

### 3.3.5 Features Analysis Whitelisting

*Chen et al (Chen, Huang and Ou, 2015)* proposed a two-stage drive-by-download attack detection mechanism to identify malicious webpages and detect drive-by-download web attacks. The first process is to identify suspicious websites based on domain reputation features. Afterwards in order to reduce the detection time, it will sandbox the suspicious ones. As the authors claiming, the WHOIS database is not reliable for their solution, therefore they are using DNS server queries and propose novel reputation attributes. The authors proposed three sets of feature selections (12 features in total), Domain-Based Features, Authority-based Features and ASN-based features. These features are, Number of CNAME records, Number of IP addresses, Numeric Ratio, Longest Meaningful Substring Ratio (LMSR), Type of Level Domain, Authority-based Features, Number of name servers, Rogue, ASN-based Features, Numbers of ASNs, Number of countries, Mode Equivalence Ratio (MER), Degree of Centralization (DoC), Corresponding country. However, Domain-Based features technique can bypass when an attacker is using services such as CloudFlare to hide their CNAME.

Zouina and Outtaj, (2017) proposed a lightweight URL phishing detection system using Support Vector Machine and similarity index. The proposed solution works with six URL

features to perform the recognition. These URL features are URL\_Size, Number\_of\_Hyphens, Number\_of\_Dots, Number\_of\_Numeric\_Chars, IP\_presence and Similarity\_IndexI. The proposed system is mapped out in Figure 3.2.



**Figure 3.2 The proposed Phishing detection system (Zouina and Outtaj, 2017)**

They used Hamming distance between the phishing website and its target and the previously mentioned extracted features from URL. The authors claim that their work is lightweight and can be integrated into Smartphone and Tables. However, Support Vector Machine has its own limitations, one of which “is the high algorithmic complexity and extensive memory requirements of the required quadratic programming” (Horváth et al., 2003). In addition, the proposed method was tested with only 2,000 records with a detection rate of 95.80%. Detection rate accuracy is too low for this amount of data (Zouina and Outtaj, 2017).

One strategy put forward by the literature (Jain and Gupta, 2016) suggests the usage of an automatically updating white-list which keeps a record of legitimate sites, and flags up sites whose URLs do not appear on it, notifying users of potential danger. Two elements are involved in the process of verifying websites to be legitimate: 1) a matching module examining domains and IP addresses, 2) analysing source code for features of hyperlinks. The study found that this method succeeded in achieving 86.02% true positive rates, with only 1.48% false negative rates.

Others have outlined a multi-tier model which classifies phishing emails for the filtering process using a system of ranked prioritisation, applying different levels of importance to message content and header. This amended algorithm managed to significantly improve false positive rates and reduce complexity. Another study (Jeeva and Rajsingh, 2016) produced an algorithm which looks for certain URL features, including keywords within the path portion, an invalid top-level domain, and transport layer security in their attempt to detect phishing websites. They noticed several other URL traits common to phishing sites: the presence of several forward slashes; the host portion containing a dot; unusual URL length. Using association rule mining, they used these variables to generate a rule for an apriority algorithm. This method resulted in the detection of approximately 93% phishing URLs.

### **3.3.6 Heuristic Approaches**

Another model proposed by the literature (He et al, 2011) adopts a heuristic approach, focusing on 12 components of websites in its analytical process. This technique borrows from established methods from the literature for a deeper meta-analysis: it adopts nine from the Anomaly Method, one from the CANTINA method and two from the Pilfer method. Using a Support Vector Machine (SVM), these 12 features were employed in a classification process designed to distinguish legitimate sites from phishing sites. In this way, the authors demonstrate how a many-pronged approach, using the findings of the literature to its advantage, can culminate in a method which produces more satisfactory results than any of its constituent methods.

Lee et al. (2015) developed a heuristic approach to anti-phishing which examines URL features identified via Google's page rankings, search suggestions, suspicious URL property values and patterns. They compared multiple machine learning algorithms and their ability to generate classifiers, and came to the same conclusion as Basnet and Doleck (2015) in finding the random forest method to be the most effective. The study found that the employment of this method resulted in a detection rate of approx. 98.23%; this is by no means an unsuccessful technique, and it may hold certain value for future retrospective meta-analyses in view of further innovation, but in its current form the level of complexity the method entails is not satisfactorily proportionate to its effectiveness in comparison with other methods.

Webpage anomalies were identified and analysed by the authors (Pan and Ding, 2006), particularly discrepancies between a website's HTTP transactions, the manner in which it identifies itself, and the apparent structural features available. Two components make up the method of analysis, and each component uses particular data sets for its analysis: 1) Identity extraction: the identity appears as a unique string which appears in its domain name, along with an abbreviation of the organisation's title. 2) Page classification, which is concerned with two elements of source structure: HTTP transactions and W3C DOM objects on web pages. Using these inputs with an SVM, the authors processed 279 phishing sites and 100 legitimate sites. The false positive rate was approximately 13%, which still leaves users contending with a considerable probability of phishing pages going undetected.

*Li et al. (2013)* developed using the Transductive Support Vector Machine; a semi-supervised learning method, as its foundation, was able to improve upon the accuracy of the SVM's detection rate by 8.3%. They employed TVSM to classify sites which had been analysed for sensitive information and web images. As evidenced by the outcome, this proved to be significantly more effective than the previous SVM model.

In 2014, others developed the anti-phishing method of generating a page signature using term frequency (Roopak and Thomas, 2014). Using this signature with a search engine to ascertain the legitimacy of websites, they employed cosine similarity and tag comparison to process the results of the search. By using Google's page-ranking data in tandem with the generated page signatures, they were able to achieve a low false positive rate and a high detection rate by contrast with previous methods.

### **3.3.7 String-matching**

The string-matching method is yet another tool for protecting users from phishing. A study (Abraham and Raj, 2014) used Edit Distance and the Longest Common Subsequence (LCS) in their process of discernment. After dividing URLs into several tokens, they processed the number of occurrences of each blacklisted token. In their experiment using these string-matching algorithms, their method produced accuracy rates of 99.5% for Edit Distance and 99.1% for LCS.

*G. Sonowal and K. Kuppusamy* proposed a multilayer model to detect phishing, titled as

PhiDMA(Phishing Detection using Multi-filter Approach) (Sonowal & Kuppusamy, 2017). The PhiDMA model incorporates five layers: Auto upgrade whitelist layer, URL features layer, Lexical signature layer, String matching layer and Accessibility Score comparison layer. The proposed model centred on both sighted and persons with visual impairments. The dimension of whitelist depends upon the number of URLs which are legitimate, that is, an increase in the size of the whitelist would assist in getting high detection accuracy. One of the significant advantages of a whitelist is that it provides the 100% accuracy rate, if the URLs list is at the scale of the web. However, a central limitation of this approach is updating. The proposed model has examined both phishing and legitimate URLs. If users visit next time an identically Legitimate URL, then the model permits them to visit the page by verifying in the whitelist, however, if identical phishing URLs, the model requires the same time to examine with all filters as previously it did. Moreover, phishing sites only live for a few days, and it has been seen that cyber communities daily detect a huge number of sites. Hence, it repeatedly requires to update daily these number of URLs in the blacklist. A prototype implementation of the proposed PhiDMA model is built with an accessible interface so that persons with visual impairments shall access it without any barrier. The result from the experiment shows that the model is capable to detect phishing sites with an accuracy of 92.72%.

### **3.3.8 URL Token-System**

Another method involves weighting URLs in a token system. Using a query webpage, Tan and Chiew (2017) picked out identity keywords and used them to generate page signatures, in turn inputting these into a search engine. This resulted in the identification of websites based on target domain name. Using standard datasets, the outcome was recorded at a rate of 92.2% true negatives and 99.2% true positives. Whereas conventionally, phishing filtration systems have been largely language-dependent, extracting data using keywords extraction algorithms, this method suggests that it is possible to achieve highly satisfying results and protect users from phishing without language-dependent methods.

### **3.3.9 Lexical Methods**

One such lexical method (Basnet and Doleck, 2015) uses a search-engine based method and uses a keyword- and reputation-focused dataset. Assessing 7 machine learning

classifiers, their study revealed the random forest classifier to be at the forefront in terms of execution, whilst Naïve Bayes was overall the worst performer. Their anti-phishing method entails the use of an expansive range of identified elements, numbering 138 in all. The result was a resounding success: a 99.8% rate of true positives, and a 99.5% rate of true negatives.

### **3.3.10 Inclusive Anti-Phishing Technology**

As researchers become more cognizant of those with differing needs, new methods have emerged which adapt to the needs of differently abled users. One such method (Sonowal and Kuppusamy, 2016) is MASPHID, which was developed and specifically tailored to the needs of those who use assisted screen readers. This model is unconventional in that it uses aural and visual parameters for its analysis. The foundation of this method is an up-to-date database of legitimate banks which operates as a whitelist for verifying URLs. When the system comes across a URL which is not whitelisted, it extracts the URL's top-level domain and takes a screenshot of the site, using these as inputs for a search engine. Since the first result is the one which is almost certain to be legitimate, the program screenshots the first page of the first result for comparison with the suspect website. Employing the root mean square method, the two sites are examined for differences. Whilst this model is chiefly applicable to those who need visual assistance due to a disability, the finding that aural and visual parameters can be effectively used to guard against phishing is interesting and suggestive of future innovation in the field.

### **3.3.11 PhishWHO**

*Choon Tan et al.* proposed PhishWHO, which is a phishing detection technique based on the difference between the target and actual identities of a webpage (Tan, Chiew, Wong, & Sze, 2016). It is an extension of our proposed phishing detection system in previous work. PhishWHO is based on a permanent phishing characteristic that stays intact over time, namely the discrepancies between the target and the actual identities of the query webpage. The target identity is defined as the domain name belonging to a legitimate brand that the phishing webpage deceptively represents, while actual identity refers to the query webpage's domain name. For legitimate webpages, the target identity often points to its own domain name, while phishing webpage does not. As such, the proposed method checks whether the query webpage is promoting itself, or promoting

another existing legitimate webpage. By applying the proposed information processing techniques, both the target identity and the actual identity can be systematically derived from the query webpage. Moreover, the proposed system can be divided into three phases. The first phase extracts identity keywords from the textual contents of the website, where a novel weighted URL tokens system based on the N-gram model is proposed. The second phase finds the target domain name by using a search engine, and the target domain name is selected based on identity-relevant features. In the final phase, a 3-tier identity matching system is proposed to determine the legitimacy of the query webpage.

### **3.3.12 Case-Based Reasoning**

Abutair and Belghith introduced a Case-Based Reasoning (CBR) Phishing Detection System (CBR-PDS) (Abutair & Belghith, 2017). It mainly depends on CBR methodology as a core part. The authors claimed that the proposed system is adaptive and dynamic as it can easily adapt to detect new phishing attacks with a relatively small data set in contrast to other classifiers that need to be heavily trained in advance. The system was tested using different scenarios on a balanced 572 phishing and legitimate URLs. CBR-PDS is tested against basic URL features and intra-URL relatedness features. The set of features can be varied, and the challenge here is how to select a minimal set of features with an appropriate weighting mechanism to effectively participate in the prediction process. A comprehensive or long set of features will add computation overhead and degrades the performance. Experiments show that the CBR-PDS system accuracy exceeds 95.62%, yet it significantly enhances the classification accuracy with a small set of features and limited data sets.

## **3.4 Spearphishing**

Hackers use spear phishing attacks to design unknown payloads that are sent to target victims as email attachments. These spear phishing attacks are still one of the preferred approaches for infiltrating target networks. The typical scenario of a spear phishing attack is an email specially crafted that is sent to a specific recipient. The recipient clicks the URL link in the mail or opens the attachment, which compromises the system. This section outlines several researches into anti-spearphishing technology, evaluating each case for its strengths and weaknesses.

## **3.5 Spearphishing Defence Case Studies**

### **3.5.1 Machine Learning**

Most of the proposed works target spear phishing emails, for instance, Dewan et al. used Symantec's Enterprise email scanning service to characterise and examine a true positive dataset of normal phishing emails, spear phishing and spam (Dewan, Kashyap and Kumaraguru, 2014). They utilised two different datasets that combined into one dataset: the two datasets comprised LinkedIn profiles and a dataset of emails (a combination of non-targeted attack and targeted attack emails). The authors proposed a model to detect spear phishing emails. The model used four ML algorithms: RF, J48 decision tree, Naïve Bayesian and decision table. The model employed a total of 27 features that consisted of 9 social features from each LinkedIn profile and 18 stylometric features from each email in the email dataset. The Weka data mining software (Hall et al, 2009) was used for the entire analysis and classification tasks. The authors applied 10-fold cross validation to test the accuracy of their classification results. The 18 stylometric features extracted from the email dataset were further classified into three categories: body features, attachment features and subject features. In order to test the models, they used a large dataset that had 9,353 non-targeted attack emails sent to 5,912 non-victims and 4,742 targeted attack emails sent to 2,434 victims. They achieved an overall maximum accuracy of 97.76% in identifying spear phishing emails. This proposed model is a good step towards detecting spear phishing emails, yet the social features set was small and only half the size of the stylometric features. Thus, the solution should employ more social features. The solution may achieve better results with SVMs. Finally, no analysis or detection approach for spear phishing emails were sent to Arabic users.

### **3.5.2 Text Mining**

Some researchers have looked at the spear phishing email text mining and URLs (Abu-Nimeh et al, 2007), (Chandrasekaran, Narayanan and Upadhyaya, 2006), (Chhabra et al., 2011). Abu-Nimeh et al. studied the performance and predictive accuracy of different ML classifiers used in text mining such as CART, LR, BART, RF, SVM and NNs. They constructed a phishing dataset of 2,889 emails, which were a set of 1,171 raw phishing emails and a set of 1,718 messages collected from their mailboxes. They constructed the dataset by stripping all attachments from the emails. They extracted the header information in order to keep the email subject and body. Then, they extracted the HTML elements and tags from the body of the emails. Afterward, they filtered out stopwords

from the plain text of the emails' bodies by employing a list of 424 commonly used English stopwords. Then, they removed the suffixes from words to get their common origin. Finally, they searched for the most frequent terms using TF-IDF (term frequency-inverse document frequency). Therefore, terms that appear often in a document and do not appear in many other documents have a higher weight. In addition, they used 43 features (keywords) for training and testing the classifiers. One binary response variable was employed to indicate either the email is phishing=1 or legitimate=0. The features represent the frequency of the "bag-of-words" that appear in phishing and legitimate emails. However, the ever-evolving techniques and language employed in phishing emails might make it hard for this approach to be effective over a long period of time or when used for detecting Arabic spear phishing, In terms spear phishing emails' URLs.

### **3.5.3 Extracting Email Features**

Using and extracting email features are considered to be a mitigation strategy by some proposed approaches; for example, Toolan and Carthy proposed a detection system that employed 40 features (Toolan and Carthy 2010) and Ma et al. proposed a classifier to detect phishing emails using hybrid features. Toolan and Carthy (Toolan and Carthy 2010) used behavioural features such as the total number of characters in the sender field, the difference between the sender's domain and the reply to the domain, the number of words in the sender field and the difference between the sender's domains and the email's model domain. These features were further divided into five distinct categories. The categories were body-based features that were extracted from the body text of the email, URL-based features which were extracted from the anchor tags in HTML emails, subject-based features that were extracted from the subject line of the email, script-based features which reflected the presence or absence of scripts in the emails and sender-based features that were extracted from the sender's email address information. However, this model cannot cope with new phishing emails and URLs as it uses a static approach. It also utilised 40 features that would have a big impact on the efficiency and performance of the detecting approach.

*Ma et al. (Ma et al, 2009)* extracted features' vectors from emails that effectively represent the instances, such as the "subject" of an email containing precisely the keywords that best characterise the email class. In addition, the absence and presence of each feature illustrated additional clues to the email's class. The classification system utilised in the detecting approach consisted of a feature generator, ML method selection,

inductor and feature evaluation. Furthermore, the detection approach derived seven features categorised into three classes, which were content features, orthographic features and derived features. It also applied five ML algorithms: decision trees, RF, MLP, naive Bayes and SVM. Their results stated that decision trees worked best in identifying phishing emails. However, the set of classification features needs to expand and consider additional features. The feature normalisation was based on the values directly derived from instances where some value is unnecessarily large. Finally, the system cannot detect new phishing attacks without extracting new features from these emails and including them.

Email filtering tools were used to detect spear phishing emails (Islam and Abawajy 2013). Islam and Abawajy proposed a multi-tier classification model for phishing email filtering (Islam and Abawajy 2013). The model classified the email message in a sequential fashion by using the first two-tier ML algorithms, and their outputs were sent to the analyser. Then, the analyser sent the outputs of the two-tier ML algorithms to the corresponding mailboxes based on the labelling of the algorithms. If the first two-tier ML algorithms misclassified the email messages, the third-tier ML algorithm would be invoked by the analyser. The third-tier ML algorithm is employed by the model to classify and send misclassified email messages to the corresponding mailboxes based on the identification. Moreover, the authors examined the effect of rescheduling the ML algorithms in a multi-tier classification process to come up with the optimum scheduling. They also proposed a method for extracting the features of phishing emails based on weights of message content and message header. The method ranked the selected features according to their priority. The results of their experiments showed that the model achieved 97% detecting accuracy with 2% false positive and 9% false negative. Although the model had good results in reducing complexity, it needs to validate some assumptions and claims such as running the test on the same set of phishing email corpora but accumulating features when phishing data is known in advance. It also has to be tested against other types of phishing emails such as emails written in Arabic.

One of the detection methods looked at phishing websites or emails employed to direct victims to these websites (Fette, Sadeh, and Tomasic, 2007). Fette et al. proposed an ML-based approach for detecting phishing attacks (Fette, Sadeh, and Tomasic, 2007). They tried to examine whether a communication is deceptive or not, based on data from the email or attack vector (an internal source) and information from external sources.

Afterward, this combined information was utilised as the input to an ML classifier to make a decision on whether the input had data generated to deceive a user. In addition, they used a collection of 10 features that mostly test URLs and the presence of JavaScript to flag emails as phishing. Nine features were extracted from the emails and the last feature was obtained from a WHOIS query. They evaluated the detection approach by using large datasets, 6,950 normal emails and 860 phishing emails. Their filter identified 96% of the phishing emails with a false negative rate of 3.6% and a false positive rate of 0.13%. Although this approach was heavily dependent on URL-based features, the URL-based features made it ineffective for detecting attachment-based attacks (spear phishing) or phishing emails that do not contain a URL. Furthermore, this approach cannot detect phishing or spear phishing emails that use Arabic to represent their text.

### **3.5.4 User Awareness Programs**

Users' training and awareness are very important for detecting spear phishing emails and promoting their level of understanding of new tricks employed by spear phishing emails. Caputo et al. conducted a largescale experiment to explore the effectiveness of embedded training (Caputo et al, 2014). The experiment tracked workers' reactions to a series of crafted spear phishing emails in a variety of immediate training and awareness activities. In addition, the experiment included four different training conditions that were:

- gain-framed and individually focused embedded training (you keep yourself from harm),
- loss-framed and individually focused embedded training (you put yourself at risk),
- gain-framed and other-focused embedded training (you keep your co-workers from harm),
- loss-framed and other-focused embedded training (you put your co-workers at risk).

The experiment was performed in an actual industrial environment using a very large sample. It was conducted within the organisation's email system and network. Thus, the authors could send and track spear phishing emails in order to monitor each employee's reactions to them. They used quite a large sample of 6,000 workers who were categorised by cumulative job experience. Their results indicated that immediate feedback and tailored framing do not suffice to increase reporting of spear phishing or reduce click rates. Although this study was conducted in a real-world scenario, there are many questions that need answers and considerations such as the fact that there was no practical

way to ensure that a participant actually finished reading the training materials completely. Finally, this work did not consider using other spear phishing emails that are written in other languages such as Arabic.

## **3.6 Smishing**

### **3.6.1 Smishing Introduction**

SMS spam is unwanted and unsolicited messages transmitted over a mobile network. The SMS mobile communication system is attractive due to its cost effectiveness by using unlimited pre-pay SMS packages. In addition, as SMS is a trusted service that makes subscribers comfortable with using it for confidential information exchange, it gives attackers a golden opportunity to acquire higher response rates than email spam.

### **3.6.2 Anti-Smishing Case Studies and End-User Behaviour**

There are a number of techniques and strategies that have been used to detect and prevent smishing attacks, for example, studying end-user behaviour towards mitigating the risks of accessing online services and facilities by using mobile devices (Yeboah-Boateng, and Amanor, 2014), analysing malicious applications used particularly for targeting Android smartphones and employing content-based technologies to detect these attacks. In addition, Osei et al. conducted a study to assess smishing and phishing attacks against mobile devices, studying the implications of end-user behaviour towards mitigating the risks of accessing online services and facilities by using mobile devices (Yeboah-Boateng, and Amanor, 2014). Their study interviewed and observed 20 end-users for their knowledge and behaviour when confronted with smishing attack situations. The samples were selected from amongst students on a university campus, friends and family. They utilized Telcos operating in Ghana to analyse and classify the datasets. The used datasets consisted of 60% men and 40% women. Microsoft Windows, Android and iOS were the OS for the selected sample of mobile devices. The results illustrated that men were more comfortable and trusting in cyberspace than women. Hence, they are more susceptible to phishing attacks than women. Moreover, 35% of the users would almost never scrutinise any messages, whilst 55% would occasionally examine the messages received as perceived threats. Although the authors used Telcos operating in Ghana to analyse and classify the datasets, the dataset was too small and cannot help in validating their study. Finally, they did not suggest any mitigation or reflect any ideas for increasing the users' awareness.

### **3.6.3 Attack Classification**

Splitting SMS phishing attacks into different classes in order to produce better detection accuracy by using a Bayesian technique was proposed by Feresca et al. (Foozy et al. 2014). The proposed solution is a rule-based technique that can separate SMS phishing from an SMS spam class and then generate an enhanced SMS phishing corpus. The authors used various SMS phishing characteristics from previous papers and they downloaded a total of 5,572 SMS from UCI's Machine Learning Repository. They claimed the classification results were 99.8064%. However, these results did not show any usefulness of this technique or how the system can be used to defend against SMS phishing attacks. Further, they claimed using various SMS phishing characteristics with winning announcement and advertisement features, yet they did not report the characteristics used or explain how the characteristics were linked with the two features. Finally, they did not consider the languages used for SMS phishing attacks.

### **3.6.4 SMS Spam Filtering**

A number of researchers have looked at SMS spam filtering, for instance, Delany et al. who discussed and analysed the current state of the art in SMS spam filtering. They focused on the content-based technologies that are becoming more and more necessary for defending against SMS spam (Delany, Buckley, and Greene, 2012). This study investigated the nature of current SMS spam by collecting a corpus of SMS spam messages and scraping messages from two public consumer complaints websites: GrumbleText and WhoCallsMe. Users of GrumbleText have the ability to report spams by forwarding them to a shortcode which preserves the original form of the message. Removing duplicates resulted in a corpus of 571 unique spam SMS messages. The WhoCallsMe website has the advantage of permitting users to report unsolicited calls and text messages that are then indexed by source phone number. Then, erroneous entries such as mismatched quotes, foreign language text or user-added parentheses were removed from the list of candidates, resulting in a list of 436 messages. The authors clustered the messages and divided the data into a flat, disjoint partition via spectral clustering methods. They found that the most common cluster was the premium rate fraud that had the clusters chat, prizes, voicemail claims and dating comprise 43.9% of the messages.

### **3.6.5 Malicious Feature Analysis**

Others investigated well-known smartphones OS such as Windows and Android. Park, Kim and Ryu proposed a solution to intercept, collect and analyse malicious applications used particularly for malicious smishing attacks targeting Android smartphones (Park, Kim, and Ryu, 2015). The solution targets malicious Android applications and detects them by using malicious feature similarity along with static analysis, which can help in blocking the installation of malicious applications on smartphones. The proposed approach consisted of a malicious application detection system of cluster-based servers and an application-based client program. The application-based client program was installed on smartphones to send anonymised information about incoming records (URL clicks, SMS/MMSs and internal mobile message) to the detection system of cluster-based servers. A server from the detection system of cluster-based servers is composed of multiple machines: a message parser, APK analyser, web crawler and APK collector. In order to validate the solution, the authors downloaded 1,200 malware samples gathered from the Android Malware Genome project (Jiang and Zhou, 2012). Twenty per cent of downloaded samples were used for detecting and 80% of the total was used as a signature. They achieved 100% detection accuracy in each APK family; however, the results were not sufficient and the samples (28 malware attacks) not enough to test the solution. In addition, no suggested improvements to static analysis were considered in this approach to improve the total efficiency of the detection mechanism which is based on signatures kept on the system.

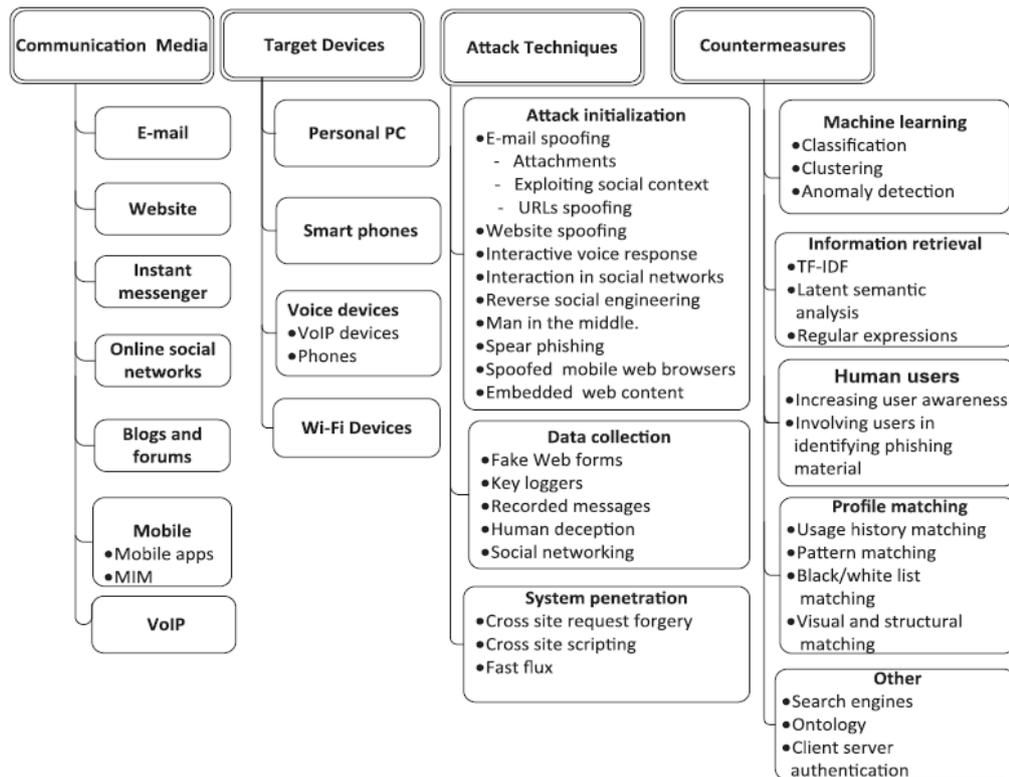
### **3.6.6 User Awareness**

*D. Goel and A. Jain*(Goel & Jain, 2018) did a detailed analysis on mobile phishing – attacking techniques and defence mechanisms. This paper gave a broad overview of mobile phishing attacks by considering various attacking techniques used by the attackers and solutions proposed by the researchers. Performance matrices and datasets used for experimental purpose have also been illustrated by the authors. In addition, they presented some of the open challenges and issues related to mobile phishing attacks. They illustrated the paper in four folds. First, they discussed in detail about mobile phishing attack, its history, motivation of attackers, and security concerns of smartphones. Then, they analysed various mobile phishing attacks and provide a taxonomy of the same. Third, they provided a taxonomy of numerous recently proposed solutions that detect and defend users from mobile phishing attacks. Finally, they discussed different issues and

challenges faced by researchers while dealing with mobile phishing attacks. Further, they also discussed datasets and evaluation matrices used by researchers for evaluating their approaches. Moreover, they had an important findings and conclusions, for instance, user education or training is necessary for creating awareness among the users so that their susceptibility to fall victim to phishing attack can be reduced, yet education alone cannot guarantee positive behaviour reaction.

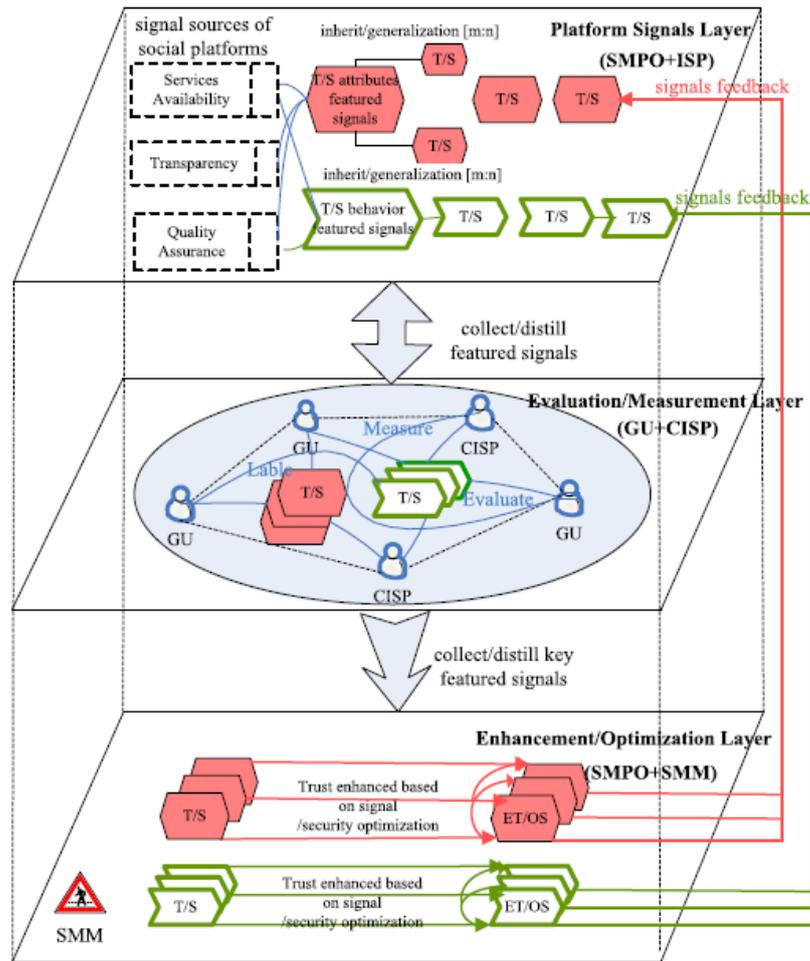
### **3.6.7 User Awareness Defence Approaches Introduction**

Aleroud and Zhou investigated phishing attacks and anti-phishing techniques developed not only in traditional environments such as e-mails and websites, but also in new environments such as mobile and social networking sites (Aleroud & Zhou, 2017). They provided a system review of extensive research on phishing techniques and countermeasures. Moreover, the authors proposed a phishing taxonomy that addresses phishing environments, techniques and corresponding countermeasures as shown in Figure 3.2. It identifies the dimensions of phishing via the process lens. Particularly, the authors identify the characteristics of phishing attacks in emergent communication media and analyse anti-phishing techniques in relation to the communication media for the first time. In view of the significant practical implications of phishing detection, they introduced a comprehensive comparison between research anti-phishing tools and another comparison between commercial anti-phishing tools. Additionally, they applied the dimensions to analyse anti-phishing tools, and ranked the techniques based on their performance. The analyses revealed several new categories of countermeasures. This research found that human users play an important part in the loop of phishing attacks, who can potentially serve as the most effective line of defence.



**Figure 3.3 The proposed phishing taxonomy (Aleroud & Zhou, 2017)**

Z. Zhanga et al. did a survey on the state-of-the-art of social media networks' security and trustworthiness particularly for the increasingly growing sophistication and variety of attacks as well as related intelligence applications (Zhang & Gupta, 2018). The authors highlighted a new direction on evaluating and measuring those fundamental platforms, meanwhile proposing a hierarchical architecture for crowd evaluations based on signalling theory and crowd computing, which is essential for social media ecosystem as shown in Figure 3.4.



**Figure 3.4 The framework of crowd evaluation and measurement for trustworthiness and security of social media platform based on featured signals (Zhang & Gupta, 2018)**

One of the important findings is that Internet users need to be well-informed about the threats that are faced by their personal and financial information. They should be able to behave securely and use reliable security measures at their aid. The direction is significantly theoretically meaningful for realizing secure interaction, sharing and digital rights management of social media content, continuously improving platform trust and security, as well as establishing a trusted and security-preserving social media ecosystem. It has also better applicable vision and practical application value for healthy, normal and rapid development of the digital media content industry. Building and strengthening trustworthiness will provide awareness and guarantee of safety to users.

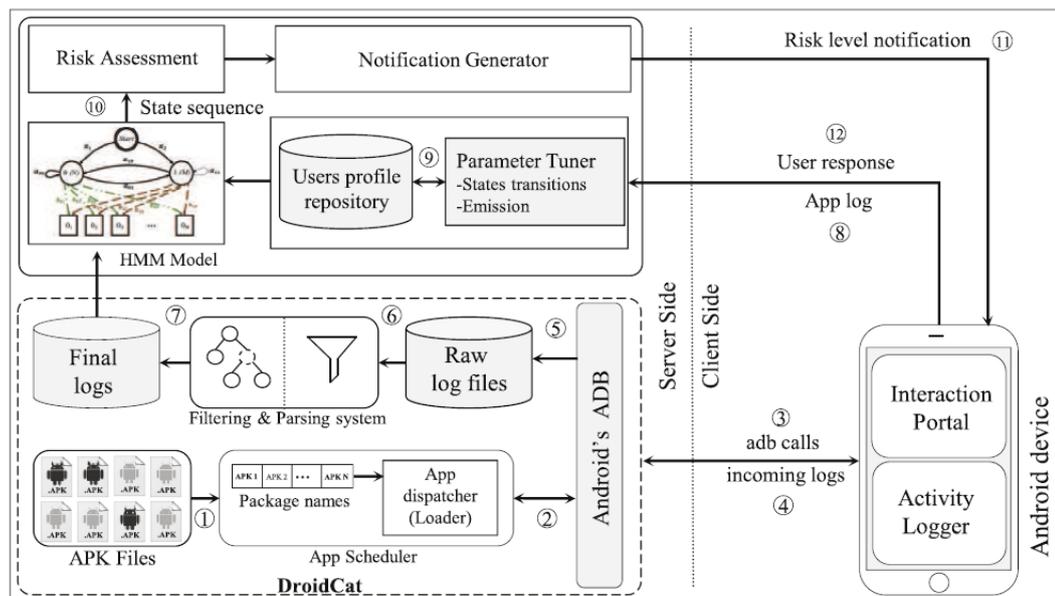
### 3.6.8 The role of user traits and related studies

*S. Curtis et al.* (Curtis, Rajivan, Jones, & Gonzalez, 2018) investigated the relationships between three personality traits—Machiavellianism, narcissism, and psychopathy and phishing effort, attack success, and end-user susceptibility to phishing emails. Participants were recruited in two stages. The first set of participants acted as attackers, creating phishing emails. The second set of participants acted as end-users, reading both benevolent and phishing emails and indicating their likely behavioural response to each email. This study finding suggest that attackers' Dark Triad scores relate to the effort that they put in writing a phishing email, but do not predict phishing success. Instead, the end users' Dark Triad scores predict the success of phishing emails. One of the interesting points is that higher levels of attacker Machiavellianism were linked to increased phishing effort, while end-user narcissism was associated to greater vulnerability when receiving phishing emails. Furthermore, this study suggests that narcissistic end-users were marginally more susceptible to phishing emails that originated from narcissistic attackers. These results have important practical implications for training, anti-phishing tool development, and policy in organizations.

*De Kimpe et al.* used an integrative lifestyle exposure model to study the effects of risky online routine activities that make a target more likely to come across a motivated offender (De Kimpe et al., 2018). Insights of the lifestyle exposure model are combined with propensity theories in order to determine which role impulsivity plays in phishing targeting. To achieve these objectives, data collected in 2016 from a representative sample (n=723) were used (De Kimpe et al., 2018). Support was found for a relationship between both online purchasing behaviour and digital copying behaviour, and phishing targeting. Moreover, a relationship was found between all online activities (except for online purchasing behaviour) and impulsivity (De Kimpe et al., 2018). The present study thus suggests that especially online shoppers and users who often share and use copied files online should be trained to deal with phishing attacks appropriately.

*B. Rashidi et al.* presented XDroid, an Android application and resource risk assessment framework based on the Hidden Markov Model (HMM) (Rashidi, Fung, & Bertino, 2017). In this approach, the authors first map the applications' behaviours into an observation set, and attach timestamps to some observations in the set. The proposed approach has an instrumentation tool that facilitates app behaviour logging in order to generate a high quality dataset for analysis as illustrated in Figure 3.4. It also utilised

comprehensive time-aware Android app behaviour analysis, which is based on the apps' intents and actions, as well as extra features that further improves detection accuracy. Further, a trained hidden Markov model, which can decide whether an app is malicious or not based on its behaviour, was used with a dynamic model that can be updated in real time to integrate users' preferences.



**Figure 3.5 The architecture of the XDroid system. (Rashidi, Fung, & Bertino, 2017)**

The authors showed that the use of temporal behaviour tracking can significantly improve the malware detection accuracy, and that the HMM can generate security alerts when suspicious behaviours are detected. Furthermore, they introduced an online learning model to integrate the input from users and provide adaptive risk assessment. Moreover, they evaluated the model through a set of experiments on the DREBIN benchmark malware dataset. The evaluation results demonstrate that the proposed model can accurately assess the risk levels of malicious applications and provide adaptive risk assessment based on user input.

In this research, K. Molinaro and M. Bolton hypothesized that phishing cues are linearly combinable, meaning a lens model analysis, a type of Judgment analysis (JA), is appropriate for evaluating phishing judgments as shown in Figure 3.4. To test this, they

analysed ten participants who judged whether emails were phishing using the double system lens model or not. Results indicated that the lens model is an appropriate means of analysing phishing judgments, primarily evidenced by the goodness of fits for both the environment model and human judgment models. They also observed varying achievement scores across participants consistent with their varying levels of performance in the judgment task. The results and how future phishing judgment research can utilize JA afforded analysis capabilities were discussed.

Alsharnouby et al. explored the strategies employed by users to identify phishing attacks ( Alsharnouby et. al., 2015). They showed participants a series of websites and asked them to identify a legitimate or fraudulent website. Moreover, the authors evaluated the effectiveness of recent changes that have been made in web browser designs to help users identify fraudulent websites. They also assessed whether users have developed improved detection strategies and mental models of phishing or not. They mentioned using eye tracking data to obtain quantitative information on which visual security indicators draw the most attention from users as they determine the legitimacy of websites. Based on their results, they identify aspects in which web browser security indicators have improved in modern web browsers, identify areas for potential improvement, and make recommendations for future designs.

### **3.6.9 Learning templates**

Authors in (Mouton, Leenen and Venter, 2016) proposed detailed social engineering attack templates that are derived from real world social engineering examples as it is necessary to have a formalised set of social engineering attack scenarios that are fully detailed in every phase and step of the process to be used in developing social engineering awareness material. Although the authors have claimed to create 10 templates which are diverse and unique, they have failed to show how these templates can be used in comparative measures.

*Konradt et al. (Konradt, Schilling, and Werners, 2016)* developed a simulation study based on the work of Fultz and Grossklags (Fultz and Grossklags, 2009). The authors of this paper extended their analysis of cybercrime from an economic view and customised their model, using it as the basis for analysis. The authors claimed that their assessment can give an insight into the perpetrator's behaviour and the effectiveness of counter-measures can be identified. The paper concluded that counter-measures are more

effective when they are targeted at the revenue stream of cybercrime attackers. Although the simulation tool demonstrated how successful the counter-measures can be if revenue streams are targeted, there was no mentioning on how effective they are in targeting digital currency such as Bitcoin.

### **3.6.10 Interactive Training Game**

*Arachchilage et al. (Arachchilage, Love, and Beznosov, 2016)* evaluated the game design framework introduced by Arachchilage and Love (Arachchilage and Love, 2013). The game was designed and developed as an educational tool to teach computer users how to thwart phishing attacks. The results showed a significant improvement in participants' phishing avoidance behaviour and suggested that participants' threat perception, safeguard effectiveness, self-efficacy, perceived severity and perceived susceptibility elements positively impact threat avoidance behaviour, whereas safeguard cost had a negative impact on it. The pilot study was targeted at university students who are likely to have knowledge of phishing threats, hence, the results might be biased. A pilot study could be targeted at a random audience in order to obtain reliable results.

### **3.6.11 Empirical studies and combined approaches**

*I. Qabajeh et al. (Qabajeh, Thabtah, & Chiclana, 2018)* reviewed and critically analysed legal, training, educational and intelligent anti-phishing approaches. More importantly, ways to combat phishing by intelligent and conventional means were highlighted, besides revealing these approaches' differences, similarities and positive and negative aspects from the user and performance prospective. Furthermore, this paper focuses on raising awareness and educating users on phishing from training and legal perspectives. This indeed will equip individuals with knowledge and skills that may prevent phishing on a wider context within the community. The authors reviewed conventional anti-phishing approaches such as law enforcement, user training, and education and then critically analysed their different methods. Then, they looked at predictive ML methods particularly rule-based methods, decision trees, associative classification, SVM, NN, and computational intelligence. They contrast the ways these methods detect phishing activities, their performance and their advantages and disadvantages. Despite the difficulty to find phishing attackers since phishing attacks have a short life span limitation, it is still crucial that law enforcement agencies improve their information sharing work as well as jurisdiction. Moreover, educating novice users using visual cues

can partly improve their abilities to detect phishing; however, many novice users are still not paying high attention to visual cues when browsing the internet which make them vulnerable to phishing attacks. Users need to be exposed to repetitive training about phishing attacks since phishers continuously change the deception tactics. Online phishing communities gather data that allow users to share information about phishing attacks such as blacklisted URLs, which is a useful information centre for users. However, this approach necessitates good awareness about web security indicators besides blacklisted URLs become outdated as updates are not performed in real-time. To sum up, anti-phishing methods based around ML especially AC and rule induction are suitable to combat phishing due to their high detection rate and more importantly the easy to understand outcomes.

*Kang Chiew et al.* presented a detailed survey of the phishing techniques and how they work (Chiew, Yong, & Tan, 2018). The interlinks between the medium of phishing, vectors or channels used and the technical approaches applied in the implementation of the phishing operations were discussed. The first interlink showed the elements in a medium which are exploited and used in a phishing attack. This allows the identification of vectors in a communication medium that is currently being exploited. By having the knowledge of the vectors being exploited, the countermeasure that targets the vector can be developed and coupled with the vector to prevent further exploitation. Further, new and vulnerable vectors can be identified from the medium, and preventive measures can be put in place to prevent exploitation of the new vectors. The second interlink gave the knowledge on how the vectors can be used to launch phishing attacks. In a phishing attack, a combination of technical approaches may be used for a better success rate. This combination of technical approaches may utilise the same vector. By knowing such combination of technical approaches, a countermeasure that targets a specific vector to tackle several technical approaches can be developed. Such countermeasures will be more holistic in nature as opposed to ad-hoc solutions. These interlinks are the characteristics of each phishing technique. Thorough understanding of existing phishing techniques is crucial for the development of holistic anti-phishing techniques to counter these phishing operations, as opposed to an ad-hoc solution that is limited and effective only to a specific case. The knowledge of such interlinks is vital to policy makers in introducing policies and guidelines to put a stop to system or infrastructure exploitation for malicious activities. Furthermore, such a review will be useful for readers from every lifestyle as well, enabling them to take precautionary and preventive actions against phishing attacks.

With such actions by the public and the availability of the anti-phishing system, the effectiveness of a phishing attack can be reduced significantly.

*E. Williams et al.* used data from two organisations that routinely handle sensitive information to address susceptibility to phishing in the workplace and the current limitation (Williams, Hinds, & Joinson, 2018). They used a new approach that enables existing theoretical concepts to be considered and new ones to be identified in relation to applied workplace settings. Further, they divided their work into two studies. In study one, nine spear phishing simulation emails sent to 62,000 employees over a six-week period were rated according to the presence of authority and urgency influence techniques. Results demonstrated that the presence of authority cues increased the likelihood that a user would click a suspicious link contained in an email. In study two, six focus groups were conducted in a second organisation to explore whether additional factors within the work environment impact employee susceptibility to spear phishing. In this study, they undertake a qualitative exploration of wider susceptibility factors related to the individual recipient, and the context that they are in by exploring employee perceptions of susceptibility within the work environment using a focus group methodology in a second organisation.

### **3.7 Existing problems and Discussion**

Qatar's economic growth has made it an attractive target for phishing, particularly spearphishing. There have been a growing number of high-profile cases of these attacks in Qatar, as outlined in the first section. The worsening situation demands a greater investment in developing defences against these attacks. The first line of defence against phishing is the user, so awareness programs which encourage the use of, for instance, two-factor authentication are crucial. Phishers intend to mine Qatar's citizens and business employees for crucial data, sensitive information and resources. A high proportion of these attacks target the financial sector, and the use of insecure WiFi as well as a refusal to acknowledge the risk of Smishing. These attacks are increasing worldwide, so the case of Qatar could provide a good example for how to deal with phishing.

As the provided glossary shows, there are many different established methods of detecting and protecting against phishing, including blacklists, heuristics, checking for visual similarity, and data mining. The chapter covers different case studies where researchers explored the efficacy of various detection methods. These studies encounter

common problems. The Structural DOM analysis method proposed by Pan and Ding (2006), for instance, struggles to cope with images, and allows attackers to sidestep its defences by reducing the number of structural features extracted. ML-based techniques used by the authors targeted various web features to different levels of success. They encountered different issues; some, like the CIA (Lin et al., 2015) worked well in the study but raised questions of practicality because of the inevitable overload incurred through implementing the system on a sustained basis. Other studies failed to satisfactorily show their methods or prove that their sample is generalizable. Whitelists are highly effective, but they need to be updated constantly in order to keep working. Overall, most of them faced the same common issue, which is failure to work with the Arabic language, which is crucial for our case study. Li et al. (2016) extract 12 different topological features from suspect websites to yield over 96% accuracy in detection, but like many of the other cases (for instance Xiang et al.'s CANTINA+ (2011)) outlined in this section, feature extraction is commonly designed around the English language and therefore cannot cope with Arabic phishing websites, which renders it useless for our case of Qatar where many sensitive websites will be in Arabic. This also points to a wider issue, which is that contents analysis often fails for non-English alphabets, and will need to be specifically designed for each language.

As section 3.8 shows, there have been instances of systems being developed which do not rely on lexical information; the URL Token-System developed by Tan and Chiew (2017) achieved high accuracy without relying on analysis of English content. We took heed of this in our developed system, which analyses emails for signs of typo-squatting, a phishing technique which uses URLs and email addresses, is suitable for use in Qatar because it uses elements of spearphishing emails which use the English alphabet, so it targets spearphishing more successfully than DNS-checking based approaches (which are commonly used by web hosts) without failing because of Arabic language emails. Unlike whitelisting methods, the proposed method does not need to be manually updated because it algorithmically checks for the kind of distortion typical of spearphishing.

The other problem with many of these approaches is that none of them achieves a 100% success rate, and this can be expected with most anti-phishing systems because, as the studies showed, machine learning has its limits and there are always vulnerabilities for phishers to learn about and take advantage of. As our data showed, whilst our technical solution offers a great improvement on typical defences offered by web-hosts,

recognising and blocking far more spearphishing emails, it is not 100% accurate. Also, many of the better systems require high processing capabilities which makes them impractical. Machine Learning can work effectively, but in many ways, it is just as valuable to invest in human learning; training users to more effectively notice and adapt to phishing and especially spearphishing is far better than simply employing technical approaches on their own. Many of the researchers (e.g. Zhang and Gupta, 2018) found that it is greatly beneficial for users to have higher awareness of internet phishing, its risks and tell-tale signs.

It was suggested that different learning methods might be employed, for instance Arachchilage, Love, and Beznosov's educational training game (2016). This, like other training programs, relies on reproducing phishing/spearphishing scenarios for internet users, creating simulated interactions where they can practise recognising fraud and threat avoidance behaviour. In order to continuously improve the design and accuracy of these scenarios and simulations, it is important to carry out data collection on user behaviour. In fact, there have been many studies which look into various aspects of user behaviour – Curtis et al. (2018) considered the role of personality traits, while De Kimpe et al. (2018) and others monitored online user behaviour to develop theories about vulnerability to phishing and the conditions that make it more likely. There have been all sorts of suggestions in the literature, including the use of eye-tracking. Clearly, there is much to be gained from collecting information about users. As suggested, phishing techniques are constantly advancing, and therefore it is necessary for the literature to keep pace. If awareness training is conducted, it should be updated frequently over the years as techniques advance, and knowledge of user habits grows.

### **3.8 Conclusion**

By taking all of these different factors into account, and learning from the challenges faced in the literature, we have chosen to opt for a double approach, combining a technical solution specifically formulated to recognise spearphishing emails with a website for user training and data collection. Our technical solution, the ECSPAD, solves one common difficulty faced by many of the ML-based systems reviewed above: their incapacity to deal with Arabic language spearphishing emails. Also, many of the systems are designed for more generalised phishing emails, whilst spearphishing requires a specialised approach, because they are specially targeted attacks. What is more, no system is 100%

accurate, and the combined problems of Qatar's economic prosperity and 'soft touch' users making it an attractive target for spearphishers means that we must tackle spearphishing specifically with our technical approach and also work on raising awareness. The ECSPAD works with the Arabic language because it analyses URL features which are in the English alphabet, without relying on content analysis. We have also developed our educational website to work alongside this, which provides information to the user, and quizzes them on their knowledge and learning. This data collection is used both to advance our knowledge of user awareness, and as a teaching implement in itself; the user can see their answers next to the real answers, allowing them to learn from their mistakes. Combining this training with the advancement of technical phishing detection methods will go a long way towards making up for the challenges and failures of ML programs, whitelisting, etc. because it creates a first line of human defence. An educated human is able to take into account many different kinds of inputs, and adapt their knowledge on a case-by-case basis, in any alphabet or language that they use, and does not need to be programmed specifically to do any of these things. As such, it is insufficient to merely focus on technical solutions. The greatest difficulty with technical solutions is that they are never perfect and must be adapted manually, whereas human learning is self-reinforcing, and makes up for failures of technology.

# CHAPTER 4

## PROPOSED METHOD

### 4.1 Introduction

This chapter discusses how the research is implemented, in other words, how the proposed method for targeted attacks prevents any suspicious emails, and how the ultimate algorithms are produced. The process of developing the proposed method and the improved algorithms are discussed in this chapter. The chapter starts with an introduction, then describes the phases of methodology design and implementation for Enterprise Credential Spearphishing Targeted Attack Detection (ECSPTAD) attacks. The research is described step-by-step, from scratch, until the final results are achieved.

### 4.2 Spearphishing

Spearphishing differs from attacks which use software and protocol weaknesses and technical vulnerabilities to infiltrate machines. The engineering that goes into a spearphishing attack can be described as social rather than technical. Spearphishing entails sending specially designed emails which are bespoke to the victim, intended to hoodwink victims into carrying out an action which benefits the predator. Due to the nature of the attack, very little technical knowledge is necessary on the part of the attacker. Unlike other types of phishing, spearphishing does not prey on the functional vulnerabilities of machines and software but rather relies on the gullibility of users, which means attacks are difficult to deflect through automated technical defence systems.

The relatively high success rate of spearphishing results from the fact that emails are easy to spoof, and the considerable time attackers invest into creating emails designed specifically for a particular victim. Because of this, as yet, effective measures or tools for identifying or defending against spearphishing do not exist. These factors make spearphishing a highly popular and attractive route for those who wish to obtain financial or data-related resources from high-level targets who represent much value to the attacker (Trend Micro, 2012).

*Credential spearphishing*, the most common form of spearphishing, involves sending spoofed emails, which ask the victim to follow links and enter their credentials into a

malicious website hosted by the attacker. This spoofed site is designed to appear indistinguishable from a recognisable, trusted site. Attachment-driven spearphishing struggles to succeed against many email providers' malware-filtration systems, which proactively check emails for malicious software. The safety of LJMU's email server, for example, is maintained by a full-time team of security staff, and regularly updates its security system and hardware.

This forces phishers to carry out expensive zero-day exploits in order to succeed against these meticulous defence systems. Conversely, the barriers set up against credential spearphishing are very low; phishers need only to cleverly construct a bespoke email and host a spoof website in order to hoodwink their victims.

### **4.3 Attack Taxonomy**

Spearphishing differs from regular phishing in specificity and selectiveness. Phishing emails are constructed for a mass audience, intended to be sent to many people in the hopes of profiting from anyone who happens to be tricked by it (Sheng et al., 2010, Tankard, 2011) whilst spearphishing emails are made bespoke to victims with particularly valuable information, capabilities or access to resources. The attacks are designed with a very specific aim in mind, which makes it possible to tailor every detail in such a way as to increase convincingness.

In order to hoodwink targets into performing actions on behalf of the phisher, spearphishing emails must instil trust by a demonstration of authority or legitimacy. Usually, this is attained by impersonating trusted entities who are already known to the target. Then, the phisher impersonating the authority figure will ask the target to carry out an action which benefits the phisher, such as transferring funds or breaching sensitive data.

The technique of impersonation in spearphishing aids in establishing the victim's trust, and also allows the attacker to evade identification and punitive consequences. Impersonation in spearphishing takes several forms, which can be used in combination:

- Spoofing the name and email address of an individual who is already trusted and known to the target in the *From* field of the spearphishing email. By closely or identically imitating the *From* field that the recipient is accustomed to seeing

when receiving an email from this trusted party, attackers are able to convince victims that they are interacting with a trusted entity (Bursztein and Eranti, 2013).

- Name spoofing without email spoofing is a means to evade certain detection systems; DKIM and DMARC scan the address line to carry out filtration, and this method does not arouse suspicion to these systems. By only spoofing the sender's name, and not the *From* email address field, phishers can circumvent safety protocols. As long as they do not notice the incorrect address, their victims believe they are interacting with a known, trusted individual.
- A previously unseen attacker selects a name and email address to put in the *From* field of the spearphishing email, where neither the name nor the email address actually match a true user's name or email address (though they might be perceived as trustworthy or similar to a real user's identity). For instance, the attacker might choose to spoof the name LJMU IT Help and the email address <helpdesk@ljmuY.com>.

Constructing names and email addresses which do not directly spoof that of a trusted individual, but rather contain elements or similarities which suggest institutional authenticity. For example, phishers might select the email address <techsupport@ebay.com> along with the name eBay Technical Support.

- Lateral attack describes the slightly more sophisticated approach of breaching one user's email and sending spearphishing emails directly from this first victim to somebody they are known and trusted by.

## 4.4 Threat Model

In this work, we specifically focus on an “*Enterprise Credential Spearphishing*” threat model, where the attacker tries to fool a targeted enterprise's victim into revealing their credentials.

In the tests that we did on the Liverpool John Moores University email system, we found that the attacker can bypass detection by changing one character of a legitimate domain name. In this test, we register the domain “*ljmuac.uk*”. The only difference between our registered domain name and the legitimate Liverpool John Moores University domain name “*ljmu.ac.uk*” is that ours has one less full stop or dot. As shown in Figure 4.1,

we sent an email from dontreply@ljmu.ac.uk<dontreply@ljmuac.uk>. In our threat model, the real email is xxx@ljmu.ac.uk, where “xxx” can be any name such as dontreply, ITHelpDesk or even a person’s name.

```
from: dontreply@ljmu.ac.uk <dontreply@ljmuac.uk>  
to: h.kolivand@ljmu.ac.uk  
date: Oct 3, 2018, 12:35 PM  
subject: Status Report  
mailed-by: ljmuac.uk
```

### **Figure 4.1 Registered domain name**

The adversary can send emails to the victim, and convince the recipient to click on URLs embedded in the adversary’s email (Figure 4.2). To impersonate a trusted entity, the attacker may set any of the email header fields to arbitrary values.

This thesis is focused on attacks which entail masquerading as a trusted entity, with the payload being a link to a credential harvesting phishing page (Figure 4.4).

Figure 4.2 shows an email we sent to LJMU students, informing them of strange internet traffic originating from their computers, and telling them that there appears to have been a small outbreak of viruses that may have spread across the network. We reassure the user that we are attempting to remove these infections, however the user must change their password immediately. Then, the user is asked to click on a link. The link redirects the user to a cloned website, (Figure 4.4) where we present a cloned version of a legitimate website (Figure 4.3). To gain more trust, we placed “https://myaccount.ljmu.ac.uk/” over the hyperlink text, which sends users to our cloned website “https://myaccount.ljmuac.uk/”.

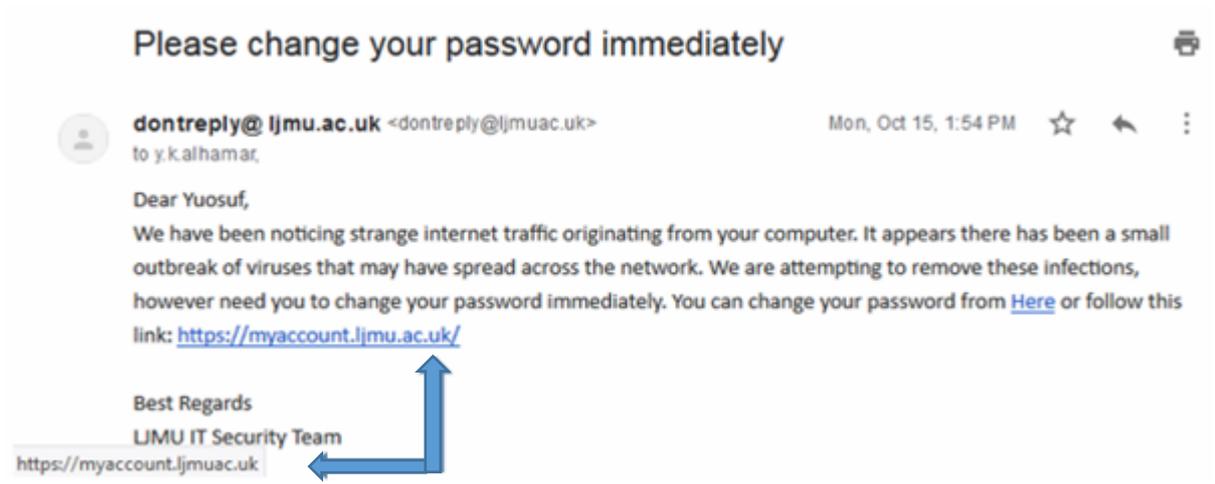


Figure 4.2 Sent email to user

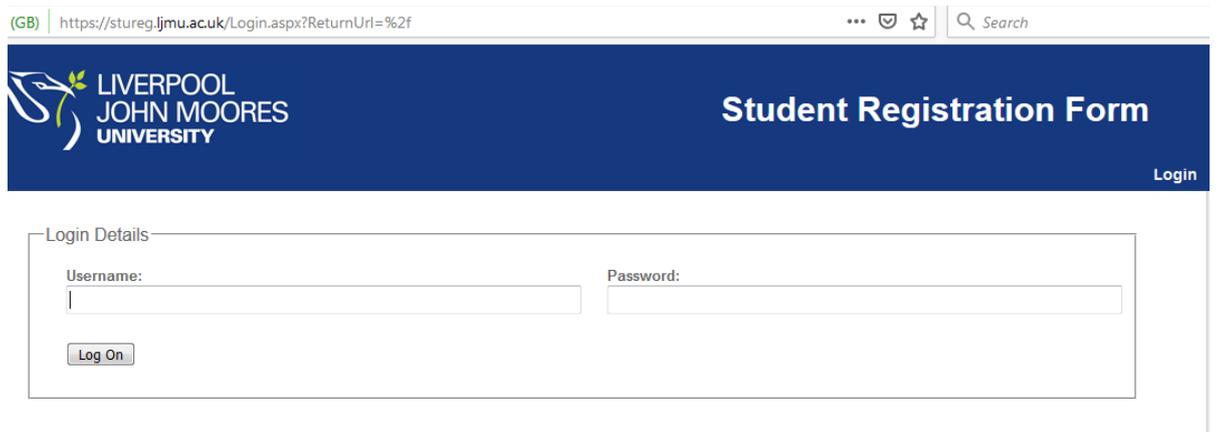


Figure 4.3 Legitimate Website

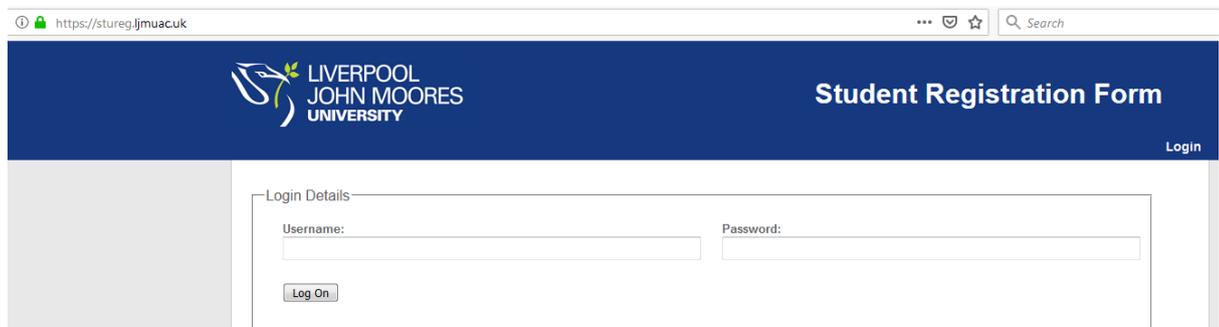


Figure 4.4 Clone website

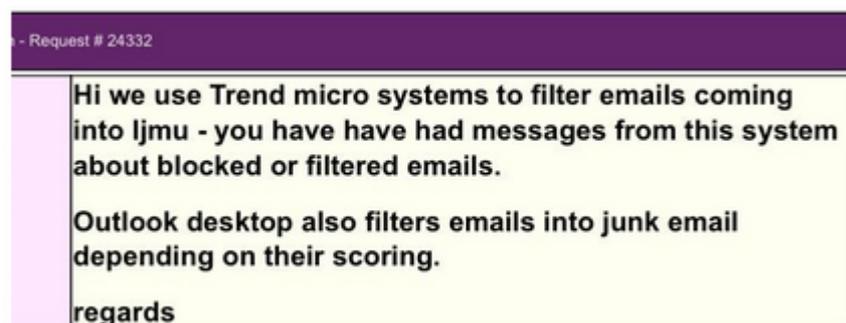
We asked 50 different people (40 students and 10 staff) to read the email and click on the link. Once they read it and opened the link, we asked if they noticed anything wrong with the email and the page. Only 2 people (1 student and 1 staff member) noticed that firstly, the sender of the email is not Liverpool John Moores University, and none of them spotted that the web page they browsed is a cloned version of a legitimate page (Figure 4.3).

As shown in Figure 4.5 we were able to obtain the users' usernames and passwords. Once the user clicks on the login button, they are redirected to the legitimate address, which in this case is "http://stureg.ljmu.ac.uk", and they think that they might have inputted their username and password incorrectly without even realising that their username and password has been stolen. As such, this spearphishing attack was successful in stealing the victim's login credentials.

```
[*] WE GOT A HIT! Printing the output:
PARAM: __LASTFOCUS=
PARAM: detail_ToolkitScriptManager1_HiddenField=
PARAM: __EVENTTARGET=
PARAM: __EVENTARGUMENT=
PARAM: __VIEWSTATE=/wEPDwUKMTYwMTg1OTk4N2QYAQUeX19Db250cm9sc1JlcXVpcm
PARAM: __VIEWSTATEGENERATOR=C2EE9ABB
PARAM: __EVENTVALIDATION=/wEdAAUSFPG18W2NaR9Tmh3oBF8Eyh1HDN25acBMNcp5
POSSIBLE USERNAME FIELD FOUND: ct100$detail$tbUsername=test
POSSIBLE PASSWORD FIELD FOUND: ct100$detail$tbPassword=test
PARAM: ct100$detail$btnSubmit=Log+On
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.
```

**Figure 4.5 Sniffed username and password**

During our test phase, we successfully bypassed the email protection that the university put in place to protect users. A dialogue was established with the university IT department, to find out what types of protection they employ and how they tackle phishing attacks (Figure 4.6).



## Figure 4.6 LJMU IT department response

Unfortunately, they had no idea what we were talking about. There is a difference between spam and phishing emails. Spam emails can be phishing emails, but spearphishing email cannot be spam, and will bypass the spam scoring system if the attacker crafts the email carefully. Therefore, the “Trend Micro Email Protection” system is impractical in guarding against spearphishing attacks on Liverpool John Moores University staff or students, as demonstrated by the fact that we successfully launched a spearphishing attack and bypassed the detection system.

During our literature review phase, we could not find any solution that tackles “*Enterprise Credential Spearphishing*”, where attackers carefully plan attacks by following these

### Steps:

**Step 1:** Identifying the victim: At the beginning of each phishing attack, an attacker needs to find a target. Since spearphishing is a targeted attack, the attacker must specifically identify the victim.

**Step 2:** Gathering information about victim: Once the attacker identifies the victim, they need to gather intel about the victim using search engines or social networks such as name, location, place of work, close friends, favourite brands, and favourite things to do.

**Step 3:** Choosing techniques: Based on the information gathered from the previous step, now the attacker chooses their attack techniques. In our threat model, the attacker has chosen spearphishing, typosquatting and credential harvesting.

**Step 4:** Preparing tools: Based on the techniques selected in step 3, the attacker now prepares the tools that are suited to the planned attack.

**Step 5:** Register domain(s): In this step, the attacker registers a domain name designed to establish the victim’s trust. For example, for a victim working in a company with the web address [www.abcd ef.co.uk](http://www.abcd ef.co.uk), the attacker will register a domain name similar to that with 1 or 2 characters different, e.g. [www.abcedcf.co.uk](http://www.abcedcf.co.uk).

**Step 6:** Craft email template: To gain more trust, the attacker must construct an email template carefully. Once a victim cannot identify anything suspicious in a spoofed email, 99% of their trust is established.

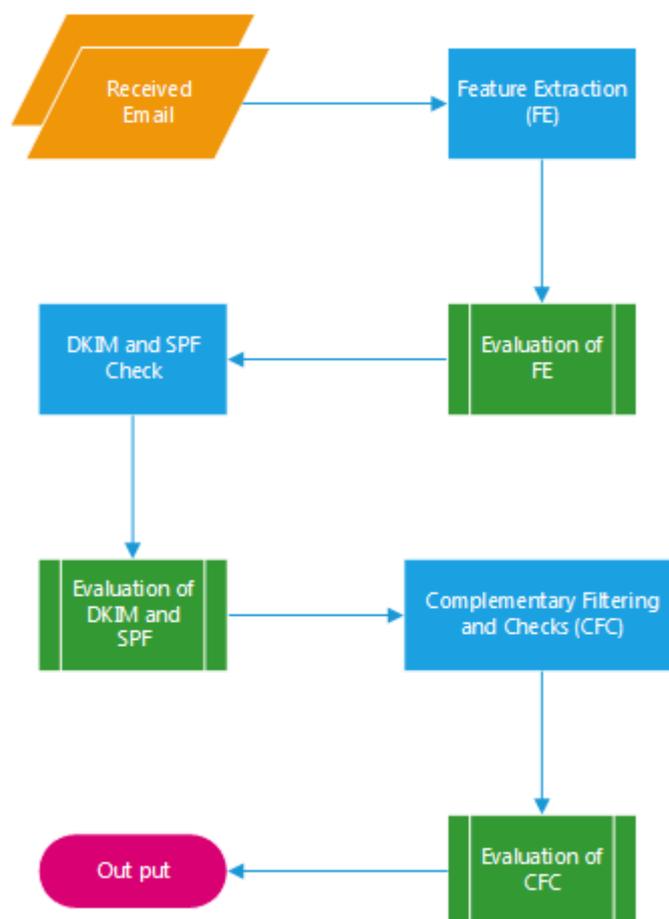
**Step 7:** Clone targeted website: Because of the nature of the techniques chosen, the attacker needs to clone the targeted website that he wants to send to the victim in order to extract their credentials.

**Step 8:** Send email: in this step, attacker sends the email.

**Step 9:** Credentials Obtained: Once victim opens the email and follows the link to the login/password page, the cloned website saves any user credentials entered.

## 4.5 Proposed Solution

To tackle this type of attack, we propose a solution that can detect an “Enterprise Credential Spearphishing” attack, where the attacker uses a similar domain name to gain the victim’s trust and to trap the victim into the attack. The proposed solution, at a high level, has four stages as illustrated in Figure 4.7.



**Figure 4.7 Overview of proposed Protection system**

As shown in Figure 4.7, the first process is features extraction, then the extracted features

are processed to calculate scores and differences. These two processes are the most important parts of our proposed solution. Once the scores and differences are calculated, the result is compared with the database and threshold values. If there is a match, an alert is created and the email is quarantined for further investigation.

## 4.6 Feature Extraction

In this process, the proposed system extracts the following features from the received email domain:

**Count number of characters ( $C_{noc}$ )** = this feature is used to extract and count the number of domain name characters.

**Count number of Unique characters ( $C_{nouc}$ )** = this feature is used to extract and count the number of unique domain name characters.

**Count number of dots ( $C_{nod}$ )** = this feature extracts and counts the number of dots in the domain name.

**Count number of numeric values ( $C_{nonv}$ )** = this feature checks to see if there are any numeric values in the received email domain and counts them.

**Count number of Hyphens ( $C_{noh}$ )** = this feature counts the number of hyphens in the domain if there are any.

**Extract Domain Extension after ( $E_{de}$ )** = this feature extracts the domain extension suffix from the received email domain.

**Count number of characters before first dot ( $C_{nocbfd}$ )** = this feature counts the number of characters before the first dot. For example, the number of characters of this domain “ljmu.ac.uk” before the first dot is not equal to “ljmuac.uk”.

**Incoming mail IP address ( $IN_{ip}$ )** = this feature retrieves the sender’s IP address.

**Valid IP address ( $V_{ip}$ )** = this feature is a list of valid email server IP addresses for the organisation.

**Similar Characters Place ( $SCP$ )** = this feature extracts and compares character placement

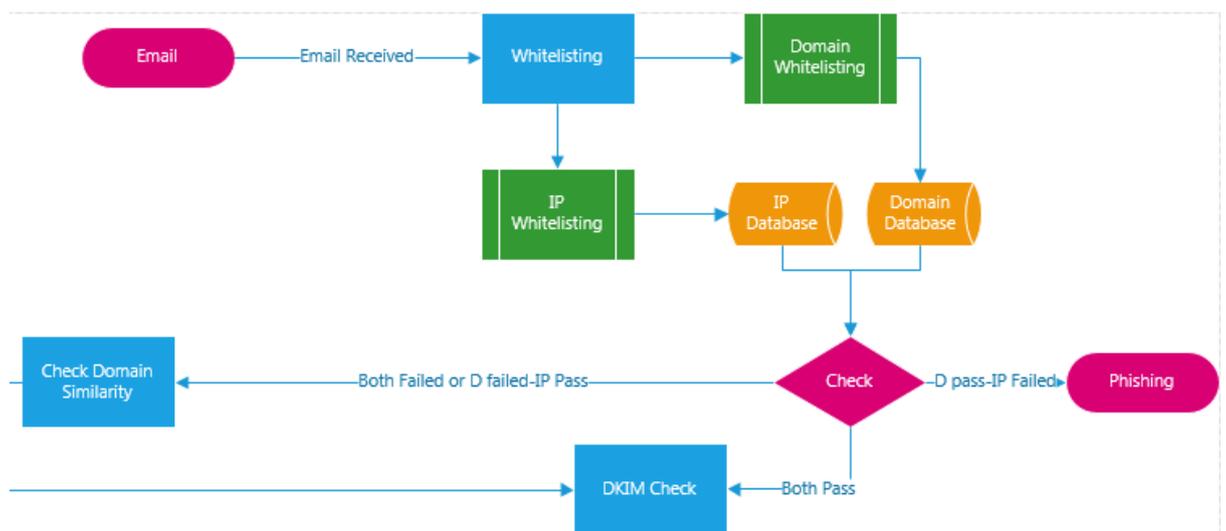
between the received email and valid domains. For example, “ljum” and “ljmu” only have two SCP matches, which are “l and j”

Similar Domain Name ( $S_{domain}$ ) = this feature compares the characters before domain extension of valid domain names (e.g. ljmu.ac.uk) with incoming domain names (e.g. ljmuac.uk).

Number of Common Characters ( $NCC$ ) = this feature extracts and compares the number of characters common between the received domain name and valid domain database. For example, “ljbcff.uk” and “ljmu.ac.uk” only have four common characters.

As shown in Figure 4.8, the proposed solution starts to work once the email is received by the system. At the first stage, the email domain is whitelisted through the first process, which is the “whitelisting” process.

This process has two sub-processes to check whether the incoming email can be whitelisted or not. The first sub-process is to check the domain against a valid domain database. This process checks if the incoming email domain name (i.e. ljmu.ac.uk) exists in the domain database. Then, the next sub-process checks the sender IP address (i.e. 1.1.1.1) against the IP database to see if the sender IP address exists in that database.



**Figure 4.8 Overview of proposed Protection system**

Afterwards, the results are compared to make a decision about the email. In the “Check” process, the system marks an email as phishing if the domain name is the same (result pass), but the IP address is different (result fail). This means an attacker is trying to spoof a valid domain name to send the phishing attack, but the IP address is not similar to the valid IPs.

If both checks fail, then the email is forwarded to another process, which is “Check Domain Similarity”. This is because neither the domain nor the IP is valid.

If the domain check result is fail but the IP address is valid, the email is still sent to the “Check Domain Similarity” process again for further examination.

If both the domain and IP pass, the proposed solution sends the email to another process named “DKIM and SPF” checker

We propose an algorithm for whitelisting the incoming email domain name. The proposed algorithm has two parts, “Function Domain Whitelisting” and “Function IP address Whitelisting”.

Algorithm 1: Whitelisting

```
Function_Domain_Whitelisting(){
```

```
def1: IF  $IN_{domain} = V_{domain}$  Then:
```

```
Pass
```

```
Else:
```

```
False
```

```
}
```

```
Function_IP address_Whitelisting(){
```

```
def2: IF  $IN_{IP} = V_{IP}$  Then:
```

```
Pass
```

```
Else:
```

*False*

}

*Function\_Domain\_Whitelisting()*: This function whitelists the domain name using the valid domain database, where  $IN_{domain}$  is the incoming email domain name and  $V_{domain}$  is a whitelisted domain in the valid domain database.

*Function\_IP address\_Whitelisting()*: This function whitelists the sender IP address (Figure 4.9) using the valid IP address database, where  $IN_{IP}$  is the sender IP address and  $V_{IP}$  is the whitelisted IP address in the valid IP address database.

```
Received: from [127.0.1.1] ([167.99.81.250])  
by smtp.gmail.com with ESMTPSA id 125-v6sm9501157wmr.2:  
for <h.kolivand@ljmu.ac.uk>  
(version=TLS1_2 cipher=ECDHE-RSA-AES128-GCM-SHA256 bit:  
Mon 03 Dec 2018 06:27:44 -0800 (PST))
```

**Figure 4.9 Send IP address**

Once an email is received, the process starts to work by checking and validating two factors. The first factor is the domain name and the second one is the sender IP address. If the result of *def1* and *def2* is pass, then the email is valid and moves to the next layer of processing, which is the Domain Keys Identified Mail (DKIM) and Sender Policy Framework (SPF). This is because if in *def1* the  $IN_{domain} = V_{domain}$ , then it means the sender domain is same as the domain in the whitelisted database. To avoid address spoofing, we check the sender IP address against the valid IP address. If in *def2* the  $IN_{IP} = V_{IP}$  then it means the email was sent from one of our trusted domains. In this case, we send the email for future checks to the DKIM and SPF process.

If *def1* pass and *def2* failed, then this means that the email is spoofed, and therefore we categorise it as phishing. This is because the pass in *def1* shows that both emails have the same domain name, but the fail in *def2* shows that the sender IP address is not valid, i.e. is not listed in the database. Therefore, it means that someone is trying to spoof the sender domain name, and the system marks the email as phishing.

If both def 1 and def 2 fail, then the email is sent to the next function, “Check Domain Similarity”.

#### Algorithm 2: Check Domain Similarity

Function Similar Character Place (SCP) () {

*def1: Find SCP*

*Read From Vdomain*

*Input INdomain*

*String SP1;*

*String SP2;*

*Counter Index = 0;*

*For I = 1 to Vdomain.length*

*For J = 1 to INdomain.length*

*IF Vdomain I = INdomain J*

*SP1.append I;*

}

Function Number of Common Character (NCC) () {

*def2: Find NCC*

*s1 = set Read From Database Vdomain;*

*s2 = set Input INdomain;*

*commonchar = s1 & s2;*

*remove\_dots = s.strip '.' for s in s2*

*IF len commonchar < 1:*

*return list set s1.intersection remove\_dots*

*else:*

*return 0*

}

Function Calculate SCP and NCC () {

*def1: Calcualte TVSCP:*

$12 \text{ len } S \text{ domain}$

*round(TVSCP)*

*def2: Calcualte SCP*

*IF RSCP  $\geq$  TVSCP Then:*

*Return 1*

*Else:*

*Return 0*

*def3: Calcualte TVNCC:*

$13 \text{ Cnouv } V \text{ domain}$

*round(TVSCP)*

*def4: Calcualte NCC*

*IF RNCC  $\geq$  TVNCC Then:*

*Return 1*

*Else:*

*Return 0*

}

Function Check Domain Similarity (CDS) () {

*def* 1:

*SCP and NCC Comapre:*

*IF SCP = 1 and NCC = 1 Then:*

*Return 1*

*IF SCP = 0 and NCC = 1 Then:*

*Return 2*

*IF SCP = 0 and NCC = 0 Then:*

*Return 3*

*IF SCP = 1 and NCC = 0 Then:*

*Return 4*

*def* 2:

*IF INSdomain = VSdomain Then:*

*Return 1*

*Else:*

*Return 0*

*IF INTLD = VTLD Then:*

*Return 1*

*Else:*

*Return 0*

*def* SDN:

*IF Vdomain = INdomain[: len Vdomain]:*

*Return 1*

*Else:*

*Return 0*

```
def Comapre1:
IF Sdomain = 1 and TLDdomain = 0 Then:
Email is Phishing
```

```
def Comapre2:
IF Sdomain = 0 and TLDdomain = 0 Then:
passing to DKIM and SPF
```

```
def Comapre3:
IF Sdomain = 0 and TLDdomain = 1 Then:
passing to DKIM and SPF
```

```
def scp_ncc_comapre 1:
IF SCP = 1 and NCC = 1 Then:
Classified as Phishing
```

```
def scp_ncc_comapre 2:
IF SCP = 0 and NCC = 1 Then:
Pass to DKIM and SPF
```

```
def scp_ncc_comapre 3:
IF SCP = 0 and NCC = 0 Then:
Pass to DKIM and SPF
```

```
def scp_ncc_comapre 4:
IF SCP = 1 and NCC = 0 Then:
Suspected as Phishing
```

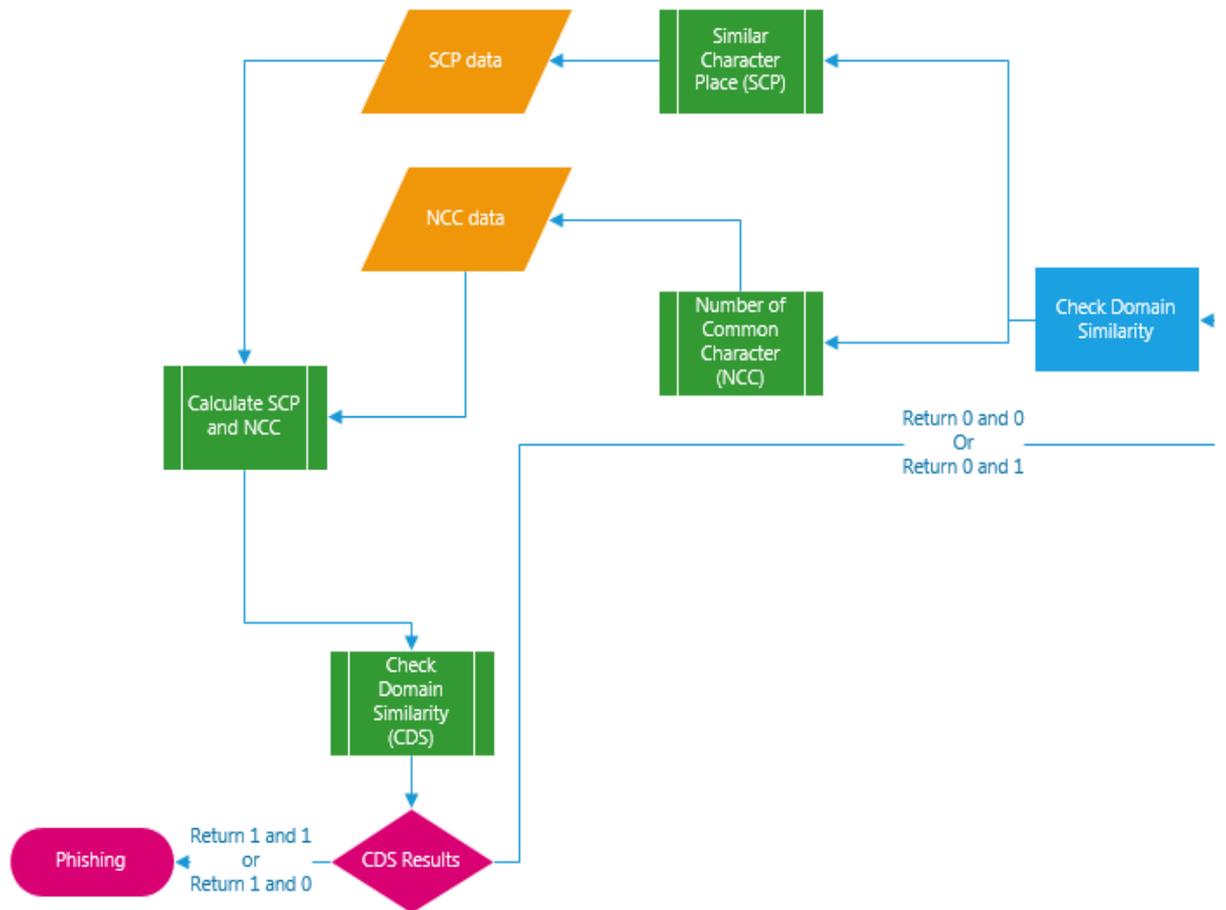
```
}
```

This process works by evaluating the incoming email domain name. As shown in Figure 4.10 this process has two sub-processes, Similar Character Place (SCP) and Number of Common Characters (NCC).

Similar Character Place (SCP) looks for common character placements between the incoming email domain name and valid domain addresses. In theory, this will help to prevent attack techniques such as “Typo squatting”. In “Typo squatting”, attackers use a similar domain to a legitimate domain. For example, an attacker might use “ljmuac.uk” as the email domain name to send an email to the victim, which is close to “ljmu.ac.uk”.

To achieve this, we proposed an algorithm named “Similar Character Place (SCP)” to find the similar character placements in both domains. If the “SCP” is more than the threshold value, it is given a “1” score, if it is less the score is “0”. The threshold value is half of the valid domain name.

As an extra security precaution, we proposed another algorithm named “Number of Common Character”. This sub-process counts the number of common characters in both domains, minimising the risk of the attacker evading detection. The idea behind this is that normally, attackers use words similar to a target address. For example, an attacker might send an email from “insatgarm.com”, trying to pretend that the email is from “instagram.com”. This domain has eight common characters with the domain “Instagram.com”. As with SCP, if the threshold is met, then the system gives a score of “1”, and if it is not met then the score is “0”. The threshold value for this process is one third of the number of characters in the valid domain address.



**Figure 4.10 Check Domain Similarity Process**

Once both Similar Character Place and Number of Common Characters are calculated, the result of this is forwarded to another sub-process called “Check Domain Similarity”. If the result of both is “1”, then the incoming email is classified as “Phishing”. This is because these figures show that the proposed sub-processes, Similar Character Place and Number of Common Character, detected a high chance of similarity to the valid domain.

If the Similar Character Place score is “1” and the Number of Common Character score is “0”, again the proposed system has detected a high chance of the incoming email having a Similar Character Place to the valid email.

If one of the SCP or both of them return “0”, then the domain will forward the email to DKIM for further examination of the domain.

*Function Similar Character Place (SCP) ():* This proposed algorithm, extracts and finds

similar character placement between an incoming email domain name and valid domain addresses in the whitelisted database. First, the proposed system reads the valid domain ( $V_{domain}$ ) from the database and retrieves the domain name from the incoming email. Then, it creates a dictionary to store the values. Then, it compares both values to find similar character placements.

For a better understating of how the process works, we created a simple output for comparing two processes. The result, shown in Figure 4.11, where “ljmu.ac.uk” is compared with “ljmuac.uk”, shows that there are only four similar character placements in both domains ['l', 'j', 'm', 'u'].

```
l -- l
j -- j
m -- m
u -- u
.   a
a   c
c   .
i, u ,j
uend( k
['l', 'j', 'm', 'u']
```

**Figure 4.11 SCP output**

*Function Number of Common Character (NCC) ()*: Once this process is finished, the next step is to find the common characters between the incoming email domain and valid domains in the whitelisted database. To be able to do this, the following algorithm has been proposed. “Set” is an unordered collection data type that is iterable, mutable, and has no duplicate elements. Python’s set class represents the mathematical notion of a set. The major advantage of using a set, as opposed to a list, is that it has a highly optimized method for checking whether a specific element is contained in the set.

To get more accurate results, the algorithm removes the dots from the result; therefore, the number of dots in this process doesn’t count. The proposed algorithm compared the following domains:

- Ljmu.ac.uk
- Liv.ac.uk

The system found five common characters between those two domains ['a', 'c', 'u', 'k', 'l'].

```
['a', 'c', 'u', 'k', 'l']  
Number of common character: 5
```

**Figure 4.12 NCC output**

*Function Calculate SCP and NCC()*: Once this feature has been extracted successfully, the next algorithm compares the result with a threshold value ( $TV_{SCP}$ ). This is half of the length of the valid domain name ( $S_{domain}$ ). For example, if  $V_{SDN} = ljmu.ac.uk$ , then  $TV_{SCP}$  is half of the length of “ljmu”, which in this case is “2”.  $R_{SCP}$  has already been extracted by *def1 : Find SCP*, which is the number of similar character places. So, if  $R_{SCP}$  is greater than or equal to  $TV_{SCP}$ , then it returns to “1” (which means it gives a “1” score), otherwise returning to “0” (which means a score of “0”). If  $TV_{SCP}$  is above “x.5”, where “x” is the number, then the proposed system rounds it to the next whole number, and if it is below “x.49” rounds to “x”.

Once the score for similar character places is calculated, then it is time to calculate the number of common characters, which is  $NCC$ , and compare it to the threshold value ( $TV_{NCC}$ ) where  $TV_{NCC}$  is normally  $1/3$  of Number  $C_{nouc}$  ( $V_{domain}$ ) Character (NCC) Characters. For example,  $V_{domain} = ljmu.ac.uk$  and  $C_{nouc}$  is 7, therefore  $TV_{NCC}$  will be “2”. If  $TV_{NCC}$  is above “x.5” where the “x” is the number, then the proposed system will round it to the next whole number, and if it is below “x.49” will round to “x”.

*Function Check Domain Similarity (CDS) ()*: The first algorithm has been proposed to compare the results of Similar Character Place (SCP) and Number of Common Character (NCC), which has been calculated by the previous Function, “Function Calculate SCP and NCC”. The algorithms “SCP” and “NCC Compare” check and compare the results from previous functions which SCP and NCC already extracted from the incoming email domain address.

### **Case 1:**

Valid Domain = ljmua.ac.uk | Incoming Domain: ljmua.ac.uk

SCP = 4 (L,J,M,U) | NCC = 7 (L,J,M,U,A,C,K)

TValue of SCP = 2 (half of the Valid Domain Name) | TValue of NCC = 2 (1/3 of Number of Valid domain Character)

SCP Algorithm return value = 1 | NCC Algorithm return value = 1

Result = the proposed system and algorithms marks this email as phishing (refer to chapter 6 for the results).

### **Case 2:**

Valid Domain = ljmua.ac.uk | Incoming Domain: lju.uk

SCP = 4 (L,U) | NCC = 7 (L,U,J,K)

TValue of SCP = 2 (half of the Valid Domain Name) | TValue of NCC = 2 (1/3 of Number of Valid domain Character)

SCP Algorithm return value = 0 | NCC Algorithm return value = 0

Result = the proposed system and algorithms will send this email to next process for further examinations as the return is 0 (refer to chapter 6 for the results).

### **Case 3:**

Valid Domain = alpine.qa | Incoming Domain: alpna.qa

SCP = 7 (a, l, p, a, ., q, a) | NCC = 6 (a, i, l, n, q, p)

TValue of SCP = round (3.5) = 4 (half of the Valid Domain Name) | TValue of NCC = 2 (1/3 of Number of Valid domain Character)

SCP Algorithm return value = 1 | NCC Algorithm return value = 1

Result = the proposed system and algorithms categorised the incoming email as phishing (refer to chapter 6 for the results).

**Case 4:**

Valid Domain = motc.gov.qa | Incoming Domain: motcogv.qa

SCP = 5 (M, O, T, C, G) | NCC = 8 (A, C, G, M, O, Q, T, V)

TValue of SCP = round (2.5) = 3 (half of the Valid Domain Name) | TValue of NCC = round (2.6) = 3 (1/3 of Number of Valid domain Character)

SCP Algorithm return value = 1 | NCC Algorithm return value = 1

Result = the proposed system and algorithms categorised the incoming email as phishing (refer to chapter 6 for the results).

The second proposed algorithm is if  $INS_{domain} = VS_{domain}$ , then return score “1”. This means if the incoming email domain name is similar to the valid domain name in the database, and if the incoming email domain tld ( $IN_{TLD}$ ) is the same as the valid domain tld ( $V_{TLD}$ ) then return “1”.

Now, the algorithm compares all calculated scores and decides whether the incoming email is phishing or not. The  $S_{domain}$  is the following:

The function *def SDN* checks similar domain names. The proposed function checks to see if  $V_{domain}$ , which is the valid domain in the database (i.e. “ljmu”) and  $IN_{domain}$ , the incoming domain name (i.e. “ljmuac”) are the same or not. That is to say, if  $V_{domain}$  is equal to  $IN_{domain}$ , but only the first number of characters, which is the length of  $V_{domain}$ .

Similar Domain Compare 1: If  $S_{domain}$  (which we already extracted with the previous algorithm) score is “1” and the  $TLD_{domain}$  score is “0”, then the email is “Phishing”. This is because it means the domain name is same. Therefore, the email is classified as “Phishing”.

Similar Domain Compare 2: If  $S_{domain}$  (which we already extracted with the previous algorithm) score were “0”, and the  $TLD_{domain}$  score were “0”, then the email is forwarded to the next process. This is because the domain name is similar, but the Top Level Domain name (TLD) is not. Therefore, the proposed system sends the email to the DKIM and SPF processes.

Similar Domain Compare 3: If  $S_{domain}$  (which we already extracted with the previous algorithm) score is “0” and  $TLD_{domain}$  score is “1”, then the email is forwarded to the next process, which is DKIM and SPF. This is because it means that the domain name is not similar, but the Top Level Domain name (TLD) is similar. Therefore, the proposed system will send the email to the DKIM and SPF processes.

SCP and NCC Compare 1: If the  $SCP$  score (which we already calculated with the previous algorithm) is “1” and  $NCC$  score is “1”, then the email is classified as “Phishing”. This is because the proposed algorithm calculated a high similarity between the incoming domain name and the valid domain name.

SCP and NCC Compare 3: If the  $SCP$  score is “0” and  $NCC$  score is “1”, then the email is sent to the next process for further examination. This is because it means the proposed algorithm found that there is a significant number of common characters between two domains.

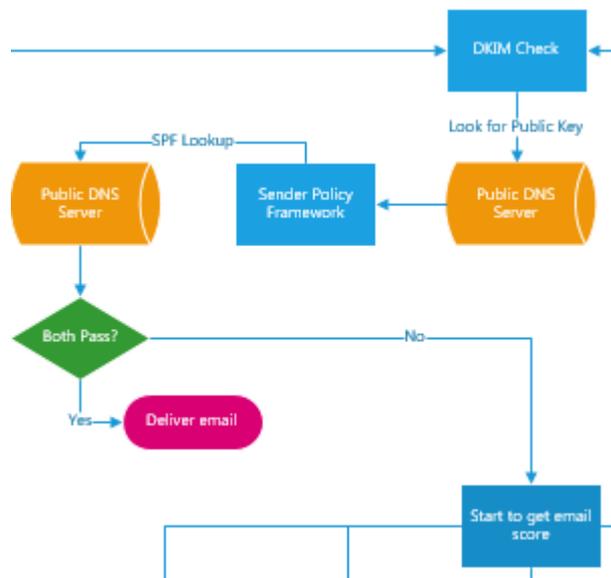
SCP and NCC Compare 3: If the  $SCP$  score is “0” and  $NCC$  score is “0”, then the email is sent to the next process for further examination. This is because the proposed algorithm threshold has not been met.

SCP and NCC Compare 5: If the  $SCP$  score is “0” and  $NCC$  score is “1”, then the email is classified as “Suspected Phishing”. This is because the proposed algorithm threshold value for similar character places has been met, therefore the email is classified as a suspect for further investigations.

## **4.7 DKIM and SPF**

This process was designed and added as an extra layer of security to make sure that the emails reaching users are 99% clean and valid.

Once an email is received, first the process checks the Domain Keys Identified Mail (DKIM) with a public DNS server. Once the result comes back from the Public DNS Server, the next process checks the Sender Policy Framework (SPF) with a Public DNS Server to hinder the ability of attackers to send email spoofing a domain name



**Figure 4.13 DKIM and SPF process**

If both the DKIM and SPF checks pass, then the system will deliver the email. This is because after the previous processes and this one, the proposed system believes that the email is 99.9% clean. However, if both of the checks or one of them failed, then an extra layer of filtering and checks are put in place to make sure that the email sender is legitimate:

**Step 1:** Read “DKIM” and “SPF” from DNS Domain

Check with Public DNS Server to see if SPF record is valid and authorised

Retrieve Public Key with Public DNS Server to verify sender key

**Step 2:** IF Both Pass = yes => Deliver Email

**Step 3:** IF Both Pass = No => Check Domain Similarity

IF either of them pass = NO => Check Domain Similarity

## 4.8 Domain Keys Identified Mail (DKIM)

is a protocol used by email systems to verify the sender and integrity of a message and prove that spammers did not modify an incoming message while in transit.

### 4.8.1 How the DKIM process works:

- The domain owner generates and publishes a cryptographic key. This is specifically formatted as a TXT record and added to a domain DNS record.
- When an email is sent from the sender server, the server generates and attaches the unique DKIM signature to the header of the email message.
- The DKIM key is used by the recipient mail servers to decrypt the message's signature and compare it against the domain DNS record. If the values match, it proves that the message is authentic and unaltered in transit, therefore, not forged or altered.

### 4.8.2 Sender Policy Framework (SPF)

Sender Policy Framework (SPF): SPF prevents spammers or attackers from sending emails with a spoofed domain name as the sender. SPF adds IP addresses to a list of servers that are authorised to send email from your domain. It verifies that messages sent from your domains originated from the listed server, which reduces the amount of backscatter that you receive. Here is how the SPF process works:

- System-admin creates a policy that defines which mail servers are authorised to send email from the domain. This SPF record is added to the domain DNS records.
- Once an email is received by the recipient mail server, it looks up the rules for the return-path (bounce) domain in the DNS record. The recipient mail server compares the IP address of the mail sender with the authorized IP addresses in the DNS record.
- The recipient mail server uses the rules specified in the sending domain's SPF DNS record to decide whether to accept, reject, or otherwise flag the email message.

An example of received email by Gmail with DKIM and SPF results is shown in (Figure 4.14).

```
dkim=pass header.i=@ljmuac.uk header.s=default header.b=dFP3P197;  
spf=pass (google.com: domain of dontreply@ljmuac.uk designates 10
```

Figure 4.14 Example from received email by Gmail

### 4.8.3 Complementary Filtering and Checks

In this process, we used an existing solution which was designed to prevent spam emails, because we believe that the same system to prevent spam can be used in conjunction with the proposed method to increase the detection rate.

If the results of DKIM and SPF failed, then the incoming email is forwarded to this process. This process has five sub-processes. An incoming email is passed to each of these five sub-processes for further checks. Each of these sub-processes has a scoring limit, which if exceeded, will categorise the email as phishing. Each filter below contributes to a SPAM/Phishing scoring. If the received email returns a total score greater than the “Pre-defined Scoring Limit”, then the message will be blocked. Compared to the Bayesian option, the Hidden Markov Model (HMM) produces results that are more exact.

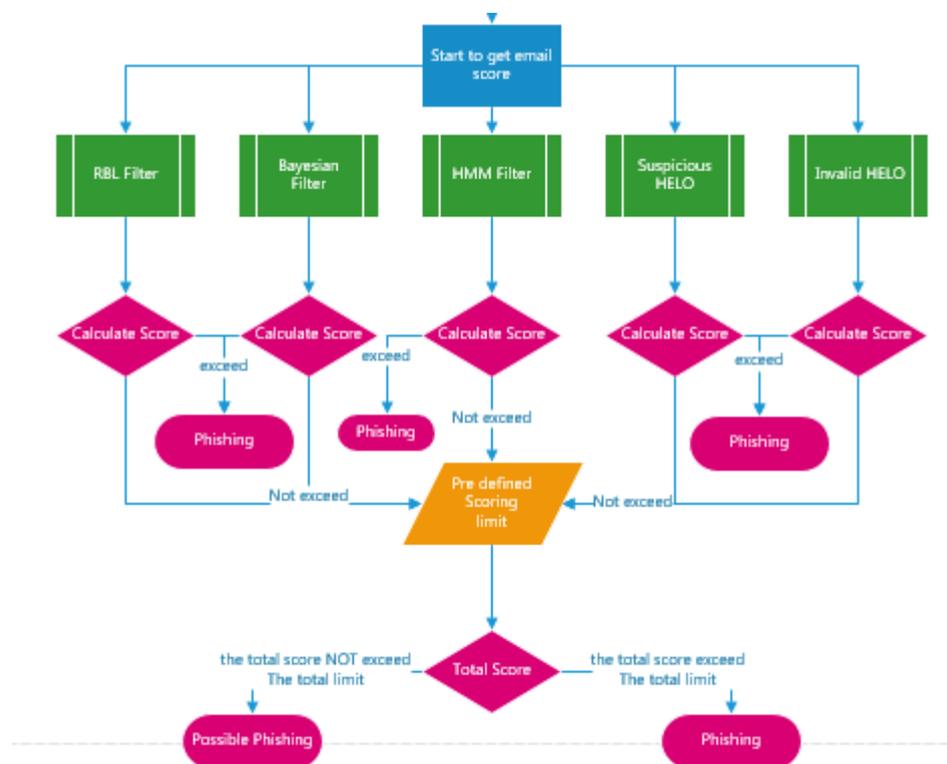


Figure 4.15 Extra Filter Check Process

### **Step 1: Check with RBL Filter**

This filter extracts the sender IP address from the email header and checks it with the configured RBL one at a time. If the check returns a positive result, it means the sender IP address is listed by one of the RBL servers and a spam score equal to the RBL server's assigned confidence level is assigned to the email.

Calculate Score:

- IF Pre-defined Score Exceed = No => Send to Total Pre-defined Score
- IF Pre-defined Score Exceed = Yes => Label Email as Phishing

### **Step 2: Check Bayesian Filter**

This scoring filter adds to a message's score if it contains specific words, and when it exceeds a pre-defined score, it categorises the message as phishing/spam. An example is "Share Password", which would surely give a high score.

Calculate Score:

- IF Pre-defined Score Exceed = No => Send to Total Pre-defined Score
- IF Pre-defined Score Exceed = Yes => Label Email as Phishing

### **Step 3: HMM Filter**

○ Calculate Score:

- IF Pre-defined Score Exceed = No => Send to Total Pre-defined Score
- IF Pre-defined Score Exceed = Yes => Label Email as Phishing

### **Step 4: Suspicious HELO**

- Calculate Score:
  - IF Pre-defined Score Exceed = No => Send to Total Pre-defined Score
  - IF Pre-defined Score Exceed = Yes => Label Email as Phishing

**Step 5: Invalid HELO**

- Calculate Score:
  - IF Pre-defined Score Exceed = No => Send to Total Pre-defined Score
  - IF Pre-defined Score Exceed = Yes => Label Email as Phishing

**Step 6: Total Score result**

- IF Total Pre-defined Score NOT Exceed = No => Label email as “Possible Phishing”
- IF Total Pre-defined Score NOT Exceed = Yes => Label email as “Phishing”

This complementary filtering along with ECSPAD creates a multi-layered solution to detecting an “Enterprise Targeted SpearPhishing Attack” with higher accuracy compared with existing methods, as compared in Chapter 6.

## 4.9 Conclusion

This chapter has presented a detailed overview of the current issues with targeted spearphishing attacks, which attackers use to bypass detection with high rates of success. This chapter presented a real-world example of targeted spearphishing attacks, where attackers use a mixture of different techniques such as Spearphishing, Typosquatting and Credential harvesting to bypass detection and conduct successful attacks.

In order to combat and detect such attacks, a multi-layered method is presented in this chapter, which has provided multiple algorithms for a multi-layered solution. The presented method was developed specifically to detect “Enterprise Targeted Spearphishing Attacks”, where attackers select their targets and harvest social networks for personal information about them to conduct a personalised attack. Their mission is to convince people to click on a link or download a malicious file. By gathering information

about target(s), an attacker can make emails that are extremely accurate and compelling.

Here, we presented a Threat Model which we used to evaluate and justify this solution. In order to have reasonable justification, we tested the presented threat model on the university email detection system, which we successfully bypassed in our test (in order to protect the university network, we tested the threat model with myself and my supervisor's email. My supervisor was aware and agreed to these experiments. In addition, no attached file was used during these experiments; only a link was contained in the emails).

The theoretical part of the aim has been successfully fulfilled using the presented method. In this regard, we have started by presenting how an Enterprise Targeted Spearphishing Attack works. We made a questionnaire for students and staff, after asking them to read an email and open a link, which asked them if there was anything wrong with this email and the page. Unfortunately, only 2 out of 50 people noticed that the webpage they were browsing was a cloned version of its legitimate page.

The presented multilayer solution could be a great help in helping users detect and avoid Enterprise Targeted Spearphishing Attacks. The proposed solution has three main processes. The first process is Feature Extraction, where we presented two main algorithms with six functions. By providing this layer, and those algorithms and functions, we successfully achieved the second objective of this research, which was detecting an Enterprise Targeted Spearphishing Attack. Then, to verify the sender, we used another layer to check the Domain Keys Identified Mail (DKIM) and Sender Policy Framework (SPF) of the sender domain against public DNS servers. In order to provide more a reliable check, we added a Complementary Filtering layer. This layer examines the sender domain with five different online directories to make sure that the sender is not listed anywhere. The enterprise email phishing detection system we aimed for has been developed (completion of Objective 4) and tested by two organisations in the UK, and two in Qatar.

# CHAPTER 5

## EMAIL PHISHING TRAINING WEBSITE

### 5.1 Introduction

In this chapter, we have provided an interactive email phishing training website to complete the presented security strategy for a Qatar organisation based on targeted email phishing which is Email Phishing Training. The Email Phishing Training project aims to provide email phishing and targeted phishing awareness to organisations and government institutions through interactive training in the State of Qatar.

The project frontend is written in typescript on top of an angular platform (Angular 7) and is powered by Asp.net Core 2.2 on the backend, using a node server as a mail sender agent. The intention is to increase and measure user awareness of email phishing and targeted phishing.

This application gathers user behaviour with certain sensors placed in the email and simulated page, which collects information on how users interact with phishing emails. Consequently, the user is shown educational interactive videos designed to improve their awareness of phishing attacks.

### 5.2 Goals

1. Start with Employee Training. Phishing awareness training starts with educating your employees on why phishing is harmful, and empowering them to detect and report phishing attempts. Depending on your organization's culture, you can deliver this initial training via a written document, an online video, company or department meetings, classroom training, of some combination of the above.
2. Create Simulated Phishing Campaigns. Simulated phishing campaigns reinforce employee training, help you understand your own risk, and improve workforce resilience. These can take many forms, such as mass phishing, spear phishing, and whaling.
3. Reinforce the Phishing Awareness Training. Nothing teaches like experience. When employees click on a link or an attachment in a simulated phishing email,

it's important to communicate (nicely, of course) to them that they have potentially put both themselves and the organization at risk. You can then display a “training page” that reinforces the dangers of phishing and reminds the employees how to report suspect emails.

4. Monitor Results and Improve. Use the results, such as the attack types that were most successful, and which teams were most vulnerable, to focus your security monitoring, strengthen your phishing awareness training, and add additional defences for phishing protection. You can also use the results to track the progress of your phishing awareness program and document improvements.

## **5.3 Methodology/Development Process**

The first step in initiating project development was to identify the broad areas of development that would be required, and then to match these to the available web technology. The website is multi-language based, and other languages can be added later.

The technologies used in project:

- Frontend Website (Angular version 7)
- Backend Website (ASP.net core 2.2)
- Service Management (Node.js)
- Server (linux nginx)
- Database Design (Mysql)

### **5.3.1 Why use asp.net core instead of other languages for server side:**

Asp.net core can handle 7+ Million HTTP requests per second from a single server. Looking at the latest run from the TechEmpower Benchmarks continuous results, ASP.NET 2.2 is the 3<sup>rd</sup> fastest webserver (0.046% off the top spot), able to respond to 7 million HTTP requests per second. Figure 5.1 compares asp.net core and other frameworks' performance in terms of response time per second and shows the benchmark of asp.net core server (kestrel) compared with other servers in terms of response time per second.

Best plaintext responses per second, Dell R440 Xeon Gold + 10 GbE (307 tests)									
Rnk	Framework	Best performance (higher is better)	Errors	Cls	Lng	Plt	FE	Aos	IA
1	ulib	7,034,945	100.0%	1	Plt	C++	Non	ULI	Lin
2	fasthttp	7,033,219	100.0%	0	Plt	Go	Non	Lin	Rea
3	libreactor	7,027,558	99.9%	0	Mcr	C	Non	Lin	Rea
4	wizzardo-http	7,026,401	99.9%	0	Mcr	Jav	Non	Lin	Rea
5	ulib-plaintext_fit	7,022,947	99.8%	0	Plt	C++	Non	ULI	Lin
6	actix-raw	7,021,312	99.8%	0	Plt	Rus	Non	act	Lin
7	aspcore	7,016,017	99.7%	0	Plt	C#	.NET	kes	Lin
8	hyper	7,013,819	99.7%	0	Mcr	Rus	Rus	Hyp	Lin

**Figure 5.1 Benchmark of asp.net core and other frameworks performance comparison**

It is also an extraordinary amount of bandwidth; enough to continuously saturate a 10GBps link. These results are with the webserver and load tester running inside Docker containers, on two different physical Linux machines connected with a 10GbE network. All this is throughput from a single server. ASP.NET Core is fast on Linux, and on Windows, compared with Other Servers How does it compare to other well-known servers?

In these “platform” comparisons, that’s: x1.78 faster than nginx, x2.93 faster than Java’s Servlet (x7.76 faster than Servlet on Tomcat), x7.36 faster than Golang’s “net/http” package, and x8.06 faster than node.js, running as a cluster of 28 processes (as node.js is single threaded).

### 5.3.2 Why use angular 7 for front end:

Angular is opinionated, giving developers defaults for things like network connectivity, state management, language choice, and building a toolchain. These defaults are continually tested and validated with each other to ensure that the Angular platform moves forward at a steady and reliable pace.

### 5.3.3 Angular is scaled

Angular was created at Google to solve Google-scale problems. To Google, this means millions of lines of code, thousands of engineers, widely varying project schedules, requirements, and workflows. The platform is designed to enable you to build and manage shared code, and to divide work amongst people holding appropriate roles. Many teams have separate designers, engineers, quality assurance, and other roles. The Component model used in Angular was designed to separate these concerns, and to allow a greater number of developers to collaborate.

### **5.3.4 Angular is Trustworthy**

Because Angular is a Google product, it is able to take advantage of Google's testing infrastructure. Every change that is made to Angular is validated against every Angular project within Google. This means that before any public release, the framework is already in use across hundreds of projects, maximizing the chance that there are no unintentional breaking changes or regressions.

### **5.3.5 Angular is Familiar**

The other common background includes developers coming from Java or C.NET. Both of these languages rely on typing, and have a centralized concept of an application that is very similar to the architecture required by Angular. Angular applications are broken into modules and components, and developers have the ability to import and export subparts of their application as needed. This is a very familiar mental model, and also helps developers get up to speed more quickly, allowing them to architect their applications successfully.

### **5.3.6 Angular has a strong Ecosystem**

There are thousands of reusable tools, libraries, and code samples across the internet for Angular and AngularJS, and a huge number of these tools have either been updated to work with Angular, or are already in the process of being updated. Developers such as VMWare, Teradata, ag-Grid, NativeScript, and others fully support Angular.

### **5.3.7 Why use MySQL as Database**

If you're looking for a free or low-cost database management system, several are available from which to choose: MySQL, PostgreSQL, SQLite, one of the free-but-unsupported engines from commercial vendors, and so forth. When you compare MySQL with other database systems, think about what's most important to you. Performance, support, features (such as SQL conformance or extensions), licensing conditions and restrictions, and price are all factors to take into account. Given these considerations, MySQL has many attractive features to offer:

- Speed: MySQL is fast. Its developers contend that MySQL is about the fastest database system you can get. You can investigate this claim by visiting <http://dev.mysql.com/tech-resources/benchmarks/>, a performance-comparison

page on the MySQL AB Web site.

- Ease of use: MySQL is a high-performance, but relatively simple database system, and is much less complex to set up and administer than larger systems.
- Query language support: MySQL understands SQL (Structured Query Language), the standard language of choice for all modern database systems.
- Capability: The MySQL server is multi-threaded, so that many clients can connect to it at the same time. Each client can use multiple databases simultaneously. You can access MySQL interactively using several interfaces that let you enter queries and view the results: command-line clients, Web browsers, or GUI clients. In addition, programming interfaces are available for many languages, such as C, Perl, Java, PHP, Python, and Ruby. You can also access MySQL using applications that support ODBC (Open Database Connectivity), a database communications protocol developed by Microsoft. This gives you the choice of using pre-packaged client software or writing your own for custom applications.
- Connectivity and security: MySQL is fully networked, and databases can be accessed from anywhere on the Internet, so you can share your data with anyone, anywhere. But MySQL has access control so that any person who shouldn't see another's data, cannot. To provide additional security, MySQL supports encrypted connections using the Secure Sockets Layer (SSL) protocol.
- Portability: MySQL runs on many varieties of Unix, as well as on other non-Unix systems, such as Windows, NetWare, and OS/2. MySQL runs on hardware from small personal computers (even palmtop devices) to high-end servers.
- Small size: MySQL has a modest distribution size, especially compared to the huge disk space footprint of certain commercial database systems.
- Availability and cost: MySQL is an Open Source project with dual licensing. First, it is available under the terms of the GNU General Public License (GPL). This means that MySQL is available without cost for most in-house uses. Second, for organizations that prefer or require formal arrangements, or that do not want to be bound by the conditions of the GPL, commercial licenses are available.
- Open distribution and source code: MySQL is easy to obtain; just use your Web browser. If you don't understand how something works, are curious about an algorithm, or want to perform a security audit, you can get the source code and

examine it. If you think you've found a bug, report it; the developers want to know.

### **5.3.8 Why use node.js as mail sender service**

Node.js is a packaged compilation of Google's V8 JavaScript engine, the libuv platform abstraction layer, and a core library, which is itself primarily written in JavaScript. Beyond that, it's worth noting that Ryan Dahl, the creator of Node.js, was aiming to create real-time websites with push capability, "inspired by applications like Gmail". In Node.js, he gave developers a tool for working in the non-blocking, event-driven I/O paradigm.

If use case does not contain CPU intensive operations, or require access to any blocking resources, you can exploit the benefits of Node.js, and enjoy fast and scalable network applications. Welcome to the real-time web.

A service to send an email concurrent with high speed and checking database simultaneously was needed, therefore node.js was best solution for this problem.

## **5.4 Data Collection**

After gaining knowledge of these types of phishing and the proposed ECSPAD the next task was to interview personnel in Qatari organisations of different sectors. The interview was carried out through a questionnaire and face to face meetings. The sectors interviewed were education, information technology and personnel from the government of Qatar. These sectors were chosen because they depend heavily on email communication to carry out their daily work from within and outside Qatar. The designed questionnaire was meant to collect data and information related to any problems facing their organisations in terms of phishing attacks. After studying the revealed problems through the interview, the next phase was to develop a web-based tool for creating an awareness training programme for Qatar government institutions and the public in general. This web-based tool was designed to be interactive to give users knowledge and demonstrate to them what email phishing entails. Users are given an opportunity to answer some basic questions through a questionnaire before starting the training. This help to present users with appropriate training based on their level. After the interactive training, users are awarded some points and later on are asked to answer some interactive questions to assess their knowledge after the training. Users are given more options to

repeat the questions whenever they prefer until they are satisfied with their results. The effectiveness of the awareness training programme will be evaluated by how users perform after their training sessions. A sample of web-based snapshot is shown in Figure 5.2 and its URL is available at: <https://atf.ljmu.ac.uk/>



**Figure 5.2 A home page of the developed Web-based tool**

### **5.4.1 Interview Guide and Focus Group Themes**

The semi-structured interview with different organizational experts focused on gathering useful information about email phishing. Main questions to experts were: (a) Are there any email phishing awareness training programmes in Qatar? (b) What are the challenges and problems facing organisations in implementing email phishing awareness training programmes? (c) What are the contributing factors affecting email phishing in the state of Qatar? (d) What are the counter processes that are available to sectors to detect email phishing and to limit its impact? (e) What is the level of public general knowledge, especially employees of organisations about the threat of email phishing in Qatar? (f) If there are email phishing awareness training programmes in some organisations, how could these programmes be improved? A total number of 154 participants were involved in this study during the interview, focus group discussions and training campaign on raising awareness regarding email phishing in the State of Qatar.

## **5.5 Results and Discussion**

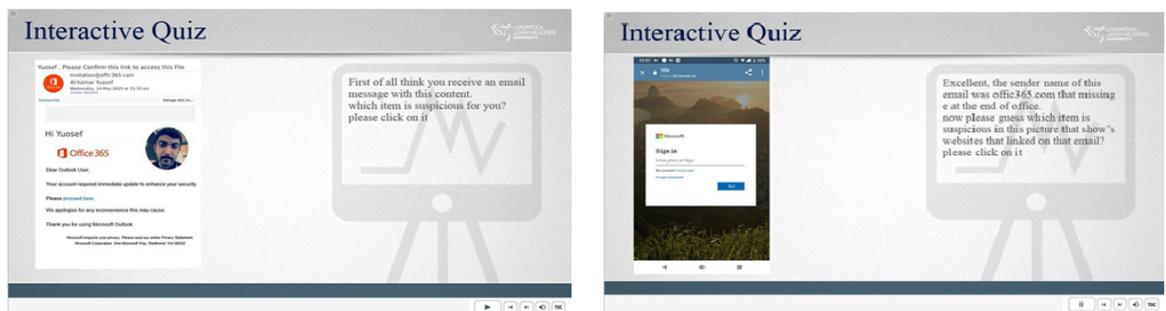
### **5.5.1 Database Design and User Experience**

A web-based tool that was used by users to answer some of questions under this study

was first designed. In terms of user experience, the website succeeds in providing a consistent and coherent user experience across the website. The website's narrative metaphor is realized effectively, in a dynamic and engaging way. Information from sensors that were put in the emails was gathered and analysed. At the end of the campaign, this information which had been analysed was presented with graphs and charts. In the user panel, users can see two types of reports; detailing reports and charts.

## 5.5.2 Interactive Quiz

In this section, an interactive quiz page as shown in Figure 5.3 was designed to help users recognize which part of an email to look at when trying to detect a phishing email, and how avoid getting fooled by phishing emails.



**Figure 5.3 An Interactive Quiz Design for the user**

Using this interaction, the user must identify and click on the items which appear suspicious in the phishing email. After clicking on links in the email, users are shown a page which is designed to deceive the user. The user learns how they can detect a page is simulated by asking questions highlighting the suspicious parts of the page.

## 5.5.3 Phishing Awareness Training

After creating a campaign and running it, each person who is fooled receives an email containing a link to an interactive online course, which is created with Adobe Captivate, and includes an online quiz at the end of the session.



**Figure 5.4 Phishing Awareness Training and Learning Objectives**

This module is highly interactive, and contains audio/visual components. A list of content will be shown to the users, as can be seen in Figure 5.4.

## 5.5.4 Database Design

A database for storing all information and data from users that interacted with the web-based tool was designed. There are three actors in the web-based tool namely: a website administrator, companies or organizations that want to check their employees' phishing awareness, and employees that are fooled by the phishing emails. The administrator only has two actions - news management and template management. The company panel or members' panel can create a campaign and import contacts who they want to send emails to, and see who gets fooled by the emails. Once the result of the campaign is shown in charts and graphs, at the end of campaign, an email is sent to users that were fooled, with a web link to our phishing awareness course. The diagram of our database design is illustrated in Figure 5.5 (Appendix A).

The main table of the designed database is the campaign, and this table is related to the following tables below: **RecipientList** by foreign key of **RecipientListId**; **FakeSender** by foreign key of **FakeSenderId**; **TimeZoneList** by enum of **TimeZoneId**; **CampaignDuration** by foreign key of **CampaignDurationId** and **RecipientList** table is related to **RecipientListDetail** by foreign key of **RecipientListDetailId**. All the campaign information is stored in this table.

## **5.6 Use Case Diagram**

There are three actors in the website - a website administrator, companies or organizations that want to check their employees' phishing awareness, and employees that are fooled by the phishing emails. The website administrator only has two actions - news management and template management.

The company panel or members' panel can create a campaign and import contacts who they want to send emails to, and see who gets fooled by the emails. Once the result of the campaign is shown in charts and graphs, at the end of campaign, an email is sent to users that were fooled, with a web link to our phishing awareness course. The use case diagram has been illustrated in Figure 5.6 (Appendix A).

## **5.7 Project Layout Diagram**

The project layout can be summarised as in Figure 5.7 (Appendix A).

## **5.8 Business Project Layout**

The project layout can be summarised as in Figure 5.8 (Appendix A).

## **Conclusion**

In this chapter, a targeted phishing email awareness strategy to organizations and government institutions through interactive training in the State of Qatar was presented. An enterprise-wide email phishing detection system to be used by organizations and individuals in the state of Qatar was developed. This detection system will help to reduce email phishing attacks in Qatar. The chapter further developed an email phishing awareness training framework to be used by organizations in the state of Qatar. The framework will help to effectively reduce the email phishing threats in Qatar. This framework will also help the Qatar government institutions and different organizations to enhance email phishing awareness for their employees through a set of recommendations. Furthermore, the chapter develops an interactive web-based tool to enhance the awareness of users in terms of targeted email phishing attacks to be used in organizations and individuals in Qatar. This interactive tool would help to train users on how to spot email phishing threats and hence reduce those threats to employees and citizens in Qatar.

# CHAPTER 6

## TEST AND RESULTS

### 6.1 Introduction

This chapter discusses the testing phase followed by the results produced. The implementation of the presented method and the algorithms are collected here and discussed in detail to show the enhancement of our technique in detecting spearphishing attacks, then followed by the awareness-training framework by performing some different tests.

The first section covers the technical proposed solution, which is called ECSPTAD (Enterprise Credential SpearPhishing Targeted Attack Detection) and the second section covers the evolution of the proposed awareness-training framework. At the end of the tests, by comparing the results, we have validated that the proposed solutions achieved the main aim of this thesis, which is to develop a solution that detects Enterprise Credential Spearphishing attack. The second aim of this thesis which is to develop an awareness-training framework for the state of Qatar to train users to reduce the impact of phishing attacks, is presented in detail. There is a proverb saying, “Prevention is better than cure”.

### 6.2 Enterprise Credential SpearPhishing Attack Detection (ECSPAD)

The proposed method will examine the received email by extracting features from the domain and evaluate it with a trusted domain database. The idea behind this method was to propose a new solution that can work like a human brain and evaluate an email domain address with a trusted one, and if it found lots of similarity between them, then categorise it as phishing email. This system will do extra checks as a method to reduce the risk of Spearphishing attack.

#### 6.2.1 `ljmu.ac.uk`

In this part, we performed a series of tests to evaluate the proposed method. In Table 6.1, we have a valid domain name set to “`ljmu.ac.uk`”. As mentioned in Chapter 4, the Similar

Character Place (SCP) Threshold Value and Number of Common Characters (NCC) will be calculated based on the valid domain name.

To calculate the SCP threshold value, we use the following proposed algorithm:

*def: Calcualte  $TV_{SCP}$ :*

$1 \frac{1}{2} \text{len } S_{domain}$

$\text{round}(TV_{SCP})$

$S_{Domain} = \text{"ljmu"}$

$\text{len } S_{domain} = 4$

$TV_{SCP} = 1 \frac{1}{2}(4) = 2$

Therefore, the SCP threshold value will be "2" for this domain.

Now it is time to calculate the NCC threshold value, we use the following proposed algorithm:

*def: Calcualte  $TV_{NCC}$ :*

$1 \frac{1}{3} C_{nouc} V_{domain}$

$\text{round } TV_{SCP}$

$V_{Domain} = \text{"ljmu.ac.uk"}$

$C_{nouc} V_{domain} = 7 - [L,J,M,U,A,C,K]$

$TV_{NCC} = 1 \frac{1}{3} 7 = 2.3$

$\text{round } TV_{SCP} = 2$

Therefore, the NCC threshold value will be “2” for this domain, as the proposed algorithm rounded “2.3” to “2”.

**Table 6.1: Case Study 1**

<b>VALID DOMAIN</b>	Ljmu.ac.uk
<b>INCOMING EMAIL</b>	Ljmuac.uk
<b>NUMBER OF COMMON CHARACTERS</b>	7
<b>NUMBER OF SIMILAR CHARACTER PLACE</b>	4
<b>SCP THRESHOLD VALUE</b>	2
<b>NCC THRESHOLD VALUE</b>	2
<b>SCP RESULT</b>	1
<b>NCC RESULT</b>	1

Once the SCP and NCC Threshold value were calculated, then we used domain “ljmuac.uk” as the phishing domain name. As results shown in

, we assume that the attacker registered the domains on

to perform the “Credential SpearPhishing Attack” by choosing the same domains to the victim domain name.

**Table 6.2: Result for case study 1**

<b>INCOMING EMAIL DOMAIN</b>	<b>Classified as Phishing?</b>
<b>LJMUAC.UK</b>	Yes

Once an email is received from “user@ljmuac.uk”, the proposed system will start to work. At the beginning, the system will extract the following features from an incoming email domain name.

**Table 6.3: Valid Domain Name (ljmu.ac.uk)**

<b><i>VCnoc</i></b>	<b>10</b>
<b><i>VCnouc</i></b>	<b>7</b>
<b><i>VDomain</i></b>	<b><i>ljmu.ac.uk</i></b>

<b>S<sub>2</sub>Domain<sub>2</sub></b>	<i>ljmu</i>
<b>VC<sub>2</sub>nod<sub>2</sub></b>	<b>2</b>
<b>VC<sub>2</sub>nonv<sub>2</sub></b>	0
<b>VC<sub>2</sub>noh<sub>2</sub></b>	<b>0</b>
<b>VE<sub>2</sub>de<sub>2</sub></b>	<i>ac.uk</i>
<b>VC<sub>2</sub>nocbfd<sub>2</sub></b>	<b>4</b>
<b>V<sub>2</sub>ip<sub>2</sub></b>	192.168.1.10

**Table 6.4: Incoming Domain Name (ljmuac.uk)**

<b>INC<sub>2</sub>noc<sub>2</sub></b>	<b>9</b>
<b>INC<sub>2</sub>nouc<sub>2</sub></b>	7
<b>IN<sub>2</sub>Domain<sub>2</sub></b>	<i>ljmuac.uk</i>
<b>S<sub>2</sub>InDomain<sub>2</sub></b>	<i>ljmuac</i>
<b>INC<sub>2</sub>nod<sub>2</sub></b>	<b>1</b>

<i>IN C nonv</i>	0
<i>INC noh</i>	0
<i>INE de</i>	<i>ac.uk</i>
<i>INC nocbfd</i>	6
<i>IN ip</i>	192.168.1.11

### Step 1: Whitelist domain

In step1, the proposed system will start to verify if the incoming email domain name is the same as the valid domain name. To carry out this operation, the following algorithm is proposed:

*def1:*

*IF IN domain = V domain Then:*

*Pass*

*Else:*

*False*

The result of this step will be “fail” as “*IN domain = ljmuac.uk*” is not same as “*ljmu.ac.uk*”.

### Step 2: Whitelist IP

In step1, the proposed system will start to verify if the incoming email domain name is the same as the valid domain name. To carry out this operation, the following algorithm is proposed:

```

def1:
IF IN[ip] = V[ip] Then:
Pass
Else:
False

```

The result for this process will be “fail” as “IN[ip] = 192.168.1.11” is not same as “V[ip] = 192.168.1.10”

Because both “Step1” and “Step2” result came back as “fail”, the email will forward to next step to preform further examinations.

### Step 3: Find Similar Character Place (SCP)

In this we proposed and algorithm to find the similar character places between V[domain] and IN[domain]. As showing in Figure 6.1, The SCP between V[domain] and IN[domain] is just 4 characters.

```

Valid Domain:  ljmu.ac.uk
Incoming Email Domain:  ljmuac.uk

l  -----  l
j  -----  j
m  -----  m
u  -----  u
.  -----  a
a  -----  c
c  -----  .
.  -----  u
u  -----  k

Similar Character Place:  ['l', 'j', 'm', 'u']

```

Figure 6.1 SCP result for case study 1

The result for this process is “4”.

### Step 4: Find Number Common Character (NCC)

Once “Step3” is finished, this step will start. This process’s proposed aim is to find the

common character(s) between the incoming email domain and the valid domain name. To get more accurate results, the proposed algorithm will remove the dots from the result. The result from this step is shown in Figure 6.2 .

```
Valid Domain: ljmu.ac.uk
Incoming Email Domain: ljmuac.uk

Common Characters: ['a', 'c', 'k', 'j', 'm', 'l', 'u']
Number Common Characters: 7
```

**Figure 6.2 NCC result for case study 1**

As shown in Figure 6.2, the result of this process is “7”

### **Step 5: SCP and NCC Calculation**

Once the features and the results from previous processes have been successfully extracted, this proposed process will calculate the threshold value for SCP and NCC. To calculate the SCP, we proposed the following algorithm:

```
def3: Calcualte SCP
IF  $R[SCP] \geq TV[SCP]$  Then:
Return 1
Else:
Return 0
```

We need to calculate the  $TV[SCP]$ . The  $TV[SCP]$  is half of the length of the valid domain name ( $S[domain] = ljmu$ ). Therefore  $TV[SCP]$  is “2”. Based on the result from “Step3” which is “4” then the result of Calculate SCP will be “1”

```
Valid Domain: ljmu.ac.uk
Incoming Email Domain: ljmuac.uk
Calcualte SCP result: 1
```

**Figure 6.3: Calculate SCP result**

Now is time for the NCC calculation process to start. The following algorithm has been

proposed where  $R_{NCC}$  is Number of Common Characters that are extracted from “Step4” and it will be compared to  $TV_{NCC}$  (threshold value) which is calculated previously.

*def4: Calcualte NCC*

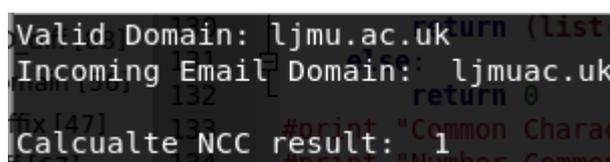
*IF  $R_{NCC} \geq TV_{NCC}$  Then:*

*Return 1*

*Else:*

*Return 0*

The  $R_{NCC}$  is “7” and the  $TV_{NCC}$  is “2”. Therefore the result of this should be “1” as Number of Common Characters is greater than the threshold value (Figure 6.4).



```
Valid Domain: ljmu.ac.uk
Incoming Email Domain: ljmuac.uk
Calcualte NCC result: 1
```

**Figure 6.4: Calculate NCC result**

### **Step 6: Check Domain Similarity**

Based on the results from previous processes, now it is time to decide whether the domain should be classified as Phishing, Suspected as Phishing or sent to the next step which is DKIM and SPF check. Based on the results, the proposed system classified the email as phishing, because  $SCP$  score is “1” and  $NCC$  score is “1” and the proposed algorithm calculated a high similarity between the incoming domain name and the valid domain name as shown Figure 6.5.

*def scp\_ncc\_comapre:*

*IF  $SCP = 1$  and  $NCC = 1$  Then:*

*Classified as Phishing*

```

Terminal
File Edit View Search Terminal Help
Valid Domain: ljmu.ac.uk
Incoming Email Domain: ljmuac.uk
Incoming email address: ljmuac.uk Classified as Phsihing

```

Figure 6.5: framework result for case study 1

## 6.2.2 instagram.com

In order to evaluate the proposed algorithms and framework, we preform an additional test with another domain. In this scenario, the user is receiving an email from “account@insatgarm.com”. The valid domain name is “instragram.com”. In this scenario, the attacker tries to perform his attack on one of the Instagram users.

To calculate the SCP threshold value, we use the following proposed algorithm:

*def: Calcualte  $TV_{SCP}$ :*

$$1 \frac{2}{len S_{domain}}$$

*round( $TV_{SCP}$ )*

$$S_{Domain} = \text{“instagram”}$$

$$len S_{domain} = 9$$

$$TV_{SCP} = 1 \frac{2}{9} = 4.5$$

$$round TV_{SCP} = 4$$

Because the  $TV_{SCP}$  is “4.5” we will round it. Therefore, the SCP threshold value will be “4” for this domain.

Now it is time to calculate the NCC threshold value, we use the following proposed algorithm:

def: Calcualte  $TV_{NCC}$ :

$$1/3 C_{nouc} V_{domain}$$

round  $TV_{SCP}$

$$V_{Domain} = \text{"Instagram.com"}$$

$$C_{nouc} V_{domain} = 10 - [a, c, g, i, m, o, n, s, r, t]$$

$$TV_{NCC} = 1/3 10 = 3.3$$

$$\text{round } TV_{NCC} = 3$$

Therefore, the NCC threshold value will be "3" for this domain, as the proposed algorithm rounded "3.3" to "3".

**Table 6.5: Case Study 2**

<b>VALID DOMAIN</b>	Instagram.com
<b>SCP THRESHOLD VALUE</b>	4
<b>NCC THRESHOLD VALUE</b>	3

Once the SCP and NCC Threshold value were calculated, then we used the domain "instagram.com" as the phishing domain name. As results shown in **Error! Reference source not found.**, we assume that the attacker registered the domains on **Error! Reference source not found.** to perform the "Credential SpearPhishing Attack" by choosing similar domains to the victim domain name.

**Table 6.6: Result for Test 2**

INCOMING EMAIL DOMAIN	CLASSIFIED AS PHISHING?
INSTAGARM.COM	Yes

Once an email is received from “account@insatgarm”, the proposed system will start to work. At the beginning, system will extract the following features from an incoming email domain name.

**Table 6.7: Valid Domain Name (Instagram.com)**

<b><i>VCnoc</i></b>	13
<b><i>VCnouc</i></b>	10
<b><i>VDomain</i></b>	<i>instagram.com</i>
<b><i>SDomain</i></b>	<i>instagram</i>
<b><i>VCnod</i></b>	1
<b><i>VCnonv</i></b>	0
<b><i>VCnoh</i></b>	0
<b><i>VEde</i></b>	<i>com</i>
<b><i>VCnocbfd</i></b>	9

**V<sub>ip</sub>**

192.168.12.100

**Table 6.8: Incoming Domain Name (insatgarm.com)**

<b>INC<sub>noc</sub></b>	13
<b>INC<sub>nouc</sub></b>	10
<b>IN<sub>Domain</sub></b>	<i>insatgarm.com</i>
<b>S<sub>InDomain</sub></b>	<i>insatgarm</i>
<b>INC<sub>nod</sub></b>	1
<b>IN C<sub>nonv</sub></b>	0
<b>INC<sub>noh</sub></b>	0
<b>INE<sub>de</sub></b>	<i>com</i>
<b>INC<sub>nocbfd</sub></b>	9
<b>IN<sub>ip</sub></b>	192.168.15.15

**Step 1: Whitelist domain**

In step1, the proposed system will start to verify if the incoming email domain name is the same as the valid domain name. To carry out this operation, the following algorithm is proposed:

```

def1:
IF INdomain = Vdomain Then:
Pass
Else:
False

```

The result of this step will be “fail” as “INdomain = insatgarm” is not same as “instagram”.

### Step 2: Whitelist IP

In step1, the proposed system will start to verify if the incoming email domain name is the same as the valid domain name. To carry out this operation, the following algorithm is proposed:

```

def1:
IF INip = Vip Then:
Pass
Else:
False

```

The result for this process will be “fail” as “INip = 192.168.12.100” is not same as “Vip = 192.168.15.15”

Because both “Step1” and “Step2” results came back as “fail”, the email will be forwarded to the next step to perform further examinations.

### Step 3: Find Similar Character Place (SCP)

In this we proposed an algorithm to find the similar character places between Vdomain and INdomain. As showing in Figure 6.6, The SCP between Vdomain and INdomain is “9” characters.

```

Valid Domain: instagram.com
Incoming Email Domain: insatgarm.com
a[3] i
i[3] i
n[4] n
s counter[5]
t d[6] a
a u[5] t
g y[10] g
r a
a r
m ----- m
. ----- .
c ----- c
o ----- o
m ----- m
Similar Character Place: ['i', 'n', 's', 'g', 'm', '.', 'c', 'o', 'm']

```

Figure 6.6: SCP result

#### Step 4: Find Number Common Character (NCC)

Once “Step3” is finished, this step will start. This process’s proposed aim is to find the common character(s) between the incoming email domain and the valid domain name. To get more accurate results, the proposed algorithm will remove the dots from the result. The result from this step is “7” as shown in Figure 6.7.

```

Valid Domain: instagram.com
Incoming Email Domain: insatgarm.com
Common Characters: ['a', 'c', 'g', 'i', 'm', 'o', 'n', 's', 'r', 't']
Number Common Characters: 10

```

Figure 6.7: NCC result

#### Step 5: SCP and NCC Calculation

Once the features and the results from previous processes have been successfully extracted, this proposed process will calculate the threshold value for SCP and NCC. To calculate the SCP, we proposed the following algorithm:

*def3: Calcualte SCP*

*IF  $R[SCP] \geq TV[SCP]$  Then:*

*Return 1*

*Else:*

*Return 0*

We need to calculate the  $TV_{SCP}$ . The  $TV_{SCP}$  is half of the length of valid domain name ( $S_{domain} = instagram$ ). Therefore  $TV_{SCP}$  is “4”. Based on the result from “Step3”, the result of Calculate SCP will be “1” as shown in Figure 6.8.

```
Valid Domain: instagram.com
Incoming Email Domain: pinsatgarm.com
123
Calculate SCP result:har1():
```

**Figure 6.8: Calculate SCP result**

Now is time for NCC calculation process to start. The following algorithm has been proposed where  $R_{NCC}$  is Number of Common Characters that are extracted from “Step4” and it will compare to  $TV_{NCC}$  (threshold value) which is calculated previously.

*def4: Calcualte NCC*

*IF  $R_{NCC} \geq TV_{NCC}$  Then:*

*Return 1*

*Else:*

*Return 0*

The  $R_{NCC}$  is “7” and the  $TV_{NCC}$  is “2”. Therefore the result of this should be “1” as Number of Common Characters is greater than the threshold value (Figure 6.9).

```
Valid Domain: instagram.com
Incoming Email Domain: insatgarm.com
131
Calculate NCC result:n 1
```

**Figure 6.9: Calculate NCC result**

## **Step 6: Check Domain Similarity**

Based on the results from previous processes, now it is time to decide whether the domain should be classified as Phishing, Suspected as Phishing or sent to the next step which is DKIM and SPF check. Based on the results, the proposed system classified the email as phishing, because *SCP* score is “1” and *NCC* score is “1” and the proposed algorithm calculated a high similarity between the incoming domain name and the valid domain name as shown in Figure 6.10.

```
def scp_ncc_comapre:
IF SCP = 1 and NCC = 1 Then:
Classified as Phishing
```

```
Valid Domain: instagram.com
Incoming Email Domain: insatgarm.com
Incoming email address: insatgarm.com Classified as Phsihing
```

**Figure 6.10 Framework result for Test 2**

### **6.2.3 Alpina.qa**

Alpina Group Ltd is a 100% Qatari diversified company operating in different industries including Food Distribution, Services, Technology, Catering & Hospitality, Construction and Real Estate.

Their main strength lies in forming strategic partnerships and long-term business relationships with customers, companies and organizations who are looking to establish a strong presence in Qatar and GCC. They are characterised by extensive knowledge, great experience that they possess and their commitment to advancing the interests of their partners. They strive for customer satisfaction through providing excellent service and a unique experience to all clients through professional and passionate teams.

Alpina Group uses email as their main line of contact with their clients and partners, therefore their email security should be able to detect email threats such as Phishing and SpearPhishing attacks.

As showing below, we registered a very similar domain to the original domain.

**Table 6.9: Valid Domain Name (alpina.qa)**

<b>VCnoc</b>	<b>9</b>
<b>VCnouc</b>	6
<b>VDomain</b>	<i>alpina.qa</i>
<b>SDomain</b>	<i>alpina</i>
<b>VCnod</b>	1
<b>VCnonv</b>	0
<b>VCnoh</b>	0
<b>VEde</b>	<i>qa</i>
<b>VCnocbfd</b>	6
<b>Vip</b>	192.168.20.100

**Table 6.10: Incoming Mail Domain (alpnia.qa)**

<b>VCnoc</b>	<b>9</b>
<b>VCnouc</b>	6
<b>VDomain</b>	<i>alpnia.qa</i>
<b>SDomain</b>	<i>alpnia</i>
<b>VCnod</b>	1
<b>VCnonv</b>	0
<b>VCnoh</b>	0
<b>VEde</b>	<i>qa</i>
<b>VCnocbfd</b>	6
<b>Vip</b>	192.168.22.100

As shown in **Error! Reference source not found.**, the proposed detection method classified the incoming email as phishing.

**Table 6.11: Alpina.qa Test**

<b>VALID DOMAIN</b>	Alpine.qa
<b>INCOMING EMAIL</b>	Alpnia.qa
<b>NUMBER OF COMMON CHARACTER</b>	6
<b>NUMBER OF SIMILAR CHARACTER PLACE</b>	4
<b>SCP THRESHOLD VALUE</b>	3
<b>NCC THRESHOLD VALUE</b>	2

<b>SCP RESULT</b>	1
<b>NCC RESULT</b>	1
<b>DETECTION RESULT</b>	Classified as Phishing

## 6.2.4 motc.gov.qa

The Ministry of Transport and Communications (MOTC) was established in 2016 working on building a robust knowledge-based economy in light of the Qatar national vision 2030 Organizing land, maritime, developing transport services and projects. Developing the information and communication sector also securing and enhancing the efficiency of ICT. The Ministry of transport and communication has a very big role in Qatar.

As show below, we got an authorisation from MOTC to be able to test this method on their email server; therefore, we registered a very similar domain to the original domain.

**Table 6.12: Valid Domain (motc.gov.qa)**

<b>VC<sup>?</sup>noc<sup>?</sup></b>	11
<b>VC<sup>?</sup>nouc<sup>?</sup></b>	8
<b>V<sup>?</sup>Domain<sup>?</sup></b>	<i>motc.gov.qa</i>
<b>S<sup>?</sup>Domain<sup>?</sup></b>	<i>motc</i>
<b>VC<sup>?</sup>nod<sup>?</sup></b>	2

<b>VC?nonv?</b>	0
<b>VC?noh?</b>	0
<b>VE?de?</b>	<i>gov.qa</i>
<b>VC?nocbfd?</b>	4
<b>V?ip?</b>	192.168.12.100

**Table 6.13: Incoming Mail Domain (motcogv.qa)**

<b>INC?noc?</b>	10
<b>INC?nouc?</b>	8
<b>IN?Domain?</b>	<i>motcogv.qa</i>
<b>S?InDomain?</b>	<i>motcogv</i>
<b>INC?nod?</b>	1
<b>IN C?nonv?</b>	0
<b>INC?noh?</b>	0

<b>IN</b> <i>de</i>	qa
<b>INC</b> <i>nocbfd</i>	7
<b>IN</b> <i>ip</i>	192.168.15.15

As shown the proposed detection method classified the incoming email as phishing.

**Table 6.14: motc.gov.qa Test Case**

<b>VALID DOMAIN</b>	<b>MOTC.GOV.QA</b>
<b>INCOMING EMAIL</b>	Motcogv.qa
<b>NUMBER OF COMMON CHARACTER</b>	8
<b>NUMBER OF SIMILAR CHARACTER PLACE</b>	4
<b>SCP THRESHOLD VALUE</b>	2
<b>NCC THRESHOLD VALUE</b>	2
<b>SCP RESULT</b>	1
<b>NCC RESULT</b>	1
<b>DETECTION RESULT</b>	Classified as Phishing

## 6.3 Comparison

In this part, we made a comparison between the results that we had from ECSPAD and other enterprise solutions and research solutions. Because the nature of the attack is a targeted attack and the victim will be selected rather than mass email sending, we perform a target test rather than analysing a database to find the phishing (Ho et al., 2017; Stringhini and Thonnard, 2015). Based on the conducted research, we could not find any solution exactly designed for Credential SpearPhishing attack.

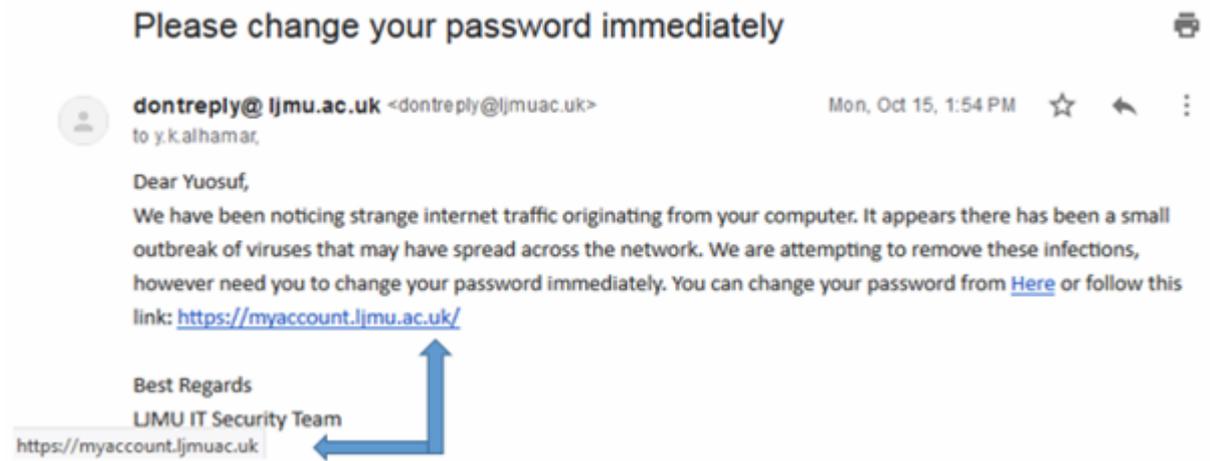
### 6.3.1 Trendmicro Email Phishing Detection

Liverpool John Moores University use TrendMicro Email Security as their enterprise approach to provide a secure environment for its email. As mentioned by TrendMicro on their website, *“A good technique for hunting and detecting suspicious domains is to also use a similar modus that cybercriminals typically employ: patterns. DNS data (i.e., passive system of record of DNS resolution data), for instance, provides information security professionals and system administrators insight on how a particular domain changes over time. Not only does this help them correlate indicators of compromise, but also provides the context needed for identifying related or additional suspicious domains. Domain registration information also helps unmask a cybercriminal's infrastructure by correlating a specific suspicious domain to others registered using similar information. (Cedric Pernet, Senior Threat Researcher at Trend Micro)”*

Trend Micro InterScan Messaging Security claims that it can stop email threads in the cloud with global threat intelligence, identifies targeted email attacks, social engineering attack and identifies targeted attack emails by correlating email components such as the header, body, and network routing.

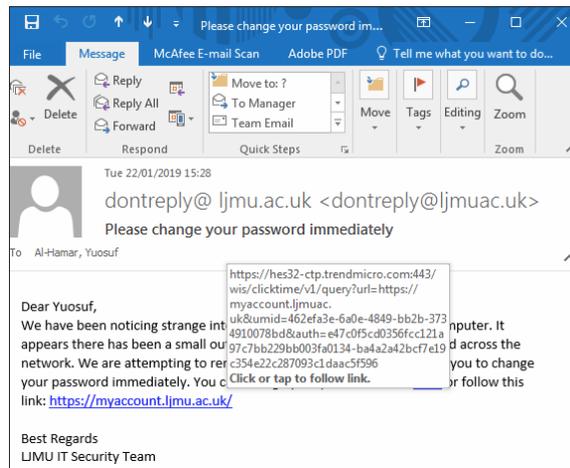
This research will prove that those claims at least are not valid for Enterprise Credential SpearPhishing attack by comparing the results of an email sent to a user in Liverpool John Moores University with TrendMicro as their email security system versus ECSPAD.

As shown in Figure 6.11, an email sent to a user (in this case myself) “y.k.alhamar@2016.ljmu.ac.uk” subject “Please change your password immediately”. In the content, we asked user to change his password due to strange internet traffic originating from his computer.



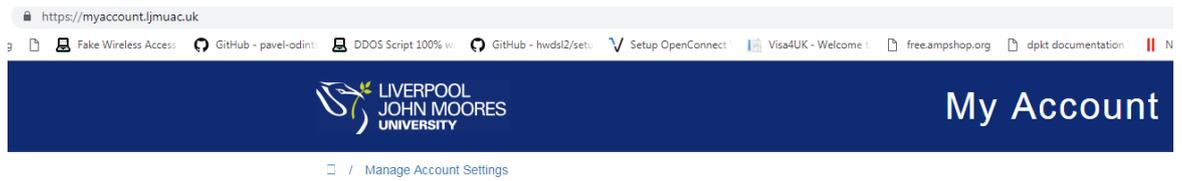
**Figure 6.11: Phishing email**

Then we asked him to follow a link to reset his password. As shown in Figure 6.12, the embedded TrendMicro email security system has a feature named “Unknown URL protection” that blocks emails with malicious URLs before delivery and re-checks URL safety when a user clicks on it.



**Figure 6.12: Phishing email content**

Once we clicked on the URL, the TrendMicro cloud threat intelligence system analysed the URL and opened it without any warning or block as shown in Figure 6.13.



**Figure 6.13: Cloned website**

For test purposes we used a test username “ljmu” and password “password” on the cloned website to get user credential details (Figure 6.14).

```
POSSIBLE USERNAME FIELD FOUND: ct100$MainContent$tbUsername=ljmu
POSSIBLE PASSWORD FIELD FOUND: ct100$MainContent$tbPassword=password
```

**Figure 6.14: User credentials**

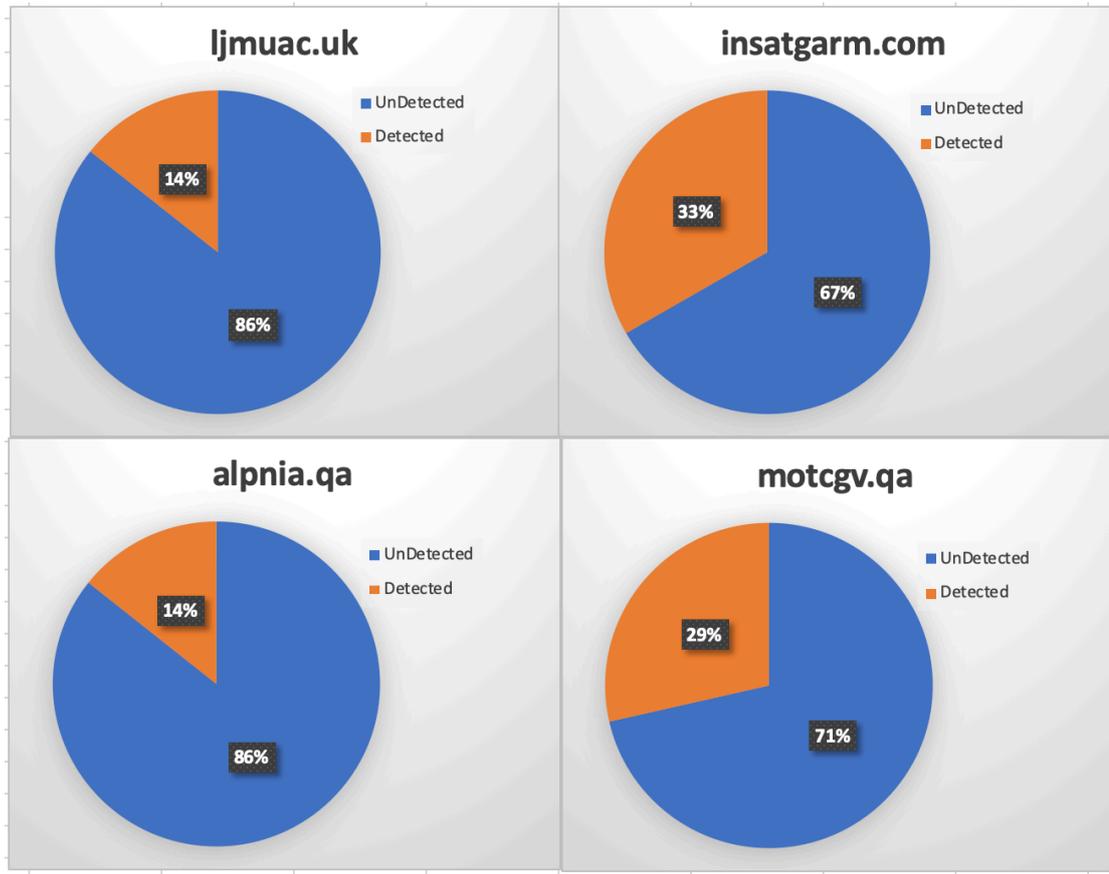


Figure 6.15 Comparison between different methods

### 6.3.2 Test Results

In this part, we ran our tests on 4 different available email services (with phishing detection). The result in **Error! Reference source not found.** shows that the only detection system that detected all of our tests is the proposed method in this thesis. However, from the result we can see that the gmail email server detection was able to detect our “Instagram.com” phishing attack and the motc.gov.qa was able to detect the attack that we sent from our registered domain “motcgv.qa”.

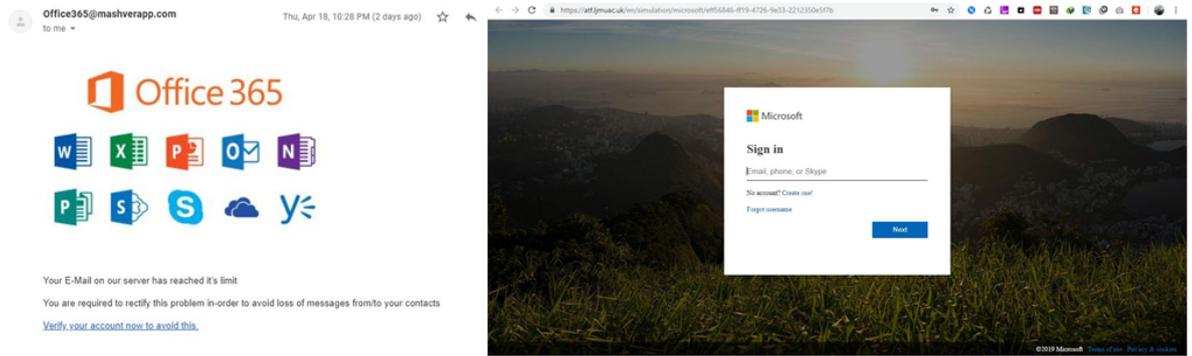
Table 6.15: Test and Results

EMAIL DETECTION	LJMUAC.UK	INSATGARM.COM	ALPNIA.QA	MOTCGV.QA
--------------------	-----------	---------------	-----------	-----------

<b>TRENDMICRO</b>	No	No	No	No
<b>YAHOO</b>	No	No	No	No
<b>LIVE</b>	No	No	No	No
<b>GMAIL</b>	No	Yes	No	No
<b>OUTLOOK</b>	No	No	No	No
<b>ECSPAD</b>	Yes	Yes	Yes	Yes
<b>COMPANY DETECTION SYSTEM</b>	No	No	No	Yes

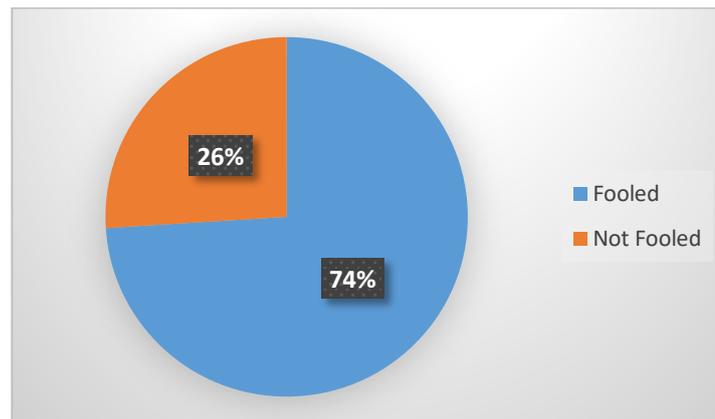
## 6.4 AWARENESS

In this part, a campaign with 154 people's emails to test how many of them are fooled was launched. This was tested with the Office 365 email template and the Office OneDrive login page. Figure 6.16 shows an email that was sent to users with a fake sender. The email tells users that their Microsoft account inbox is full and they must follow the link to avoid loss of messages. After clicking on the link, a page is shown, as you see in Figure 6.16. This page is simulated by Microsoft Office 365, and has a GUID in the link for detecting which users click the link.



**Figure 6.16 An Office 365 email template and the Office OneDrive login page**

If someone submits their information into the simulated login page it means they have been fooled, and a link about the learning course is sent to that user. A total number of 154 people were tested, and 74% of people got fooled as shown in Figure 6.17.

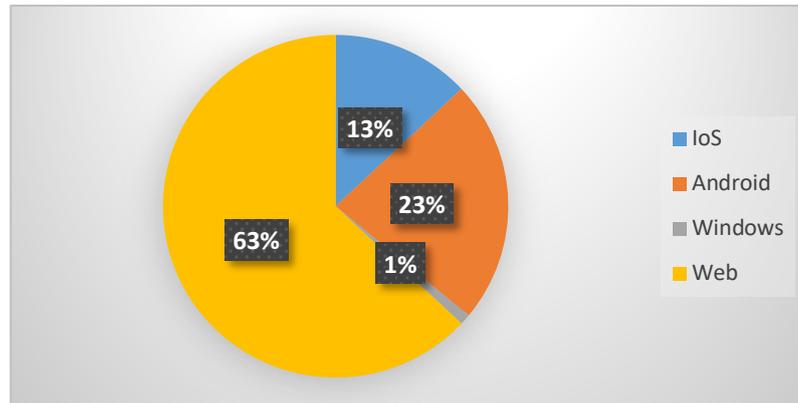


**Figure 6.17 Percentage of people that get fooled by email phishing**

Id	CreatedAt	CampaignId	RecipientId	Delivered	Opened	Username	Device	Agent
0232c565-1ff5-467f-abe0-9bcb32fd61b0	25/04/2019 18:07	8bab569e-b103-4786-99eb-b3d8ab837442	4c08f2f1-760c-4156-92d0-1236d7e15650	True	null	null	null	null
02d4a37a-3f21-4585-8952-86b773f1dfa5	25/04/2019 18:07	8bab569e-b103-4786-99eb-b3d8ab837442	90ee6e28-fa3a-4a04-819d-a91ec4f3456f	True	null	null	null	null
0468813b-24a9-4f77-b18d-7faf4f5e4783	25/04/2019 18:07	8bab569e-b103-4786-99eb-b3d8ab837442	358ec36a-a733-462b-acba-da3106cf6d5b	True	null	null	null	null
062291f6-ef84-4cf5-8bf9-61a4f97d57b3	25/04/2019 18:07	8bab569e-b103-4786-99eb-b3d8ab837442	07f740ee-328d-49b4-adb7-de36c469eb51	True	25/04/2019 18:08	null	null	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.103 Safari/537.36
0ac27868-fcd4-4e64-88f3-80427e29ad6e	25/04/2019 18:07	8bab569e-b103-4786-99eb-b3d8ab837442	944726c8-07bd-46fe-97cc-ff75a04d0f55	True	26/04/2019 00:10	null	null	null
0c796adc-4d6a-43b2-9933-dd79815d4c03	25/04/2019 18:07	8bab569e-b103-4786-99eb-b3d8ab837442	3789d90e-f072-438d-93e3-307dc6f1d686	True	null	null	null	null

**Figure 6.18 Data collected using the designed web-based tool**

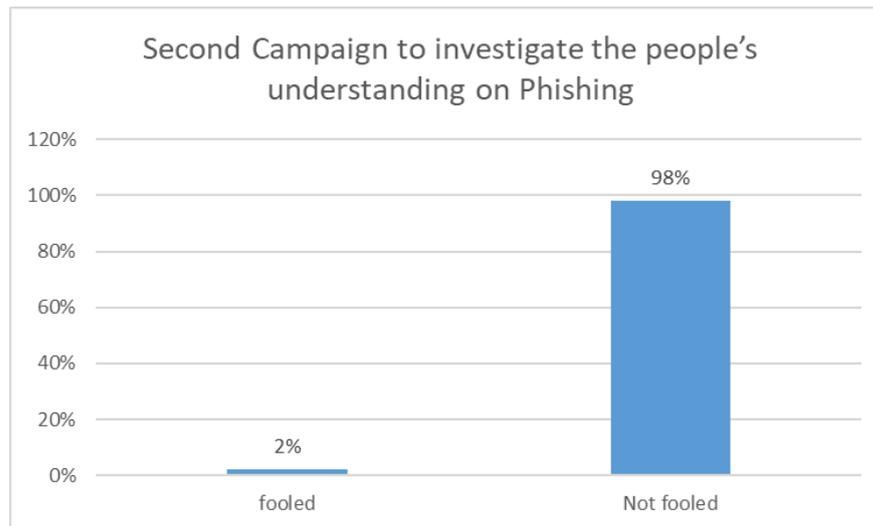
Figure 6.18 shows the screenshot of data collected from a sample of 154 people that participated in this study. Not the sample shown in Figure 6.19 is only based on the first participants. A type of browser, IP address, email address and an operating system that a particular user was using were collected.



**Figure 6.18 The number of people who were fooled using different operating system**

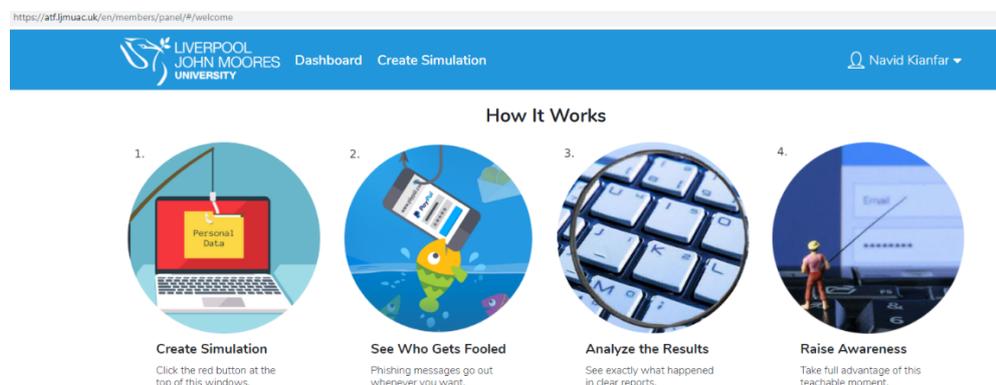
Figure 6.19 shows people that got fooled with the OS type that was used during this test investigation. It can be shown that 13% used IOS, 63% web, 1 % windows phone and 23% android.

It is important to note that, people who got fooled in the first campaign received an email after the campaign finished that contained a link to the course that was mentioned above; in the online course users were taught that they must check the sender name before opening an email and must check the URL of the website after login to that site. In that aspect, users take an exam at the end of the course to ensure that they have learned the objectives that were talked about in online course. After this campaign, a campaign with these 154 people was started again with a PayPal template using a similar fake sender name. An interesting outcome is that only 2% of all participants got fooled and submitted login form, while others based on awareness of online courses, suspected the second campaign and did not open the link.



**Figure 6.19 Second Campaign to investigate the people's understanding on Phishing**

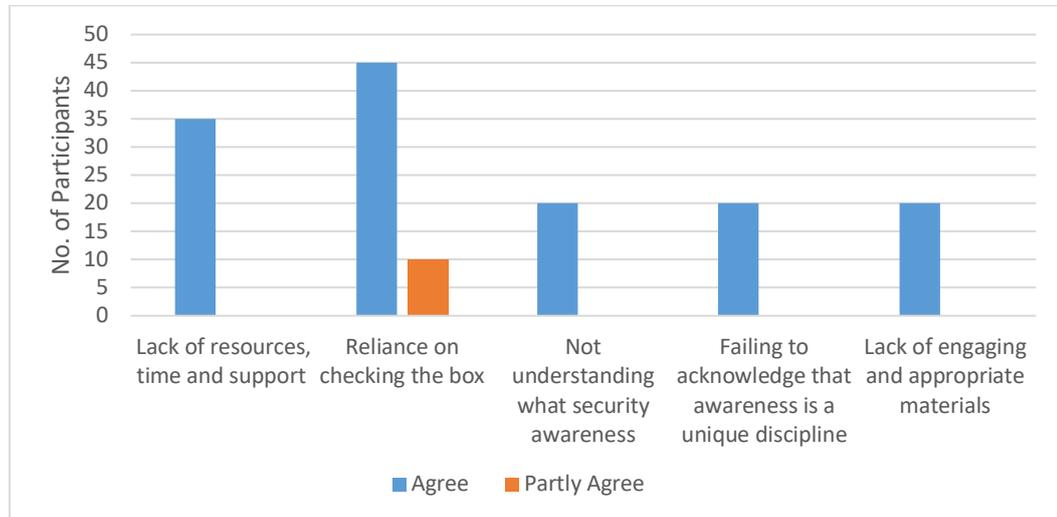
As shown in Figure 6.20, the results indicate that, this campaign achieved 98% of participants whose awareness was improved about phishing through learning videos and an interactive quiz from the first campaign. One of the participants who get fooled clicked by mistake and the rest couldn't realize if this was the same trap. Figure 6.21 illustrates the step-by-step on how our web-based tool works. Personal data for creating a simulation and investigating who get fooled based on the wrong email that was sent to them were collected. The data as presented above was then analysed. Based on the results, an awareness training programme to employees of the relevant organization or company was conducted. It is worth mentioning that, this tool is very useful in raising awareness about email phishing as detailed above.



**Figure 6.20: A Snapshot of Web-based Tool Steps for Raising Phishing Awareness to Organizational employees**

## 6.4.1 Interview Results

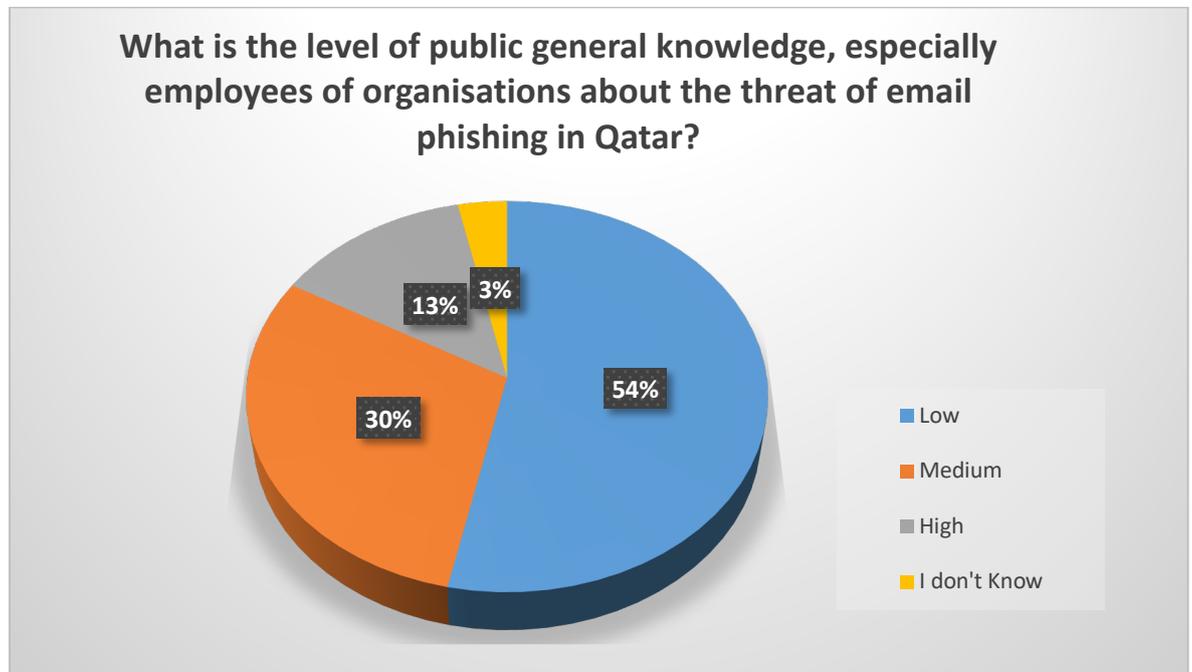
In this subsection, results that were conducted during the interview with different experts from the 154 people that participated in this study are presented. As shown in Figure 6.23 30% of all the participants relying on checking the email box indicates a major challenge while 23.33% indicated lack of resources, time and support to employees on raising an awareness regarding email phishing. 23.33% employees and experts have no knowledge of security awareness as indicated and 23.33% cannot acknowledge that awareness is a unique discipline that needs to be considered. The lack of engaging and appropriate materials is another factor that contributes to not implementing email phishing awareness training programmes in Qatar as indicated by 23.33% of all the participants. However, 6.67% of all the participants show they partly agree with the factor of engaging on checking the email box as the challenge of implementing email awareness training programmes. It is our expectations that, the developed framework and the participation of these numbers of experts will be very beneficial in understanding the overall notion of email phishing awareness in the State of Qatar.



**Figure 6.21: Challenges of implementing Email Phishing Awareness Training Programmes in Qatar**

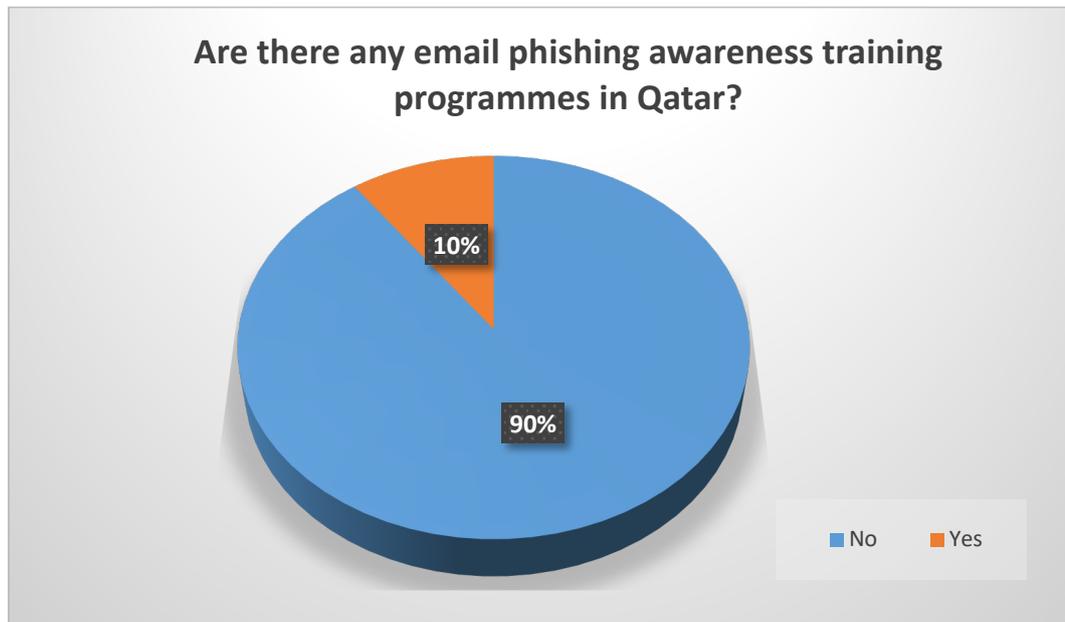
Figure 6.22 indicates the number of participants who acknowledged whether there are any email phishing awareness programmes in Qatar. Out of 154 people under this study, 54% indicated that there was a low level of training programmes while 30% indicated

medium followed by 13% of people who indicated that there is a high level of training programmes in Qatar. It is clear from the figure that no training programmes are being conducted to employees and experts in the State of Qatar regarding email phishing awareness.



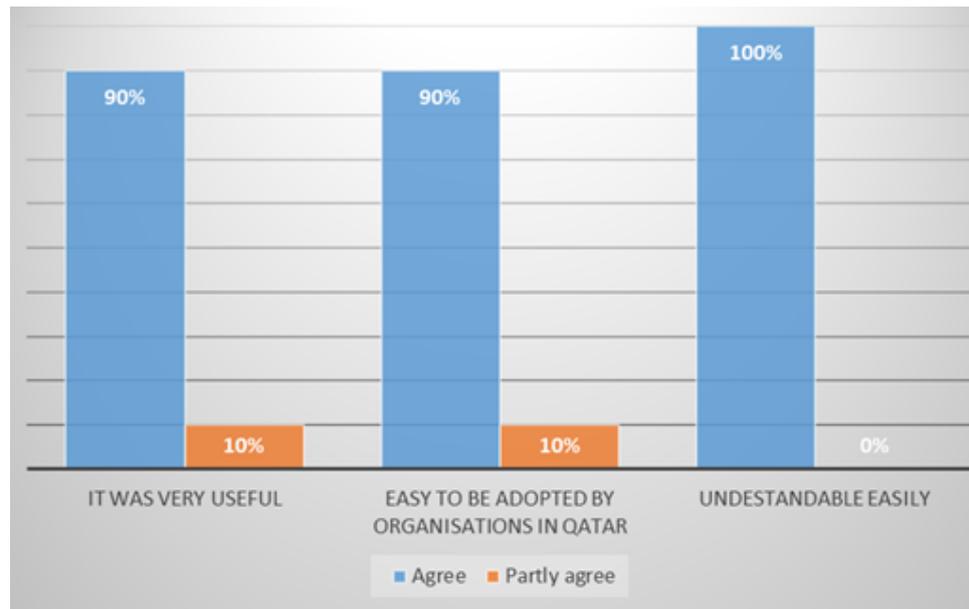
**Figure 6.22: Level of Public Knowledge regarding email phishing**

Figure 6.23 shows responses from participants who were asked to indicate whether there are any email phishing awareness training programmes in Qatar. 90% respondents indicated that there were no awareness training programmes in Qatar while only 10% indicated that there were email phishing programmes. Our work becomes among the first study in Qatar to raise email phishing training programmes which will help employees to understand and know more about email phishing and therefore prevent some of intruders who would be able to do money laundering or fraud in different organizations.



**Figure 6.23: Response on Email Phishing Awareness Training Programmes in Qatar**

Figure 6.24 shows the response from 10 managers who are experts in IT and who evaluated our proposed frameworks. Three different criteria to organizational managers to evaluate the developed platform for raising email phishing awareness whether was very useful, easy to adopt by organizations in Qatar or was understandable easily by non-expert were stipulated. As shown in Figure 5.18, 90% acknowledged that the framework was very useful while 90% agree that, it is easy to be adopted by different organizations in Qatar. All 10 managers (2 from public universities (director and deputy director of IT services), 3 from the oil and gas industry (IT manager, senior security officer and senior network administrator), and 2 from government ICT institution (Chief security officer and director of information system) and , 2 from the hospitality industry (IT manager and chief operation officer) and 1 from a health organization (Director of IT services)) further agreed that, the developed framework is easy to understand while conducting training to employees. It is our expectation that, the developed framework will be useful and of help to the government of Qatar in reducing the risk of email phishing attacks to organization employees. Note that, these are three different aspects that are evaluated separately by 10 managers.



**Figure 6.24 :Evaluation of the proposed email phishing framework for raising awareness**

## 6.4.2 Discussion

Email phishing attacks are increasingly becoming a big concern to people, governmental and non-governmental organization. Web browsers and emails applications have become more sophisticated in detecting email phishing, yet despite all these advances, people who use these services are not aware and there is little in terms of awareness campaigns or research that prevent people from being duped or being victims of phishing emails or websites. Therefore, it is important that people are aware of the existing email phishing threats and the most common techniques that are used by attackers, to be able to protect themselves from those types of phishing attack and to understand how they should all be contributing to ensure security (Flowerday and Tuyikeze, 2016). It has been proven that employees should be trained and become aware of email phishing threats. If this is not achieved, then threat levels will not be contained and any detection system that is meant to secure organizations will always be inadequate (Okeke and Shah, 2016).

## 6.5 Recommendations and Study Impacts

### 6.5.1 Government

The key reasons for conducting this research are based on the fact that email phishing is increasingly becoming more sophisticated and government institutions are the main target of attacks. In Qatar, the rapid increase of using email as the official means of

communications in the government institutions means that the threat of email phishing should not be underestimated. Therefore it is important to know the threat to Qatar in order to propose and recommend training awareness to fit the purposes and recommendations that can help fight the menace of email phishing. The particular importance is to ensure that the general population is aware of the threat of email phishing. In addition, it is envisaged that the outcomes of this study can eventually lead to people being more aware of email phishing threats that will eventually reduce the chances of successful attacks.

### **6.5.2 ICT Sector**

Another key issue that this study hopes to achieve is to determine the level of compliance of institutions and stake holders when it comes to making sure that people are secure. This is why talking to relevant personnel is vital to how data collection is conducted in this study. The outcomes of the study would benefit the companies in Qatar, especially the IT sectors.

### **6.5.3 Institutions**

It is strongly believed that this study targets a very important issue to the government and different institutions in Qatar. One of the institutions is the Qatar news agency that has been hacked recently which had a big effect on the country. The results under this study and the recommendations herein will help every sector in Qatar and even citizens to provide processes that can help in reducing the threat of email phishing which will also change the way of thinking when they receive an email and how to react. The recommendations will help in developing key email phishing awareness training programmes to ensure the reduction of phishing email attacks.

### **6.5.4 Research and Academic Institutions**

There are several researchers investigating the impact of email phishing threats to security and how these threats can be reduced. The investigation also shows that Qatar is facing an enormous email phishing threat, which is a big concern for government and non-governmental organizations and will put them under risk while their employees are not well-educated and trained in security wise. However, the recommendations will be specific and tailored to the needs of Qatar with the peculiar email phishing problems as they relate to the country's needs. The interviews that have been conducted with government and non-government organizations have discussed the major problems in Qatar in awareness and the training that is needed. It is given that if the recommendations

of this research are taken into consideration and implemented it will raise the awareness to the issues and significantly reduce the chance of a successful attack.

## 6.6 Conclusion

By comparing the results from ESCPAD and TrendMicro Email Security for Liverpool John Moores University, we successfully presented that TrendMicro failed to detect an Enterprise Credential SpearPhishing attack that we designed and launched on the Liverpool John Moores University email system. A successful SpearPhishing attack on the Liverpool John Moores University email system could be a catastrophic event as it could lead to credential theft, identity theft, Malware download and Ransomware attack. The attack method proposed in this thesis showed how an enterprise security system like TrendMicro could be vulnerable to SpearPhishing attack.

We continuously sent those emails in 4 months from Oct 2018 on average 10 emails per month, the last test done on 22/01/2019, which clearly show that the TrendMicro intelligence security system is unable even to determine the pattern of our attacks.

By repeating the test, but with different domains on five popular and widely used Email systems and mail servers, we found that 4 out of five are vulnerable to such phishing attacks.

As shown in Figure 6.15, for domain “ljmuac.uk” only the proposed system, ESCPAD was able to detect the Enterprise Credential Spearphishing attack. In addition, we found that the Ministry of Transport and Communications in Qatar detected our attack method when we used .qa domain, but when we changed it to ljmuac.uk, it easily bypassed the detection.

This result has been changed for the second test with “insatgram.com”. In this test only the proposed method and Gmail was able to detect the attack, unfortunately, we could not find what phishing detection mechanism is used by Gmail to detect our phishing attack. However, after doing further tests, we found that Gmail did a content analysis, therefore it classified the first email received as phishing.

As the result shows, ECSPAD performed a good detection in comparison to five standard and widely used email systems (with built-in Phishing Detection Mechanism).

# CHAPTER 7

## CONCLUSION

### 7.1 Overall Conclusion

This thesis has proposed two novel strategies for protection against email phishing threats, in the context of Qatar's wealth, making it an attractive target for phishing attackers. These strategies are aimed at reducing the risk of resources and information being compromised to attackers. One of these is a technical approach called ECSPAD (Enterprise Credential Spearphishing Attack Credential), and the other is a social awareness-based solution, the development of an interactive website designed to train users in Qatar to recognise and defend against spearphishing email attacks. This chapter provides a brief overview of what has been discussed, considering the contributions, and limitations of this research. It also provides suggestions for further research in the area, to clarify anything left unclear by this research.

As email becomes more present in modern life, making communication easy for users and becoming an integral part of business, the incentive for spearphishing increases and attacks have become more common. Globally, awareness is becoming ever more important as spearphishing costs economies huge sums of money. Whilst policies have been introduced by various national bodies to educate users and make them more guarded against spearphishing attacks, the case of Qatar demands further attention. Spearphishing in Qatar has become a growing concern over recent years; Qatar's economic success has made it an attractive pool of phishing victims, while research has suggested that Qatar's email users are more likely to believe spearphishing attacks because they have a greater tendency to be fooled. Resultantly, this thesis takes a double approach, reviewing the literature before presenting a technical as well as a social-awareness based solution.

The research started by presenting a background on phishing defence methods. It first provides information on the ever-evolving nature of phishing attacks, which can trick even knowledgeable users. Then, several research studies on the efficiency of various technical solutions are summarised. By analysing features such as URL and email address, blacklisting can be used to flag up suspicious sites to the user. Then, investigating why financial institutions are the most vulnerable to phishing attacks, providing an

overview of high-profile cases of resource-loss due to spearphishing, in order to highlight the sophistication users must anticipate with phishing attacks. Several research studies into the effectiveness of different technical solutions (e.g. data-mining) are summarised for an insight into the benefits and limits of technical solutions (Xiang et al., 2011, Chen et al., 2011, Jeeva and Rajsingh, 2016, Ramesh et al., 2014, Thomas et al., 2011). Then, we tried to provide a glossary of different phishing attack techniques, summarising research and showing diagrams of how these techniques work. This provides a base of awareness from which to develop defensive solutions, which is the aim we achieved through this research.

The first objective of this research was to research and identify the limitations and lack of existing solutions. This objective has been achieved by conducting a literature review on phishing emails impacts in the state of Qatar and the world in general. The outcomes of this objective are in chapter 2 and 3

We have presented the specific case of Qatar, outlining the necessity and urgency of this research. Several research studies which test the success of phishing detection methods are recounted and summarised. These methods are technical systems which inspect emails in a variety of ways, such as analysing webpage features for evidence of phishing, in order to flag up fraudulent emails. Then, it provides a glossary of phishing defence methods, covering the pros and cons of popular methods and techniques, which are currently in use. The characteristics of spearphishing require a different defence mechanism from, for instance, smishing (SMS phishing). Each method of phishing has its own telltale signs, and this chapter summarises many research studies into the success of different defences. Then, the chapter explores the angle of user awareness as a defence mechanism; it outlines the literature's research into the importance of personality and evaluates the success of various user awareness training programs. User awareness is crucial because a savvy human acting as the first line of defence is less likely to be confounded by updated phishing mechanisms, or new formats. This contrasts with technical solutions, which must be updated constantly to remain efficient.

This research has addressed the initial aims, which were to develop two phishing defence methods, a technical one and an awareness program designed specifically for the needs of Qatar. As mentioned, Qatar presents a unique case because it is highly attractive to spearphishers. Our tests show that spearphishing, due to its highly targeted nature, is

unlikely to be detected by traditional DNS-checking methods, so it requires content-analysis technology coupled with higher user awareness. This thesis' first unique research contribution is the development of ECSPAD. It begins by bringing the conclusions drawn from the research of the previous chapters to bear on the development of this new system. The unique nature of spearphishing demands the adoption of a specific attack taxonomy and threat model, revolving around user-targeted spoofed emails. The chapter presents our original case study, where a sample of 50 LJMU members were tested on their response to a phishing email. The failure of various technical solutions employed at LJMU against spearphishing is explained, and used as a basis for the proposed solution, which is a system that detects Enterprise Credentials Spearphishing, where attackers use a domain name very similar to a legitimate site. The system entails feature extraction, and two algorithms for whitelisting and domain similarity checking. A test we ran of this method is detailed. Then, Domain Keys Identified Mail and Sender Policy Framework are checked as an extra layer of protection. These are all combined with existing checks to provide a more comprehensive defence.

The second objective is to present a new technique for spearphishing targeted attacks taking domain similarity into consideration, this technique is used for Qatar domains which are targeted attacked by the hackers. The main technology companies have yet to include the proposed domain similarity checks because this is left to the existing laws (Passing Off) in developed countries such as in the US and UK on similarity in domain names. If an organisation has a domain name very similar to that owned by another organisation then an organisation can be challenged for passing off and will be forced to change the domain name (Wang, 2006).

The second original contribution is an Email Phishing Training Website, an interface developed to teach users about phishing and learn about user habits. The methodology of constructing the interface is outlined. The chapter presents the flow wireframe for the website, and sequence diagrams for how users interact with the program. It also provides a discussion on the success and limitations of the program. The program is evaluated against the aims, which were to provide training, test, and anonymously collect data.

The third objective was to design and develop an email phishing training awareness

programme to be used by organisations in the state of Qatar, this objective has been achieved by having interviews with different organizations in different sectors, government, IT, research and academic, information technology and institution. After gathering information from each organization we found that there is a lack of awareness in phishing general and specifically targeted attacks, therefore an interactive website has been developed to raise the awareness in organizations in the state of Qatar.

Our research displayed the results of our original study on how well users and email hosts can detect and prevent spearphishing attacks. We spoof an email claiming to be from Instagram, changing one letter as our research showed is common. The results are compared to existing spearphishing defence methods, especially LJMU's Trend Micro, which originally failed to capture our spoofed email. It is also compared to popular web hosts' defence mechanisms. The relative success of ECSPAD is evaluated.

The fourth objective was to develop an enterprise email phishing detection system to be used by organizations and individuals in the state of Qatar. This system was discussed and recommended in the interviews by the organizations in Qatar. The system will allow the employees to be more secure and any similar domain name will not be received by the hacker.

And finally, the last objective is to evaluate the effectiveness of the proposed email phishing awareness programme, applications and the enterprise-wide detection system in the state of Qatar, therefore we have tested and evaluated the proposed solution and shown a high percentage in the result.

In this stage, we have successfully met all our aims and objectives mentioned in chapter one (1.5)

## **7.2 Research Contributions**

This thesis has several unique contributions, which serve to inform future literature and practice in spearphishing defence methods. The contributions of this research can be summarised as follows:

Spearphishing emails are highly targeted, tailor-made attacks. The case of Qatar requires special attention. Because of the economic success of the region, combined with what we

have called a soft touch in Qatari email users, Qatar's residents and businesses are attractive targets for spearphishing attackers. By raising user awareness through training, developing knowledge of user practices through testing and anonymous data collection, and actively seeking to create more sophisticated technical defence mechanisms which go beyond basic Domain Name System (DNS) checks to respond to the sophisticated techniques phishers are using and by taking a novel two-pronged approach, developing both a technical solution and a human awareness-training program, the research serves to address both sides of the problem.

As our research and testing of LJMU's TrendMicro Email Security System showed, it is insufficient to simply rely on checking DNS data when dealing with the growing danger of Enterprise Credential Spearphishing attacks. Over 10 months, their system was not sophisticated enough to detect our test attacks, even though they had a consistent pattern. Several popular email hosts were also unable to detect this type of attack, except Gmail, which we discovered to be employing a content analysis method. This means further methods of email analysis must be developed to go beyond DNS checking and adapt to the changing methods which lead many attackers to success.

In our case, we chose to focus on the Enterprise Credential attack method, which closely spoofs the appearance of legitimate URLs and email addresses when sending targeted attacks. These kinds of attacks could have harsh consequences if key credentials or data are compromised. The ECSPAD succeeded in detecting this type of attack more effectively than the TrendMicro system, and also performed well compared to the detection systems of popular email hosts.

The awareness training program that we designed using HTML and JavaScript succeeded in providing a stimulating and informative user experience, which operated technically without any problems. The Model-View-Controller architecture employed provided a sustainable codebase, and by using Drupal and a central MySQL database allowed for users to learn from their mistakes by comparing their answers to the real ones. The program allows for anonymous data collection, which aids in further understanding the scope of user behaviour and knowledge, and may be used as the basis of further research. This data is presented in a user-friendly way with charts and graphs.

### 7.3 Difficulties and solutions

This thesis has addressed several difficulties which often prevent the success of phishing deflection approaches. Firstly, that popular web hosts are failing to protect against phishing attacks, because they are not responding to the sophistication of targeted spearphishing attackers, who use techniques such as Enterprise Credential Spearphishing to slip past DNS checkers and fool users. Although we address only one such method, our results show that by specifically targeting and taking into account the sophisticated techniques phishers are using, we can increase the success of our detection methods. The second problem that prevention schemes face is that they are not backed up by user-awareness training; humans are the first line of defence when it comes to detecting false material. If more employees and citizens, especially in the case of Qatar, were wary and alert to emails asking for their credentials, this could combine well with technical solutions to reduce incidences of phishing attacks. In our study, we were limited by the time frame as the datasets generated by the anonymous data collection element of the program can be used to further expand our knowledge and make awareness programs more comprehensive and widespread.

Lack of understanding on how end users learn can lead to ineffective email phishing awareness training programmes. In this study we have taken the effectiveness of the training by measuring who completed training, test results and feedback. However, we did not take into consideration end users' learning experience in the awareness training programme. This means the one-size fits all delivery strategy was applied. Awareness training programme needs to be personalized for end users to align with their time availability, leaning needs, generation, interests and their literacy in technology. Answers to the following questions can lead to effective awareness training programmes.

- What parts of awareness training programme are most effective and least effective?
- What parts of awareness training programme most engage end users and what parts disconnect them?

## 7.4 Future work

The glossaries of phishing attacks, different models of attack taxonomies, and prevention systems should provide a basis for the development of further technical tools to be used in combination with what has been developed already so that the technology of fighting cybercrime may keep pace with its perpetrators. We have shown that by focusing on one element of Spearphishing emails it is possible to attain a good rate of detection compared with commonly used defence systems. DNS checkers cannot suffice as the Internet-Of-Things advances and much of our global economy exists in the online sphere. There must be more development in research in this field, so that collective knowledge may combine to advance the field of phishing detection in general and create a safer world. By focusing on specific attack types, such as typosquatting (Banerjee et al., 2008) mobile phone phishing (Tufts, 2012) or tabnapping (Raskin, 2010), we can move beyond generalised checks which more often than not allow more specialised attacks to slip through.

When we conducted our Instagram spearphishing tests, summarised above, the only popular web host who successfully detected and prevented the spoofed email was Gmail, and we researched this to discover that Gmail employs a content analysis method. Unlike LJMU's defence methods, which only check DNS, content analysis was able to prevent users from receiving a malicious email. Other research has also shown that a combined approach which analyses web elements and features such as URL as well as DNS, is more successful at preventing spearphishing attacks; Li et al. (2015) used a hybrid system, employing imaging techniques for detection. They addressed one particular telltale sign of phishing, and were able to increase success. The ECSPAD addresses the technique of typosquatting (Banerjee et al., 2008) where phishers use addresses and URLs which closely resemble legitimate websites; we created an algorithm which checks specifically for this trick, and managed to double the prevention rates of most email providers in our study. Other techniques need to be developed which address specific elements of phishing emails, such as use of language or generic openings

As suggested, the datasets generated during the testing process for the user-awareness training program may form the basis for future development of user-awareness. By implementing this program and others like it, using the development methodology laid out in this thesis as a blueprint, it is possible for organisations to conduct their own training awareness schemes and advance the literature forward into an age of greater user-

savviness and lower incidences of credentials, confidential information and resources being compromised through successful spearphishing attacks. The summarised literature and glossaries provided in this thesis provide a comprehensive overview of the fundamentals as well as the latest in phishing attack and defence methods, which may be used to create further informational material, and even provide more advanced training to those in high positions who would stand to compromise crucial company information if they were attacked, i.e. attractive victims for whaling.

The user awareness training method we used, an interactive website, is just one method of increasing user awareness in an engaging way whilst anonymously collecting data on user knowledge and awareness. Other methods include interactive animations which have an educational design, games where the user encounters simulated phishing situations, and informational videos designed in such a way that they keep the attention of the viewer. All of these methods have the potential to reach different learners. Ideally, a combination of all or some of these methods should be pursued to guarantee greater user awareness.

## REFERENCES

- Abraham, D. and Raj, N.S., 2014, September. Approximate string-matching algorithm for phishing detection. *In Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on)* (pp. 2285-2290). IEEE.
- Abu-Nimeh, S., Nappa, D., Wang, X. and Nair, S., 2007, October. A comparison of machine learning techniques for phishing detection. *In Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit* (pp. 60-69). ACM.
- Abutair, H. Y. A., & Belghith, A., 2017. Using Case-Based Reasoning for Phishing Detection. *Procedia Computer Science*, 109, 281–288. doi:10.1016/j.procs.2017.05.352
- Adebowale, M. A., Lwin, K. T., Sánchez, E., & Hossain, M. A., 2019. Intelligent web-phishing detection and protection scheme using integrated features of Images, frames and text. *Expert Systems with Applications*, 115(December 2017), 300–313. doi:10.1016/j.eswa.2018.07.067
- Akhawe, D. and Felt, A.P., 2013, August. Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness. *In USENIX security symposium* (Vol. 13).
- Akhawe, D., He, W., Li, Z., Moazzezi, R. and Song, D., 2014. Clickjacking Revisited: A Perceptual View of {UI} Security. *In 8th {USENIX} Workshop on Offensive Technologies ({WOOT} 14)*.
- Aldawood, H. and Skinner, G., 2018, December. Educating and Raising Awareness on Cyber Security Social Engineering: A Literature Review. *In 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)* (pp. 62-68).

IEEE.

Al-Hamar, M. and Khalid, M., 2010. "Reducing the risk of e-mail phishing in the state of Qatar through an effective awareness framework", *Thesis submitted at Loughborough University, UK, 2010.*

Al-Hamar, M., Dawson, R. and Guan, L., 2010, June. A culture of trust threatens security and privacy in Qatar. *In 2010 10th IEEE International Conference on Computer and Information Technology* (pp. 991-995). IEEE.

Al-Hamar, M., Dawson, R. and Al-Hamar, J., 2011. The need for education on phishing: a survey comparison of the UK and Qatar. *Campus-Wide Information Systems*, 28(5), pp.308-319.

Al-Mohannadi, H., Awan, I., Al Hamar, J., Al Hamar, Y., Shah, M. and Musa, A., 2018, August. Understanding Awareness of Cyber Security Threat Among IT Employees. *In 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)* (pp. 188-192). IEEE.

Aleroud, A., & Zhou, L., 2017. Phishing environments, techniques, and countermeasures: A survey. *Computers and Security*, 68, 160–196. doi:10.1016/j.cose.2017.04.006

Almomani, A., Gupta, B.B., Atawneh, S., Meulenberg, A. and Almomani, E., 2013. A survey of phishing email filtering techniques. *IEEE communications surveys & tutorials*, 15(4), pp.2070-2090.

Alomari, E., Manickam, S., Gupta, B.B., Singh, P. and Anbar, M., 2014, February. Design, deployment and use of HTTP-based botnet (HBB) testbed. *In Advanced Communication Technology (ICACT), 2014 16th International Conference on* (pp. 1265-1269). IEEE.

Alotaibi, M., Furnell, S. and Clarke, N., 2016, December. Information security policies: a review of challenges and influencing factors. *In 2016 11th International Conference for Internet Technology and Secured Transactions (ICITST)* (pp. 352-358). IEEE.

Alsharnouby, M., Alaca, F. and Chiasson, S., 2015. Why phishing still works: User strategies for combating phishing attacks. *International Journal of Human-Computer Studies*, 82, pp.69-82.

APWG, 2014. Phishing Activity Trends Report (4th Quarter 2014). Unifying the Global Response To Cybercrime. [online] APWG. Available at: [http://docs.apwg.org/reports/apwg\\_trends\\_report\\_q4\\_2014.pdf](http://docs.apwg.org/reports/apwg_trends_report_q4_2014.pdf). [Accessed: 10th April 2019]

APWG, 2015. Phishing Activity Trends Report (4th Quarter 2015). Unifying the Global Response To Cybercrime. [online] APWG. Available at: [http://docs.apwg.org/reports/apwg\\_trends\\_report\\_q4\\_2015.pdf](http://docs.apwg.org/reports/apwg_trends_report_q4_2015.pdf). [Accessed: 10th April 2019]

Arachchilage, N.A.G. and Love, S., 2013. A game design framework for avoiding phishing attacks. *Computers in Human Behavior*, 29(3), pp.706-714.

Arachchilage, N.A.G., Love, S. and Beznosov, K., 2016. Phishing threat avoidance behaviour: An empirical investigation. *Computers in Human Behavior*, 60, pp.185-197.

Banday, M.T. and Qadri, J.A., 2011. Phishing-A growing threat to e-commerce. arXiv preprint arXiv:1112.5732.

Banerjee, A., Barman, D., Faloutsos, M. and Bhuyan, L.N., 2008, April. Cyber-fraud is one typo away. In *INFOCOM 2008. The 27th Conference on Computer Communications*. IEEE (pp. 1939-1947). IEEE

Basnet, R.B. and Doleck, T., 2015, February. Towards developing a tool to detect phishing URLs: a machine learning approach. In *2015 IEEE International Conference on Computational Intelligence & Communication Technology* (pp. 220-223). IEEE.

Basnet, R., Mukkamala, S. and Sung, A.H., 2008. Detection of phishing attacks: A machine learning approach. In *Soft Computing Applications in Industry* (pp. 373-383). Springer, Berlin, Heidelberg.

Bursztein, E. and Eranti, V., 2013. Internet-wide efforts to fight email phishing are working. Google Security Blog, [Online]. Available at <https://security.googleblog.com/2013/12/internet-wide-efforts-to-fight-email.html>.

[Accessed: 5th May 2019]

Caputo, D.D., Pfleeger, S.L., Freeman, J.D. and Johnson, M.E., 2014. Going spear phishing: Exploring embedded training and awareness. *IEEE Security & Privacy*, 12(1), pp.28-38.

Chandrasekaran, M., Narayanan, K. and Upadhyaya, S., 2006, June. Phishing email detection based on structural properties. *In NYS cyber security conference* (Vol. 3).

Chaudhry, J.A., Chaudhry, S.A. and Rittenhouse, R.G., 2016. Phishing attacks and defenses. *International Journal of Security and Its Applications*, 10(1), pp.247-256.

Chen, C.M., Huang, J.J. and Ou, Y.H., 2015. Efficient suspicious URL filtering based on reputation. *Journal of Information Security and Applications*, 20, pp.26-36.

Chen, X., Bose, I., Leung, A.C.M. and Guo, C., 2011. Assessing the severity of phishing attacks: A hybrid data mining approach. *Decision Support Systems*, 50(4), pp.662-672.

Chhabra, S., Aggarwal, A., Benevenuto, F. and Kumaraguru, P., 2011, September. Phish/\$ocial: the phishing landscape through short urls. *In Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference* (pp. 92-101). ACM.

Chiew, K. L., Yong, K. S. C., & Tan, C. L. 2018. A survey of phishing attacks: Their types, vectors and technical approaches. *Expert Systems with Applications*, 106, 1–20. doi:10.1016/j.eswa.2018.03.050

Choi, Y.S., He, S., Lee, G.M., Leung, C.M., Whinston, A. and Zhuang, Y., 2017, December. A Pan-Asian Field Experiment of Organizational Awareness to Information Security and Preparedness Against Cybercrime. *In The Workshop on Information Technologies and Systems* (WITS 2017).

Cova, M., Kruegel, C. and Vigna, G., 2010, April. Detection and analysis of drive-by-download attacks and malicious JavaScript code. *In Proceedings of the 19th international conference on World wide web* (pp. 281-290). ACM.

Cruz, A. ed., 2016. *Digital and Social Media Marketing-a Results-driven Approach*. Taylor & Francis Limited.

Curtis, S. R., Rajivan, P., Jones, D. N., & Gonzalez, C. (2018). Phishing attempts among the dark triad: Patterns of attack and vulnerability. *Computers in Human Behavior*, 87(May), 174–182. doi:10.1016/j.chb.2018.05.037

Da Veiga, A. and Eloff, J.H., 2010. A framework and assessment instrument for information security culture. *Computers & Security*, 29(2), pp.196-207.

Davinson, N. and Sillence, E., 2010. It won't happen to me: Promoting secure behaviour among internet users. *Computers in Human Behavior*, 26(6), pp.1739-1747.

De Kimpe, L., Walrave, M., Hardyns, W., Pauwels, L., & Ponnet, K. (2018). You've got mail! Explaining individual differences in becoming a phishing target. *Telematics and Informatics*, 35(5), 1277–1287. doi:10.1016/j.tele.2018.02.009

Delany, S.J., Buckley, M. and Greene, D., 2012. SMS spam filtering: methods and data. *Expert Systems with Applications*, 39(10), pp.9899-9908.

Dewan, P., Kashyap, A. and Kumaraguru, P., 2014, September. Analyzing social and stylometric features to identify spear phishing emails. *In Electronic Crime Research (eCrime), 2014 APWG Symposium on* (pp. 1-13). IEEE.

Dhamija, R., Tygar, J.D. and Hearst, M., 2006, April. Why phishing works. *In Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp. 581-590). ACM.

DigiCert (2009). Phishing: A primer on what phishing is and how it works. [Online] Available at [https://www.digicert.com/news/DigiCert\\_Phishing\\_White\\_Paper.pdf](https://www.digicert.com/news/DigiCert_Phishing_White_Paper.pdf) [Accessed: 10<sup>th</sup> April 2019]

Dormann, W., & Manion, A., 2004. Vulnerability note vu#490708: Microsoft internet explorer window.createpopup() method creates chromeless windows, [Online] Available at <https://www.kb.cert.org/vuls/id/490708> [Accessed on: Apr. 19, 2017].

Dormann, W., 2005. Vulnerability note vu#356600: Microsoft internet explorer dhtml editing activex control contains a cross-domain vulnerability. [Online] available at, <https://www.kb.cert.org/vuls/id/356600> [Accessed on: Apr. 19, 2017]

Downs, J.S., Holbrook, M.B. and Cranor, L.F., 2006, July. Decision strategies and susceptibility to phishing. *In Proceedings of the second symposium on Usable privacy and security* (pp. 79-90). ACM.

Elledge, A., 2007. Phishing: An analysis of a growing threat. Sans Institute. Retrieved February, 2, p.2016.

Emigh, A., 2005. Online identity theft: Phishing technology, chokepoints and countermeasures. *ITTC Report on Online Identity Theft Technology and Countermeasures*, 1–58.

FBI, 2017. Business e-mail compromise: Cyber-enabled financial fraud on the rise globally. [Online] Available at <https://www.fbi.gov/news/stories/business-e-mail-compromise-on-the-rise> [Accessed on: May 12, 2017].

Felt, A.P. and Wagner, D., 2011. Phishing on mobile devices. *In Proceedings of the w2sp'11: Web 2.0 security and privacy*.

Felten, E.W. and Schneider, M.A., 2000, November. Timing attacks on web privacy. *In Proceedings of the 7th ACM conference on Computer and communications security* (pp. 25-32). ACM.

Flowerday, S.V. and Tuyikeze, T., 2016. Information security policy development and implementation: The what, how and who. *computers & security*, 61, pp.169-183.

Fette, I., Sadeh, N. and Tomasic, A., 2007, May. Learning to detect phishing emails. *In Proceedings of the 16th international conference on World Wide Web* (pp. 649-656). ACM.

Foozy, M., Feresia, C., Ahmad, R. and Abdollah, M.F., 2014. A practical rule based technique by splitting SMS phishing from SMS spam for better accuracy in mobile device. *International Review on Computers and Software*, 9(10), pp.1776-1782.

Fu, A.Y., Wenyin, L. and Deng, X., 2006. Detecting phishing web pages with visual similarity assessment based on earth mover's distance (EMD). *IEEE transactions on dependable and secure computing*, 3(4), pp.301-311.

Fultz, N. and Grossklags, J., 2009, February. Blue versus red: Towards a model of distributed security attacks. *In International Conference on Financial Cryptography and Data Security* (pp. 167-183). Springer, Berlin, Heidelberg.

Gascon, H., Ullrich, S., Stritter, B. and Rieck, K., 2018, September. Reading between the lines: content-agnostic detection of spear-phishing emails. *In International Symposium on Research in Attacks, Intrusions, and Defenses*(pp. 69-91). Springer, Cham.

Gelernter, N. and Herzberg, A., 2016, April. Tell me about yourself: The malicious captcha attack. *In Proceedings of the 25th International Conference on World Wide Web* (pp. 999-1008)..

Gibbs, S. 2016. Dropbox hack leads to leaking of 68m user passwords on the inter-net. [Online] Available at <https://www.theguardian.com/technology/2016/aug/31/dropbox-hack-passwords-68m-data-breach> [Accessed on: Apr. 21, 2017]

Gibb, R. 2015. What is a transparent proxy? [Online] Available at: <https://>

www.maxcdn.com/one/visual-glossary/transparent-proxy [Accessed Apr. 28, 2017]

Goel, D., & Jain, A. K. 2018. Mobile phishing attacks and defence mechanisms: State of art and open research challenges. *Computers and Security*, 73, 519–544. doi:10.1016/j.cose.2017.12.006

Gruschka, N. and Iacono, L.L., 2009, July. Vulnerable cloud: Soap message security validation revisited. In *Web Services, 2009. ICWS 2009. IEEE International Conference on* (pp. 625-631). IEEE.

Gruschka, N. and Jensen, M., 2010, July. Attack surfaces: A taxonomy for attacks on cloud services. In *2010 IEEE 3rd international conference on cloud computing* (pp. 276-279). IEEE.

Gudkova, D., Vergelis, M., Demidova, N. and Shcherbakova, T., 2016. *Spam and Phishing in Q2 2016*. Kaspersky Lab, 18.

Gupta, B.B., Tewari, A., Jain, A.K. and Agrawal, D.P., 2017. Fighting against phishing attacks: state of the art and future challenges. *Neural Computing and Applications*, 28(12), pp.3629-3654.

Halevi, T., Memon, N. and Nov, O., 2015. Spear-phishing in the wild: A real-world study of personality, phishing self-efficacy and vulnerability to spear-phishing attacks. *Phishing Self-Efficacy and Vulnerability to Spear-Phishing Attacks* (January 2, 2015).

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I.H., 2009. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1), pp.10-18.

Hayashi, N. 2014. New phishing technique outfoxes site owners: Operation Huyao. [Online] Available at <http://blog.trendmicro.com/trendlabs-security-intelligence/new-phishing-technique-outfoxes-site-owners-operation-huyao/> [Accessed 27<sup>th</sup> March 2017]

He, M., Horng, S.J., Fan, P., Khan, M.K., Run, R.S., Lai, J.L., Chen, R.J. and Sutanto, A., 2011. An efficient phishing webpage detector. *Expert systems with applications*, 38(10), pp.12018-12027.

Heartfield, R. and Loukas, G., 2016. A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks. *ACM Computing Surveys (CSUR)*, 48(3), p.37.

Henderson, N. 2017. Hackers target docuSign, BT customers with phishing emails. [Online] Available at Hackers target docuSign, BT customers with phishing emails [Accessed 29<sup>th</sup> May 2017]

Ho, G., Sharma, A., Javed, M., Paxson, V. and Wagner, D., 2017. Detecting credential spearphishing in enterprise settings. In *26th {USENIX} Security Symposium ( {USENIX} Security 17)* (pp. 469-485).

Hong, J., 2012. The state of phishing attacks. *Communications of the ACM*, 55(1), pp.74-81.

Horváth, G., Suykens, J., Basu, S., Micchelli, C., Vandewalle, J., 2003. Advances in learning theory: Methods, models and applications. NATO science series III: *Computer and systems sciences*, vol. 190. Amsterdam: IOS Press.

Huang, C.Y., Ma, S.P. and Chen, K.T., 2011. Using one-time passwords to prevent password phishing attacks. *Journal of Network and Computer Applications*, 34(4), pp.1292-1301

Huang, L.S., Moshchuk, A., Wang, H.J., Schechter, S. and Jackson, C., 2012, August. Clickjacking: Attacks and Defenses. In *USENIX Security Symposium* (pp. 413-428).

Islam, R. and Abawajy, J., 2013. A multi-tier phishing detection and filtering approach. *Journal of Network and Computer Applications*, 36(1), pp.324-335.

J2 Global, I. 2017. How eFax works. [Online] Available at: <https://www.efax.com/how->

it- works. [Accessed Apr. 13, 2017]

Jackson, C., Simon, D.R., Tan, D.S. and Barth, A., 2007, February. An evaluation of extended validation and picture-in-picture phishing attacks. *In International Conference on Financial Cryptography and Data Security* (pp. 281-293). Springer, Berlin, Heidelberg.

Jagatic, T.N., Johnson, N.A., Jakobsson, M. and Menczer, F., 2007. Social phishing. *Communications of the ACM*, 50(10), pp.94-100.

Jain, A.K. and Gupta, B.B., 2016. A novel approach to protect against phishing attacks at client side using auto-updated white-list. *EURASIP Journal on Information Security*, 2016(1), p.9.

Jakobsson, M., 2018. Two-factor in authentication—the rise in SMS phishing attacks. *Computer Fraud & Security*, 2018(6), pp.6-8.

Jakobsson, M. and Ratkiewicz, J., 2006, May. Designing ethical phishing experiments: a study of (ROT13) rOnl query features. *In Proceedings of the 15th international conference on World Wide Web* (pp. 513-522). ACM.

James, L., 2005. *Phishing exposed*. Elsevier.

James, J., Sandhya, L. and Thomas, C., 2013, December. Detection of phishing URLs using machine learning techniques. *In Control Communication and Computing (ICCC), 2013 International Conference on* (pp. 304-309). IEEE.

Jeeva, S.C. and Rajsingh, E.B., 2016. Intelligent phishing url detection using association rule mining. *Human-centric Computing and Information Sciences*, 6(1), p.10.

Jiang, X. and Zhou, Y., 2012, May. Dissecting android malware: Characterization and evolution. *In 2012 IEEE Symposium on Security and Privacy* (pp. 95-109). IEEE.

Kals, S., Kirda, E., Kruegel, C. and Jovanovic, N., 2006, May. Secubat: a web vulnerability scanner. *In Proceedings of the 15th international conference on World Wide Web* (pp. 247-256). ACM.

Kamat, P. and Gautam, A.S., 2018. Recent Trends in the Era of Cybercrime and the Measures to Control Them. *In Handbook of e-Business Security* (pp. 243-258). Auerbach Publications.

Khonji.org, 2019. *Anti-Phishing Studies*. [online] Available at: [http://khonji.org/phishing\\_studies.html](http://khonji.org/phishing_studies.html) [Accessed 12 Feb. 2019].

Konradt, C., Schilling, A. and Werners, B., 2016. Phishing: An economic analysis of cybercrime perpetrators. *Computers & Security*, 58, pp.39-46.

Kovacs, E. 2016. *Linux trojan takes screenshots every 30 seconds*. [Online] Available at: <http://www.securityweek.com/linux-trojan-takes-screenshots-every-30-seconds> [Accessed 26 April 2017],

Kovacs, E., 2017. Amnesty warns of phishing attacks on Qatar activists. [Online] (updated 15 Feb. 2017) Available at: <https://www.securityweek.com/amnesty-warns-phishing-attacks-qatar-activists> [Accessed 30 Jun.2018]

Krombholz, K., Hobel, H., Huber, M. and Weippl, E., 2015. Advanced social engineering attacks. *Journal of Information Security and applications*, 22, pp.113-122.

LaForge, A. 2016. Flash and chrome. [Online] Available at: <https://blog.google/products/chrome/flash-and-chrome/> [Accessed 19 April 2017]

Kuosmanen, V. 2017. *Browser autofill phishing*. [Online] Available at: <https://www.kuusmanen.fi/2017/05/15/browser-autofill-phishing/>

[//github.com/anttiviljami/browser-autofill-phishing](https://github.com/anttiviljami/browser-autofill-phishing) [Accessed 19 April 2017]

Landwehr, C.E., Bull, A.R., McDermott, J.P. and Choi, W.S., 1994. A taxonomy of computer program security flaws. *ACM Computing Surveys (CSUR)*, 26(3), pp.211-254.

Laszka, A., Lou, J., and Vorobeychik, Y., 2016. Multi-defender strategic filtering against spear-phishing attacks. Published in: *AAAI'16 Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (pp. 537-543)

Lee, J.L., Kim, D.H. and Chang-Hoon, L., 2015. Heuristic-based approach for phishing site detection using url features. *In Proc. of the Third Intl. Conf. on Advances in Computing, Electronics and Electrical Technology-CEET* (pp. 131-135).

Li, Y., Chu, S. and Xiao, R., 2015. A pharming attack hybrid detection model based on IP addresses and web content. *Optik-International Journal for Light and Electron Optics*, 126(2), pp.234-239.

Li, Y., Xiao, R., Feng, J. and Zhao, L., 2013. A semi-supervised learning approach for detection of phishing webpages. *Optik-International Journal for Light and Electron Optics*, 124(23), pp.6027-6033.

Li, Y., Yang, L. and Ding, J., 2016. A minimum enclosing ball-based support vector machine approach for detection of phishing websites. *Optik-International Journal for Light and Electron Optics*, 127(1), pp.345-351.

Lin, C.H., Tien, C.W., Chen, C.W., Tien, C.W. and Pao, H.K., 2015, September. Efficient spear-phishing threat detection using hypervisor monitor. *In Security Technology (ICCST), 2015 International Carnahan Conference on* (pp. 299-303). IEEE.

Lininger, R. and Dean, R., 2005. Phishing: Cutting the Identity Theft Line. *Wiley, publishing Inc.* Indianapolis, Indiana, USA.

Ma, Q., Johnston, A.C. and Pearson, J.M., 2008. Information security management objectives and practices: a parsimonious framework. *Information Management & Computer Security*, 16(3), pp.251-270.

Ma, L., Ofoghi, B., Watters, P. and Brown, S., 2009, July. Detecting phishing emails using hybrid features. *In Ubiquitous, Autonomic and Trusted Computing, 2009. UIC-ATC'09. Symposia and Workshops on* (pp. 493-497). IEEE.

Mahemoff, M. 2009. *Explaining the “don’t click” clickjacking tweetbomb*. [Online] Available at : <http://softwareas.com/explaining-the-dont-click-clickjacking-tweetbomb/> [Accessed 22 May 2017],

Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A., & Liang, Z. 2018. Detecting Phishing Websites via Aggregation Analysis of Page Layouts. *Procedia Computer Science*, 129, 224–230. doi:10.1016/j.procs.2018.03.053

Marxism. 2016. Instant messaging: Why it is so popular? [Online] Available at: <http://www.ict-pulse.com/2016/02/instant-messaging-popular> [Accessed 17 April 2017]

McElwee, S., Murphy, G. and Shelton, P., 2018, April. Influencing Outcomes and Behaviors in Simulated Phishing Exercises. *In SoutheastCon 2018* (pp. 1-6). IEEE.

McIntosh, M. and Austel, P., 2005, November. XML signature element wrapping attacks and countermeasures. *In Proceedings of the 2005 workshop on Secure web services* (pp. 20-27). ACM.

Medvet, E., Kirda, E. and Kruegel, C., 2008, September. Visual-similarity-based phishing detection. *In Proceedings of the 4th international conference on Security and privacy in communication networks* (p. 22). ACM.

Microsoft, 2005. Anti-phishing white paper. [Online] Available at: [http://www-pc.uni-regensburg.de/systemsw/ie70/Anti-phishing\\_White\\_Paper.doc](http://www-pc.uni-regensburg.de/systemsw/ie70/Anti-phishing_White_Paper.doc) [Accessed 17 April 2017]

Millettary, J., 2013. Technical trends in phishing attacks. *United State Computer Emer-*

*gency Readiness Team (US-CERT)* , 1–17 .

Mishra, A. and Gupta, B.B., 2014, August. Hybrid solution to detect and filter zero-day phishing attacks. *In Proceedings of the Second International Conference on Emerging Research in Computing, Information, Communication and Applications* (pp. 373-379).

Mitnick, K.D. and Simon, W.L., 2011. *The art of deception: Controlling the human element of security*. John Wiley & Sons.

Mohammad, R.M., Thabtah, F. and McCluskey, L., 2015. Tutorial and critical analysis of phishing websites methods. *Computer Science Review*, 17, pp.1-24.

Moreno, M., 2016. Malware as a service: As easy as it gets. [Online] Available at: <https://www.webroot.com/blog/2016/03/31/malware-service-easy-gets/> [Accessed May 2017]

Mouton, F., Leenen, L. and Venter, H.S., 2016. Social engineering attack examples, templates and scenarios. *Computers & Security*, 59, pp.186-209.

Nagunwa, T., 2014. Behind identity theft and fraud in cyberspace: the current landscape of phishing vectors. *International Journal of Cyber-Security and Digital Forensics (IJCSDF)*, 3(1), pp.72-83.

Nikiforakis, N., Balduzzi, M., Desmet, L., Piessens, F. and Joosen, W., 2014, October. Soundsquatting: Uncovering the use of homophones in domain squatting. *In International Conference on Information Security* (pp. 291-308). Springer, Cham.

Niu, Y., Hsu, F. and Chen, H., 2008, April. *iPhish: Phishing Vulnerabilities on Consumer Electronics*. In UPSEC.

Okeke, R. and Shah, M., 2016. *Information Theft Prevention: Theory and Practice*. Routledge.

Ollmann, G., 2004. *The Phishing Guide—Understanding & Preventing Phishing Attacks*. NGS Software Insight Security Research.

Pan, Y. and Ding, X., 2006, December. Anomaly based web phishing page detection. In *null* (pp. 381-392). IEEE.

PandaLabs (2015). PandaLabs' annual report 2015. Technical Report . Panda Security .  
Patil, D. R. , & Patil, J. B. (2015). Survey on malicious web pages detection techniques. *International Journal of u- and e- Service, Science and Technology.*, 8 (5), 195–206 .

Park, W., Kim, S.J. and Ryu, W., 2015, October. Detecting malware with similarity to Android applications. In *Information and Communication Technology Convergence (ICTC), 2015 International Conference on* (pp. 1249-1251). IEEE.

Parsons, K., McCormac, A., Pattinson, M., Butavicius, M. and Jerram, C., 2015. The design of phishing studies: Challenges for researchers. *Computers & Security*, 52, pp.194-206

Patil, D.R. and Patil, J.B., 2015. Survey on malicious web pages detection techniques. *International Journal of U-and E-service, Science and Technology*, 8(5), pp.195-206.

PhishLabs, 2017. *2017 phishing trends & intelligence –Hacking the human. Technical Report*. PhishLabs.

Qabajeh, I., Thabtah, F., & Chiclana, F., 2018. A recent review of conventional vs. automated cybersecurity anti-phishing techniques. *Computer Science Review*, 29, 44–55. doi:10.1016/j.cosrev.2018.05.003

Rader, M. and Rahman, S., 2015. Exploring historical and emerging phishing techniques and mitigating the associated security risks. *arXiv preprint arXiv:1512.00082*.

Ramesh, G., Krishnamurthi, I. and Kumar, K.S.S., 2014. An efficacious method for detecting phishing webpages through target domain identification. *Decision Support Systems*, 61, pp.12-22.

RapidMiner, 2019. Lightning Fast Data Science Platform for Teams | RapidMiner©. [online] Available at: <https://rapidminer.com> [Accessed 12 Feb. 2019].

Rashidi, B., Fung, C., & Bertino, E., 2017. Android resource usage risk assessment using hidden Markov model and online learning. *Computers & Security*, 65, 90–107. doi:10.1016/j.cose.2016.11.006

Raskin, A., 2010. Tabnabbing: A new type of phishing attack. [Online] Available at: <http://www.azarask.in/blog/post/a-new-type-of-phishing-attack/> [Accessed 11 May 2017]

Rietta, F.S., 2006, March. Application layer intrusion detection for SQL injection. *In Proceedings of the 44th annual Southeast regional conference* (pp. 531-536). ACM.

Roopak, S. and Thomas, T., 2014. A Novel Phishing Page Detection Mechanism Using HTML Source Code Comparison and Cosine Similarity. *2014 Fourth International Conference on Advances in Computing and Communications*. IEEE

RSA, 2016. *A decade of phishing. Technical Report*. RSA FraudAction Intelligence

Ruderman, J., 2016. *Same-origin policy*. [Online] Available at: [https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin\\_policy](https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy) [Accessed 25 April 2017]

Rydstedt, G., Gourdin, B., Bursztein, E. and Boneh, D., 2010, August. Framing attacks on smart phones and dumb routers: tap-jacking and geo-localization attacks. *In Proceedings of the 4th USENIX conference on Offensive technologies* (pp. 1-8). USENIX Association.

Saad, R.M., Almomani, A., Altaher, A., Gupta, B.B. and Manickam, S., 2014. ICMPv6 flood attack detection using DENFIS algorithms. *Indian Journal of Science and Technology*, 7(2), pp.168-173.

Salem, O., Hossain, A. and Kamala, M., 2010, June. Awareness program and AI based tool to reduce risk of phishing attacks. *In Computer and Information Technology (CIT)*,

2010 IEEE 10th International Conference on (pp. 1418-1423). IEEE.

Sanglerdsinlapachai, N. and Rungsawang, A., 2010, January. Using domain top-page similarity feature in machine learning-based web phishing detection. *In Knowledge Discovery and Data Mining, 2010. WKDD'10. Third International Conference on* (pp. 187-190). IEEE.

Schiller, C. and Binkley, J.R., 2011. *Botnets: The killer web applications*. Elsevier.

Schneider, F., Provos, N., Moll, R., Chew, M. and Rakowski, B., 2007. Phishing protection design documentation. [Online] Available at: [https://wiki.mozilla.org/Phishing\\_Protection:\\_Design\\_Documentation](https://wiki.mozilla.org/Phishing_Protection:_Design_Documentation) [Accessed 20 May 2018]

Sharifi, M. and Siadati, S.H., 2008, March. A phishing sites blacklist generator. *In Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on* (pp. 840-843). IEEE.

Sheng, S., Holbrook, M., Kumaraguru, P., Cranor, L.F. and Downs, J., 2010, April. Who falls for phish?: a demographic analysis of phishing susceptibility and effectiveness of interventions. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 373-382). ACM.

Sheng, S., Magnien, B., Kumaraguru, P., Acquisti, A., Cranor, L.F., Hong, J. and Nunge, E., 2007, July. Anti-phishing phil: the design and evaluation of a game that teaches people not to fall for phish. *In Proceedings of the 3rd symposium on Usable privacy and security* (pp. 88-99). ACM.

Sheng, S., Wardman, B., Warner, G., Cranor, L., Hong, J. and Zhang, C., 2009. An empirical analysis of phishing blacklists.

Silic, M. and Back, A., 2016. The dark side of social networking sites: Understanding phishing risks. *Computers in Human Behavior*, 60, pp.35-43.

Singh, N.P., 1970. Online frauds in banks with phishing. *The Journal of Internet Banking and Commerce*, 12(2), pp.1-27.

Singh, V. and Pandey, S.K., 2014, October. A comparative study of cloud security ontologies. In *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization* (pp. 1-6). IEEE.

Sinha, A., Haddad, I., Nightingale, T., Rushing, R. and Thomas, D., 2006, April. Wireless intrusion protection system using distributed collaborative intelligence. In *2006 IEEE International Performance Computing and Communications Conference* (p. 79). IEEE.

Skopik, F., Pahi, T. and Leitner, M., 2017. Situational Awareness for Strategic Decision Making on a National Level. In *Collaborative Cyber Threat Intelligence* (pp. 225-276). Auerbach Publications.

Sonowal, G., & Kuppusamy, K. S., 2017. PhiDMA - A phishing detection model with multi-filter approach. *Journal of King Saud University - Computer and Information Sciences*. doi:10.1016/j.jksuci.2017.07.005

Song, Y., Yang, C. and Gu, G., 2010, June. Who is peeping at your passwords at Starbucks?—To catch an evil twin access point. In *Dependable Systems and Networks (DSN), 2010 IEEE/IFIP International Conference on* (pp. 323-332). IEEE.

Sonowal, G. and Kuppusamy, K.S., 2016, August. MASPHID: a model to assist screen reader users for detecting phishing sites using aural and visual similarity measures. In *Proceedings of the International Conference on Informatics and Analytics* (p. 87). ACM.

Sood, A.K. and Enbody, R.J., 2011. Malvertising—exploiting web advertising. *Computer Fraud & Security*, 2011(4), pp.11-16.

Spring, T., 2017. Latest tax scams include phishing lures, mal- ware. [Online] Available at: <https://threatpost.com/latest-tax-scams-include-phishing-lures-malware/124431/> [Accessed 26 April 2017]

Stringhini, G. and Thonnard, O., 2015, July. That ain't you: Blocking spearphishing through behavioral modelling. *In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment* (pp. 78-97). Springer, Cham.

Suganya, V., 2016. A Review on Phishing Attacks and Various Anti Phishing Techniques. *International Journal of Computer Applications*, 139(1), pp.20-23.

Symantec, W.H., 2011. Advanced Persistent Threats: A Symantec Perspective. *Symantec World Headquarters*.

Symantec, 2013. *Internet security threat report 2013*. Technical Report. Symantec Corporation

Symantec, 2016. *Internet security threat report 2016*. Technical Report. Symantec Corporation.

Tabassum, A., Mustafa, M.S. and Al Maadeed, S.A., 2018, March. The need for a global response against cybercrime: Qatar as a case study. *In 2018 6th International Symposium on Digital Forensic and Security (ISDFS)* (pp. 1-6). IEEE.

Tan, C. L., Chiew, K. L., Wong, K. S., & Sze, S. N., 2016. PhishWHO: Phishing webpage detection via identity keywords extraction and target domain name finder. *Decision Support Systems*, 88, 18–27. doi:10.1016/j.dss.2016.05.005

Tan, C.L. and Chiew, K.L., 2017. Phishing webpage detection using weighted URL tokens for identity keywords retrieval. *In 9th International Conference on Robotic, Vision, Signal Processing and Power Applications* (pp. 133-139). Springer, Singapore.

Tankard, C., 2011. Advanced persistent threats and how to monitor and deter them. *Network security*, 2011(8), pp.16-19

Teufel, S., Burri, R. and Teufel, B., 2018, May. Cybersecurity Guideline for the Utility

Business a Swiss Approach. *In 2018 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)* (pp. 1-6). IEEE.

Tewari, A., Jain, A.K. and Gupta, B.B., 2016. Recent survey of various defence mechanisms against phishing attacks. *Journal of Information Privacy and Security*, 12(1), pp.3-13.

Thomas, K., Grier, C., Ma, J., Paxson, V. and Song, D., 2011, May. Design and evaluation of a real-time URL spam filtering service. *In Security and Privacy (SP), 2011 IEEE Symposium on* (pp. 447-462). IEEE.

Toolan, F. and Carthy, J., 2010, October. Feature selection for spam and phishing detection. *In eCrime Researchers Summit (eCrime), 2010* (pp. 1-12). IEEE.

Trend Micro, 2012. Spear-phishing email: Most favored APT attack bait. [Online] Available at: <http://www.trendmicro.com/au/cloud-content/us/pdfs/security-intelligence/white-papers/wp-spear-phishing-email-most-favored-apt-attack-bait.pdf> (accessed 1 October 2018).

TrendLabs, 2012. Evolved Threats in a “Post-PC” World. [Online] Available at: <https://www.trendmicro.de/cloud-content/us/pdfs/security-intelligence/reports/rpt-evolved-threats-in-a-post-pc-world.pdf> [Accessed 1 October 2018]

TrueSoftware, 2017. Mypublicwifi: *Turn your computer into a WIFI access point with firewall and URL tracking.* [Online] Available at: <http://www.mypublicwifi.com/publicwifi/en/index.html> [Accessed 14 June 2017]

Tufts, A., 2012. *How to protect your android against “Smishing”.* [Online] Available at:

<https://www.oneclickroot.com/how-to/how-to-protect-your-android-against-smishing/> [Accessed 3 May 2017]

Varghese, J., 2017. Qatar faced 93,570 *phishing attacks in first quarter of 2017*. [Online] (updated 12 May 2017) Available at: <<http://m.gulf-times.com/story/547784/Qatar-faced-93-570-phishing-attacks-in-first-quart>> [Accessed 30 Jun. 2018]

VeriSign, 2012. *2012 iDefense cyber threats and trends*. White Paper . VeriSign, Inc.

Vural, I. and Venter, H., 2011, January. Detecting mobile spam botnets using artificial immune systems. In *IFIP International Conference on Digital Forensics* (pp. 183-192). Springer, Berlin, Heidelberg.

Wang, F.F., 2006. Domain names management and legal protection. *International Journal of Information Management*, 26(2), pp.116-127.

Wang, Y.M., Beck, D., Wang, J., Verbowski, C. and Daniels, B., 2006. Strider Typo-Patrol: *Discovery and Analysis of Systematic Typo-Squatting*. SRUTI, 6, pp.31-36

Waziri, I., 2015, November. Website forgery: Understanding phishing attacks and nontechnical countermeasures. In *Cyber Security and Cloud Computing (CSCloud), 2015 IEEE 2nd International Conference on* (pp. 445-450). IEEE.

Weins, K. 2016. *Cloud computing trends: 2016 state of the cloud survey*. [Online] Available at: <http://www.rightscale.com/blog/cloud-industry-insights/cloud-computing-trends-2016-state-cloud-survey> [Accessed 10 July 2018]

Whittaker, B., 2017. Phishing attacks target activists over Qatar and Egypt. [Online] (updated 15 Feb. 2017) Available at: <<http://al-bab.com/blog/2017/02/phishing-attacks-target-activists-over-qatar-and-egypt>> [Accessed 30 Jun. 2018]

Whittaker, C., Ryner, B. and Nazif, M., 2010, February. Large-Scale Automatic

Classification of Phishing Pages. *In NDSS* (Vol. 10, p. 2010).

Whitehead, G.P., 2017. *10 phishing examples in 2017 that targeted small businesses*. [Online] (updated 29 Aug. 2017) Available at: <<https://smallbiztrends.com/2017/08/phishing-examples-small-business.html>> [Accessed 30 Jun. 2018]

Williams, E. J., Hinds, J., & Joinson, A. N. 2018. Exploring susceptibility to phishing in the workplace. *International Journal of Human Computer Studies*, 120(April), 1–13. doi:10.1016/j.ijhcs.2018.06.004

World Population Review. 2019. *Qatar Population*. [ONLINE] Available at: <http://worldpopulationreview.com/countries/qatar-population/>. [Accessed 12 August 2019].

Xiang, G., Hong, J., Rose, C.P. and Cranor, L., 2011. Cantina+: A feature-rich machine learning framework for detecting phishing web sites. *ACM Transactions on Information and System Security (TISSEC)*, 14(2), p.21.

Xing, X., Meng, W., Lee, B., Weinsberg, U., Sheth, A., Perdisci, R. and Lee, W., 2015, May. Understanding malvertising through ad-injecting browser extensions. *In Proceedings of the 24th international conference on world wide web* (pp. 1286-1295). International World Wide Web Conferences Steering Committee.

Xu, Z. and Zhu, S., 2012, August. Abusing Notification Services on Smartphones for Phishing and Spamming. *In WOOT* (pp. 1-11).

Yadav, N. , & Nagpal, B. 2016. Study on clickjacking attack. *International Journal of Emerging Research in Management and Technology*, 5 (6), 37–41 .

Yeboah-Boateng, E.O. and Amanor, P.M., 2014. Phishing, SMiShing & Vishing: an assessment of threats against mobile devices. *Journal of Emerging Trends in Computing*

*and Information Sciences*, 5(4), pp.297-307.

Yue, C., 2013. The devil is phishing: Rethinking web single sign-on systems security. In Presented as part of the 6th {USENIX} *Workshop on Large-Scale Exploits and Emergent Threats*.

Yue, C. and Wang, H., 2010. BogusBiter: A transparent protection against phishing attacks. *ACM Transactions on Internet Technology (TOIT)*, 10(2), p.6.

Zhang, Z., & Gupta, B. B., 2018. Social media security and trustworthiness : Overview and new direction. *Future Generation Computer Systems*, 86, 914–925. doi:10.1016/j.future.2016.10.007

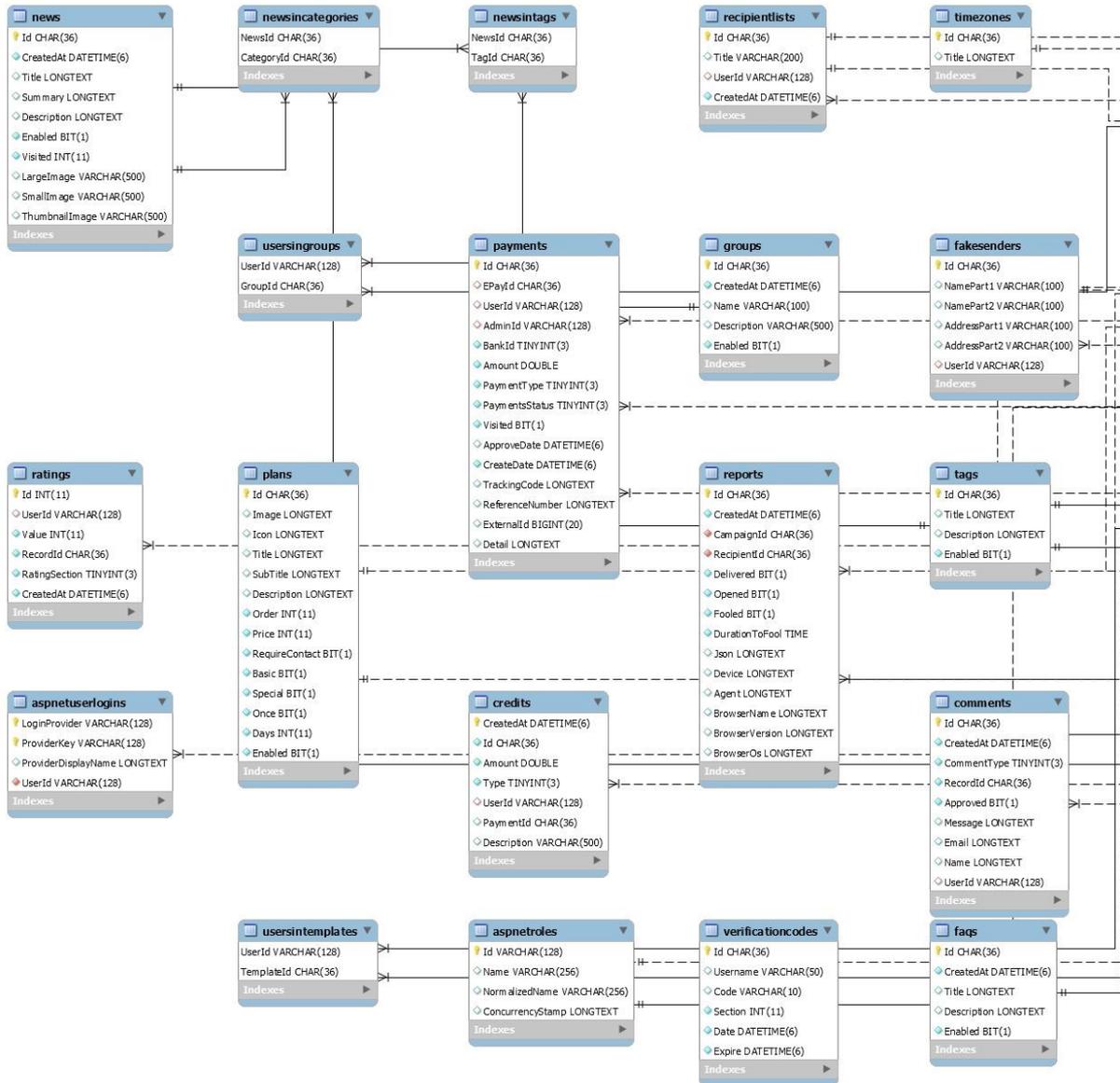
Zhou, S., 2015. A Survey on Fast-flux Attacks. *Information Security Journal: A Global Perspective*, 24(4-6), pp.79-97.

Zouina, M. and Outtaj, B., 2017. A novel lightweight URL phishing detection system using SVM and similarity index. *Human-centric Computing and Information Sciences*, 7(1), p.17

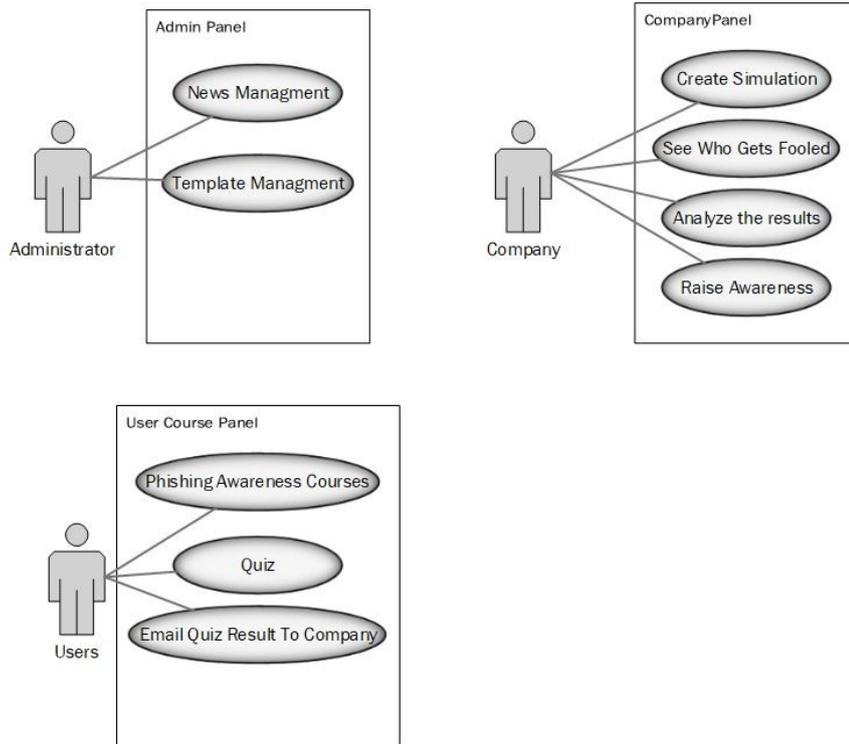


# APPENDIX A

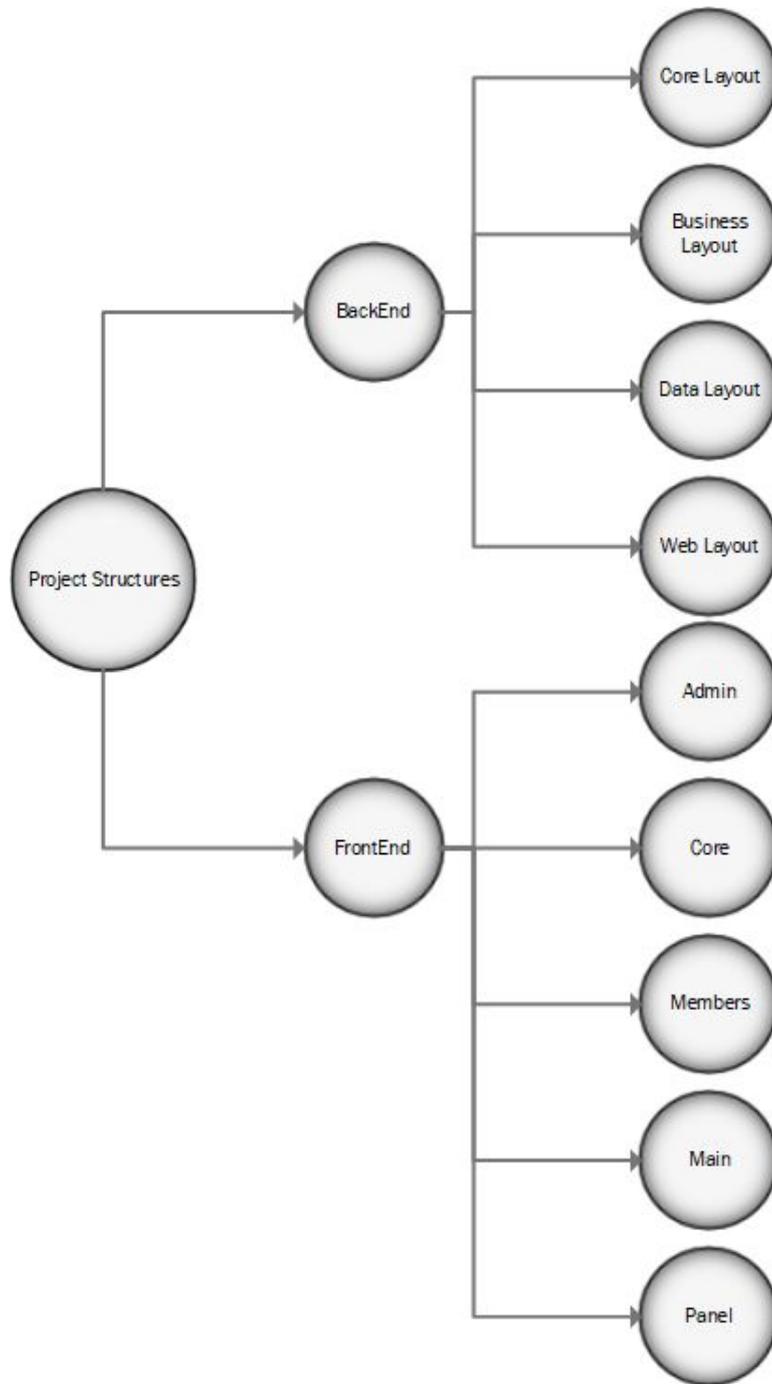
## DATABASE DESIGN AND USE CASES



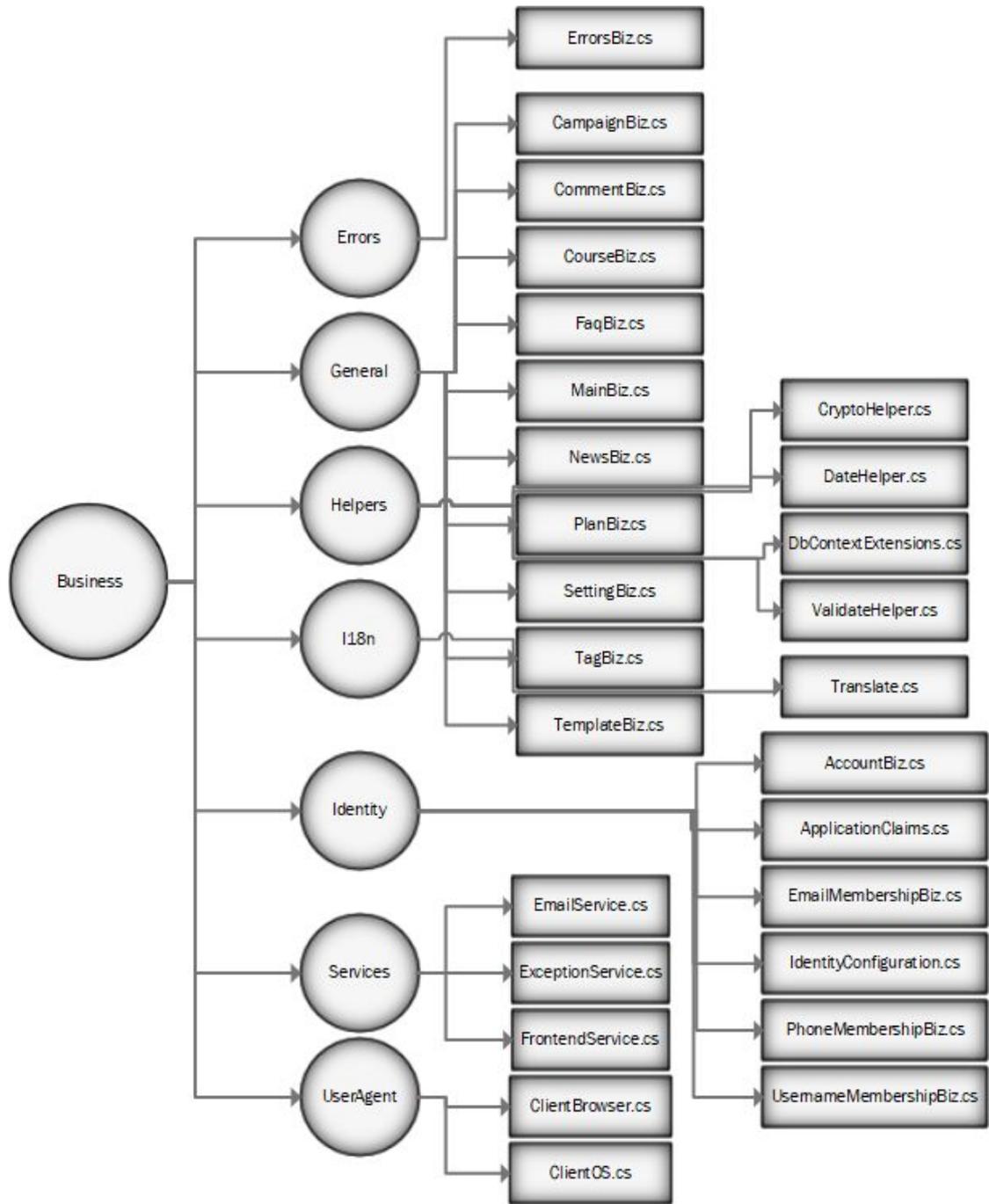
**Figure 5.1 A Database Design for Storing Data from users of our Web-based Tool.**



**Figure 5.2 Use case diagram of the project**



**Figure 5.3 The project's layout**



**Figure 5.4 The main classes of businesses**