AN AGENT-BASED PHISHING ATTACK MODEL FROM A HUMAN-ORGANIZATIONAL-TECHNICAL PERSPECTIVE

MELTEM EMINE MUTLUTÜRK

BOĞAZİÇİ UNIVERSITY

2023

AN AGENT-BASED PHISHING ATTACK MODEL FROM A HUMAN-ORGANIZATIONAL-TECHNICAL PERSPECTIVE

Thesis submitted to the

Institute for Graduate Studies in Social Sciences

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Management Information Systems

by

Meltem Emine Mutlutürk

Boğaziçi University

2023

DECLARATION OF ORIGINALITY

I, Meltem Emine Mutlutürk, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution.
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature……

Date……

ABSTRACT

An Agent-Based Phishing Attack Model from A Human-Organizational-Technical Perspective

In the rapidly evolving digital landscape, cybersecurity has emerged as a significant concern for organizations. This thesis delves into the intricate dynamics of malware-based phishing attacks on enterprise computer networks. Utilizing the robust methodological tool of Agent-Based Modelling (ABM), the research is firmly rooted in socio-technical theory and the concept of complex adaptive systems (CAS). The study meticulously examines the pivotal role of human factors, particularly awareness training and the credibility of phishing emails, in determining susceptibility to phishing attacks. Also, it underscores the significant impact of technological countermeasures, including the strategic deployment of Endpoint Detection and Response (EDR) solutions and the implementation of a hybrid antivirus scan policy, in mitigating infection rates. By seamlessly integrating human behaviour with socio-technical dimensions, the research provides a nuanced, comprehensive understanding of cybersecurity threats. The findings underscore the necessity for a balanced, holistic approach that equally prioritizes human behaviour and technological measures. This approach is crucial to enhance organizational resilience against relentless cyber threats. The insights gained from this research offer invaluable guidance for organizations striving to navigate the complex cybersecurity challenges in today's increasingly digital age.

ÖZET

İnsan-Kurumsal-Teknik Açıdan Bir Ajan Tabanlı Phishing Saldırı Modeli

Hızla gelişen dijital dünyada, siber güvenlik işletmeler için önemli bir sorun haline gelmiştir. Bu tez, kurumsal bilgisayar ağlarına yönelik kötü amaçlı yazılım tabanlı phishing saldırılarının karmaşık dinamiklerine derinlemesine iner. Araştırma, Ajan Tabanlı Modelleme gibi sağlam bir metodolojik araç kullanarak, sosyo-teknik teori ve karmaşık uyarlanabilir sistemler kavramında sıkıca kök salmıştır. Çalışma, phishing saldırılarına karşı hassasiyeti belirlemede, özellikle farkındalık eğitimi ve phishing e-postalarının güvenilirliği gibi insan faktörlerinin hayati rolünü titizlikle inceler. Ayrıca, bulaşma oranlarını hafifletmede, Endpoint Detection and Response (EDR) çözümlerinin stratejik dağıtımı ve hibrit bir antivirüs tarama politikasının uygulanması dahil teknolojik önlemlerin önemli etkisini vurgular. Araştırma, insan davranışını sosyo-teknik boyutlarla sorunsuz bir şekilde entegre ederek, siber güvenlik tehditlerine dair nüanslı, kapsamlı bir anlayış sağlar. Bulgular, insan davranışını ve teknolojik önlemleri eşit derecede önceliklendiren dengeli, bütünsel bir yaklaşımın gerekliliğini vurgular. Bu yaklaşım, acımasız siber tehditlere karşı kurumsal direnci artırmak için hayati öneme sahiptir. Bu araştırmadan elde edilen sonuçlar, bugünün giderek dijitalleşen çağında karmaşık siber güvenlik zorluklarını yönetmeye çalışan organizasyonlar için rehberlik sunar.

CURRICULUM VITAE

NAME: Meltem Emine Mutlutürk

DEGREES AWARDED

PhD in Management Information Systems, 2023, Boğaziçi University

MSc in Management Information Systems, 2016, Dokuz Eylül University

BA in Econometrics, 2013, Dokuz Eylül University

AREAS OF SPECIAL INTEREST

Agent-based modelling, Cybersecurity, Data visualization, Simulation, Data Mining Techniques

PROFESSIONAL EXPERIENCE

Research Assistant, Department of Management Information Systems, Boğaziçi University, 2016 –

IS consultant, Vore Logistic Construction Petro chemistry, 2020 - 2022

PUBLICATIONS

International Journal Articles

Kor, B., Wakkee, I., Mutluturk, M. (2020), "An Investigation of Factors Influencing Entrepreneurial Intention amongst University Students", Journal of Higher Education Theory and Practice, Vol: 20, No: 1, pp. 70-86.

Ozturan, M., Mutlutürk, M., Çeken, B., Sarı, B. (2019), "Evaluating the information systems integration maturity level of travel agencies", Information Technology & Tourism, Vol: 21, No: 2, pp. 237-257 (SSCI).

Mutlutürk, M., Mardikyan, S. (2018), "Analysing Factors Affecting the Individual Entrepreneurial Orientation of University Students", Journal of Entrepreneurship Education, Vol: 21, No: Special Issue, pp. 1-15 (Scopus).

Kor, B., Mutluturk, M. (2017), "Business & Management Studies: An International Journal", Business & Management Studies: An International Journal, Vol: 5, No: 3, pp. 525-547.

Book Chapters

Mutlutürk, M. (2022). Industry 4.0 for smart cities. Artificial Intelligence Perspective for Smart Cities, 55-73.

Dursun, S.M., Mutluturk, M., Taskin, N., Metin, B. (2022), "An Overview of the IT Risk Management Methodologies for Securing Information Assets", ss. 30-47, IGI Global, Pennsylvania, USA. DOI: 10.4018/978-1-7998-7943-5.ch002.

Mutluturk, M., Kor, B., Metin, B. (2021), "The role of Edge/Fog Computing Security in IoT and Industry 4.0 Infrastructures: Edge/Fog-based Security in Internet of Things", ss. 211-222, IGI Global, USA. ISBN13: 9781799877400, DOI: 10.4018/978-1-7998-7740-0.ch014.

Cimen, B., Mutluturk, M., Kocak, E., Metin, B. (2020), "A Hybrid Asset-Based IT Risk Management Framework", IGI Global, USA.

International Conference Proceedings

Metin, B., Mutluturk, M. (2019), "Perceived Smart-Phones Security In Digital Life", The 13th Mediterranean Conference on Information Systems and The 16th Conference of The Italian Chapter of AIS, Naples, Italy, September 27-28.

Ozdemir, S., Mutluturk, M., Kor, B., Metin, B. (2019), "Country Of Origin Criteria for Digitalization With National IT Products", 6. International Management Information

Systems Conference, İstanbul, Turkey, 09-12 October.

Metin, B., Mutluturk, M., Badur, B. (2019), "IT Risk Modeling for Effective Information Security Management", 6th International Conference on Automation, Control Engineering & Computer Science (ACECS-2019), İstanbul, Turkey, October 1-3.

Rabea, A., Mutluturk, M., Metin, B. (2019), "Privacy Concerns on Mobile Applications for Google Play Store Market", 11th International Conference on Electrical and Electronics Engineering, :ELECO, Bursa, Turkey, 28-30 November.

Kor, B., Wakkee, I., Mutluturk, M. (2019), "The Development of Entrepreneurial Intention amongst University Students: The Influence of Individual and Contextual Factors", Interdisciplinary European Conference on Entrepreneurship Research (IECER): Entrepreneurship for a Better Future.

Kutlu, B., Ceken, B., Mutluturk, M., Turkmen, C. (2018), "A Meta-Analysis of Social Media Learning Studies in Educational Research", Fifth International Management Information Systems Conference, Ankara, Turkey, 24-26 October.

ACKNOWLEDGEMENTS

This has been a long time in the making. First and foremost, I am truly grateful for the beacon of light that is my thesis advisor, Prof. Bilgin Metin, for his unwavering support, guidance, and mentorship throughout my PhD journey. Words cannot begin to express how much I appreciate your support, I am truly fortunate to have had the opportunity to work with and learn from you in all aspects, not only research. Thank you for being the epitome of positivity and hope.

I would also like to express my gratitude to my thesis committee members, Prof. Vahap Tecim, Prof. Bertan Badur, Assoc. Prof. Mehmet Nafiz Aydın, and Assist. Prof. Nazım Taşkın, for their valuable feedback, and constructive criticism. Their expertise and suggestions have greatly contributed to the quality and depth of my research.

I must extend further gratitude to Prof. Bertan Badur, as he has provided much more support and guidance to me and this thesis than is possible to measure. I am, and will forever be in awe of your knowledge and wisdom.

Along with Prof. Vahap Tecim, I would like to express my deepest appreciation to Prof. Çiğdem Tarhan and Assoc. Prof. Can Aydın; without your encouragement, I wouldn't be where I am today.

A special thank you to Dr. Osman Yücel, for helping me become the turtle tamer that I am. I am especially grateful for the support of all my bestest friends (you know who you are), colleagues past and present, students and professors of Bogazici University department of Management Information Systems. You made working as a research assistant and this PhD much more pleasant, and I have learned a lot from every single one of you.

Now, on to the people that deserve to have a whole PhD thesis written about them, this acknowledgement doesn’t do them justice, my family. I am forever indebted to my loving parents and brother, for their unconditional love, support, and belief in me and my abilities even at times I didn’t. They have been my safe harbour during the most challenging of times.

This journey has not been without its difficulties, but it has provided me with countless opportunities for personal and professional growth. The lessons I have learned, the resilience I have developed, and the friendships I have made will remain with me for a lifetime.

Here's to being stardust in the galaxy of science.

To my brother, mum, and dad,

For their unconditional love

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ABM: Agent-based Modelling

APTs: Advanced persistent threats

CAS: Complex Adaptive System

DDOS: Distributed-denial-of-service

DOS: Denial-of-service

EDR: Endpoint Detection and Response

HOT: Human, Organizational, and Technical

IS: Information System

SIR: Susceptible-Infected-Recovered

SMEs: Small and Medium Enterprises

CHAPTER 1

INTRODUCTION

"Today, to him gazing south with a new-born need stirring in his heart, the clear sky over their long low outline seemed to pulsate with promise; today, the unseen was everything. The unknown, the only real fact of life."

— Kenneth Grahame, The Wind in the Willows

The increasing dependence on digital technologies in modern organizations has heightened their vulnerability to cyberattacks (Ponsard, Grandclaudon, & Bal, 2019; Eian, Yong, Li, Qi, & Fatima, 2020). Cybercriminals have become increasingly sophisticated in their methods (Katterbauer, Hassan, & Cleenewerck, 2022; Khan et al., 2022), employing advanced techniques and technologies to bypass traditional security measures, such as firewalls and antivirus software (Neghina & Scarlat, 2013). Furthermore, the rapid growth of digital technology and the increasing reliance on online services have broadened the potential attack surface for these threat actors, elevating the vulnerability of organizations to cyberattacks (Garello & Mousavi, 2021).

Among the myriad of cyber threats faced by organizations, phishing and malware attacks pose substantial challenges to the security and stability of their networks. The repercussions of these threats are often severe, potentially causing significant financial and operational damage, data breaches, and even impacting compliance with data protection and cybersecurity regulations (Ibrahim, Thiruvady, Schneider, & Abdelrazek, 2020). Given the complexity and scale of these threats,

understanding the dynamics of phishing attacks and malware propagation within organizational networks has become a critical area of study.

The primary aim of this research is to delve into the effects of a malware-based phishing attack on an enterprise computer network. Notably, the success of phishing attacks often hinges on the human element, as phishing campaigns often employ deception and manipulation, making system users the most vulnerable link in the cybersecurity chain (Pfleeger & Caputo, 2012; Bada, Sasse, & Nurse, 2019). This suggests that cybersecurity threats are not solely technological problems but also involve human and organizational elements.

This study adopts an integrative approach, examining the interplay between human behaviour, technology, and organizational elements. For this purpose, our research employs Agent-Based Modelling (ABM) as a central methodological tool. ABM is a computational modelling technique that facilitates the simulation of individual agents' behaviours and their interactions (Bonabeau, 2002; Wilensky & Rand, 2015), offering several advantages in understanding phishing attacks and their dynamics (Burns, Posey, Courtney, Roberts, & Nanayakkara, 2017).

To explore the complicated dynamics of phishing attacks and malware propagation within enterprise computer networks, this research is guided by two key theoretical perspectives; socio-technical theory and the concept of complex adaptive systems (CAS). Frequently, cybersecurity is primarily approached as a technological problem and human and organizational factors are overlooked or underemphasized. The socio-technical theory posits that the effective functioning and security of complex systems, such as organizations and their information systems, depend on the alignment and interaction between social elements and technical aspects (Appelbaum, 1997;

Malatji, Von Solms, & Marnewick, 2019). In the context of phishing attacks and malware, socio-technical theory highlights the need to consider both the human and technical dimensions of cybersecurity. Factors such as user awareness, training, and organizational culture can influence the susceptibility of individuals to phishing attacks (Karamagi, 2022), while the effectiveness of technical security measures, such as firewalls and antivirus software, can determine the potential for malware propagation within the network (Shim, 2015). CAS provides a lens to examine the behaviour and evolution of systems composed of numerous interacting components or agents, such as organizational networks (Anderson, 1999).

1.1 Research objectives

This research aims to capture individual agent behaviours, such as user susceptibility to phishing emails, malware infection rates, and the effectiveness of security measures. ABM allows for the representation of dynamic interactions between agents (Crooks & Heppenstall, 2011; Kaniyamattam, 2022). By simulating these interactions, the research can gain insights into how the collective behaviour of agents within the network influences the overall impact and outcomes of phishing attacks and malware dissemination.

The significance of studying the impact of phishing attacks and malware propagation on enterprise networks is immense, given the increasing reliance of organizations on digital infrastructure. The successful execution of a phishing attack or malware intrusion can disrupt an organization's operations, leading to the unauthorized access of sensitive data, damaging an organization's reputation, and resulting in

significant financial losses (Gupta, Tewari, Jain & Agrawal, 2017; Jain & Gupta, 2022; Tasmin, Sarmin, Shalehin, & Haque, 2022).

Given the intricacies of phishing attacks and malware propagation within an enterprise network, the following research question has been established to guide this research:

RQ: How do human, organizational, and technical elements of enterprise networks affect phishing attacks and malware propagation?

This research aims to provide a holistic understanding of phishing attacks and malware propagation in enterprise networks by incorporating human behaviour, technology, and organizational factors. It addresses a significant gap in the existing literature and responds to the practical needs of organizations navigating the complexities of the digital age.

In conclusion, this research, employing ABM and guided by socio-technical theory and the perspective of CAS, contributes to a deeper understanding of the effects of a malware-based phishing attack on an organizational network. By capturing the complex dynamics and consequences of these threats, it offers valuable insights for organizations to improve their cybersecurity measures.

1.2 Structure of the thesis

In the upcoming chapters, the research will explore the foundational concepts and relevant literature in order to provide a thorough background. Next, it will outline the methodology employed in the study, providing detailed insights into the design of the model. Subsequently, the research will present the findings, offering a comprehensive

overview of the results. Finally, it will discuss the proposed model, analysing their implications for theory, practice, and future research endeavours.

CHAPTER 2

BACKGROUND AND RELATED WORK

"Not all those who wander are lost."

- J.R.R. Tolkien, "The Riddle of Strider"

This chapter aims to provide a holistic overview of the state-of-the-art in cybersecurity research, setting the stage for the methodology of this thesis.

For this purpose, this chapter provides a comprehensive review of the existing literature pertinent to the research objectives outlined in the previous chapter. First, the fundamental aspects of cybersecurity are addressed, highlighting its importance and the associated threats. Second, the discussion delves into the specifics of key cybersecurity threats, focusing on phishing and malware-based attacks, and the various measures employed for their mitigation. Third, the discussion takes a deep dive into the domain of phishing attacks, scrutinizing the content and the design aspects of phishing emails, the role of malware in these attacks, and the influence of personality traits and cognitive biases on susceptibility to phishing. This is followed by a detailed exploration of the role of human factors in cybersecurity, discussing the impact of awareness levels and cognitive biases on cybersecurity behaviour. Fourth, the investigation of cybersecurity within the framework of complex adaptive systems and socio-technical theory occurs, intending to understand the interaction between technical and social aspects in a cybersecurity context. Fifth, agent-based modelling in cybersecurity is introduced, discussing its advantages, its application in the context of complex adaptive systems,

and its potential for studying phishing attacks and malware propagation. Finally, the chapter summarizes the insights gathered, identifies the gaps in the existing literature, and outlines the direction for future research in this domain.

2.1 An overview of cybersecurity

Cybersecurity has recently become a pressing concern for organizations worldwide in the digital era. The term "Digital Age" captures the current era, where digital technology forms the backbone of most activities (Pokrivcakova, 2017), and cybersecurity has become a critical concern due to the widespread digitization of data and processes (Rodrigues, 2017). It helps to emphasize the modern context and the evolving nature of the threats organizations face. As digital networks and systems become more integral to business operations, the risk of cyber threats grows correspondingly (Rodrigues, 2017).

Cybersecurity, in its broadest sense, is about protecting digital assets from threats and maintaining the normal functioning of systems and networks (Whyte, Dagher, & Hagenah, 2023). A commonly accepted model for understanding cybersecurity is through the lens of Confidentiality, Integrity, and Availability, often referred to as the CIA triad (Ham, 2021).

Confidentiality means ensuring that data and information are accessible only to those authorized to view them. Unauthorized access, whether by an internal or external party, can result in the disclosure of sensitive information which can be damaging to both individuals and organizations. Cybersecurity measures that support confidentiality include encryption, access controls, and two-factor authentication (Scheponik et al., 2016).

Integrity ensures the accuracy and consistency of data over its entire lifecycle. Data should remain original and should not be altered or tampered with, without authorization (Puianu, Flangea, Marinescu, & Marinescu, 2017). A breach of integrity could involve an unauthorized party changing or deleting data, disrupting operations or leading to incorrect decisions based on inaccurate data. Measures to maintain integrity include digital signatures and version controls (Hartono, Holsapple, Kim, Na, & Simpson, 2014).

Availability refers to ensuring that information and systems are accessible and operational when needed by authorized users (Aslan, Aktuğ, Ozkan-Okay, Yilmaz, & Akin, 2023). A denial-of-service (DOS) attack, for example, seeks to disrupt the availability of a service, making it inaccessible to users. To maintain availability, measures such as redundancies, failover systems, and distributed-denial-of-service (DDoS) protection can be used (Hartono et al., 2014).

In essence, cybersecurity is a balance and combination of these three principles. It's about protecting sensitive data (confidentiality), ensuring it is not tampered with (integrity), and making sure it's always accessible when needed (availability).

Cybersecurity in organizations is paramount; it safeguards sensitive data, maintains privacy, and guarantees systems and data integrity (Aslan et al., 2023). With organizations now more interconnected than ever, storing and processing vast amounts of sensitive data digitally, they have become attractive targets for cybercriminals. In such a connected world, a single vulnerable system can potentially expose the entire network to a cyberattack (Aslan, et al., 2023). Therefore, robust cybersecurity measures are crucial, not only for the protection of individual systems but also for maintaining the overall health and functionality of the entire organizational infrastructure.

The absence of robust cybersecurity can lead organizations to face potential losses on multiple fronts, such as financial losses, operational disruptions, and loss of intellectual property. Furthermore, data breaches can result in reputational damage, loss of customer trust, and potential legal ramifications (Ibrahim, Thiruvady, Schneider, & Abdelrazek, 2020).

2.2 Key cybersecurity threats

Organizations encounter various cybersecurity threats, each with unique characteristics and potential impacts. Among the most common and damaging are phishing attacks and malware propagation, although other prevalent threats include ransomware attacks, DoS attacks, and advanced persistent threats (APTs). Each threat poses unique challenges and necessitates different countermeasures, highlighting the need for a comprehensive, multi-faceted approach to organizational cybersecurity (Babate, Musa, Kida, & Saidu, 2015).

2.2.1 Phishing attacks

Phishing attacks have evolved significantly since their emergence in the mid-1990s (Rekouche, 2011). The term "phishing" was coined to describe the act of "fishing" for sensitive information from unsuspecting users by tricking them into providing it (Purkait, Kumar De, & Suar, 2014). Early phishing attacks primarily targeted users through email, directing them to fake websites that look legitimate, such as banks or online services, to steal their login credentials and other sensitive data (Hong, 2012).

Over time, phishing attacks have become more sophisticated and diversified. Spear-phishing, for instance, is a targeted form of phishing that involves crafting

personalized emails to specific individuals or organizations, often using information gathered from social media and other sources to increase the attack's credibility (Chiew, Yong, & Tan, 2018). Whaling is another variation that targets high-profile individuals, such as executives, in an attempt to gain access to valuable data or financial resources (Pienta, Thatcher, & Johnston, 2020).

Phishing attacks have also extended beyond email to exploit other communication channels, such as social media, instant messaging, and SMS text messages (smishing) (Hong, 2012). Additionally, some phishing attacks employ malicious attachments or links to deliver malware to the target's device, further compromising the security of the individual or organization (Lam & Kettani, 2019).

Phishing is often the initial vector for more advanced and damaging attacks, such as those involving malware or ransomware. The success of phishing heavily relies on manipulating human behaviours and decision-making processes, exploiting users' trust and lack of awareness about the tell-tale signs of a phishing attempt (Raijvan & Gonzalez, 2018; Alhogail & Alsabih, 2021). Table 1 provides a summary of the phishing attacks provided in this section.

Table 1. Summary of Phishing Attacks

Phishing Attack Type	Target	Attack Surface	Attack Vector	Key Aspects
Email Phishing	Individuals / Companies	Email Systems	Email Links	Deceptive emails designed to trick recipients into revealing sensitive information
Spear Phishing	Specific Individuals / Companies	Email Systems	Email Links	Targeted phishing emails appearing to be from a trusted source
Whaling	High-level Executives	Email Systems	Email Links	Highly targeted phishing aimed at senior executives
Smishing	Mobile Phone Users	SMS Systems	SMS Links	Phishing via SMS messages leading to a malicious website
Vishing	General Public	VoIP Systems	Voice Calls	Phishing via voice calls using scare tactics and urgency

2.2.2 Malware attacks

Malware, short for malicious software, is another common cyber threat. It includes any software intentionally developed to disrupt, damage, or illicitly access computer systems and networks. Malware can be delivered to a system via various means, often embedded within phishing emails or malicious websites (Namanya, Cullen, Awan, & Disso, 2018). The propagation of malware involves spreading these malicious programs across networks, often exploiting computer vulnerabilities or human factors. The types of malware include viruses, worms, Trojans, ransomware, and spyware, each with its unique characteristics and propagation techniques (Namanya et al., 2018).

Besides phishing, malware propagation techniques have evolved in sophistication, exploiting system vulnerabilities and human factors. The techniques include:

- Email attachments: Malware can be hidden in email attachments, infecting the system when opened (Chiew, Yong, & Tan, 2018).
- Drive-by downloads: These occur when a user unknowingly "invites" a malware by visiting an infected website, viewing an infected email message, or clicking on a deceptive pop-up window (Provos, Mavrommatis, Rajab, & Monrose, 2008).
- Social engineering: Attackers may use social engineering tactics, such as spear phishing, to trick users into downloading and installing malware (Hong, 2012).
- Malvertising involves injecting malicious or malware-laden advertisements into legitimate online advertising networks and web pages (Kumar, Rautaray, & Pandey, 2017).
- Watering hole attacks: In this method, the attacker guesses or observes which websites the group often uses and infects those sites with malware. The strategy is to infect a specific group of end users (Caltagirone, Pendergast, & Betz, 2013).

Malware can be classified into various categories based on their behaviour, objectives, and propagation methods (Namanya et al., 2018). Some common types include:

- Viruses: Malicious programs that self-replicate by infecting other files and can cause various types of damage (Madihah & Nazean, 2006).
- Worms: Self-replicating programs that propagating across networks, often exploiting vulnerabilities to infect other systems (Madihah & Nazean, 2006).
- Trojans: Malware disguised as legitimate software that can provide unauthorized access to a user's device or perform other malicious actions (Madihah & Nazean, 2006).

- Ransomware: Malware that encrypts a user's data or locks their device, demanding a ransom payment for the decryption key or unlock code (Continella et al., 2016).
- Spyware: Software that secretly monitors and collects user information, such as keystrokes, browsing habits, or personal data (Podder, Mondal, Bharati, & Paul, 2021).

Types of malware have been summarized in Table 2.

Table 2. Summary of Types of Malware

Malware Type	Target	Attack Surface	Attack Vector	Key Aspects
Viruses	Computer Systems, networks	Operating Systems	Infected Files / Software	Self-replicating code that modifies programs and files
Worms	Networks, devices	Operating Systems, Network Services	Network Propagation	Self-replicating code that propagates through network connections
Trojans	Inividuals, businesses	Operating Systems	Malicious Software	Malware disguised as legitimate software, often providing remote control to attacker
Ransomware	Individuals, businesses, government	Operating Systems	Email, Exploits, Malicious Software	Malware that encrypts user’s data until a ransom is paid
Spyware	Individuals, businesses	Operating Systems	Various methods	Malware that collects and sends user data to a third party

Computer worms are of significant importance as this study has focused on the propagation of a worm within an enterprise network. A computer worm can be described as a self-replicating software that spreads autonomously across network connections, typically causing damage. Originating from an initial machine, the worm leverages the network to spread copies of itself to other devices, resulting in an exponential growth in the number of existing worms (Kamal, Ali, Alani, & Abdulmajed, 2016).

Worms comprise several elements. They are capable of spreading to new vulnerable hosts, infecting those hosts, and initiating the propagation cycle once more. In terms of propagation, a worm may randomly jump into a new host to infect. For malicious worms, this random selection of potential victims is a simple way of expanding its influence. Alternatively, some worms scan their local network to find possible vulnerable hosts. If the worm bypasses a firewall and enters a different network, its spread can be swift. This technique also enables faster infection rates, since systems with similar IP addresses are likely closely linked within the network topology (Twardus, 2005).

Infection of new hosts by malicious worms is typically achieved through an exploitable flaw in a network service or multiple services already running on the targeted system. This vulnerability could be a buffer overflow in a network service, which enables the worm to execute arbitrary code with high-level privileges. Some worms, like Nimda, are multipartite, employing several infection vectors. To protect a system from a multipartite worm, all infection vectors must be accounted for. Certain worms carry a payload that could be a backdoor program, allowing remote access to the compromised system by a malicious user. Alternatively, the payload could be a virus the worm assists in spreading to new systems, DoS attack set to be executed by all infected

systems at a specific date and time. Some worms don't carry payloads and exist solely to propagate (Twardus, 2005).

Worms can be sorted into three broad categories for easier analysis: email worms (including other client application worms), Windows file-sharing worms, and traditional worms. This doesn't imply a rigid classification, as many worms can be included in multiple categories, like Nimda, which falls into all three (Kienzle & Elder, 2003).

Email worms and other client application worms represent a significant part of network-aware malicious code since email is crucial for personal and business life. Many worms use other file-sharing applications, various peer-to-peer file sharing systems, to spread in a way similar to email worms. They lure users into executing untrusted files and don't seem to have significantly evolved from email worms, so they are considered in the same category.

Windows file sharing worms take advantage of the widespread SMB protocol, while traditional worms resemble the model of the 1988 Morris worm. They mostly connect directly via TCP/IP-based protocols, exploit vulnerabilities, don't typically need user intervention, and utilize propagation vectors other than email and Windows file sharing (Kienzle & Elder, 2003).

Significant worm attacks (summary can be found in Table 3) in security history include:

The Morris Worm: Released in 1988, this is one of the first worms to spread across the Internet, causing significant disruption.

The ILOVEYOU Worm: Released in 2000, it spread via email and affected millions of computers worldwide, leading to billions of dollars in damage.

The Slammer Worm: Released in 2003, it spread rapidly and infected hundreds of thousands of computers within minutes, causing significant Internet disruption.
The Stuxnet Worm: Discovered in 2010, it was designed to specifically target and disrupt Iran's nuclear program. It was a significant development in the field of cybersecurity, as it represented a shift towards cyber warfare (Shah, Shah, Shah, & Kanai, 2017).

Discovered in 2016, the Mirai worm was designed to attack Internet of Things (IoT) devices such as IP cameras, taking advantage of their default or weak passwords. This resulted in a widespread infection, leading to the creation of a botnet utilized to conduct Distributed Denial of Service (DDoS) attacks. The emergence of the Mirai worm underlined the security flaws inherent in IoT devices and highlighted the significance of enhancing their security (Lingenfelter, Vakilinia, & Sengupta, 2020).

Table 3. Summary of Significant Worm Attacks

Worm Name	Target	Attack Surface	Attack Vector	Key Aspects
Morris	UNIX Systems	Internet, Email Systems	Exploit in the UNIX sendmail and finger programs	One of the first worms to spread across the internet; developed by Robert T. Morris in 1998
ILOVEYOU	Microsoft Windows Systems	Email systems, File sharing	Email attachment, File sharing	Tricked users into opening an email attachment named “LOVE-LETTER-FOR-YOU.txt.vbs”, causing massive damage globally in 2000
Slammer	Servers running Microsoft SQL Server 2000	SQL Servers	Buffer overflow in Microsoft SQL Server	Rapid propagation in 2003, caused internet outages and slowed down general internet traffic
Stuxnet	Windows systems running Siemens Step7 software	Industrial Control Systems	USB drives, Zero-day exploits	Specifically targeted Iranian nuclear facilities in 2010, causing physical damage
Mirai	Internet of Things (IoT) devices	IoT devices with default usernames and passwords	Telnet	Created a large botnet in 2016 used for distributed denial of service (DDOS) attacks

2.3 Human, organizational, and technological countermeasures

As the threat landscape evolves, so too do the measures employed to counteract them (Nagunwa, 2014). It is well-acknowledged that organizations face significant risks due to the actions and behaviours of their employees (Thomson, Von Solms, & Louw, 2006). A nuanced understanding of human behaviour provides valuable insights into why individuals fall prey to phishing emails (Beznosov, & Beznosova, 2007), and technological flaws can often be traced back to human errors (Hinson, 2003). Despite this, companies tend to disproportionately focus on technological solutions for information and data security, overlooking the substantial risk posed by employee behaviour. Cyber attackers frequently exploit human's good intentions and social vulnerabilities more readily than they bypass technological defences (Mitnick & Simon, 2003).

A majority of employees struggle to distinguish between legitimate and fraudulent websites (Dhamija, Tygar, & Hearst, 2006), and many can become so engrossed in their work duties that they neglect browser security indicators. Security measures should not be restricted to only physical and technical measures as even technical experts may have misconceptions that such steps alone are sufficient for safeguarding both them and average consumers (Jakobsson, 2007). Issues like social engineering attacks and non-compliance with organizational security policies are being increasingly spotlighted as security concerns due to a lack of security awareness. This further emphasizes that technical solutions can only be as effective as the individuals who utilize and manage them, highlighting the need for a holistic approach to information security (Colwill, 2009).

Given that most security threats exploit human behaviour, an organizational framework integrating human, organizational, and technological (HOT) dimensions is essential to combat them (Frauenstein & Von Solms, 2013). In a model depicted in Figure 1, these dimensions' function independently due to limited interaction and communication, providing only a 'single layer' defence. This structure allows phishing attacks to multiply if one dimension weakens, thus compromising the others. A more robust approach, involves an 'in-depth' defence model, where multiple barriers work in concert as a defence. Technical measures such as hardware and software solutions, including firewalls, Intrusion Detection and Prevention Systems (IDPS), antivirus and antimalware software, and patch management, are crucial components of cybersecurity strategies.

However, the efficacy of these measures greatly depends on their integration with other organizational aspects (Jain & Gupta, 2022). This integration is encapsulated in the HOT framework as suggested by Frauenstein & Holms (2013). In this concept, the human element often serves as the initial line of defence against phishing, while technological tools provide the final layer of protection. This approach offers a comprehensive solution to phishing threats, as illustrated in Figure 1.

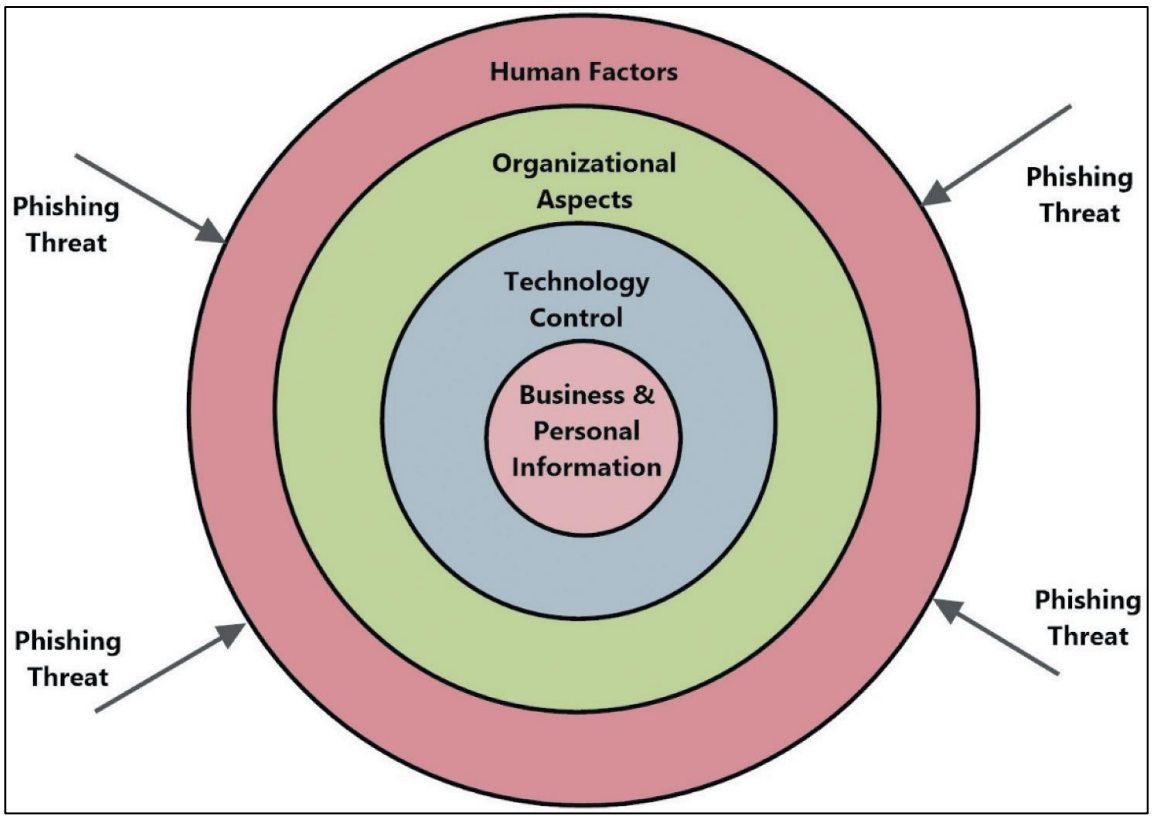

Figure 1. HOT model dimensions (Jain & Gupta, 2022)

2.3.1 The human element

Effective cybersecurity extends beyond technical and procedural measures to involve improving the human element. Strategies like game-based training are employed to teach employees to identify, counteract, and report phishing attacks (Jain & Gupta, 2022).

The human dimension necessitates effective awareness, education, and training to fortify the 'human firewall' and cultivate information security behaviour in the organization. Phishing attacks primarily expose the human dimension, jeopardizing technology and organizational dimensions. Therefore, education must permeate all

dimensions to ensure optimal functioning of the HOT dimensions, facilitating sufficient risk mitigation against phishing attacks (Frauenstein & Von Solms, 2013).

Security awareness programs can be invaluable in this regard. Such programs (Cobb, 2010) foster trust between employers and employees by providing clarity on the purpose of security protocols. This mutual understanding can lead to fewer accidental breaches and improved detection and reporting of malicious activities.

2.3.2 Organizational elements

Complementing human-focused measures, organisational components involve incorporating policies and procedures at various levels, including human resources, requirements, access control, and website access. Adherence to various Enterprise Risk Management frameworks, such as the Control Objectives for Information and Related Technology (COBIT), is encouraged to guide IT control (Jain & Gupta, 2022).

Organizations should foster a culture of information security, acknowledging the importance of enhancing and securing their data. They should lead initiatives to prevent and manage potential security threats, achievable through robust physical and software measures. Emphasizing the need for comprehensive policies and solid procedures to guard against phishing attacks is crucial. Once implemented, these measures will exert a positive influence on both the technological and human elements of the organization (Frauenstein & Von Solms, 2009).

2.3.3 Technological elements

From a technological standpoint, identity theft attacks due to phishing attacks can be mitigated using two-factor authentication, where the first credential is typically a

username-password combination, and the second one is found on the user's digital device. Enterprises are also encouraged to deploy various tools such as antivirus software, anti-phishing toolbars, and web application firewalls, and to maintain updated operating systems and browsers (Alhogail & Alsabih, 2021; Shahbaznezhad, Kolini, & Rashidirad, 2021).

Incorporating a robust endpoint security strategy is essential for organizations, considering that the endpoint is a common avenue for hackers to access the entire network (Parmar, 2012). A robust endpoint security strategy goes beyond traditional blacklist-based solutions, given their limitations, and should adopt a layered approach known as 'defence in depth.' This includes firewalls and antivirus solutions as a basic level of security, but must also include stronger layers like application whitelisting and system restore methods. Whitelisting identifies which programs are allowed to run, providing better assurance against unknown malware. System restore methods, on the other hand, allow for quick removal of unwanted or malicious software that may have bypassed other security measures (Parmar, 2012).

Endpoint Detection and Response (EDR) controls are crucial in this context. EDRs aid in quickly detecting and responding to advanced persistent threats (APTs) and help reduce alert fatigue (Kamruzzaman et al., 2022). EDR platforms enhance the capabilities of traditional endpoint protection platforms by identifying unknown threats that antivirus software cannot detect alone (Liggett, 2018). They set themselves apart from traditional endpoint protection tools with their unique ability to detect emerging threats and trigger alerts based on suspicious patterns, signatures, or activity (Liggett, 2018).

EDR systems leverage machine learning and AI methods to find new patterns and correlations, improving their ability to detect unknown threats (Karantzas & Patsakis, 2021). These systems can enhance the capabilities of Security Operation Centers (SOCs), alerting both users and response teams to emerging threats. However, EDR systems also face challenges such as generating many false positives, necessitating continual improvements in the technology and the introduction of newer methods such as Tactical Provenance Graphs (TPG) to reduce false-positive rates (Karantzas & Patsakis, 2021).

Even with robust technical security measures in place, the human factor remains crucial to the overall security of an organization. Investing in training for IT teams and raising user awareness about potential threats are critical aspects of improving an organization's overall security (Karantzas & Patsakis, 2021).

Simultaneously, proactive measures to counter phishing attacks are essential. This includes scanning emails for malicious content and sharing the findings with the entire organization, ensuring everyone benefits from this knowledge. Security Awareness Training (SAT), which educates employees about potential cyber threats, is another effective tool to defend against phishing attacks (Kohavi, 2021).

Moreover, holistic defences against phishing should be in place. Multi-tiered countermeasures that involve technical, individual, and organizational approaches have proven effective against such threats (Pienta et al., 2018). These measures span from the use of firewalls, encryption software, and two-factor authentication to SETA (Security Education Training and Awareness) programs, motivation and training, to legal and policy frameworks to deal with successful attacks (Pienta et al., 2018).

Lastly, in an organizational context, both detective and preventive countermeasures play a pivotal role in mitigating phishing attacks. While detective countermeasures track and audit users' activities, preventive measures block access to phishing emails, promoting safe practices among employees (Shahbaznezhad et al., 2021). These measures, combined with the human and organizational components, help fortify the defence against phishing and other cybersecurity threats (Jain & Gupta, 2022). In conclusion, achieving comprehensive cybersecurity necessitates a multidimensional approach, combining technical, procedural, and human-focused measures. The HOT framework proposed by Frauenstein & Von Solms (2013) offers a robust and effective defence strategy against evolving cybersecurity threats. Constant communication and education across all these dimensions are integral to cultivating a resilient and adaptable defence strategy.

2.4 Susceptibility to phishing attacks

User susceptibility to phishing refers to the likelihood of an individual falling for a phishing attack (Anawar, Kunasegaran, Mas'ud, & Zakaria, 2019).

Prior studies such as that of Frank, Jaeger, and Ranft (2022), Valecha, Gonzalez, Mock, Golob, and Raghav (2020), and Frauenstein (2019) suggest that contextual elements such as the sender's address and related visual signs can impact a user's response to an email. Indicators of susceptibility to phishing are associated with psychosocial factors, which include individual personality traits and corresponding interpersonal behaviours. Personality traits, including the Big Five - openness, conscientiousness, extraversion, agreeableness, and neuroticism - are inherent aspects of a person's character and are known to shape their interactions, decision-making

processes, and reactions to job uncertainties or stresses. Social engineering attacks are designed to take advantage of these personality traits.

2.4.1 Content and design of phishing emails

Certain studies have delved into how the physical characteristics of phishing emails affect user vulnerability to these attacks. Jakobsson and Ratkiewicz (2006) conducted an experiment wherein they sent phishing emails containing intentionally suspicious-looking links, including spelling errors, escape characters, and exposed IP addresses in the URL. Their results revealed that between four and 14 percent of users still clicked on these deceptive links. In a separate study, Jakobsson, Tsow, Shah, Blevis, and Lim (2007) explored how trust markers in emails and webpages influenced their trustworthiness. They concluded that 1) a sophisticated layout and legal disclaimers promote trust, 2) an excessive focus on security can be detrimental, 3) individuals rely heavily on URLs to determine trust, and 4) third-party endorsements' effectiveness depends on recognition of the endorsing party's name.

In a different research, Wang, Herath, Chen, Vishwanath, and Rao (2012) investigated how users react to visual hints and phishing detection indicators within phishing messages and how it influences their decision-making. They discovered that attentiveness to emotional triggers, phishing detection signs, and users' phishing awareness significantly impacts phishing detection. Harrison, Vishwanath, Ng, and Rao (2015) determined that suggestive information about social presence promotes heuristic decision-making and amplifies susceptibility to phishing. They manipulated social presence with visual hints such as the university logo, security logo, and clickable chat icons.

Phishing emails utilize a combination of misleading content and intricate design to exploit human vulnerabilities, thereby tricking users into jeopardizing their own or their organization's security. This is achieved by using various tactics that play on these vulnerabilities (Goel, Williams, & Dincelli, 2017).

One way they manipulate victims is through the use of urgent language in content. Phishing emails often convey a sense of urgency to incite immediate action, capitalizing on a user's scarcity bias, a phenomenon where a higher value is placed on resources or opportunities deemed to be in short supply (Verma & Hossain, 2014).

Another common tactic is impersonation, where phishers pose as trusted entities such as banks, service providers, or colleagues within an organization. This strategy is used to earn the user's trust and bypass any initial scepticism, thereby making the scam more convincing (Gupta, Gupta, Ahamad, & Kumaraguru, 2015). Moreover, these malicious emails may use threats or rewards to manipulate the recipient's emotions and judgment. They might threaten account suspension or offer enticing rewards like prize money, exploiting human tendencies such as loss aversion or greed, making the phishing attempt more effective (Harrison, Svetieva, & Vishwanath, 2016).

Regarding design, phishing emails often mimic the visual branding of legitimate organizations. This includes elements such as logos, colour schemes, and layout, which are replicated to lend credibility to the deceptive email (Dhamija, Tygar, & Hearst, 2006). Furthermore, hidden URLs and disguised links are used to further fool the user. Links within these emails might be made to appear as legitimate URLs or use characters that look similar visually, thereby deceiving users into clicking on them, leading to potentially disastrous security breaches. These deceptive practices showcase the

sophisticated and manipulative nature of phishing attacks, highlighting the importance of awareness and education in combating this threat (Chiew et al., 2018).

2.4.2 The influence of personality traits on susceptibility to phishing attacks

Human factors play a significant role in determining the success or failure of phishing attacks. One such factor is the influence of personality traits on an individual's susceptibility to these attacks. Research has shown that certain personality traits, such as conscientiousness, agreeableness, and openness to experience, can affect an individual's likelihood of falling victim to phishing attempts (Halevi, Memon, & Nov, 2015). For instance, individuals with higher levels of conscientiousness may be more vigilant in scrutinizing emails and less likely to click on suspicious links (Vishwanath, 2015). Conversely, individuals with high levels of agreeableness may be more trusting of others and, therefore, more susceptible to phishing attacks that exploit this trust (Sumner, Byers, & Shearing, 2011).

Parrish, Bailey, and Courtney (2009) proposed potential connections between the Big Five personality traits and susceptibility to phishing, though there was limited empirical data to validate these hypotheses. According to their research; people who are curious and eager to explore the content behind a hyperlink might be more susceptible to phishing attacks (openness), individuals who are detail-oriented may be able to discern subtle distinctions between genuine emails and phishing attempts (conscientiousness), a desire for attention and acceptance might increase a person's vulnerability to phishing (extraversion), a tendency to trust false assertions, comply with instructions to click links, and a selfless inclination to believe in a scam can all amplify susceptibility to

phishing (agreeableness), and those who are nervously vigilant and reluctant to share information may be more resistant to phishing attempts (neuroticism).

The study of Ge, Lu, Cui, Chen, and Qu (2021) explores the factors that make individuals susceptible to phishing attacks, also focusing specifically on the influence of the Big Five personality traits, a person's knowledge and experience, and the cognitive processing involved when dealing with emails. Drawing from a sample of 414 Chinese participants, the research incorporated various metrics like the Big Five Personality Inventory (BFI-44), Mail Elaboration Scale (MES), Web Experience Questionnaire, Experience with Electronic Mail Scale, and Knowledge and Technical Background Test.

Results suggested that people with low levels of conscientiousness and openness, and high neuroticism were more prone to falling victim to phishing attacks. Experience with the internet and knowledge about computers and web were also found to be significant factors that indirectly affected susceptibility to phishing by influencing how one elaborates emails. Furthermore, the likelihood of someone seeking more information or deleting an email is indicative of their sensitivity in judging email legitimacy. The study unveils the role cognitive processing plays in mediating the relationship between individual characteristics and phishing susceptibility, providing valuable insights for future research in this domain and potential applications in creating effective phishing risk interventions or training programs.

Understanding the relationship between personality traits and phishing susceptibility can help organizations develop targeted training and awareness programs to reduce their employees' vulnerability to phishing attacks.

2.4.3 Awareness levels and their impact on cybersecurity behaviour

Awareness of cybersecurity threats and best practices is another crucial human factor influencing individuals' cybersecurity behaviour (Nurse, 2021). The research has shown that individuals with higher levels of cybersecurity awareness are less likely to engage in risky behaviours, such as clicking on suspicious links or downloading unverified attachments. Organizations can improve their employees' cybersecurity awareness through training programs, awareness campaigns, and ongoing communication regarding current threats and best practices. By increasing their employees' awareness levels, organizations can not only reduce the likelihood of successful phishing attacks but also foster a culture of cybersecurity responsibility that helps protect the organization as a whole (Yan, 2019).

2.4.4 The role of cognitive biases in decision-making related to cybersecurity threats

Cognitive biases can significantly impact individuals' decision-making processes, especially in the context of cybersecurity threats. These biases can lead to irrational decisions, such as disregarding warning signs, trusting fraudulent messages, or underestimating the risks associated with certain behaviours (McAlaney & Benson, 2020).

In the context of phishing attacks, cognitive biases provide insight into how people process emails and whether a person clicks on links (Yang et al., 2022). This understanding can also inform the design of user interfaces and warning systems that help users make better-informed decisions when faced with phishing attacks and other cybersecurity threats (Chen, Zahedi, & Abbasi, 2011).

In conclusion, the role of human factors in cybersecurity research is crucial for understanding the complex dynamics of phishing attacks and other threats. By incorporating insights from this research into their cybersecurity strategies, organizations can better protect themselves against the ever-evolving landscape of cyber threats and enhance their overall cybersecurity resilience.

2.5 Socio-technical theory

Socio-technical theory highlights the interdependence of social and technical components within complex systems like organizations and their information systems (Baxter and Sommerville, 2011).

When applied to cybersecurity, socio-technical theory provides a comprehensive approach to understanding and addressing complex cybersecurity challenges (Bada et al., 2019). Optimizing the technical system involves implementing appropriate security measures, but these solutions must be designed to fit the needs and capabilities of the users. Similarly, optimizing the social system involves understanding and addressing the human factors that contribute to cybersecurity risks, such as conducting regular cybersecurity awareness training and fostering a security-conscious culture (Puhakainen & Siponen, 2010). Building upon socio-technical theory, in a phishing attack context, it can be said that an employee's inclination to click on a phishing email, and hence, adherence to email security policy, is influenced by technological, individual, and organizational factors (Shahbaznezhad et al., 2021).

Socio-technical approaches have been applied in various phishing and malware research studies, emphasizing the need to consider both human and technological factors when addressing these threats. Ostby and Kowalski (2021) evaluated how organizations

managed data breaches during the COVID-19 crisis, using Gjøvik municipality as a case study. This organization closed their email system and shut down macros to prevent a crisis overload. The study analysed their decision in a socio-technical and crisis management context, proposing a similar approach for other organizations to prevent both data breaches and a crisis overload simultaneously.

The H2020 PANACEA project (Anastasopolou et al., 2020) introduced a socio-technical modelling approach to capture health service specificities and map cybersecurity interventions, thereby offering a useful tool for both public and private healthcare organizations. A socio-technical approach incorporating adequate system protection, reliable system defence, continuous monitoring, and rapid response was proposed to mitigate ransomware threats in the study of Sittig and Singh (2016).

Another study (Mwakalinga, & Kowalski, 2011) presented a socio-technical security model for systems defence, proposing security as a function of the states an attacker can produce over the states that can be controlled in defence.

Finally, considering the vulnerability of small and medium enterprises (SMEs) to cyber threats, Perozzo, Zaghloul, and Ravarini (2022) proposed a CyberSecurity Readiness Model for SMEs (CSRM-SME) based on a Socio-Technical view of organizations. The model was tested on three SMEs to assess their cybersecurity readiness and to further understand the environment and strategies adopted to prevent and manage cyber-attacks. This study highlighted the overlooked aspect of cybersecurity readiness in SMEs, despite them being among the most vulnerable and least mature in terms of cybersecurity resilience and risk.

2.6 Complex adaptive systems

Complex adaptive systems (CAS) comprise numerous interacting components that adapt based on their experiences and interactions (Holland, 2006). CAS instances can be found in various contexts, such as global economies, social systems, organizations, ecosystems, cultures, political landscapes, technologies, traffic, and weather patterns (Burns et al., 2017).

Choi, Dooley, and Rungtusanatham (2001) reviewed the CAS literature and developed a comprehensive framework encompassing its elements and features. This framework is made up of three key interrelated elements: (1) internal mechanisms, (2) co-evolution, and (3) environment. In a CAS, the behaviour of agents is typically directed by fairly simple rules (i.e., internal mechanisms) leading to the emergence of 'self-organized' behaviour patterns. For instance, in a capitalist economy, business firms' inclination to maximize profits and consumers to optimize utility results in efficient resource allocation. The presence of non-linearity, extensive connectivity, and dynamism create a 'rugged environment' where it is challenging, if not impossible, to devise optimal mathematical models and solutions.

2.6.1 Cybersecurity as a complex adaptive system

Securing IS is a crucial responsibility in any organization and it involves managing the interaction between a host of players, which include both well-meaning and malicious insiders in the organization as well as external parties. This means that the environment where security needs to be maintained constitutes a CAS where organizational agents interact with each other and their surroundings regularly. These agents and their

environment mutually adapt and evolve as security within the organization emerges through adaptive behaviours.

Defining a complex system universally is challenging. Usually, such a system is marked by nonlinear, emergent, and adaptive behaviour (Miller & Page, 2009). In a cyber-context, this implies a dense network of interrelations that make up a cyber-ecosystem. The true complexity of this system comes to the forefront when we take into account the interactions between its dimensions from both internal dynamics and its relation to the information landscape.

Seeing how a dynamic system evolves over time requires a holistic approach, keeping in account all possible variables to figure out its adaptivity or response to transformations. This panoramic perspective fosters the idea of a cyber-ecosystem as a unique instance of a complex system, essentially a Complex Adaptive Ecosystem – comparable to a CAS.

Acknowledging the human aspect in cyberspace paints a picture of a coherent system comprising technical and social factors that consistently interact in a resilient, sustained manner. It's essential to understand that the structure and behaviour of the cyber ecosystem changes over time, impacting its success or failure. But, the alterations in the cyber ecosystem's structure and behaviour trigger certain challenges, particularly the cognitive complexities linked to the development of cyber awareness and understanding (Olagbemiro, 2019).

The development of both cyber awareness and understanding hinges significantly on the ability to detect intruders and anomalous conditions. Furthermore, cyber understanding is dependent on the capability to analyse and correlate the information gained from the observations to understand the sources and intent of an attack,

facilitating a response to the threat. The growth of cyber awareness and understanding remains vitally important, and can only be tackled when there is a profound recognition that cyberspace is a complex adaptive system.

Given the crucial role of the human dimension, cyberspace as a CAS within a larger information landscape essentially morphs into a socio-ecological ecosystem. With this recognition comes the understanding that the traditional attributes of a CAS, such as nonlinearity, emergent behaviour, and the balance between chaos and non-chaos, are directly applicable in cybersecurity operations (Olagbemiro, 2019).

2.7 Agent-based modelling in cybersecurity: an overview

Agent-based modelling (ABM) is a computational method that enables the simulation of actions and interactions of autonomous agents, aiming to assess their effects on the system as a whole (Bonabeau, 2002). The agents can represent anything from cells in a biological organism to individuals or organizations in a society. At its core, agent-based modelling involves creating “agents”, each of which is programmed with certain behaviours, objectives, and decision-making rules. These agents then interact with each other and their environment in a virtual “world”, and their behaviours and interactions can be observed and analysed to gain insights into the system's behaviour (Macal & North, 2010). ABM enables the application of CAS methodologies. These can tackle the behaviour exhibited by each participant within complex systems (North, Macal, & Campbell, 2005).

One of the main advantages of agent-based modelling is its ability to capture the complexity and heterogeneity of real-world systems. Unlike traditional modelling approaches that often rely on aggregate-level data and assumptions of homogeneity,

agent-based models can incorporate the diversity and complexity of individual behaviours and interactions, making them particularly suitable for studying complex adaptive systems like organizational networks (Crooks & Heppenstall, 2011).

ABM is an emerging and valuable method that is yet to be widely utilized within the Information Systems (IS) field. It assists in studying the complex interplay and resulting emergent patterns of micro-level agent behaviours within a CAS over time. Even though it's a relatively novel approach within IS, initial studies highlight its potential for probing into matters associated with IS security (Burns et al., 2017).

Firstly, Rajivan, Janssen, and Cooke (2013) focused on the dynamics within cyber defence analyst teams has shown the significance of ABM in this field. The researchers developed an ABM to investigate how different team sizes and collaboration strategies affect performance outcomes, such as the accurate processing of intrusion alerts. The results indicated that specific collaboration strategies enhance performance and that larger team sizes can hinder performance. This study also showcased the potential of ABM as a methodology for investigating team processes within cyber defence.

Thompson and Morris-King (2018) applied ABM to the complex environment of mobile tactical networks. In these networks, the military-inspired hierarchical command structures and spatial aspects add layers of complexity that are well-suited to ABM. The researchers simulated military units' operations on a synthetic battlefield and used the model to study malware spread and the impact of hierarchy and cybersecurity policies on it. This work concluded that ABM is highly suitable for representing these intricate structures and encouraged others to incorporate its key elements in similar studies.

Mohammed, Gunasekaran, Mostafa, Mustafa, and Abd Ghani (2018) aimed to overcome existing limitations in spam detection by including visual information and text-based analysis in the spam filtering process. This model, called the Multi-Natural Language Anti-Spam (MNLAS), was implemented using a Java environment and a dataset of 200 emails to detect and filter different types of spam.

Lastly, Burns et al. (2017) posits that the management of information security can be understood as a CAS. In such a system, the interactions among individuals and their environment at a micro-level form the overall security posture at a macro-level. This research presented agent-based models to illustrate simple phishing problems and simulate organizational security outcomes using different theoretical security approaches. It proposed the use of ABM to model the complexity of information security risks and organizational responses.

Despite the evident potential of ABM in understanding complex systems like cybersecurity, its application in this field, particularly in studying phishing attacks, remains remarkably limited. The scarcity of ABM-based research in phishing attacks denotes a significant gap in the current academic literature and provides a compelling justification for this study.

The use of ABM in cybersecurity primarily focuses on the analysis of the impacts of diverse strategies and structures on cybersecurity posture. The singular agent-based phishing attack model created by Burns et al. (2017) signifies the emerging trend of ABM applications in this specific arena. However, their work did not delve into the nuanced effects of individual characteristics like personality traits on the dynamics of phishing attacks, which is an area our study addresses.

By incorporating personality traits within an agent-based model to simulate malware-based phishing attacks and worm propagation within an organizational network, this study seeks to contribute to the emerging literature in this domain. The exploration of personality traits as factors influencing user responses to phishing attempts enriches the complexity of the simulation, creating a more realistic representation of an organization's cybersecurity landscape.

Furthermore, the unique integration of individual, technological, and organizational factors in our agent-based model will add depth and breadth to our understanding of the multifaceted nature of cybersecurity. This comprehensive approach transcends the traditional technocentric perspectives of cybersecurity, offering a holistic view that better reflects the socio-technical reality of modern organizations.

2.7.1 Agent-based modelling tools

There is a wide range of tools available for constructing agent-based models. Each tool has its own advantages and disadvantages, providing researchers with flexibility in their choice of modelling platform. Several modelling platforms specifically designed for agent-based modelling exist, offering built-in features that assist in model development, execution, visualization, data collection, and analysis. The five widely used agent-based modelling platforms seen in the literature are: Multi-Agent Simulator of Neighbourhoods (MASON), NetLogo, Repast, Swarm, and Java Swarm. The selection of a modelling platform depends on understanding the strengths and limitations of each platform and aligning them with the requirements of the specific agent-based model under development (Railsback, Lytinen, & Jackson, 2006).

Among the available options, NetLogo was chosen as the most suitable platform/language for this research project for several reasons. Firstly, NetLogo provides thorough, clear, and comprehensive documentation, ensuring ease of use and a smooth learning curve. Secondly, its programming syntax and semantics are straightforward and user-friendly. Additionally, NetLogo offers a customizable user interface with useful components tailored for agent-based modelling. The platform also includes built-in features and tools that facilitate the creation and analysis of agent-based models. NetLogo's popularity in the research community, extensive library of model examples, and its maturity as a platform further supported its selection. Lastly, NetLogo is freely available for use, making it accessible to researchers without financial constraints. Table 4 summarizes the strengths and weaknesses of the aforementioned ABM tools as well as their practical implications.

Table 4. Comparison of Key Characteristics, Strengths, Weaknesses, and Practical Applications of Popular ABM Tools

ABM Tool	Strengths	Weaknesses	Practical Applications
MASON	Provides a robust, fast, and scalable simulation environment. Strong visualization capabilities.	Lower-level programming may be difficult for non-programmers.	Best for larger, complex simulations that require efficient execution and strong visualization.
Netlogo	Easy-to-learn and use, suitable for educational purposes. Well-documented with an extensive library of existing models.	Less suitable for very large or computationally intensive simulations.	Best for beginners, educators, and rapid prototype development.
Repast	Offers rich set of features and tools for complex simulations. Flexible, with support for several programming languages.	Higher learning curve compared to tools like Netlogo.	Ideal for users comfortable with Java or Python seeking comprehensive features for complex simulations.
Swarm	Developed for complex adaptive systems. Allows low-level customization.	Less active community and documentation. Not as user-friendly as some alternatives.	Best for users comfortable with Objective-C and need high customizability.
Java Swarm	A Swarm port for Java. Inherits Swarm's features and strengths while utilizing Java's widespread usage.	As with Swarm, it has less active community and support.	Good choice for users familiar with Java who need high customizability and are comfortable navigating less documentation.

2.7.2 The potential of agent-based modelling for studying phishing attacks and malware propagation

In the context of this study, agent-based modelling can be used to examine the dynamics of phishing attacks and malware propagation in an organizational network. By creating agents that represent different elements of the system, such as users, devices, and phishing emails, and programming them with behaviours and decision-making rules based theory, an agent-based model can simulate the process of a phishing attack and the subsequent propagation of malware in the network. This can provide valuable insights into the factors that contribute to the success or failure of such attacks and the potential effectiveness of different countermeasures.

One of the key strengths of ABM lies in its ability to represent emergent phenomena (Fievet & Sornette, 2018), which result from the interactions and adaptations of individual agents within the system (Kang & Aldstadt, 2019). ABM provides a valuable platform for testing and evaluating the effectiveness of various countermeasures and strategies to mitigate phishing attacks and malware threats. Lastly, ABM allows for integrating human factors, such as personality traits, awareness levels, and cognitive biases, as well as socio-technical dimensions of phishing attacks (Bennett, 2021).

In summary, agent-based modelling offers a powerful tool for studying the dynamics of phishing attacks and malware propagation in organizational networks. By capturing the complexity and dynamics of these threats, agent-based models can provide insights that can inform the development of more effective cybersecurity strategies. Despite the advancements in our understanding of cybersecurity threats and mitigation strategies, several gaps remain in the current body of research. There is potential for

more extensive application of this method, especially in the context of phishing attacks and malware threats. There is also a need for more studies that combine agent-based modelling with socio-technical theory and the concept of complex adaptive systems to provide a more comprehensive understanding of cyber threats.

After the extensive review of the current body of knowledge on cybersecurity, phishing attacks, malware propagation, complex adaptive systems, socio-technical theory, and agent-based modelling, it becomes clear that while considerable progress has been made in these fields, significant gaps still need to be addressed. This study aims to address these gaps by applying a socio-technical perspective and the concept of complex adaptive systems to study phishing attacks and malware threats in organizational networks.

CHAPTER 3

AGENT-BASED PHISHING ATTACK MODEL DESIGN

"He who loves practice without theory is like the sailor who boards ship without a rudder and compass and never knows where he may cast."

— Leonardo da Vinci

This chapter comprehensively explores the elements and processes for modelling the propagation of malware-based phishing attacks within organizational networks. It highlights the phishing attack model components using agent-based modelling. In an agent-based model, agents are distinct entities that have specific properties, behaviours, and decision-making capabilities. In our model, an agent is a computer-user pair. Agent state transitions refer to changes in the status, properties, or behaviours of agents over time. These transactions can be influenced by internal rules within the agent, interactions with other agents, or changes in the environment that is the network topology in our case. How malware propagation affects agent state transitions, is a key aspect of the proposed model.

The chapter begins by examining two distinct network topologies - hierarchical and random, and their effects on worm propagation. The hierarchical network mirrors an organizational structure with parent-child nodes, symbolizing department heads and team members. Conversely, the random network exemplifies a flat organizational layout where all nodes have an equal probability of being linked. Studying these topologies aids in understanding the influence of network structure on the spread of malware-based phishing attacks.

Following this, the chapter delves into the role of Endpoint Detection and Response (EDR) tools and antivirus software in network defence. It discusses how these tools can enhance resilience to attacks and the significance of their strategic placement within the network. Moreover, the chapter touches on different antivirus scanning policies and the effectiveness of IT teams in controlling infections.

Adapting the Susceptible-Infected-Recovered (SIR) model widely used in epidemiology and public health research, the chapter further discusses the agent state transitions, which serve as the foundation of this model. The agents' states range from susceptible to infected, detected, known-susceptible, and resistant. The intricate transition process among these states is illustrated, emphasizing the dynamic nature of malware propagation within a network.

3.1 The proposed phishing attack model elements

The configurable parameters of the model encompass human, organizational, and technical components. A depiction of the model's configurable components can be found in Figure 2. Each group of parameters is independently configurable, however, certain components are interconnected and impact one another such as network topology and EDR placement.

The components chosen for the model were guided by several factors. Firstly, a literature review was conducted to identify key components that have been commonly used in similar studies and models. These were then evaluated based on their relevance to the specific context of our research - in this case, the simulation of malware-based phishing attacks and worm propagation within an organizational network.

Secondly, due to the complex nature of phishing attacks, the components were selected to capture a broad range of factors influencing the success of these attacks. These include technological factors, human factors, and organizational factors, aligning with the socio-technical systems approach adopted in our research.

While the selected components are comprehensive and align with the objective of our research, it is important to note that these are not the only components possible or available. Agent-based modelling is highly flexible and can incorporate a wide variety of parameters. However, it was necessary to balance the complexity of the model with its interpretability and computational feasibility. Including too many components could make the model overly complex and difficult to interpret, while also increasing the computational demands.

Finally, the choice of components was validated through a cybersecurity and research expert in the field to ensure that the chosen components are not only theoretically sound but also practical and realistic in the context of phishing attacks in organizational networks.

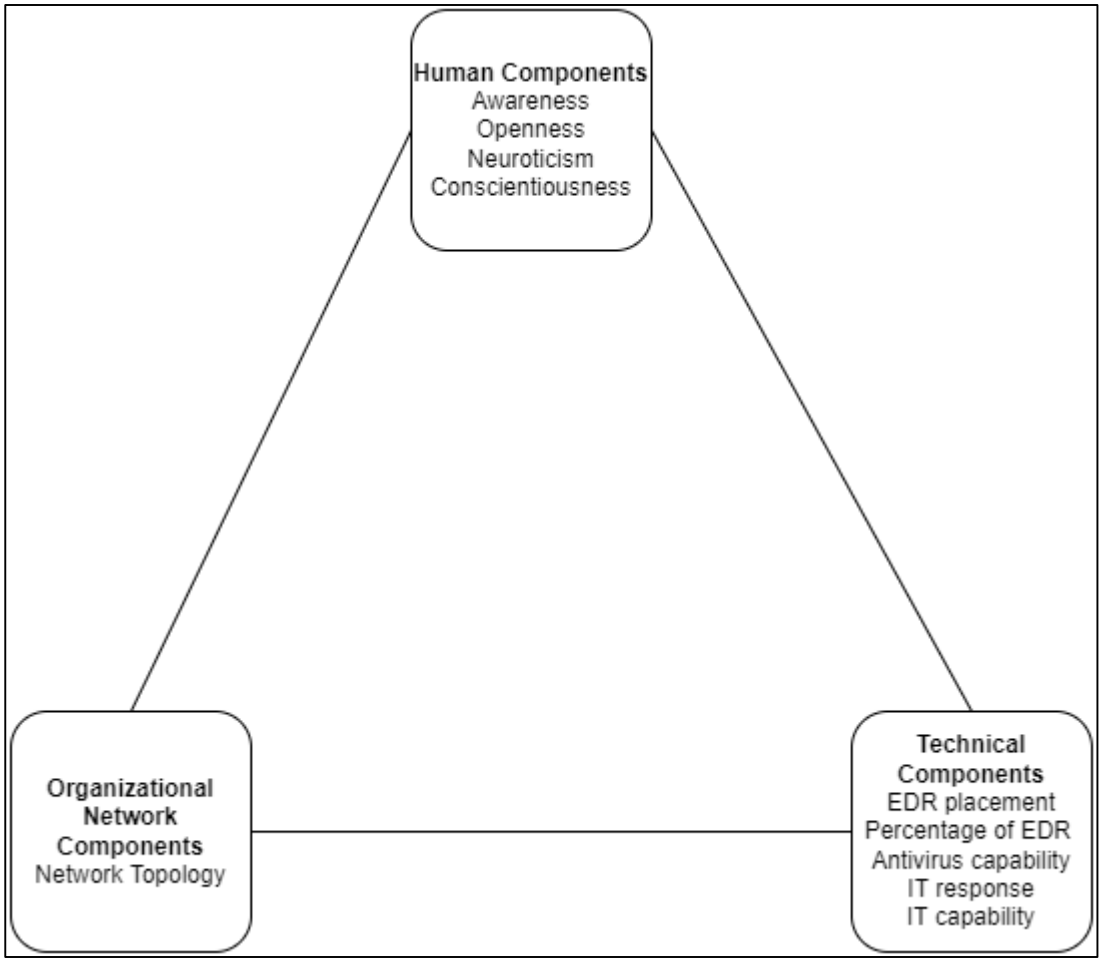

Figure 2. Configurable HOT model components

3.1.1 Human components of the proposed model

A comprehensive exploration of the literature regarding the influence of unique personality traits reveals no agreement on which traits notably impact susceptibility to phishing attacks. However, according to the literature, our model includes the personality traits that significantly affect susceptibility to phishing email attacks. As per the study by Ge, Lu, Cui, Chen, and Qu (2021), it is suggested that neuroticism increases the likelihood of further email scrutiny for additional data, suggesting that high neuroticism correlates with victim personality.

High conscientiousness reduces phishing susceptibility by enhancing email elaboration. Other research (Yeo & Neal, 2008) confirms that conscientiousness contributes to improved email elaboration. Email elaboration refers to the frequency of detailed cognitive processing of email content, specifically, careful consideration of the merits of email information and correlating these cues with prior know experiences, and beliefs. Higher openness, along with awareness, has been found to positively influence email elaboration.

Using the coefficients corresponding to these traits to the log-odds of further inspecting a phishing email, formulas were devised to calculate the email elaboration (ME), additional email checking (CE), and the likelihood of perceiving an email as phishing (PP) for each agent:

$$ME(i) = w_c * conscientiousness(i) + w_o * openness(i) + w_a * awareness(i) \quad (1)$$

$$CE(i) = w_n * neuroticism(i) + w_{me} * ME(i) \quad (2)$$

$$PP(i) = {}^{1}/_{(1 + e^{(-k*CE(i)+a)})} \quad (3)$$

Where w represents the coefficients (weights) of each trait. Ultimately, using the phishing perception probability, a risk probability (P) is computed for each agent:

$$P(i) = 1 - PP(i) \quad (4)$$

The higher the likelihood of perceiving an email as phishing, the lower the risk of clicking on it.

This network includes humans with various characteristics based on parameters such as awareness, neuroticism, openness, and conscientiousness. Personality traits are generally measured on a discrete scale (like the 1-5 scale often used in the Big Five personality test). The Poisson process is a discrete process (Zhang, Zhao, & Chang, 2012), making it well-suited for these parameters. Consequently, this model employs a conditional Poisson distribution to create a network filled with various human characteristics. The above formulas account for these personality traits, making this model more reflective of actual organizations.

Lastly, as mentioned in the background and related work section, the content and design of phishing emails impact user susceptibility to phishing attacks. Hence, the phishing email parameters, mailDesignQuality and mailContentQuality, represent the average quality of the phishing email in question. This parameter ranges from 1-5, and this model assumes that the middle value does not impact a user's susceptibility to the phishing email. Values higher than this midpoint increase the likelihood of an agent clicking on a phishing email's malicious link, while values lower than the midpoint increase the perception of the email as phishing, consequently reducing the likelihood of an agent clicking on a malicious link. The average of these values are named as "credibility".

The following algorithm (Algorithm 1) depicts the process of the effect of mail credibility on a person's risk probability.

Algorithm 1 Mail Credibility

if $\boldsymbol{credibility > 3}$ **then**
  $\boldsymbol{P(i)} \leftarrow \boldsymbol{P(i)} + ((\boldsymbol{credibility} - 3) * \boldsymbol{credibilityCoef})$
else if $\boldsymbol{credibility < 3}$ **then**
  $\boldsymbol{P(i)} \leftarrow \boldsymbol{P(i)} - ((3 - \boldsymbol{credibility}) * \boldsymbol{credibilityCoef})$
else
  continue
end if

3.1.2 Organizational network components of the proposed model

The thesis examines how different organizational network topologies influence the propagation of malware-based phishing attacks. We study worm propagation across two distinct network topologies. These two distinct network types, which can be generated from their corresponding Network submodels in the Netlogo simulator, serve as the basis for our experiments.

The first network type is hierarchical, resembling a tree-like structure is handled, where parent nodes are connected to child nodes. This design simulates an organizational hierarchy with parent nodes symbolizing department heads and child nodes representing team members. Secondly, random network topology is considered. The random network topology embodies a flatter organizational structure, devoid of any top-level nodes, and every node shares the same probability of connection.

The chosen network type not only impacts the spread of malware-based phishing but also alters the worm's scanning policy. Within the hierarchical network, the probability of worm spread hinges on direction, which is derived from the network's depth based on organizational levels. This network operates on the assumption that a top-down infection holds a higher likelihood of propagation. This is attributed to the worm's behaviour within the network, replicating emails and dispatching them to the

contact addresses of infected agents. This network presumes emails originating from higher authority levels are perceived as more legitimate, enhancing their propagation potential. Additionally, the network assumes a worm's likelihood of propagation improves with proximity to the host IP addresses. Consequently, the network employs a weighted distance-based approach, fostering random interdepartmental node connections, potentially spreading infection in both top-down and bottom-up directions.

The core idea lies in mirroring the essence of a real-world organizational structure. Here, the agents represent individuals within an organization, and the links delineate the interaction or relationship that exists between these individuals. The construct of levels within this model denotes the hierarchical strata typically found in organizational setups, with individuals at higher levels (closer to 0) reflecting positions of higher authority or rank.

One of the key features of this model is the concept of "weight" associated with each link. Interpreted within the context of an organizational framework, this "weight" corresponds to the intensity or significance of the relationship between two interconnected individuals. The model has been designed to assign a higher weight to relationships existing between individuals placed at successive hierarchical levels (akin to a manager and their direct subordinates), while a lower weight has been assigned to relationships amongst individuals occupying the same hierarchical level (equivalent to peer-level interactions).

An essential component of this model is the calculation of the shortest path between any two individuals, taking into consideration the weights associated with the relationships involved. This computation mirrors the concept of "communication distance" in real-world organizational scenarios. Here, a shorter weighted distance

signifies a more potent or direct communication pathway. The model presupposes a higher likelihood of interaction or influence between individuals situated closer to each other in terms of this weighted distance. For instance, in a typical organization, a subordinate is more likely to be influenced by their direct supervisor (owing to the shorter weighted distance) than by another individual situated several hierarchical levels away.

In contrast, the random network assumes that all nodes have an equal chance of spreading infection. This is because every node is connected in the same manner, and each connection has the same likelihood. Figure 3 provides the visuals for the network topologies.

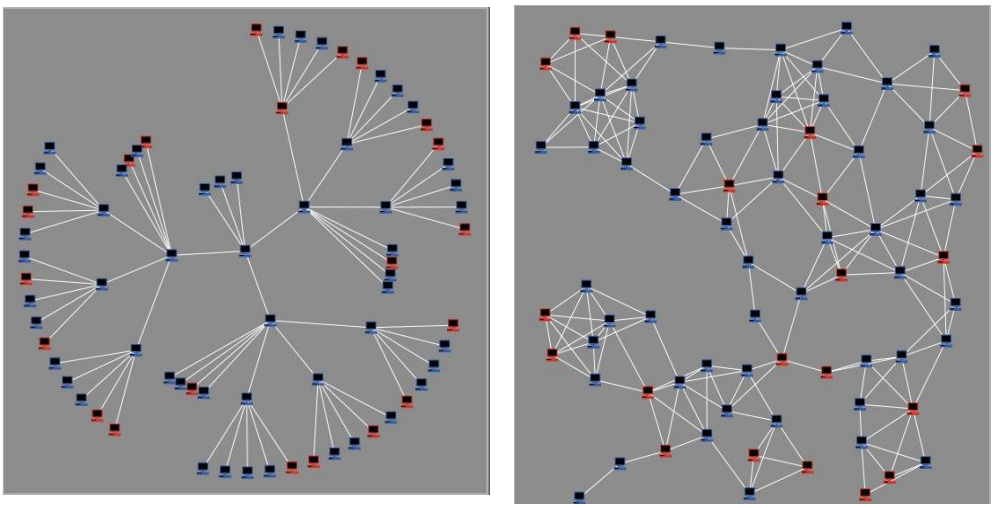

Figure 3. Hierarchical (left), Random (right)

3.1.3 Technical components of the proposed model

The spread of malicious software, such as worms, across a network is closely related to the precautions taken by organizations, especially the use of end-point security solutions like antivirus and EDR. Considering end-point security, the proposed model includes

technical parameters such as EDR and antivirus tools. Furthermore, the capability of IT personnel and response likelihood which may depend on the number of computers per IT personnel is also key.

3.1.3.1 EDR policy

EDR tools function by continuous surveillance of endpoint devices' activities, and raising threat alerts when they detect potential malicious behaviour (Hassan, Bates, & Marino, 2020). Our model presupposes that agents equipped with EDR display greater resistance. The adjustable parameters related to EDR encompass the proportion of agents possessing EDR and the specific EDR placement strategy employed.

Given the substantial financial investment required for EDR tools, not all organizations can afford to equip every computer agent within their network. Thus, the strategic placement of EDR is pivotal in maximizing effectiveness while minimizing expenditure. There are two primary EDR placement strategies to consider: hierarchical or random. Hierarchical placement situates EDR primarily at pivotal points or "parent" nodes. If the number of nodes set to house EDR tools (determined by the EDR percentage parameter) exceeds the count of parent nodes, the remaining EDR tools are dispersed randomly. This approach is only compatible with the hierarchical network topology. Conversely, random EDR placement disperses EDR tools indiscriminately, a strategy that can be utilized across both network topologies.

3.1.3.2 Antivirus policy

The model's antivirus and IT parameters play a significant role in the network's infection cleaning rate. Two configurable aspects are attached to the antivirus tool: the likelihood

of the antivirus effectively cleaning worms and the antivirus tool's scanning policy as set by the organization. The success rate of antivirus in eradicating an infection directly impacts the infection rate within the network.

Organizations can select from three possible antivirus scanning frequency policies: quick scan, deep scan, and complete scan. The quick scan policy assumes daily network scans by the antivirus tool. On the other hand, deep scan policies depend on the network infrastructure and the scanning policy, offering a comprehensive network scan every t days, where t differs from one organization to another. Lastly, the complete scan is a hybrid method, combining elements of both quick and deep scan policies. This method implies daily quick scans supplemented by in-depth scans at a specific frequency.

Each method carries inherent trade-offs. Quick scans, while faster and less demanding on the network infrastructure, offer lower probabilities of detecting infected agents. In contrast, deep scans, although more resource-intensive and potentially disruptive to network functionality, yield higher probabilities of discovering infected nodes.

Two configurable parameters are associated with IT: response and capability. Response refers to the likelihood of an IT team attending to a detected infected node, while capability indicates the success probability of the IT team in cleaning the infected node.

3.2 Agent state transitions

This model is structured on the foundation of the SIR model, a straightforward mathematical paradigm widely employed in epidemiology to comprehend disease

propagation within a population. The model segregates the population into three categories (Palomo-Briones, Siller, & Grignard, 2022):

Susceptible (S): This category includes individuals who have yet to be infected but are susceptible to the disease.

Infected (I): This group encompasses individuals currently infected with the disease, who are capable of transmitting it to susceptible individuals.

Recovered (R): This segment comprises individuals who have been infected, have since recovered, and are now immune to the disease.

This thesis study has adapted and expanded upon the SIR model to suitably simulate the propagation of malware-based phishing attacks within an organizational setting. The state transition diagram for this proposed model is given in Figure 4.

The initial state, Susceptible (S), corresponds to the condition of an agent that has not yet encountered a worm but could become infected eventually. The subsequent state, Infected (I), represents agents that have interacted with a worm, become infected, and can now disseminate the worm to other susceptible agents. Every susceptible agent (S) is susceptible to infection with an average probability (γ) per unit time, potentially due to contact with infected computers within the network. This probability γ hinges on the direction of spread, network topology, and the computer security level.

As the worm spreads across the network, antivirus software might detect an infected agent with a probability α, transitioning the agent into the Detected (D) state. Infected agents that evade detection ($1 - \alpha$) can be regarded as asymptomatic infections, significantly contributing to the spread rate as they continue to infect other susceptible agents.

Detected agents stand a chance to be cleared of infection (albeit temporarily), contingent on the capabilities of the antivirus software and the IT team. Agents that have been successfully cleaned transition to a Known-Susceptible state (KS), maintaining the same probability of future re-infection (γ). The probabilities of an agent transitioning from the Detected (D) to Known-Susceptible (KS) state are β_{α} and β_{IT}, which represent the likelihood of antivirus software or the IT team clearing the infection, respectively.

Detected agents also possess a probability Ɛ of becoming Resistant (R), dependent on the IT team's effectiveness in cleaning the infected node. Agents that cannot be cleaned due to infection complexities are assumed to be disconnected from the network and isolated, thereby considered resistant as they can no longer spread the infection to other agents. Lastly, detected agents that haven't been isolated from the network remain infected, posing a risk to other susceptible and known-susceptible agents within the network. η denotes the duration of unit time that a previously detected agent remains infected. A diagram illustrating the state transitions of agents is provided in Figure 4.

The quantity of nodes within each compartment dynamically fluctuates over time. Consequently, five variables, S(t), I(t), D(t), KS(t), and R(t), are introduced to denote the numbers of susceptible, infected, detected, known-susceptible, and resistant nodes at a specific time t, respectively. The network size at time t is represented by

$$N(t) = S(t) + I(t) + D(t) + KS(t) + R(t) + R(t) \qquad (5)$$

This model assumes that the network is static, indicating that no new agents enter or exit the network at any given time. Therefore, the SIDKsR model essentially presents

a novel framework for modelling the propagation of a malware-based phishing attack within an enterprise network.

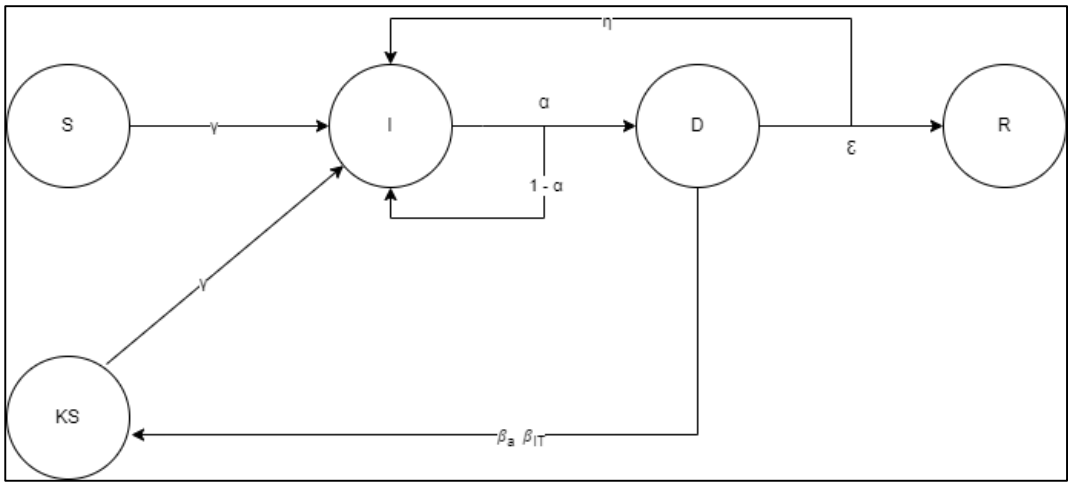

Figure 4. Agent state transitions diagram

We can represent the state transitions mathematically with the following differential equations:

$$\frac{dS(t)}{dt} = -\gamma * S(t) * I(t) \quad (6)$$

$$\frac{dI(t)}{dt} = \gamma * S(t) * I(t) - \alpha * I(t) + \gamma * KS(t) * I(t) - \eta * I(t) \quad (7)$$

$$\frac{dD(t)}{dt} = \alpha * I(t) - (\beta_a + \beta_{IT} + \varepsilon) * D(t) + \eta * I(t) \quad (8)$$

$$\frac{dKS(t)}{dt} = (\beta_a + \beta_{IT}) * D(t) - \gamma * KS(t) * I(t) \quad (9)$$

$$\frac{dR(t)}{dt} = \varepsilon * D(t) \quad (10)$$

Equation 6 gives us the rate at which Susceptible nodes are becoming Infected. The negative sign indicates that the number of Susceptible nodes is decreasing. It

assumes that each interaction between a Susceptible and an Infected node has a constant probability γ of causing an infection.

Equation 7 gives us the rate at which nodes are becoming Infected or being detected as Infected. It includes terms for Susceptible nodes becoming Infected (γS(t)I(t)), Infected nodes being Detected (αI(t)), Known-Susceptible nodes becoming Infected again (γKS(t)I(t)), and Detected nodes that are still Infected (ηI(t)).

Equation 8 depicts the rate at which nodes are being Detected as Infected or moving to other states from Detected. It includes terms for Infected nodes being Detected (αI(t)), Detected nodes becoming Known-Susceptible or Resistant (β_aD(t), β_ITD(t), εD(t)), and Detected nodes that are still Infected (η*I(t)).

Equation 9 tells us the rate at which nodes are becoming Known-Susceptible or being reinfected. It includes terms for Detected nodes becoming Known-Susceptible due to antivirus or IT actions ($\beta_a D(t)$, $\beta_{IT} D(t)$), and Known-Susceptible nodes being reinfected (γ*KS(t)*I(t)).

Lastly, equation 10 depicts the rate at which nodes are becoming Resistant. It includes a term for Detected nodes becoming Resistant due to IT actions (ε*D(t)). Table 5 provides the notations for the state transitions.

Table 5. State Transition Notations

Notation	Description
γ	Probability of becoming infected
α	Probability of being detected
1 – α	Probability of not being detected
Ɛ	Probability of becoming resistant
η	Period of infection of detected agent
β_a	Probability of an infected agent being cleaned by antivirus software
β_{IT}	Probability of an infected agent being cleaned by IT team

This model assumes that the incubation period of the infection is insignificant, meaning an agent gets infected and recovers either permanently (resistant) or temporarily (known-susceptible) instantly. However, for this study, it's crucial to factor in the duration an agent requires to transition from one state to another, referred to as transition time.

The algorithm for the state transitions can be found below (Algorithm 2).

Algorithm 2 State Transitions

for each *agent* in network:
  if agent.state $=$ "$Susceptible$" **then**
    if $\boldsymbol{\gamma} > \boldsymbol{random}(\mathbf{0},\mathbf{1})$ **then**
      agent.state $=$ "$Infected$"
    else
      continue
    end if
  else if agent.state $=$ "$Infected$" **then**
    if $\boldsymbol{\alpha} > \boldsymbol{random}(\mathbf{0},\mathbf{1})$ **then**
      agent.state $=$ "$Detected$"
    else
      continue
    end if
  else if agent.state $=$ "$Detected$" **then**
    if $\boldsymbol{\beta_{\alpha}} > \boldsymbol{random}(\mathbf{0},\mathbf{1})$ **then**
      agent.state $=$ "$Known - Susceptible$"
    else if $\boldsymbol{\beta_{IT}} > \boldsymbol{random}(\mathbf{0},\mathbf{1})$ **then**
      agent.state $=$ "$Known - Susceptible$"
    else if $\mathcal{E} > \boldsymbol{random}(\mathbf{0},\mathbf{1})$ **then**
      agent.state $=$ "$Resistant$"
    end if
  else if agent.state $=$ "$Known - Susceptible$" **then**
    if $\boldsymbol{\gamma} > \boldsymbol{random}(\mathbf{0},\mathbf{1})$ **then**
      agent.state $=$ "$Infected$"
    else
      continue
    end if
  end if
end for

3.3 Model outputs

The model generates a range of outputs at each tick, which can be analysed for understanding the dynamics and effectiveness of various defensive strategies. These outputs include: number of phished agents, initial infected agents, infected-per-tick, peak-simultaneous-infections, never-infected, computers infected more than once, mean times infected, states of computers (S, I, D, KS, R), number of EDR solutions, number of computers detected by antivirus software, computers cleaned by antivirus, computers cleaned by IT personnel, duration of infection, time from infection to detection, time from detection to clean. This model also calculates the significant costs that go along with such a complex system. Cost of infection and cost of investment are the main cost factors. Cost of infection depicts the cost of downtime of the system based on the number of nodes infected and the duration that computers are incapacitated. The second cost factor is investment costs. This costs comprises of cost of antivirus software, cost of EDR solutions, cost of IT personnel, and cost of awareness training.

Among the main output, number of infected agents and total time of infection; the cost of infection and cost of investment plays a crucial role in determining which strategies to implement. All model outputs can be found in Table 6.

Table 6. Model Outputs

Output	Description
Tick	The total number of ticks it took for the system to be cleared
Initial-infected	The number of agents that clicked on the phishing email and subsequently became infected
Detected-computers	The number of infected agents that were detected by the antivirus software
Infected-per-tick	The number of newly infected agents within the network
Total-infected	The cumulative total number of infected agents
Susceptible	The number of susceptible agents within the network
Peak-simultaneous-infections	The maximum number of infected agents at a single tick
Av-cleaned-computers	The number of computers cleaned of infection by antivirus software
IT-cleaned-computers	The number of computers cleaned of infection by IT personnel
Resistant	The number of resistant agents within the network
Phish-click	The number of agents that clicked on the phishing email
Mean-alarm	The average duration from detection of infection to clean
Known-susceptible	Agents that have been cleaned of infection by antivirus or IT but are still susceptible
Never-infected	Agents that have never been infected
Times-infected-more-1	The number of agents that have been infected more than once
Max-no-times-infected	The maximum number of times any agent has been infected
Avg-days-inf	The average number of days of infection for the network
Edr	The number of edr solutions in place
Mean-infection-detection	The average duration of agents becoming infected to being detected by antivirus software
Mean-infection-duration	The average duration of infection
Cost-of-infection	The total cost of infection
Cost-of-investment	the total cost of investment including IT team cost, software cost, and awareness training cost

3.3.1 Cost factors

This model includes various costs associated with managing and mitigating malware-based phishing attacks. The cost elements encapsulated within this model are broad, comprising costs related to IT personnel, software tools, and user business disruptions, as well as costs like awareness training. These costs consist of expenditures on IT team

efforts for cleaning and isolating infections, the employment of software tools including antivirus and EDR solutions, and the consequences of user business disruptions caused by infections. Other costs focus on investment in awareness training to improve user behaviour in the face of phishing threats. The aim is to determine the cost of investment and the cost of infection for various strategies, thereby facilitating more informed decision-making concerning cybersecurity investment and strategy optimization.

Cost of infection considers the downtime of a systems' components, namely the user's loss of productivity. Cost of investment on the other hand, includes all technical components found within the model such as antivirus and EDR tools and IT personnel.

In summary, this chapter offers an in-depth examination of the complex components and transitions involved in modelling malware-based phishing attacks within an organizational context. It highlights the importance of strategic security tool deployment, network structure understanding, and comprehensive analysis of agent state transitions as well as cost factors for effective network defence. The agent-based model explained in this chapter was implemented using Netlogo software (Wilensky, 1999) the codes of which can be found in Appendix A.

CHAPTER 4

PHISHING ATTACK SCENARIOS AND EVALUATION

"Now, here, you see," says the Red Queen, "it takes all the running you can do, to keep in the same place."

– Lewis Carroll, Alice in Wonderland

This chapter examines the fundamental building blocks of our model, primarily focusing on the design and examination of various phishing attack scenarios. This serves as the cornerstone of our agent-based model and allows us to meticulously scrutinize the effectiveness of multiple mitigation strategies in the context of a highly dynamic organizational computer network. The chapter further dissects these scenarios through local sensitivity analyses, adjusting one parameter at a time while keeping the remaining parameters constant. By doing so, we're able to assess the robustness and resilience of different countermeasures against phishing attacks, uncovering their strengths, weaknesses, and potential areas for improvement. This analytical approach assists us in offering well-founded insights that can drive the development of more robust anti-phishing techniques and policies.

The scenarios are based on different versions of the original model. The scenarios manipulate human-centred, IT, and network components. The properties of these components vary to simulate the behaviours of real organizational networks. The scenarios measure the number of days it takes to clear the system of the virus, the number of network components infected, and the cost of investment along with the cost

of infection. The research describes and analyses the outputs from the different scenarios to provide possible mitigation strategies and policies.

4.1 General assumptions of all phishing attack scenarios

The following assumptions hold for all subsequent phishing attack scenarios of the agent-based model.

I. The models run on a discrete time scale where each tick represents a day
II. The computer security level of all agents are the same
III. All agents are assumed to be susceptible at initialization except for the agents with EDR solutions in place
IV. The size of the network is the same with 73 agents (representative of a small to mid-sized enterprise)
V. There is only an initial attack. The models simulate the propagation and behaviour of a single malware-based phishing attack
VI. The model runs for a maximum of 365 ticks (days)

All scenarios where run 20 times using BehaviorSpace, part of the Netlogo software used for running experiments of models created in Netlogo. Our choice of conducting 20 runs for each scenario in BehaviorSpace was guided by the following principles. First, from the standpoint of statistical robustness, multiple runs are essential in a stochastic simulation to estimate the mean and variability of outputs (Lee et al., 2015; Seri & Secchi, 2017), and 20 runs were found to be an adequate number to provide these estimates without overtaxing computational resources. Second, considering computational feasibility, more runs would necessitate higher computational

power and time, thus we found that 20 runs provided an optimal balance. Lastly, our decision was also informed by the observed stability of our model in preliminary tests, which suggested minimal benefits from numerous additional runs.

4.2 Phishing attack scenario 1: the baseline case

This scenario comprises the baseline for all models. The model starts by initializing the agent and global parameters. These parameters can be found in Table 7.

Table 7. Scenario 1 Parameters

Parameter	Value	Description
NetworkType	Hierarchical	Hierarchical network topology
Openness	3	Average value of openness
Conscientiousness	3	Average value of conscientiousness
Neuroticism	3	Average value of neuroticism
Awareness	3	Average value of awareness
mailDesignQuality	3	Quality of design cues of phishing email
mailContentQuality	3	Quality of content of phishing email
β_a	0.75	Probability of antivirus tool cleaning infected agent
$\beta_{IT\text{-}R}$	0.4	Probability of IT team responding to an infected agent
$\beta_{IT\text{-}C}$	0.5	Probability of IT team cleaning an infected agent
%-edr	0.15	Percentage of agents with an EDR solution in place
edr-placement	Random	EDR solutions are placed at random
virus-scan-policy	Complete	Complete scan policy in place
virus-check-frequency	7	Deep scan is conducted at every 7^{th} tick

The Setup procedure of the model (see Appendix B for flowchart) first initializes the network topology as well as the placement of EDR finally initiating the initial infection procedure. Algorithm 3 provides the algorithm for the setup procedure.

Algorithm 3 Setup Procedure

```
if network − type = "hierarchical" then
    call network − hier
    call create − random − links
else
    call network − rand
end if

call initialize − globals
call initialize − turtle − variables

set no − of − edr − solutions to round (edr − percentage *
count of turtles)
if edr − placement = "hierarchical" then
    call hierarchical edr
else
    call random edr
end if

call initial infection
```

The initial infection is based on the probability of individual agents clicking on the phishing attack email subsequently, the phished agents' computer security is compared with the worm strength. If the computer security is not sufficient compared to the strength of the worm in question, the computer transitions to the Infected state. The agents that have been infected in the setup procedure make up the number of initially infected agents. Algorithm 4 provides the algorithm for the initial infection procedure.

Algorithm 4 Initial Infection

```
if P(i) > random(0, 1) then
    if computerSecurity < wormStrength then
        become infected
    else
        continue
    end if
else
    continue
end if
```

Figure 5. gives the percentage of users that have been phished and the percentage of computers that have been infected. This base model does not take phishing email credibility into account and user personality traits are set to average values. Based on these values, 32% of users fell for the phishing emails and subsequently 28% of the computers in the network have been infected. The computers infected are the computers of the phished 28% of users.

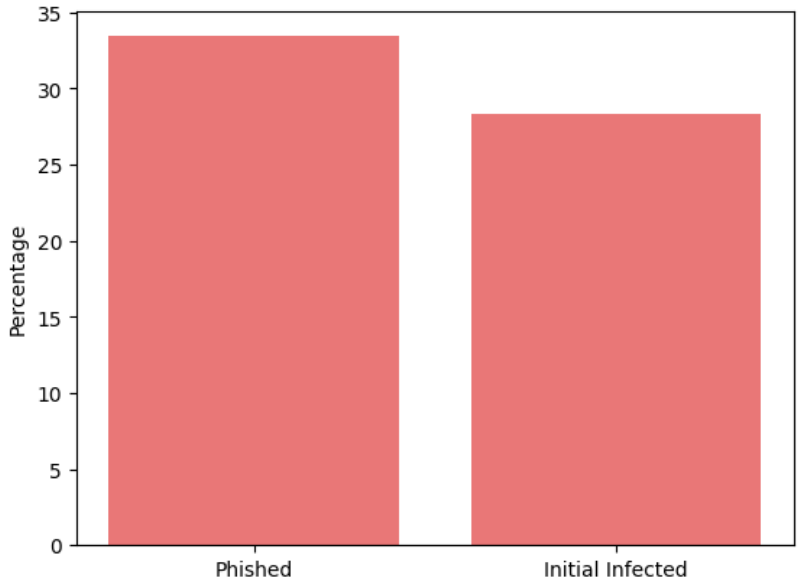

Figure 5. Percentage of users phished and initial infected

The model goes on to conduct the Go procedure (see Appendix B for flowchart) which runs the virus spread and virus scan submodels. If the network has been cleared of infections before 365 ticks, the model provides a system clear message and provides various outputs for analysis provided in-depth in the previous chapter. Algorithm 5 provides the algorithm for the Go procedure.

Algorithm 5 Go Procedure

```
   if ticks = 365 then
      stop
   end if

   if ticks = 0 and all turtles infected = False then
      display "Phishing Attack Failed."
      stop
   end if

   if ticks > 0 and all turtles infected = False then
      display "System Clear."
      stop
   end if

for each turtle:
     virus − check − timer = virus − check − timer + 1
      if virus − check − timer mod(virus − check − frequency) = 0 then
         set virus − check − timer = 0
      end if
   end for

   set infected − per − tick = 0

   if network − type = "hierarchical" then
      for each turtle with infected = True
         for each network − link −
neighbour of turtle with resistant and infected = False
            if neighbour − id > turtle − id or neighbour − level = turtle −
level and neighbour − group = turtle − group then
               call spread − virus2
            else
               call spread − virus
            end if
         end for
      end for

      for each turtle with infected = True
         for each random − link −
neighbour of turtle with resistant and infected = False
            call spread − virus − random
          end for
      end for

   ticks = ticks + 1
   call calculate − costs
```

Figure 6 provides the values of agents in different states; Infected, Susceptible, Known Susceptible, and Resistant. As expected, over time (tick), the number of infected and known susceptible decrease as the system is cleaned of the infection. Whereas, the number of resistant agents increase as time goes by. The number of susceptible decrease and then remain at a constant level till the end of the run. This is due to the number of EDR solutions in place. At setup, all agents are assumed to be susceptible except for the agents with EDR in place. As the model progresses, the infection spreads to susceptible agents, making them transition from susceptible to infected state, therefore lowering the number of susceptible. The number of agents in the Susceptible state lowers at each tick until it reaches the number of EDR solutions. These agents will never be infected and will remain susceptible.

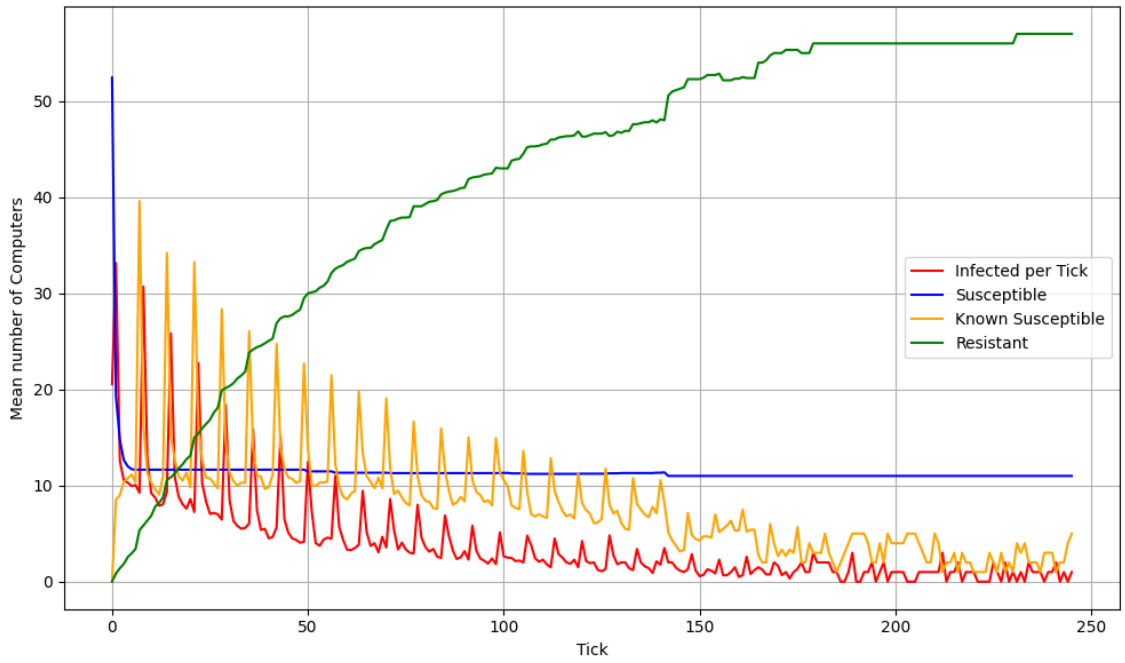

Figure 6. Agent states for base scenario

4.3 Scenarios for assessing the effect of awareness training and phishing mail factors on being phished

The aim of these scenarios are to understand the effect of human factors on the outcome of user susceptibility to phishing email attacks. The scenarios were based on the base scenario changing only one parameter at a time to evaluate the impact of the change in a specific parameter on the number of phished users.

4.3.1 Awareness training

Using the base model, one-at-a-time sensitivity analysis was conducted by changing one parameter at a time while keeping all other parameters constant. We run the base model for each value of awareness each for 20 runs keeping all other parameters constant. The base model was run changing the awareness parameter while keeping all other parameters the same as the base model values. The percentage of phished users is given for all values of awareness in Table 8.

Table 8. Local Sensitivity Analysis of the Awareness Parameter

Scenario	Value	% Phished
Base Model	3	33.43
Scenario 1	1	47.06
Scenario 2	2	39.63
Scenario 3	4	26.60
Scenario 4	5	25.76

Figure 7 displays the percentages of phished users as a boxplot. It is evident that when the user's awareness averages on the lowest level (1), the percentage of phished users increases to nearly half of the agents. While increasing awareness levels of users by way of awareness training decreases the percentage of users being phished. This

outcome is assuming that the content and design of the phishing mail attack does not have an effect on likelihood of being phished.

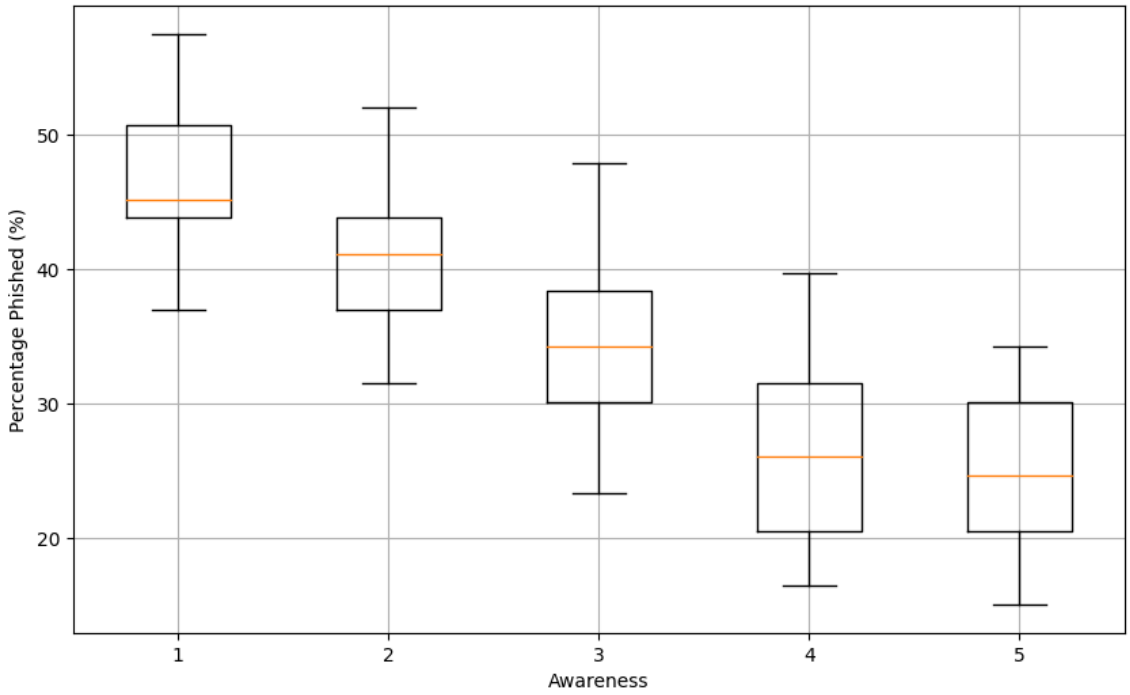

Figure 7. Percentage phished based on awareness levels

4.3.2 Phishing attack email content and design

We also run the same model for each value of phishing mail factors as these are assumed to influence a user's perception of a phishing email. We run the base model for the lowest (1) and highest (5) average value of content and design for 20 runs keeping all other parameters constant. The scenarios and the average percentage of users phished can be found in Table 9. We name the average of email content and design variables as "credibility".

Table 9. Percentage Phished Based on Email Credibility

Scenario	Value	% Phished
Base Model	3	33.43
Scenario 1	1	22.49
Scenario 2	5	88.30

Figure 8. depicts the percentages found in Table 4 as a boxplot. It can be seen that the credibility of a phishing email positively affects the users' perception of said email. The better the design and content of the phishing attack email, the higher the credibility and the more convincing the email is to users. The higher the credibility, the higher the likelihood of a user perceiving the email as legitimate, subsequently clicking on any malicious links and infecting the network.

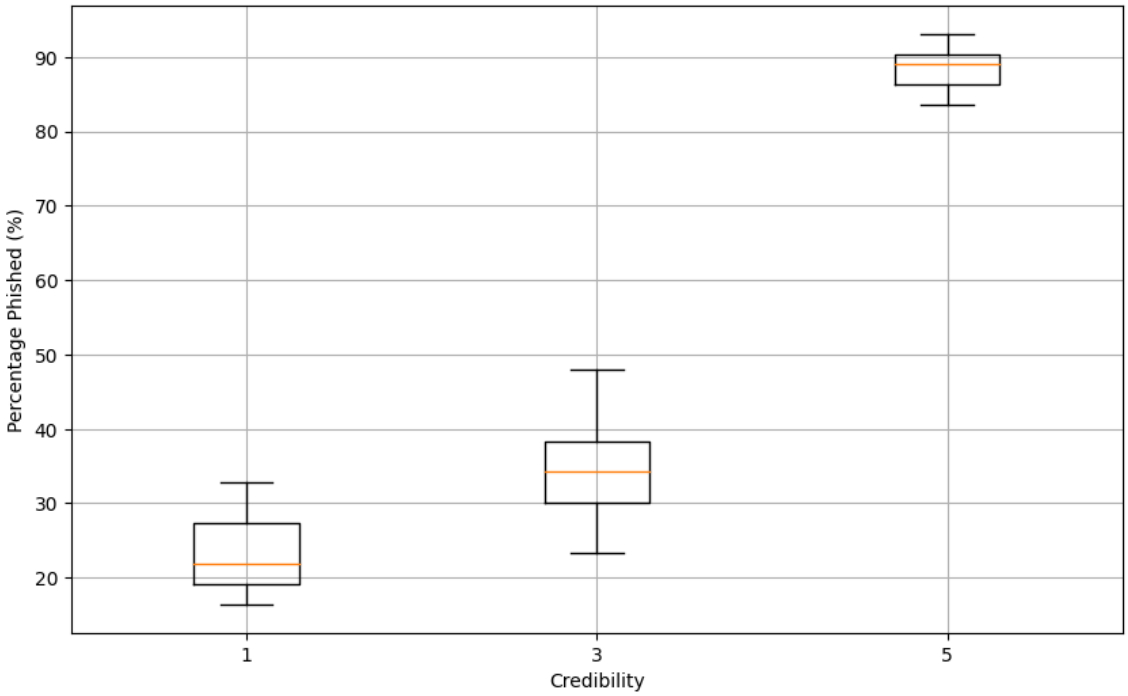

Figure 8. Percentage phished based on email credibility

4.4 Scenarios assessing the effect of network topology

Two scenarios were run 20 times each to understand the effect network topology has on the number of infected agents as well as the spread of infection. The base model network topology is hierarchical. This base model was compared with the random network topology model. All other parameter values are the same.

Looking at Figure 9, it can be said that the random network topology has lower number of infected agents throughout the runs compared to the hierarchical network topology. This can be due to the fact that with the hierarchical network topology there are more ways for the infection to spread among agents as well as the different and higher probability of spread. The random network topology assumes that all nodes have the same probability of spread and these nodes are randomly connected.

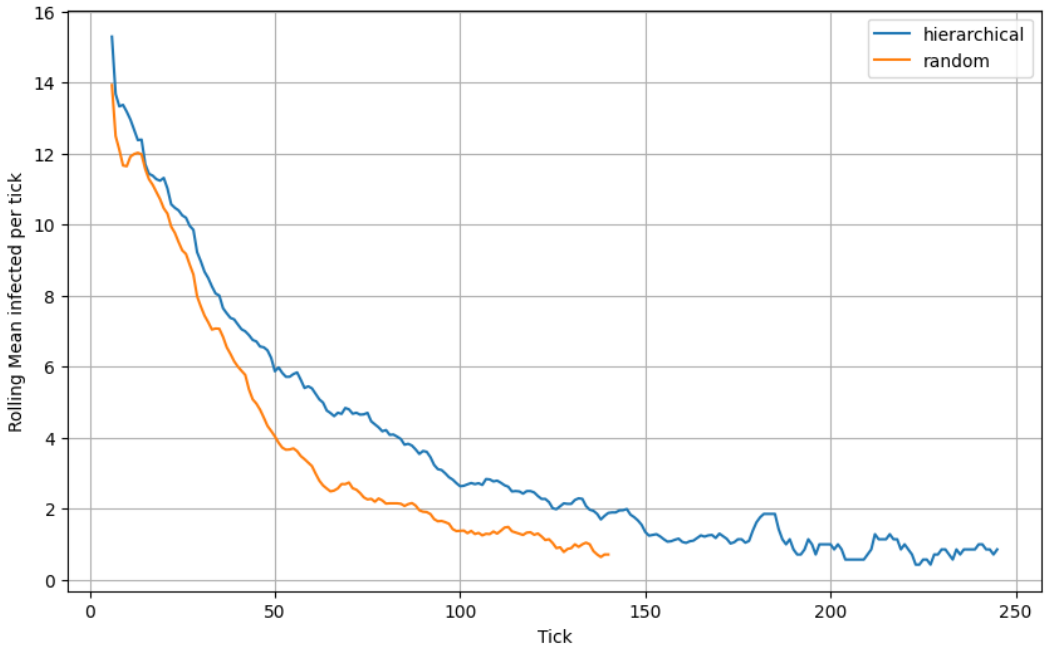

Figure 9. Average number of infected agents for hierarchical and random network topology

However, Figure 10 shows that the mean duration of infection is slightly higher for the random network topology.

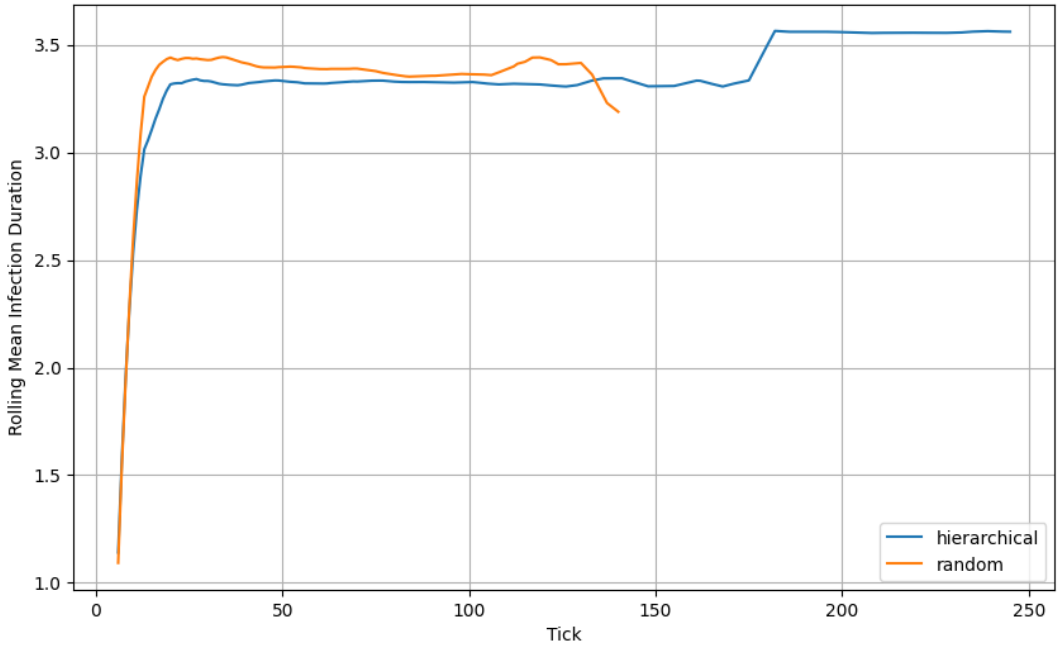

Figure 10. Mean infection duration for hierarchical and random network topology

4.5 Scenarios assessing the effect of antivirus policy

These scenarios compare the different types of scan policy an organization can implement. The base model utilized a complete scan policy approach. This is a hybrid of quick and deep scan where a quick scan of the network is conducted at every tick and a deep scan is conducted every t ticks based on the deep scan frequency, which is another parameter that has been changed to understand its effect on multiple output measures.

Firstly, two scenarios based on the deep scan frequency parameter are run. The base model assumes a deep scan is conducted at every 7 ticks. The deep scan has a higher probability of detecting infected agents than quick scan, however it is costlier on the network as it relies on more computational power.

Figure 11 provides the average number of infected agents over time for deep scan policy of t = 7 (base scenario) over t = 2. As is clear, the number of infected agents drops significantly for a policy of deep scan every 2 ticks. The scenario employing a more frequent deep scan policy also is quicker to clear the network than the base model scenario. This is due to the higher probability of detecting and therefore cleaning infected nodes. The base model allows the infection to spread faster as it only scans the network every 7 ticks, making it harder to clear the network.

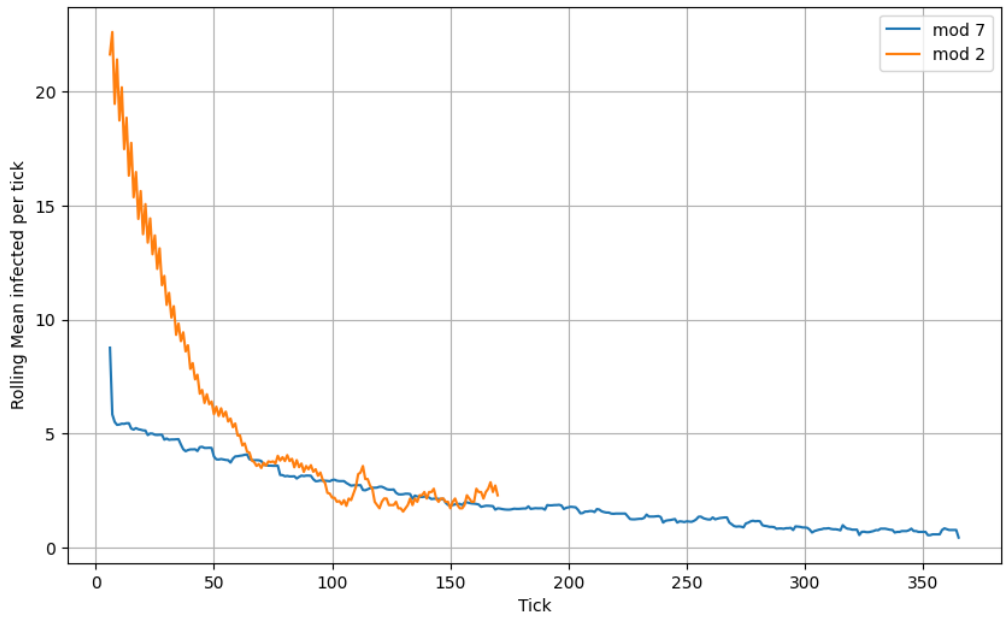

Figure 11. Average infected based on deep scan frequency

The average infection duration (Figure 12) corroborates the previous explanation. As it is clear that the average duration of infection for agents is significantly higher than that of the more frequent scan policy scenario. Again, this is due to only scanning the network for infections every 7 ticks.

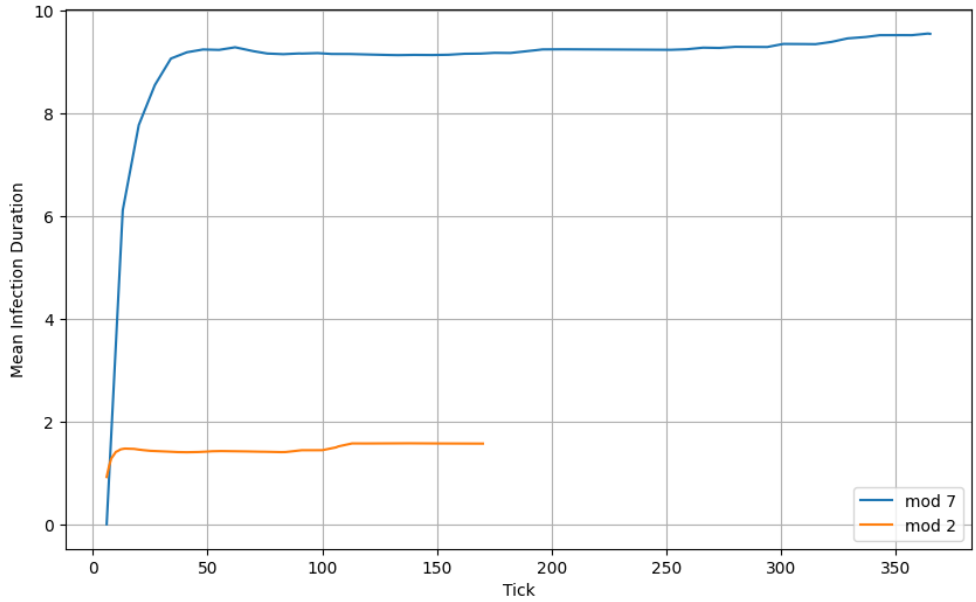

Figure 12. Average infection duration based on deep scan frequency

The next scenarios compare the different scan types to understand how they affect the number of infections, infection duration, average time from infection to detection, and average time from detection to clean. Figure 13 compares the three scan types in terms of number of infected agents. While the complete scan type – the base model scenario- starts with a high number of infections compare to quick and deep scan policies, it rapidly decreases and enables the network to be cleaned faster than the other two scan types as mean infection duration is the shortest (Figure 14).

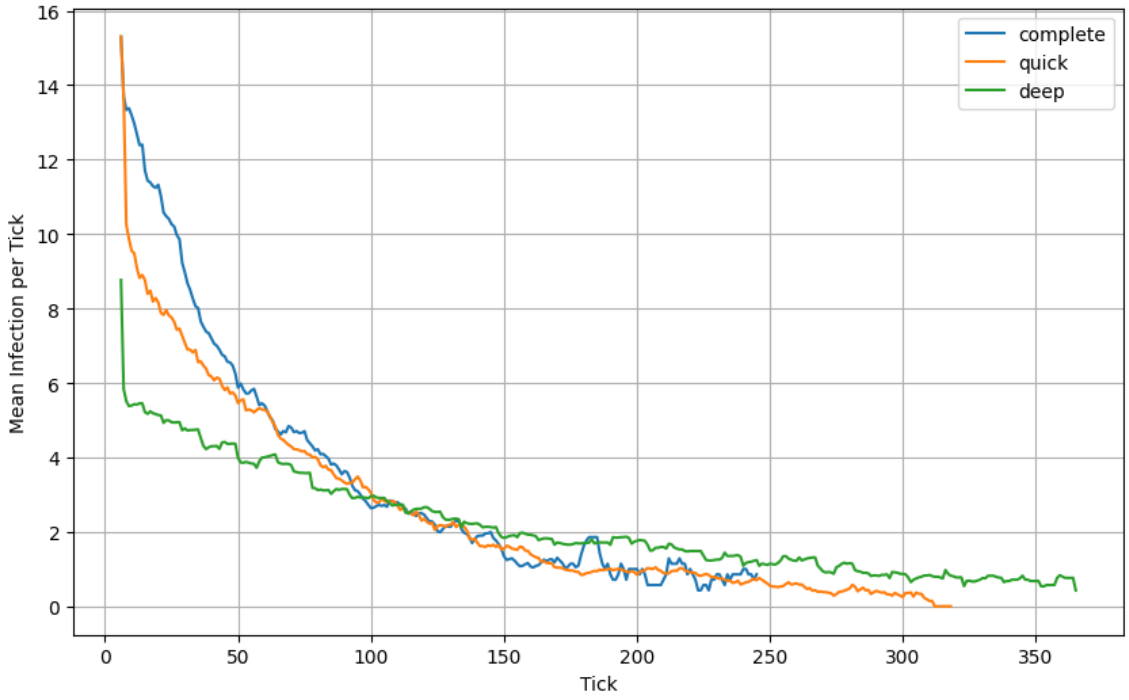

Figure 13. Average infections based on scan types

On the contrary, deep scan starts with a lower number of infections, however takes longer to clean the network overall as the mean infection duration is the longest of the three. Based on these two measures (number infected and infection duration), it can be said that employing a hybrid scan policy is the most effective in terms of infection duration.

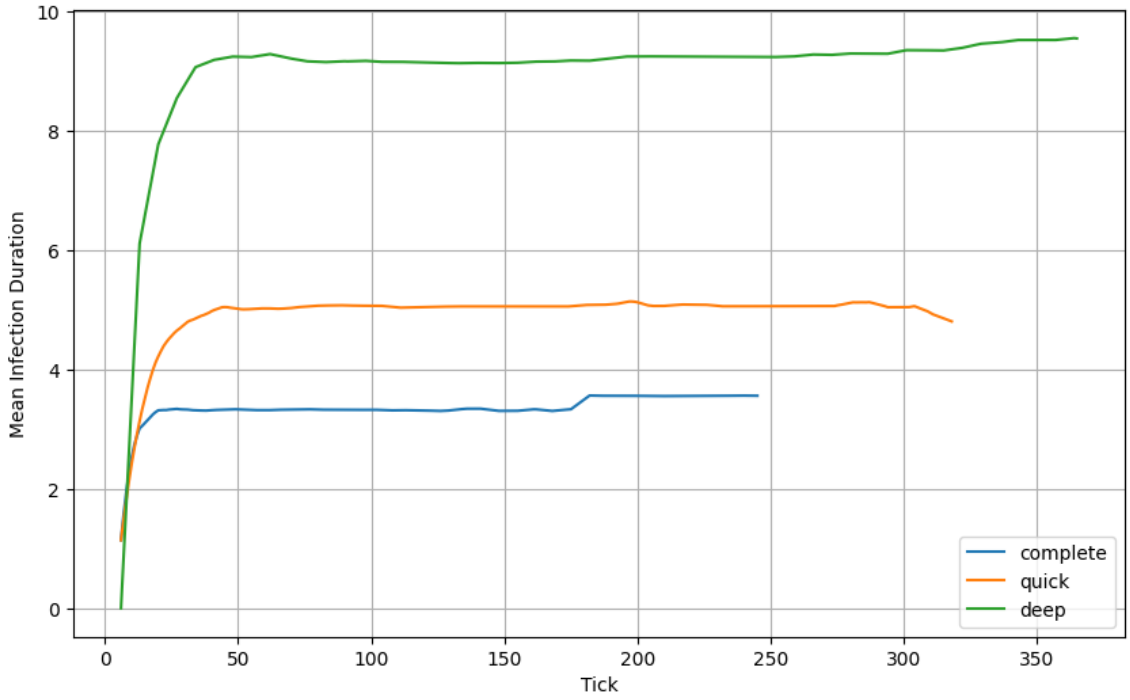

Figure 14. Average infection duration based on scan types

In terms of average time from infection to detection (Figure 15); the hybrid scan policy (complete) performs better as it produces shorter infection to detection times. For the hybrid scan policy, an infected agent is in the infected state for an average of three ticks, while an infected agent is in this state for an average of 8 ticks.

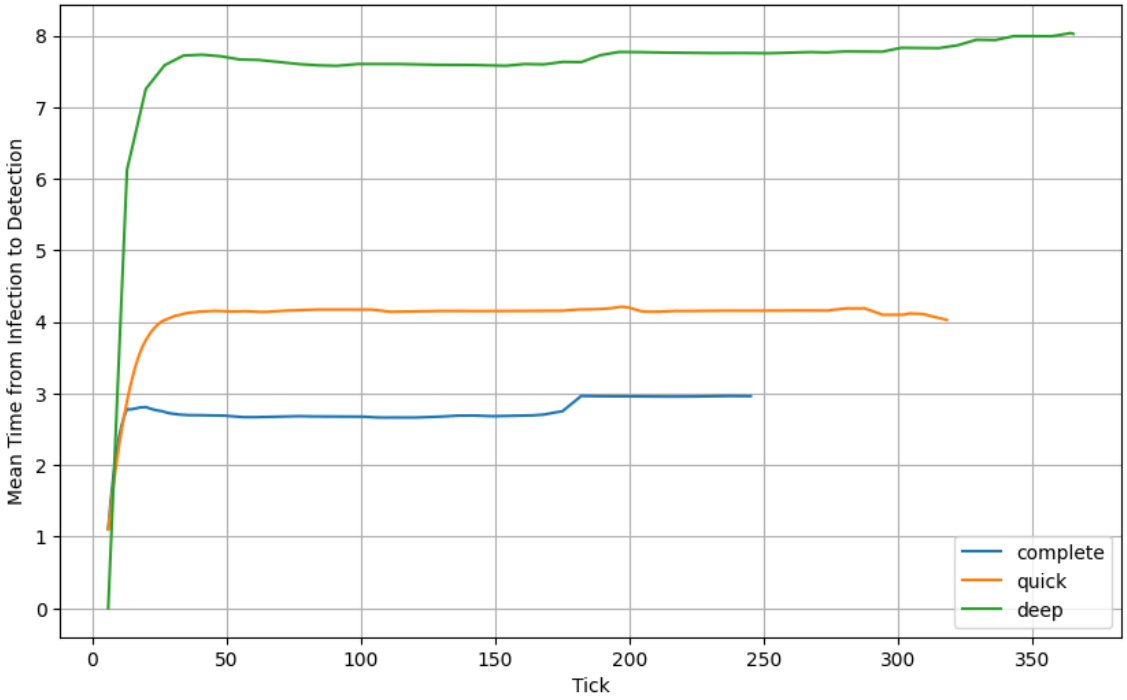

Figure 15. Average duration from infection to detected based on scan type

Figure 16. provides a boxplot of the cost of infection for different scan types. It makes sense that the average cost of infection is highest for the deep scan policy as this policy has the highest number of infections and the longest infection duration which are essential elements in the calculation of cost of infection.

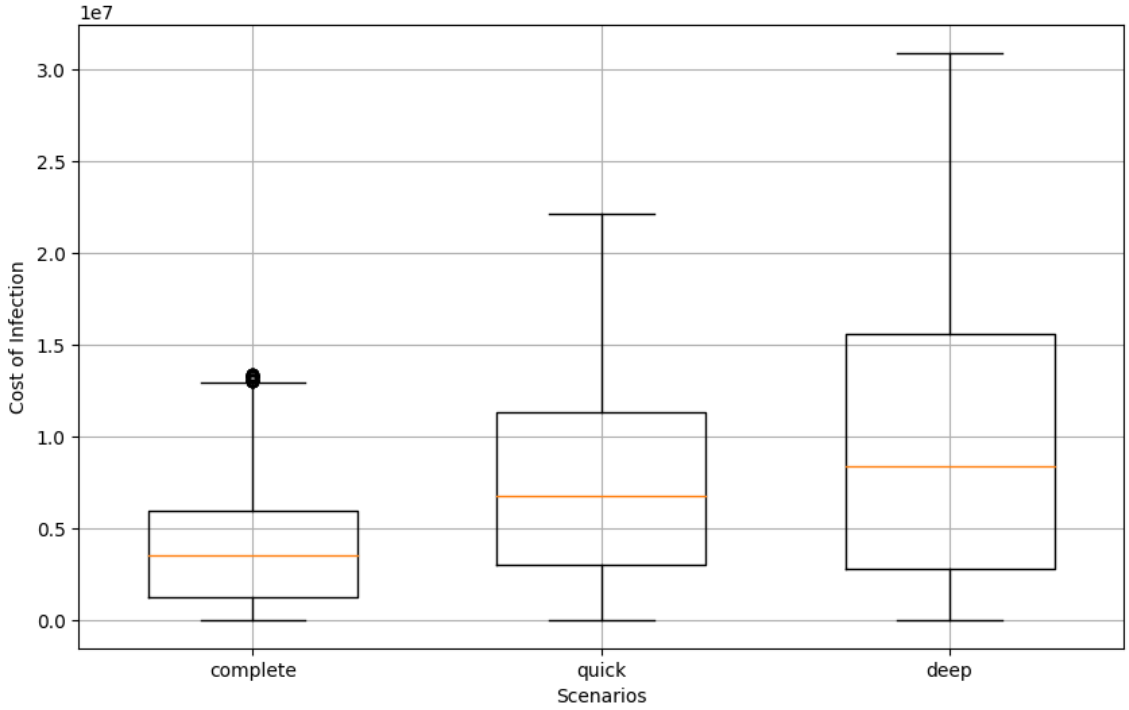

Figure 16. Boxplot of cost of infection for scan types

Overall, based on the assessment of different metrics, it can be said that employing a hybrid scan policy provides the best of both worlds as it cuts the infection duration significantly as well as the average cost of infection.

4.6 Scenarios assessing the effects of EDR policy

Another countermeasure to ensure a secure network in terms of malware attacks is the investment in EDR solutions. Although costly, these solutions can have a great impact in eradicating or slowing down the spread of infection within a system. This is based on two parameters; the first being the number of EDR solutions to put in place, and the placement of the solutions within the network.

4.6.1 Placement of EDR

To assess the effect of the placement of EDR solutions within a network, a scenario keeping all base model parameters constant while only changing the placement of EDR to hierarchical was executed. This scenario was compared with the base model where the placement policy is random.

Figure 17 pictures the average number of infected nodes for both placement policies. The number of infected nodes is lower for hierarchical placement as this placement method places the EDR solutions at the "top" of the hierarchy which prevents the worm from spreading faster as usually in these types of networks –and in the case of this model- spread from the top-down has a higher probability than bottom-up. So, strategically placing EDR solutions at more "important" nodes, hinders the spread of such infections.

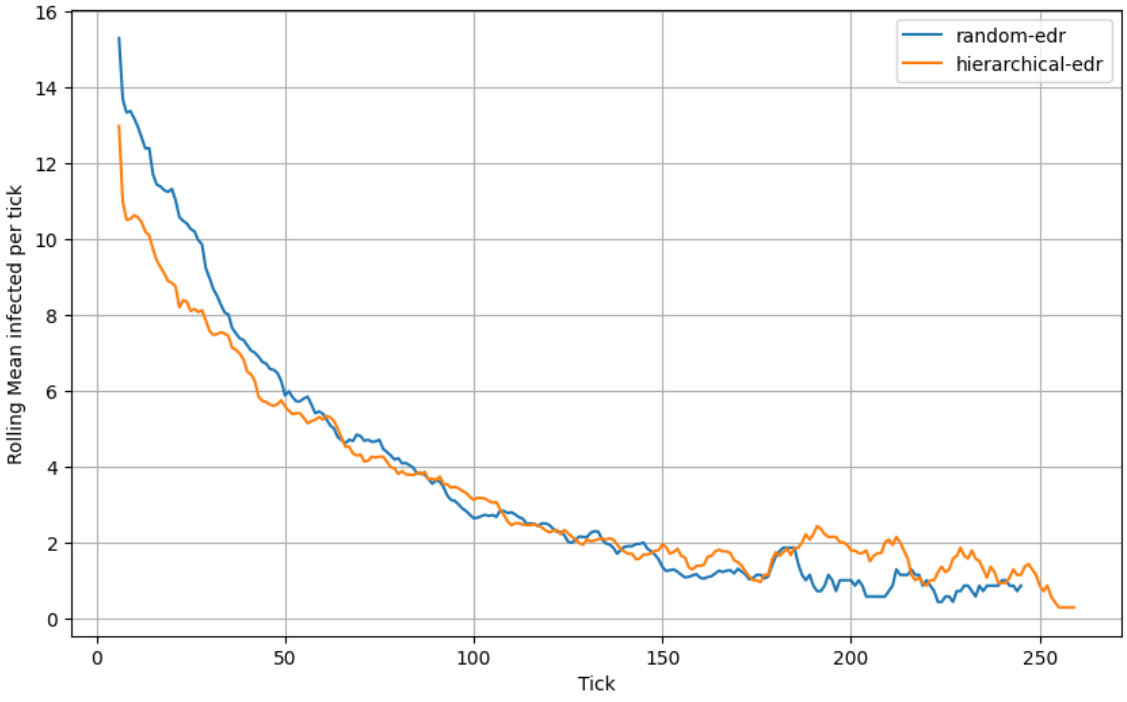

Figure 17. Average infected based on placement of EDR

4.6.2 Number of EDR solutions

Another factor within the employment of EDR solutions is the number of solutions to place in the network. The base model provides a percentage of EDR parameter which takes the number of computers in the network into account. The base model assumed 15% of computers within the network had an EDR solution in place. The scenario to be compared increase the percentage to 30% to assess the effects of the number of EDR solutions on the spread of infection within a network.

Figure 18 provides the average infected agents per tick based on the two scenarios. It is evident that increasing the number of EDR solutions in the network decreases the number of infected nodes.

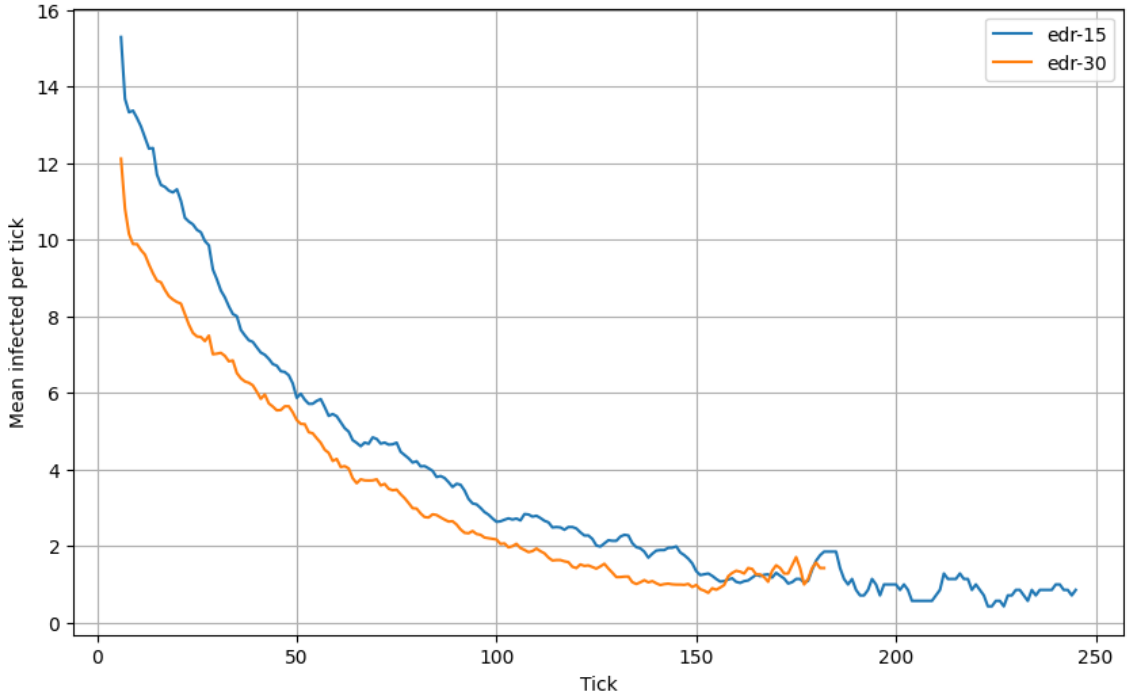

Figure 18. Average infected based on number of EDR solutions

4.7 Scenarios assessing the effect of IT response policy to infected computers

The following scenarios were executed to examine the effect of IT personnel response likelihood and capability on malware propagation. For this purpose, firstly, the response likelihood for IT personnel parameter was changed keeping all other base scenario parameters constant. Subsequently, IT capability parameter was changed keeping all other base scenario parameters constant.

4.7.1 Response likelihood and capability

Firstly, local sensitivity was conducted for the response likelihood parameter, keeping all other parameters constant at the base scenario values. The base scenario response likelihood, which is the likelihood that an IT personnel responding to a detected computer, is 0.4. We experimented with likelihoods of 0.5 and 0.3. Looking at the average duration of infection for these three scenarios (Figure 19), we can see that the higher the likelihood of IT responding to an incident, the lower the duration of infection and vice versa.

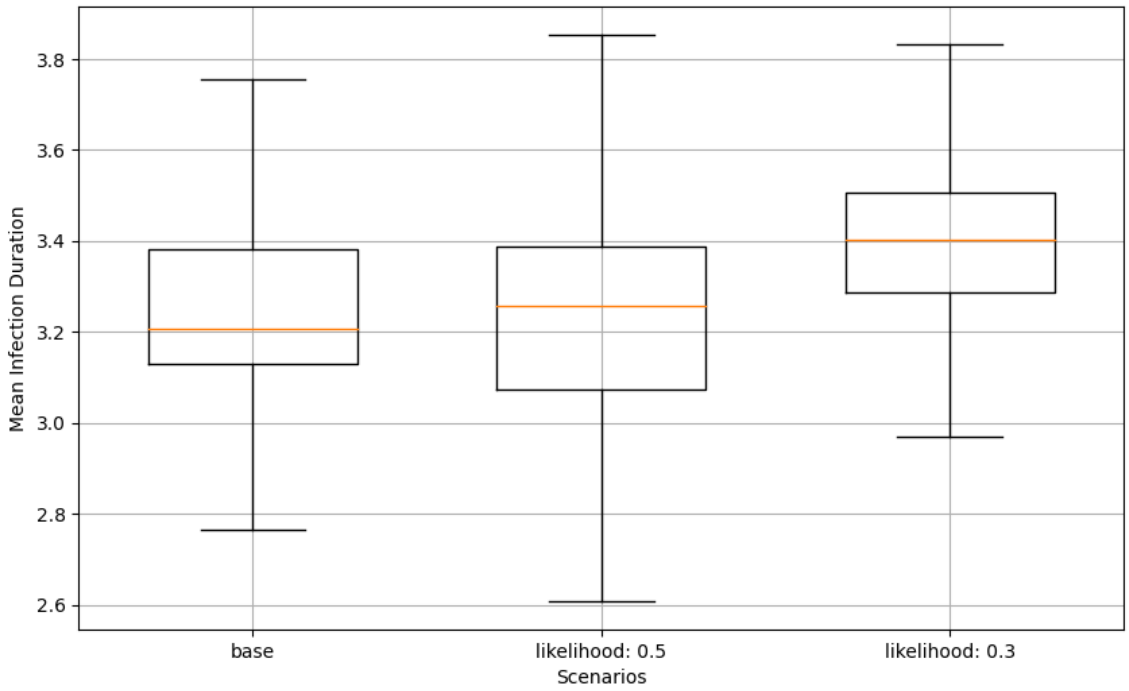

Figure 19. Boxplot of infection duration based on response likelihood

We also examined the percentage of detected computers that have been cleaned by IT personnel based on these scenarios. Again, a higher likelihood of responding to a detected computer yields a higher cleaning percentage for IT (Figure 20).

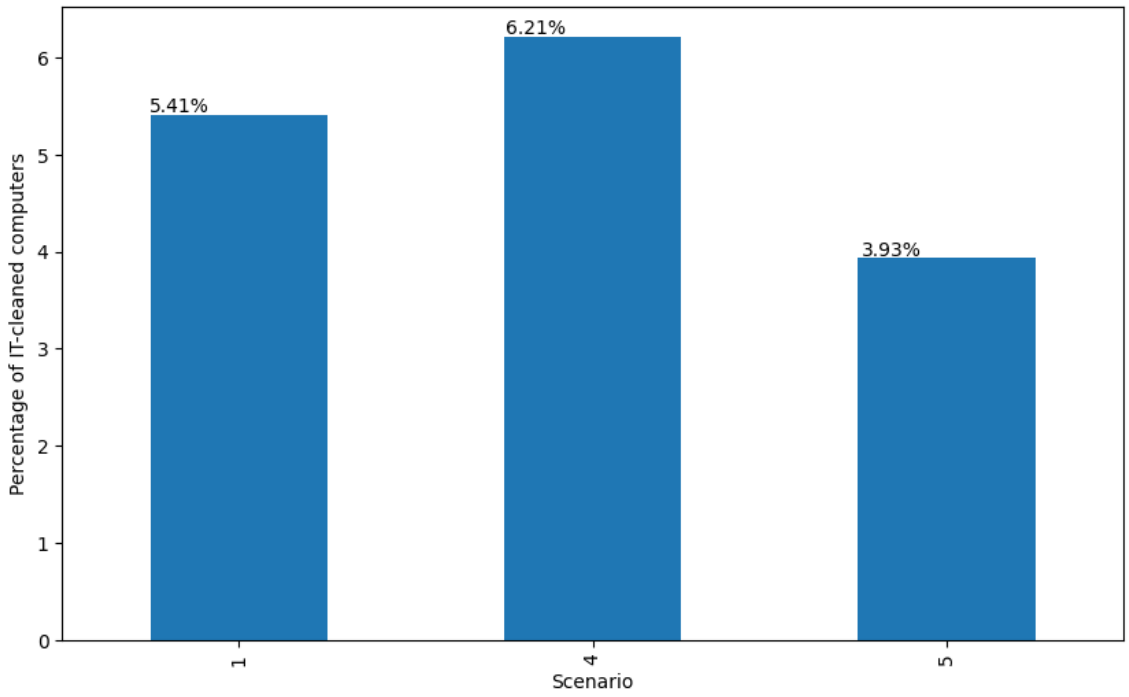

Figure 20. Percentage of computers cleaned by IT based on response likelihood

However, in terms of cleaning an infected computer, likelihood of response on its own may not be sufficient as the capability of the IT personnel responding to the incident comes into play. Therefore, the assess the effect of IT capability on the rate of computers cleared of infection by IT personnel, the capability parameter values were tweaked keeping all other based scenario parameters constant. The base scenario capability value is 0.5. We evaluated the base scenario against two different values of capability, 0.6 and 0.4. Figure 21 presents the percentage of computers cleaned by IT out of the total detected computers. It is clear that scenario 6 (capability: 0.6) yields a higher rate of cleaning by IT than the other two scenarios.

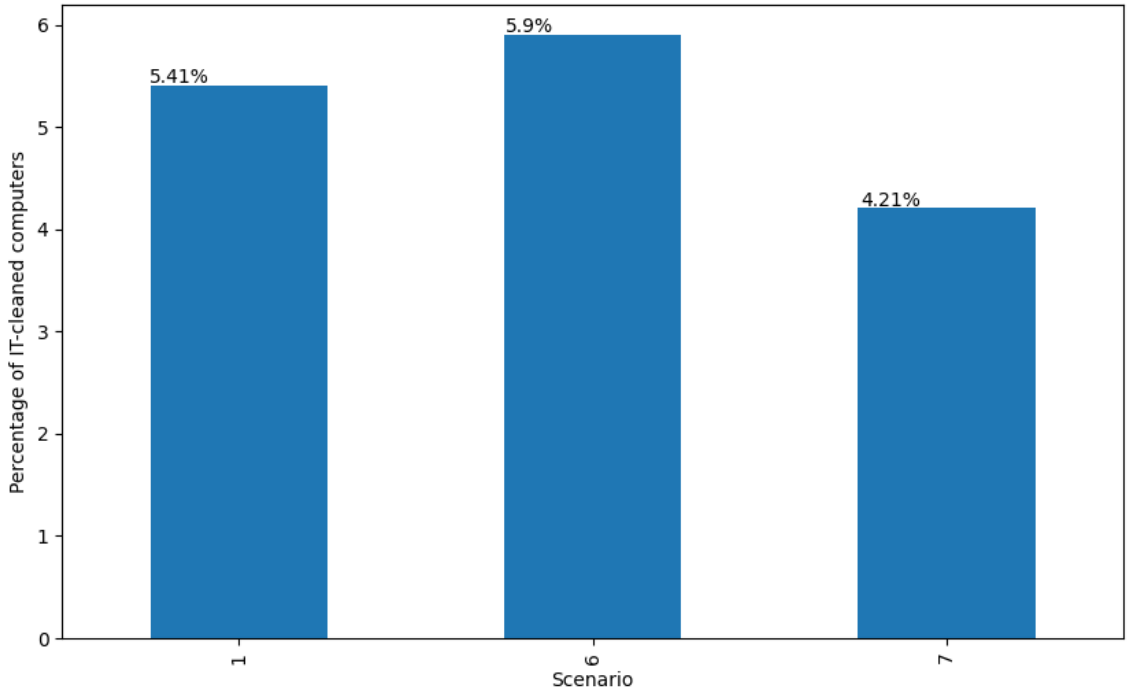

Figure 21. Percentage of computers cleaned by IT based on IT capability

4.7.2 Exponential response likelihood

Lastly, a scenario was prepared that not only takes IT response likelihood into account but supports the notion that once a computer has been detected as infected by an antivirus software but not been cleaned, IT personnel should employ a response policy of prioritizing such computers. Therefore, the response likelihood should be higher. The longer the computers remain infected, the higher the likelihood of a IT personnel prioritizing this computer over others. Algorithm 6 provides the procedure for calculating a new response likelihood based on the period of infection of a detected computer. This new response likelihood exponentially decreases the likelihood of an IT personnel not responding to an incident, therefore, subsequently increasing the likelihood of responding to a detected computer based on the duration of its infection.

Algorithm 6 IT Response Likelihood

$q \leftarrow 1 - responseLikelihood$
if $\eta > 0$ **then**
    $\eta ResponseLikelihood \leftarrow 1 - q^{\eta}$ **then**
    if $\eta ResponseLikelihood > random(0,1)$ **then**
        if $\beta_{IT} > random(0,1)$ **then**
            become known-susceptible
        else
            become resistant
        end if
    else
        continue
    end if
else
    continue
end if

Figure 22 compares the previous scenarios based on IT response likelihood with the new response likelihood scenario. The former provides the duration from the detection of an infected computer to the network becoming clean while the latter provides the infection duration. The two figures show similar patterns between the four scenarios in question. It can be said that prioritising computers that remain infected lowers the duration of infection within the network as well as the time taken from detection of an infected computer to cleaning.

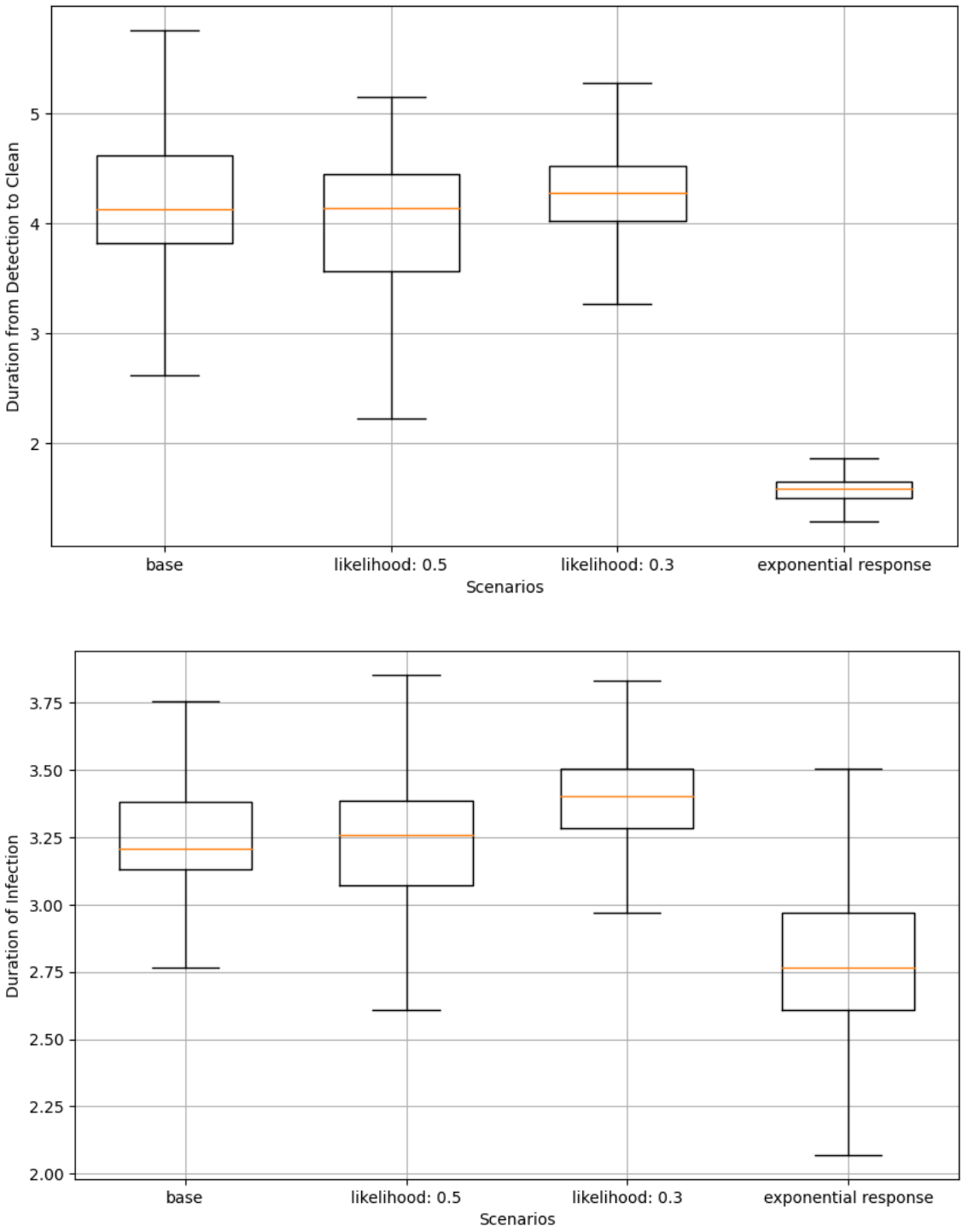

Figure 22. Boxplot of duration from detection to clean (top), Average infection duration (bottom)

4.8 Scenario 0: worst case scenario

Finally, a worst case scenario was executed. This scenario assumed that no mitigation strategies were in place. All software, IT, and awareness parameter values were set to the lowest possible value. When run, this scenario reached its peak infection very quickly, at close to two ticks, meaning that the infection spread throughout the network swiftly. This is of course due to there being no scanning policies, no EDR solutions set in place and all users having the lowest possible level of awareness, making them more vulnerable to the phishing attack.

The absence of any mitigation strategy is apparent when looking at Figure 23 as there are no agents in resistant and known susceptible states. The initial agents that are in susceptible state quickly transition to infected and state there for the duration of the run as there are no detection and cleaning measure put in place. As the model only runs to 365 ticks, this scenario was cut off before the network was cleared of infection.

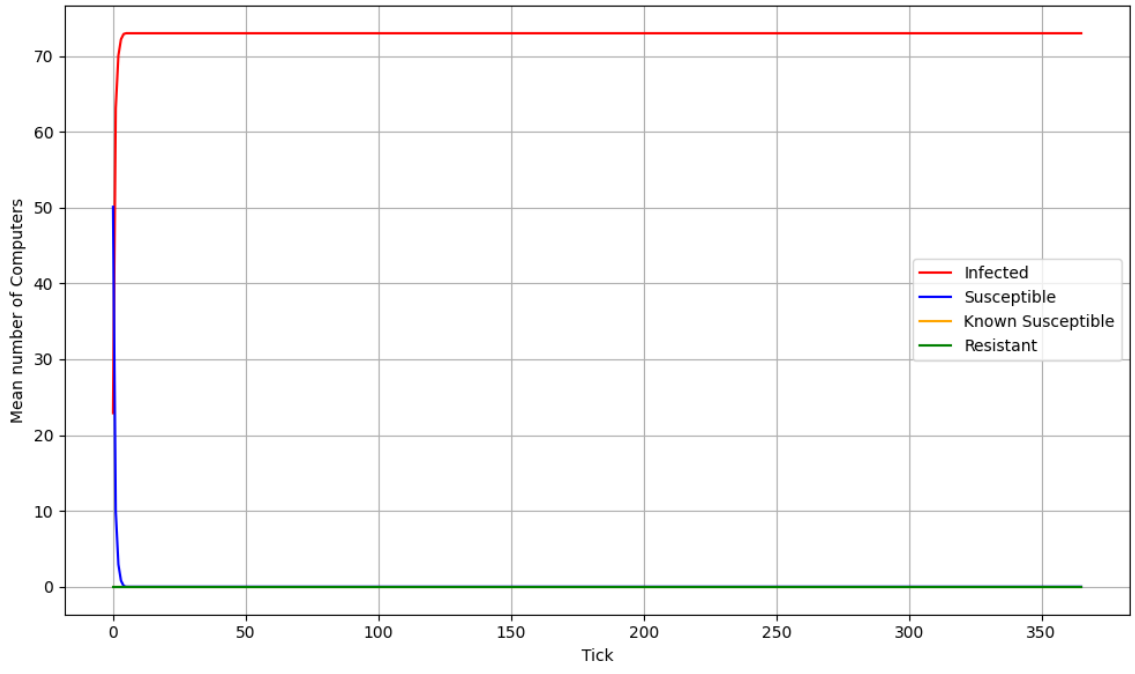

Figure 23. Number of components in each state for worst case scenario

4.9 Comparison of all scenarios

Examining all the scenarios provided to assess various policies and mitigation strategies within themselves has aided in understanding which measures are significant in mitigating malware-based threats. This section provides a general comparison of all scenarios in question. The id's of the scenarios and their corresponding descriptions along with all outputs of the scenarios can be found in Appendix C.

First we compare the scenarios based on average overall time for the network to be cleared of infection (Figure 24).

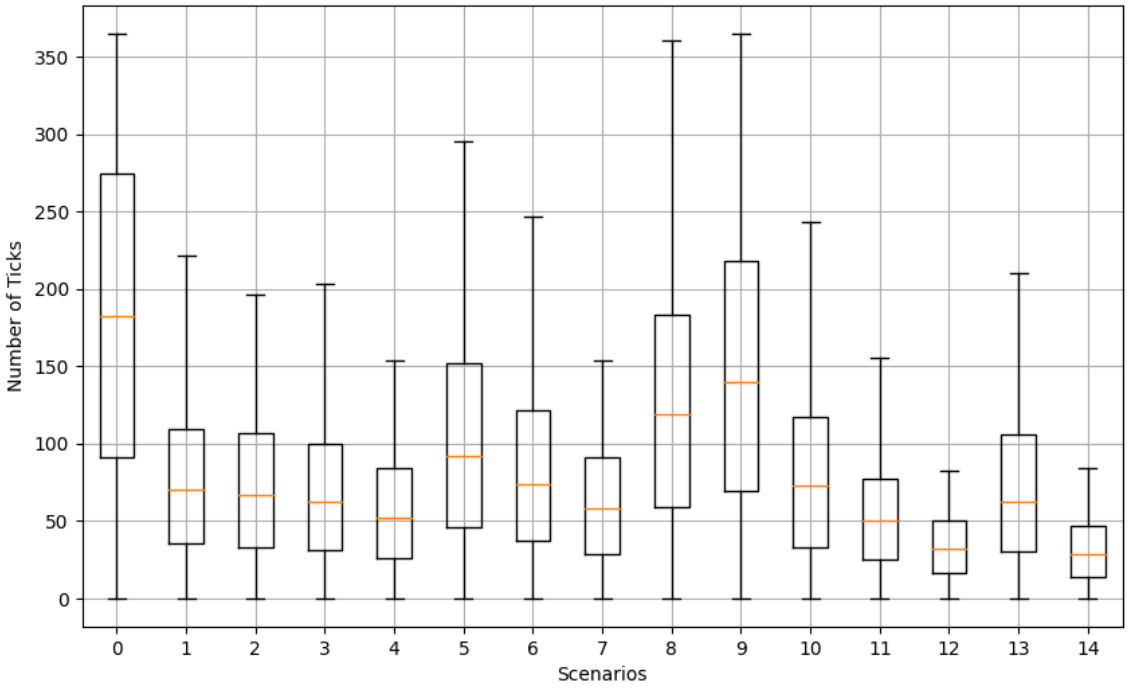

Figure 24. Boxplot of ticks to clear network for all scenarios

While scenario 0 – the worst case- had the longest duration (365) to clean the network, which is actually not the case as this was the cut-off for the model, scenario 14 –exponential response likelihood- was the quickest to clear the network of infection.

Next, we compare the percentage of computers infected for each scenario. Again, the worst case scenario living up to its name and purpose, due to not having any mitigation measures in place, has all agents of the network infected.

Based on infected percentage (Figure 25), the strategy of increasing the number of EDR solutions seems to have a positive effect as this scenario (13) has the lowest percentage of infected agents. However, a more in-depth examination of the chart can tell that most scenarios are close together and based around the mean percentage of infected agents.

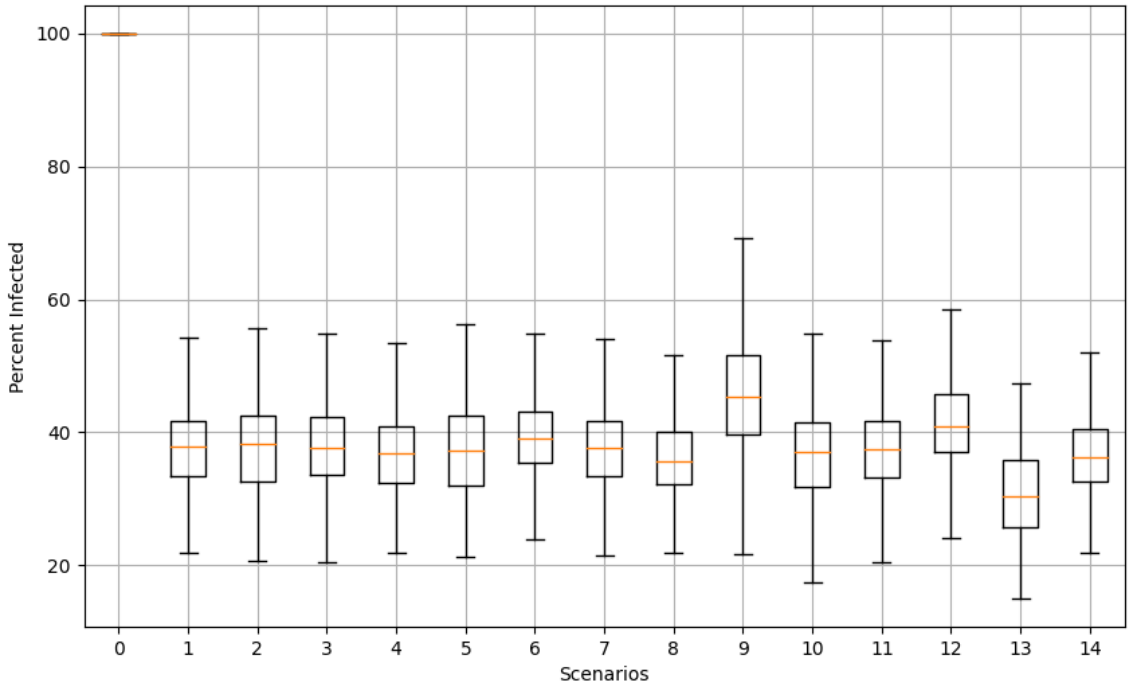

Figure 25. Percentage infected for all scenarios

We further compare the scenarios in terms of peak simultaneous infection. Peak simultaneous infection refers to the maximum number of computers that are infected at the same time. When a virus, malware, or any malicious software starts to propagate across a network, it begins to infect the connected systems. Over time, the number of

infected systems increases. At a certain point, the number of infected systems reaches a maximum before starting to decrease, either due to the deployment of security measures (like antivirus software or EDR) or because the malware has already infected all possible systems. This highest point of infection, when the most systems are infected at the same time, is what's referred to as the "peak simultaneous infection".

This concept is especially crucial in network security and incident response, as it represents the severity of the infection outbreak. It helps security professionals to assess the impact of the attack and guides in the development of appropriate measures to counter the spread of the malware and to prevent future incidents. It is also a valuable metric for testing the robustness of a network's security infrastructure and protocols.

Figure 26 provides the both the maximum number of computers that have been infected at the same time as well as the number of ticks it took to reach peak simultaneous infection. While scenario 11, which represents a random network, has a peak simultaneous infection value on the mean line, the rate of infection is slower than all other scenarios. On the other hand, scenario 13, the EDR policy scenario, has the lowest number of simultaneous infections. The spread rate of the infection was the fastest in scenario 12, the scenario with a frequent deep scan policy in place.

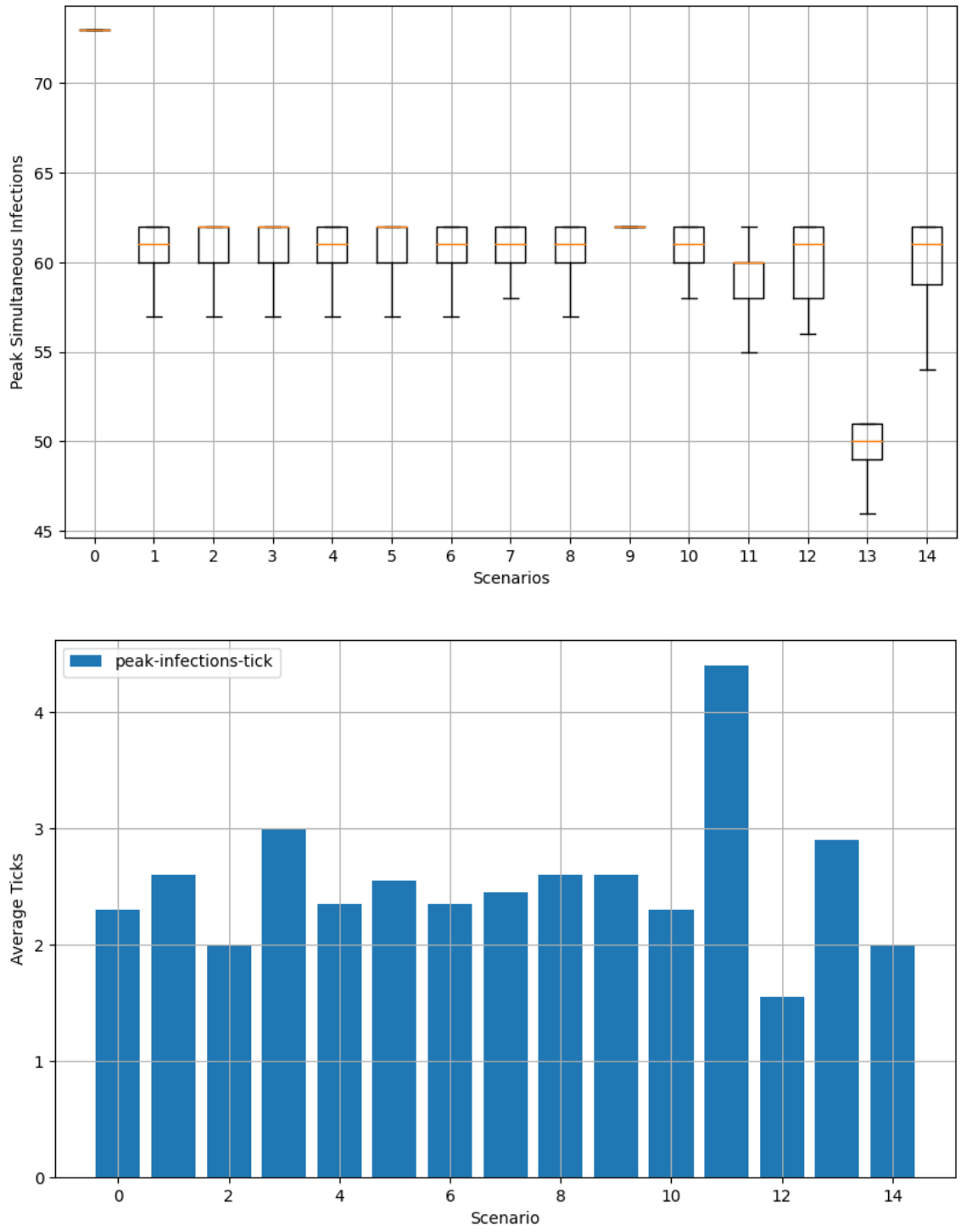

Figure 26. Peak simultaneous infections (top) and rate of spread (bottom) for all scenarios

Comparison of strategies and policies using the metrics provided while insightful also has drawbacks. One of the most critical metrics for an organization is cost, therefore, while many of these scenarios may seem appropriate to implement, they may not be cost effective. Figure 27 provides the ratio of investment cost to cost of infection for all scenarios. This ratio represents the investment efficiency against infection costs. For example, this ratio is the highest for scenario 13 (EDR policy). The ratio of 0.15 implies that for every unit of cost incurred due to infection, roughly 0.15 units of investment are being made in strategies aimed at preventing or mitigating these infections.

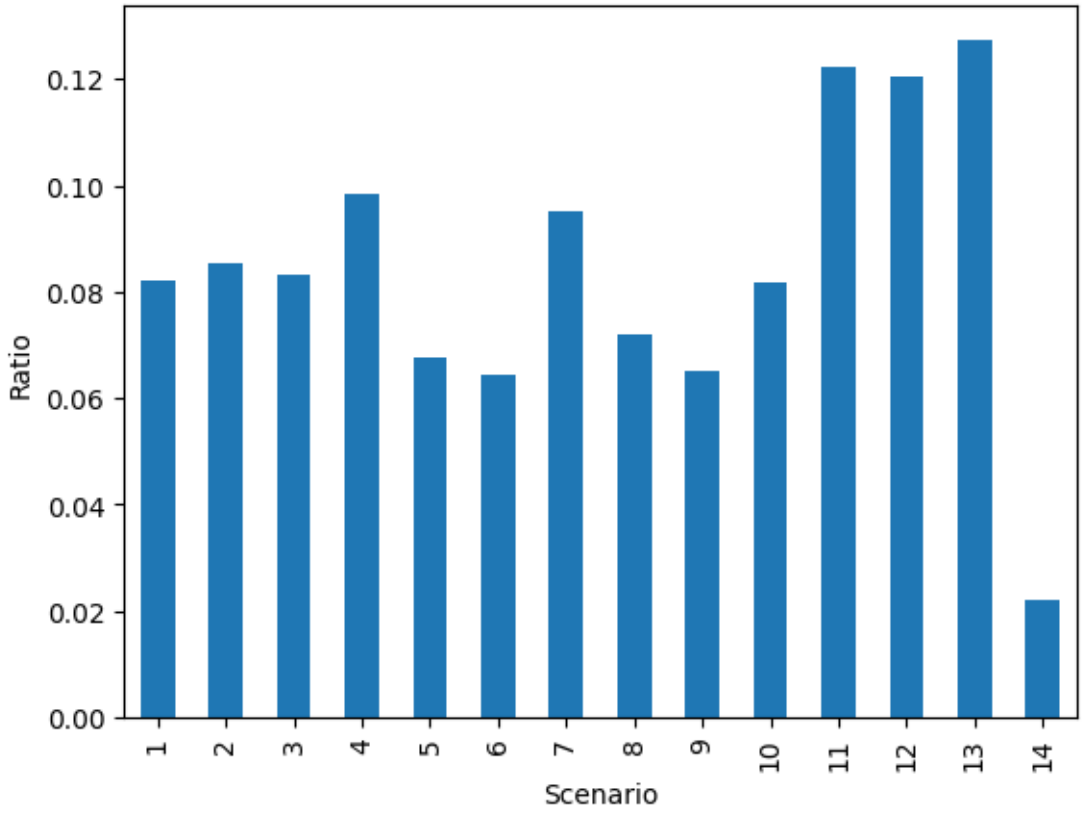

Figure 27. Cost ratio for all scenarios

In practical terms, it means that the cost of investment is significantly lower than the cost of infections. This could be interpreted in a few ways:

I. If the investment is effectively reducing the number of infections, then a low ratio is positive. It means that high degree of prevention or mitigation is being achieved with relatively low investment.

II. Conversely, if the number of infections remains high despite the investments, this ratio could be an indication that there is underinvestment in cybersecurity measures relative to the severity and cost of infections.

III. It may also indicate that the cost of infections is extremely high, in which case additional investment in prevention and mitigation strategies could potentially be needed.

To interpret this ratio accurately, it's essential to also consider other factors such as the number of infections, the effectiveness of the strategies employed, the consequences of the infections, and the nature of the costs involved.

CHAPTER 5

DISCUSSION AND CONCLUSION

"I know one thing, that is I know nothing." - Socrates

With the rise of the digital era, the cybersecurity landscape has grown increasingly complex, characterized by constant evolution and expansion. Phishing attacks, especially those employing malware, form a significant part of this landscape and pose substantial threats to organizations across the globe. Our approach to this issue was established by a recognition that cybersecurity threats are not merely technological problems, but instead encompass complex interactions of human and organizational factors. This realisation steered the research towards a detailed examination of the interaction between these components, employing Agent-Based Modelling (ABM) as our key methodological tool. ABM, a computational modelling technique, facilitates the simulation of individual agents' behaviours and their interactions within a system. By leveraging ABM, we successfully captured individual agent behaviours, represented dynamic interactions, simulated emergent phenomena, and tested various countermeasures and strategies aimed at mitigating cyber threats.

A significant aspect of this research was the integration of socio-technical dimensions of phishing attacks into the model. Grounded in the theoretical perspectives of socio-technical theory and the concept of complex adaptive systems (CAS), this innovative approach provided a holistic perspective on the issue. By emphasizing the importance of integrating and interacting social and technical aspects for effective cybersecurity, the research shed light on the significance of human factors, such as user

awareness, training, and personality traits. Moreover, the role of technical security measures was also underlined, together leading to a more nuanced understanding of the vulnerabilities that exist within an organizational network. These insights illuminated potential areas where mitigation strategies could be most effective.

Our study suggests that a balanced approach, giving equal importance to both human behaviour and technological measures, can significantly enhance the organization's resilience to cyber threats.

To test the model, we simulated a variety of phishing attack scenarios and conducted local sensitivity analyses. The variables examined included human behaviour, technical and IT measures, and network components. The results of these simulations provided invaluable insights into the resilience of different countermeasures to phishing attacks. Notably, even minor adjustments in these variables led to significant changes in the propagation of malware-based phishing attacks, reflecting the sensitivity of organizational networks to small changes.

This study makes several significant contributions to the field of cybersecurity:

Theoretical contribution: This research makes significant theoretical contributions to the understanding of malware-based phishing attacks within organizational networks. By integrating socio-technical theory and the concept of complex adaptive systems (CAS), the study provides an integrative approach to understanding the complex interplay of factors involved in malware-based phishing attacks, bridging the gap between technical and human elements. This integration provides a more comprehensive and holistic perspective on effective cybersecurity strategies. The findings shed light on the vulnerabilities that exist within organizational

networks and highlight the need for an integrated approach that considers both the technical and human dimensions of cybersecurity.

By integrating socio-technical theory, CAS, and the agent-based modelling approach, this research contributes to advancing our theoretical understanding of malware-based phishing attacks within organizational networks. The insights gained from this study can guide future research efforts, inform the development of theoretical frameworks, and enhance the overall knowledge base in the field of cybersecurity.

Methodological contribution: The study employs ABM as a key methodological tool. By simulating individual agent behaviours and their interactions, the model captures the dynamics and emergent phenomena of malware-based phishing attacks. The use of ABM allows for a comprehensive examination of the complex web of factors involved in these attacks and enables the testing and evaluation of various defence strategies. To the best of the authors knowledge, this is the first study to incorporate human factors (namely personality traits) into an agent-based model to examine the effects of these factors on malware-based phishing attacks.

The agent-based model developed in this research serves as a valuable tool for future research in the field. The model provides a robust and flexible framework for simulating a variety of phishing attack scenarios and evaluating different countermeasures. It enables researchers to gain insights into how small changes in human behaviour, IT measures, and network components can lead to substantial differences in attack propagation and impact. This sensitivity to small changes aligns with the CAS function of operating at the edge of chaos (Turner, Baker, & Morris, 2018), based on the butterfly effect described by Olbolensky (2014) supporting the notion that even minor tweaks within a system can lead to significant shifts in outcomes.

Practical contribution: The research provides practical implications for organizations striving to defend against cyber threats. By identifying the key factors that influence the spread and impact of malware-based phishing attacks, organizations can develop more effective strategies to mitigate these threats. The agent-based model developed in this study serves as a valuable tool for simulating different attack scenarios and assessing the potential effects of various defence strategies. This application aids in evidence-based decision-making, allowing organizations to proactively prepare for potential threats and enhance the resilience of their digital infrastructure.

These practical contributions bear significant implications for businesses. First, they illuminate the vulnerabilities inherent in an organizational network and provide insights into effective mitigation strategies. The emphasis on human factors suggests businesses should prioritize user awareness and training as part of their cybersecurity strategy. Furthermore, the results highlight the effectiveness of technical security measures, such as Endpoint Detection and Response (EDR) tools, in limiting the spread of phishing attacks. This aligns with the literature stating that the use of such tools will result in the decrease of IT staff fatigue from frequent and repetitive alerts, and a reduction of average response time subsequently increasing the detection of attacks and shortening the network infection time (Kamruzzaman, Ismat, Brickley, Liu, & Thakur, 2022). The agent-based model developed in this study could serve as a strategic tool for businesses to simulate different phishing attack scenarios, analyse their potential impacts, and test the impact of various countermeasures.

The findings of this study contribute to the existing body of knowledge by providing a more comprehensive understanding of phishing attacks and malware

propagation, and by demonstrating the potential of agent-based modelling as a tool for studying these threats.

5.1 Limitations and future work

This thesis is not without its limitations that provide several avenues for future research.

5.1.1 Limitations

Despite the significant contributions made by this study, it is not without limitations. One of the primary limitations is the inherent simplification of the real world within the model. Although the model attempted to incorporate a comprehensive range of factors including human behaviour, IT measures, and network components, the complexity and unpredictability of these aspects in real-world scenarios may not be fully captured. Human behaviour, for example, is highly intricate and often affected by a myriad of factors that are difficult to account for in a model. Although this research attempted to encapsulate some of these complexities, it could not cover every aspect. Moreover, the model assumes a certain level of consistency in human behaviour, which may not always hold true in real-world scenarios.

The theoretical nature of the model constitutes another limitation. Although we ran a variety of simulations to test the model, it remains a theoretical representation of reality. Therefore, its real-world applicability might be restricted. The findings and insights accumulated from this study would benefit significantly from empirical validation through application and testing in real-world contexts.

Future work should also address the scope of the model, as it does not account for all possible defensive strategies against phishing attacks. Given the rapidly evolving

nature of cyber threats, future research could consider a broader range of variables and mitigation strategies. There are many other innovative security solutions, such as advanced threat intelligence or multi-factor authentication, which were not considered in the model but could potentially have significant impacts on phishing attack mitigation. More comprehensive incorporation of human behaviour complexities and the extension of the model to other types of cyberattacks would undoubtedly strengthen the model's validity and robustness.

Lastly, the model assumes and is based on a single malware-based phishing attack on a network. Therefore, modelling the worm of a single attack and how it propagates throughout the network. Future studies could incorporate multiple attacks at various times with varying levels of malware to allow for more complex variations of these types of attacks.

5.1.2 Future work

Given the limitations of this study, several avenues for future research emerge. One significant area to explore is the integration of more variables that were not considered in this model. These could include variations in network size, nature of malware, the complexity of IT security measures, or the sophistication level of phishing techniques. By examining these variables, future research could offer a more detailed picture of phishing attack propagation and the effectiveness of mitigation strategies. Improvements in capturing the complexity of human behaviour could also be a promising direction for future work. Human behaviour is highly nuanced and influenced by a range of factors, from individual characteristics and experiences to cultural and societal influences. Therefore, research that seeks to better account for these complexities could lead to

more accurate and useful models. Future studies could also focus on extending the model to other types of cyberattacks. This would provide a more comprehensive understanding of the cybersecurity landscape and inform more effective defence strategies. For example, the model could be adapted to simulate ransomware attacks, distributed denial-of-service (DDoS) attacks, or other forms of cyber threats.

Finally, future research could benefit from the practical application of the model. Testing the model in real-world settings and validating its predictions against actual outcomes would greatly enhance its reliability and robustness. Moreover, such studies could provide invaluable insights into the real-world efficacy of different mitigation strategies, thus contributing to the development of more effective cybersecurity policies and practices.

5.2 Conclusion

Our research has made a significant contribution to the understanding of malware-based phishing attacks within organizational networks. Through the use of Agent-Based Modelling, we have developed a comprehensive framework to dissect the intricate socio-technical dynamics involved in these attacks. The agent-based model serves as a valuable tool for future investigations in this critical field, providing a foundation for further explorations and innovations.

The practical implications of our research are evident in the guidance it offers to professionals in designing and implementing robust cybersecurity strategies. By acknowledging the socio-technical dimensions of cybersecurity threats, organizations can create a safer and more secure digital working environment. The findings and insights derived from our research have the potential to shape more resilient

cybersecurity strategies and policies. The development and validation of the agent-based model showcase the potential of advanced computational models in managing and decision-making within information systems.

In summary, our research offers both theoretical and practical insights into the dynamics of malware-based phishing attacks within organizational networks. By employing an ABM approach and considering the complex dynamics between human behaviour, technological measures, and organizational factors, we have made significant contributions to the understanding of these threats. The practical implications of our research guide organizations in defending against cyber threats, while the theoretical implications highlight the importance of theories and frameworks in addressing these challenges.

As the cyber landscape continues to evolve, studies like ours become increasingly essential to ensure the security of digital infrastructure. Our research aims to contribute to the creation of safer and more secure digital environment for organizations. By shedding light on the human, technological, and organizational factors that underpin these threats, our research provides a comprehensive picture that aids in the formulation of robust cybersecurity strategies. Ultimately, our goal is to provide a relatively simple and adjustable model to support the continued growth and development of the cybersecurity field by enhancing the understanding and management of the complex threats posed by phishing attacks and malware. The agent-based model developed in this thesis serves as a valuable tool for future research. It presents a robust and flexible framework for simulating a variety of phishing attack scenarios and evaluating different countermeasures. Future research can refine, expand, and apply this

model to gain further insights into the dynamics of phishing attacks and other cybersecurity threats.

APPENDIX A

NETLOGO SOURCE CODE

```
extensions [rnd nw]


turtles-own
[
  infected?
  resistant?
  virus-check-timer   ;number of ticks since last virus-check
  computer_resistance   ;resistance degree of the computer
  level
  parent
  group
  prob
  days_inf            ;number of days spent infected
  mail_elaboration
  prob_of_check_human_factor
  edr?
  scan?
  neuroticism_t
  openness_t
  conscientiousness_t
  awareness_t
  human_risk
```

phishing_perception

alarm

av-clean?

IT-clean?

head

times-infected

click?

infected-by

infected-when

prob-list

cleaned-when

detected-when

random-link-prob

infection-duration

infection-to-detection

alarm-list

]

undirected-link-breed [network-links network-link]

undirected-link-breed [random-links random-link]

links-own [weight]

globals [

mailQuality ;determines the attractiveness of phishing email

susceptible

resistant

infected

known-susceptible

percentage_susceptible

percentage_resistant

percentage_infected

percentage_knownsusceptible

weakHumanCount

weakComputerCount

;ticks

;cost-of-inv

;cost-of-inf

avg_days_inf

;cost-of-attack

lvl

max-alarm

max-alarm-count

mean-alarm

max-times-infected

mean-times-infected

;virus-check-frequency

infected-list

p

dist

peak-simultaneous-infections

;cost-of-edr

no-of-edr-solutions

cost-of-training

cost-of-IT

cost-of-antivirus

detected-computers

av-cleaned-computers

IT-cleaned-computers

isolated-computers

total-isolated

total-detected

;num-scan

det-comp

av-clean-comp

IT-clean-comp

;num-anti

;num-IT

total-infected

infected-per-tick

initial-infected

mean-infection-detection

mean-infection-duration

mean-human-risk

overall-risk

avg-comp-security

total-cost

total-av-clean

total-IT-clean

no-of-IT-personnel

computer_strength

av-clean-rate

IT-clean-rate

resistant-rate

user-business-disruption

IT-team-cost

software-tool-cost

awareness-training-cost

cost-of-investment

cost-of-infection

d_up

o_r

avg-downtime

d_ap

d_et

d_at

```
  d_av
  d_edr
  sum-alarm
  total-infected-rate
  user-business-disruption-factor
  IT-team-cost-factor
  software-tool-cost-factor
  edr-tool-cost

]

to initialize-globals

  set mailQuality (mailDesignQuality + mailContentQuality) / 2
  set computer_strength (computer-security-level + up-to-dateness) / 2
  ;set virus-check-frequency 7
  set infected-list []
  set max-alarm 0
  ;set no-of-IT-personnel round(count turtles / (1 / response-likelihood))
  ;set no-of-IT-personnel round((count turtles * response-likelihood) / IT-capability)
  set no-of-IT-personnel 15
  set detected-computers 0

end
```

```
to initialize-turtle-variables

  ask turtles [
    set SIZE 1.5
    set edr? false
    set scan? false
    set av-clean? false
    set IT-clean? false
    set click? false
    set infected-by []
    set infected-when []
    set prob-list []
    set cleaned-when []
    ;set num-scan []
    set det-comp []
    set detected-when []
    set av-clean-comp []
    set IT-clean-comp []
    ;set num-anti []
    ;set num-IT []
    set alarm-list []
    set infection-duration []
```

```
set infection-to-detection []
set computer_resistance computer_strength

set neuroticism_t random-poisson (neuroticism - 1)
set conscientiousness_t random-poisson (conscientiousness - 1)
set openness_t random-poisson (openness - 1)
set awareness_t random-poisson (awareness - 1)
while [neuroticism_t > 4
or
conscientiousness_t > 4
or
openness_t > 4
or
awareness_t > 4]
[set neuroticism_t random-poisson (neuroticism - 1)
 set conscientiousness_t random-poisson conscientiousness - 1
 set openness_t random-poisson openness - 1
 set awareness_t random-poisson awareness - 1]
 set neuroticism_t (neuroticism_t + 1)
 set conscientiousness_t (conscientiousness_t + 1)
 set openness_t (openness_t + 1)
 set awareness_t (awareness_t + 1)
```

```
    set mail_elaboration (conscientiousness_t * 0.17) + ( openness_t * 0.19) +
(awareness_t * 0.28)
    set prob_of_check_human_factor (0.11 * neuroticism_t) + (0.16 * awareness_t) +
(0.19 * mail_elaboration)
    set phishing_perception 1 / (1 + e ^ ( -1.77004 * prob_of_check_human_factor +
1.1748 ))]

end



to setup
 clear-all
 ;random-seed 12345
 set-default-shape turtles "computer workstation"
 ask patches [
  set pcolor gray]



  ifelse network-type = "hierarchical"[
  network-hier
  create-random-links
 ][
  network-rand
 ]
```

```
  initialize-globals
  initialize-turtle-variables

  set no-of-edr-solutions round (edr-percentage * count turtles)
  ifelse edr-placement = "hierarchical" [
    edr-hier
  ][
    edr-rand]

  reset-ticks
  initial-infection

  set peak-simultaneous-infections count turtles with [infected?]

  ask links [ set color white ]

  ask turtles [
  if (edr?) and (infected?)
    [set color yellow]]


end
```

```
to initial-infection


 ask turtles [
 set human_risk 1 - phishing_perception
    (ifelse mailQuality > 3 [set human_risk human_risk + ((mailQuality - 3) * 0.1145)]
     mailQuality < 3 [set human_risk human_risk - ((3 - mailQuality) * 0.1145)]
    [set human_risk human_risk])]


  set mean-human-risk mean [human_risk] of turtles


  ask turtles
 [if random-float 1 < human_risk
 [set weakHumanCount (weakHumanCount + 1)
  set click? true
  if wormStrength > computer_resistance
      [become-infected
       set total-infected total-infected + 1
       set infected-per-tick infected-per-tick + 1
       set infected-list lput who infected-list
       set infected-by lput who infected-by
       set infected-when lput ticks infected-when
       set times-infected times-infected + 1
       set days_inf (days_inf + 1)
```

```
        set weakComputerCount (weakComputerCount + 1)]]]
   set initial-infected count turtles with [infected?]

end

to edr-hier
  if (count turtles with [head = 1]) >= round (edr-percentage * count turtles) [
    ask n-of no-of-edr-solutions turtles with [head = 1]
  [
    set computer_resistance 9
    set edr? true
  ]
  ]
  if (count turtles with [head = 1]) < no-of-edr-solutions [
    ask n-of (count turtles with [head = 1]) turtles with [head = 1]
   [
    set computer_resistance 9
    set edr? true
    ]

    ask n-of ((no-of-edr-solutions) - (count turtles with [head = 1])) turtles with [head = 0]
      [
    set computer_resistance 9
    set edr? true
```

```
    ]
    ]
end

to edr-rand ;random placement of edr solution
  ask n-of no-of-edr-solutions turtles
      [
    set computer_resistance 9
    set edr? true
    ]
end



to network-hier

 set lvl 1
 create-turtles 1 [
    set level 0
    become-susceptible
    set head 1
   ]

    let n 0
```

```
while [n <= lvl][
  ask turtles with [level = n][
    hatch 3
    [
      set parent [who] of myself


      create-network-link-with myself [set weight 1.5]
      set level n + 1
      set group [who] of myself
      set parent myself
      set head 1
 ]
]
  set n n + 1




]


set n 0


  ask turtles with [head = 1][
    hatch level + 3
    [
```

```
      set parent [who] of myself

      create-network-link-with myself [set weight 1]
      set level [level] of myself
      set group [who] of myself
      set parent myself
      set head 0
    ]
  ]
  layout-radial turtles links turtle(0)


  ask turtles [set random-link-prob []]

  ask turtles [

   ask other turtles[

     set dist nw:weighted-distance-to myself weight
     set p 0.8 * (0.25 ^ (dist - 2))
     set random-link-prob lput p random-link-prob
     ;print (word "turtle1 " [who] of myself word" turtle2 " [who] of self word" distance "
dist word" p: " p)
    ]
```

```
  ]

end

to create-random-links

  ask turtles [
    ask other turtles with [not link-neighbor? myself][

          if random-float 1 < p [
            create-random-link-with myself
          ]
    ;print (word "turtle1 " [who] of myself word" turtle2 " [who] of self word" distance "
dist word" p: " p)
    ;print (word "turtle2 " [who] of self)
    ;print (word "distance " dist)
    ;print (p)
        ]
      ]

  layout-radial turtles links turtle(0)

end
```

```
to network-rand ;create random network

    create-turtles number-of-nodes
  [

   setxy (random-xcor * 0.95) (random-ycor * 0.95)
   become-susceptible

  ]

  let num-links (average-node-degree * number-of-nodes) / 2
  while [count links < num-links ]
  [
   ask one-of turtles
   [
    let choice (min-one-of (other turtles with [not link-neighbor? myself])
              [distance myself])
    if choice != nobody [ create-link-with choice ]
   ]
  ]
  ; make the network look a little prettier
  repeat 10
  [
   layout-spring turtles links 0.3 (world-width / (sqrt number-of-nodes)) 1
```

```
  ]
end

to go
  ;random-seed 12345
  if ticks = 365 ;simulation runs for 365 days
  [
    ;calculate-costs
    stop
  ]
  if (all? turtles [not infected?]) and (ticks = 0)
  [user-message ("Phising attack failed.") stop]
  if (all? turtles [not infected?]) and (ticks > 0)
  [user-message ("System Clear.")



    stop
]

  ask turtles
  [
     set virus-check-timer virus-check-timer + 1
     if virus-check-timer mod virus-check-frequency = 0
       [ set virus-check-timer 0 ]
```

```
  ]
  set infected-per-tick 0

  ifelse network-type = "hierarchical" [
  ask turtles with [infected?]
  [ask network-link-neighbors with [(not resistant?) and (not infected?)]
    [ifelse (who > [who] of myself) or (level = [level] of myself and who = [group] of
myself )
      [spread-virus2]
       [spread-virus]]]
  ask turtles with [infected?]
  [ask random-link-neighbors with [(not resistant?) and (not infected?)]
    [spread-virus-random]]
  ][
     ask turtles with [infected?]
  [ask link-neighbors with [(not resistant?) and (not infected?)]
    [spread-virus-random]]
  ]

  let current-simultaneous-infections count turtles with [infected?]
  if current-simultaneous-infections > peak-simultaneous-infections [
    set peak-simultaneous-infections current-simultaneous-infections
  ]
```

```
set av-cleaned-computers 0
set IT-cleaned-computers 0
set detected-computers 0
set isolated-computers 0
virus-scan

set det-comp lput detected-computers det-comp
set av-clean-comp lput av-cleaned-computers av-clean-comp
set IT-clean-comp lput IT-cleaned-computers IT-clean-comp

update-globals

ask turtles with [infected?]
[set days_inf (days_inf + 1)]

;print (word "Infected turtles: " infected-list)

if any? turtles with [not empty? alarm-list][
  set max-alarm max [max alarm-list] of turtles with [not empty? alarm-list]
  set max-alarm-count count turtles with [member? max-alarm alarm-list]
  set mean-alarm mean[ mean alarm-list] of turtles with [not empty? alarm-list]

]
```

```
  if any? turtles with [not empty? infection-to-detection][
  set mean-infection-detection mean[ mean infection-to-detection] of turtles with [not
empty? infection-to-detection]
  ]
    if any? turtles with [not empty? infection-duration][
  set mean-infection-duration mean[ mean infection-duration] of turtles with [not empty?
infection-duration]
  ]

  set total-av-clean total-av-clean + av-cleaned-computers
  set total-IT-clean total-IT-clean + IT-cleaned-computers
  set total-isolated total-isolated + isolated-computers
  set total-detected total-detected + detected-computers
  set mean-times-infected mean [times-infected] of turtles

  tick

  calculate-costs
end

to become-infected
  set infected? true
  set resistant? false
  set color red
```

```
  set scan? false
  set av-clean? false
  set IT-clean? false
  set alarm 0
end

to become-susceptible
  set infected? false
  set resistant? false
  set color blue

end

to become-resistant
  set infected? false
  set resistant? true
  set color green
  ask my-links [ set color green - 2 ]
  set scan? false
  set av-clean? false
  set IT-clean? false
  if alarm > 0 [
  set alarm-list lput alarm alarm-list
  set alarm 0
```

```
  ]
  set computer_resistance 10
end

to become-knownsusceptible
  set infected? false
  set resistant? false
  set color orange
  set scan? false
  set av-clean? false
  set IT-clean? false
  if alarm > 0 [
  set alarm-list lput alarm alarm-list
  set alarm 0
  ]
end

to spread-virus

     if random-float 1 < 0.2
       [if computer_resistance < wormStrength
          [ become-infected
       set total-infected total-infected + 1
       set infected-per-tick infected-per-tick + 1
```

```
      set infected-when lput (ticks + 1) infected-when
      set infected-list lput who infected-list
    set times-infected times-infected + 1
    set infected-by lput myself infected-by
    set prob-list lput 0.2 prob-list]
        ]
  set prob 0.2

end

to spread-virus2
  if random-float 1 < 0.8
  [if computer_resistance < wormStrength
    [become-infected
     set total-infected total-infected + 1
     set infected-per-tick infected-per-tick + 1
     set infected-when lput (ticks + 1) infected-when
     set infected-list lput who infected-list
  set times-infected times-infected + 1
  set infected-by lput myself infected-by
  set prob-list lput 0.8 prob-list]]
  set prob 0.8

end
```

```
to spread-virus-random
  if random-float 1 < 0.6
  [if computer_resistance < wormStrength
    [become-infected
      set total-infected total-infected + 1
      set infected-per-tick infected-per-tick + 1
      set infected-when lput (ticks + 1) infected-when
      set infected-list lput who infected-list
      set times-infected times-infected + 1
      set infected-by lput myself infected-by
  set prob-list lput 0.6 prob-list]]
set prob 0.6

end

to virus-checks-q

    if random-float 1 < 0.2 [

    set scan? true
    set detected-computers (detected-computers + 1)
    ;ifelse alarm > 0 [
     ; set detected-when replace-item (length detected-when - 1) detected-when (ticks + 1)
```

```
    ;][
    ;set detected-when lput (ticks + 1) detected-when
    ;]

    if alarm = 0 [set detected-when lput (ticks + 1) detected-when
    ;set num-scan lput [who] of turtles with [scan?] num-scan
      set infection-to-detection lput (last detected-when - last infected-when) infection-to-
detection]
      ifelse random-float 1 < av-clean-prob [
      set av-clean? true
      ;set num-anti lput [who] of turtles with [antivirus-clean?] num-anti
      become-knownsusceptible
      set av-cleaned-computers (av-cleaned-computers + 1)
      set cleaned-when lput (ticks + 1) cleaned-when
      set infection-duration lput (last cleaned-when - last infected-when) infection-
duration

      ] [
        ifelse random-float 1 < response-likelihood [
          ifelse random-float 1 < IT-capability [
          set IT-clean? true
          ;set num-IT lput [who] of turtles with [IT-clean?] num-IT
          become-knownsusceptible
          set IT-cleaned-computers (IT-cleaned-computers + 1)
```

```
        set cleaned-when lput (ticks + 1) cleaned-when
        set infection-duration lput (last cleaned-when - last infected-when) infection-
duration
        ] [
          become-resistant
        set cleaned-when lput (ticks + 1) cleaned-when
        set isolated-computers (isolated-computers + 1)
        ;set cleaned-computers (cleaned-computers + 1)
        set infection-duration lput (last cleaned-when - last infected-when) infection-
duration
        ]
       ] [
        set alarm alarm + 1
       ]


     ]
   ]
    if (scan?) and (last detected-when != ticks + 1)[
   set alarm alarm + 1
 ]


end
```

```
to virus-checks-d

   if random-float 1 < 0.8 [
   set scan? true
   set detected-computers (detected-computers + 1)
   ;ifelse alarm > 0 [
    ; set detected-when replace-item (length detected-when - 1) detected-when (ticks + 1)
   ;][
   ;set detected-when lput (ticks + 1) detected-when
   ;]

   if alarm = 0 [set detected-when lput (ticks + 1) detected-when

   ;set num-detected lput [who] of turtles with [scan?] num-detected
     set infection-to-detection lput (last detected-when - last infected-when) infection-to-
detection]
     ifelse random-float 1 < av-clean-prob [
     ;set num-av lput [who] of turtles with [av-clean?] num-av
     become-knownsusceptible
     set av-clean? true
     set av-cleaned-computers (av-cleaned-computers + 1)
     set cleaned-when lput (ticks + 1) cleaned-when
     set infection-duration lput (last cleaned-when - last infected-when) infection-
duration
```

```
    ] [
      ifelse random-float 1 < response-likelihood [
        ifelse random-float 1 < IT-capability [
        ;set num-IT lput [who] of turtles with [IT-clean?] num-IT
         become-knownsusceptible
        set IT-clean? true
        set IT-cleaned-computers (IT-cleaned-computers + 1)
        set cleaned-when lput (ticks + 1) cleaned-when
        set infection-duration lput (last cleaned-when - last infected-when) infection-
duration
        ] [
          become-resistant
        set cleaned-when lput (ticks + 1) cleaned-when
        set isolated-computers (isolated-computers + 1)
        ;set cleaned-computers (cleaned-computers + 1)
        set infection-duration lput (last cleaned-when - last infected-when) infection-
duration
        ]
       ] [
        set alarm alarm + 1
       ]

    ]
  ]
```

```
  if (scan?) and (last detected-when != ticks + 1) [
    set alarm alarm + 1
  ]

end

to virus-scan
  if virus-scan-type = "quick"
  [quick]
  if virus-scan-type = "deep"
  [deep]
  if virus-scan-type = "complete"
  [complete]
end



to quick

  ask turtles with [infected?][
    virus-checks-q]

end

to deep
```

```
  ask turtles with [infected?][
    if virus-check-timer = 0 [
  virus-checks-d
    ]
  ]

end

to complete
  ask turtles with [infected?][
  ifelse virus-check-timer = 0 [
  virus-checks-d
  ][
  virus-checks-q
  ]]
end



to calculate-costs
  set d_up 172
  set o_r 1.3
  set avg-downtime avg_days_inf
  set d_ap 218
```

```
  set d_et 0.12
  set d_at 0.1
  set d_av 0.12
  set d_edr 0.21
  set sum-alarm sum [sum alarm-list] of turtles with [not empty? alarm-list]
  set total-infected-rate total-infected / mean-times-infected
  set av-clean-rate total-av-clean / total-detected
  set IT-clean-rate total-IT-clean / total-detected
  set resistant-rate total-isolated / total-detected
  set user-business-disruption-factor d_up * o_r * avg-downtime
  set IT-team-cost-factor d_ap * o_r
  set software-tool-cost-factor d_av * o_r
  set edr-tool-cost d_edr * no-of-edr-solutions * ticks
  set user-business-disruption (total-infected * user-business-disruption-factor); + ((no-
of-IT-personnel * IT-team-cost-factor * IT-clean-rate) + (no-of-IT-personnel * IT-team-
cost-factor * resistant-rate)) * avg-downtime
  set IT-team-cost (no-of-IT-personnel * IT-team-cost-factor) * ticks
  set software-tool-cost software-tool-cost-factor * count turtles * ticks  + edr-tool-cost
  set awareness-training-cost count turtles * d_at * awareness * ticks
  set cost-of-investment IT-team-cost + software-tool-cost + awareness-training-cost
  set cost-of-infection user-business-disruption

end
```

```
to update-globals ;; function to count number of susceptible, resistant and infected agent momentarily
  set susceptible count turtles with [color = blue]
  set known-susceptible count turtles with [color = orange]
  set resistant count turtles with [color = green]
  set infected count turtles with [color = red]
  set percentage_susceptible 100 * count turtles with [color = blue] / count turtles
  set percentage_knownsusceptible 100 * count turtles with [color = orange] / count turtles
  set percentage_resistant 100 * count turtles with [color = green] / count turtles
  set percentage_infected 100 * count turtles with [color = red] / count turtles
  set avg_days_inf (mean [days_inf] of turtles)

end
```

APPENDIX B

NETLOGO MODEL PROCEDURE FLOWCHARTS

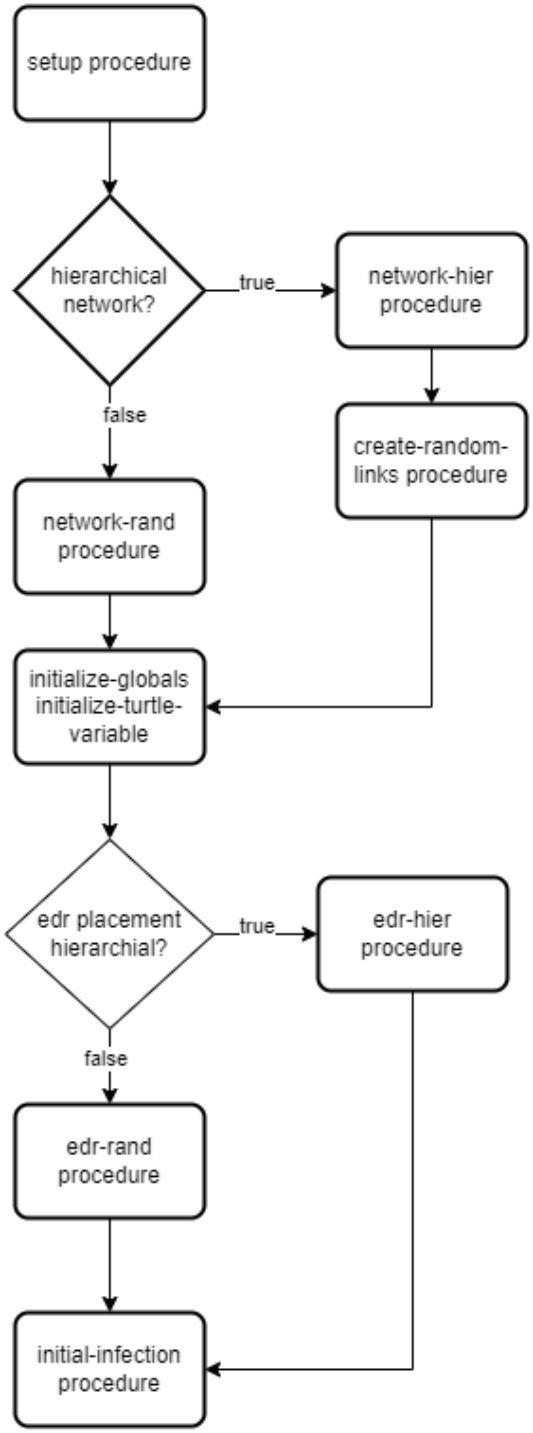

Figure B1. Setup procedure

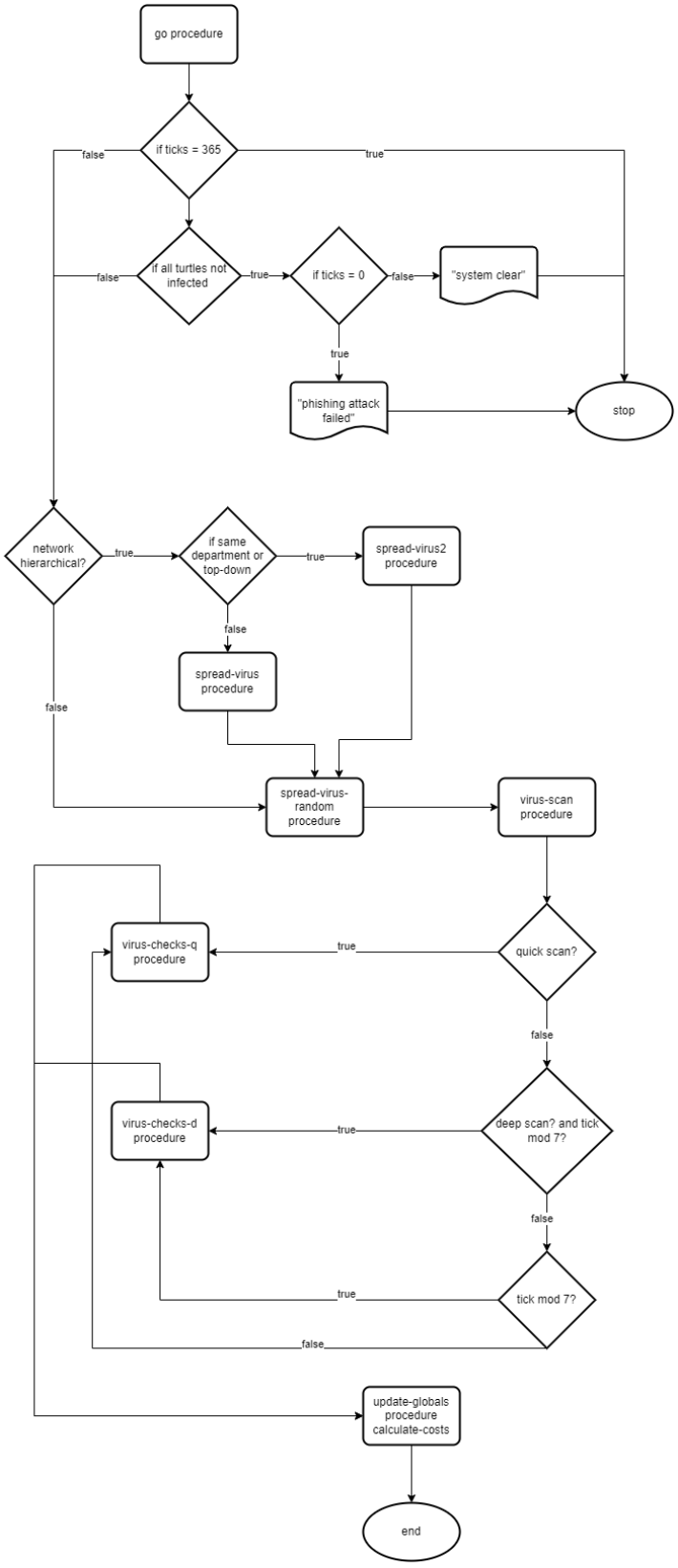

Figure B2. Go procedure

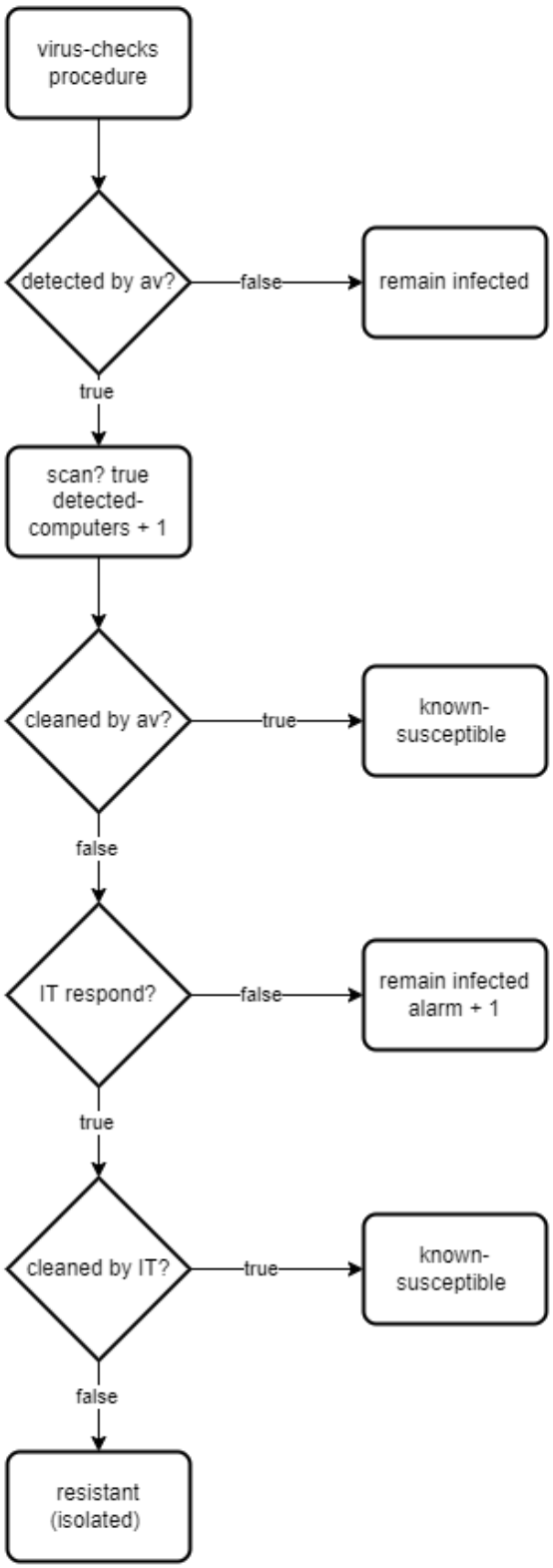

Figure B3. Virus checks procedure

APPENDIX C

AGENT-BASED MODEL SCENARIO OUTPUTS

Table C1. All outputs of scenarios (average of 20 runs)

scenario-id	ticks	initial-infected	detected-computers	infected-per-tick	susceptible	peak-simultaneous-infections	peak-infection-tick	av-cleaned-computers	IT-cleaned-computers	resistant	phished	mean-detection-to-clean	known-susceptible	max-no-times-infected	avg_days_inf	no-of-edr	mean-infection-detection	mean-infection-duration	cost-of-infection	cost-of-investment
0	365	23	14.55	0.20	0.17	73.00	2.30	0.00	0.00	0.00	22.90	0.00	0.00	1.00	181.63	0.00	4.30	0.00	2,964,569.19	781,882.56
1	132	21	6.78	5.76	11.80	60.15	2.60	5.06	0.37	32.54	24.41	3.94	10.67	22.33	24.71	11.00	2.62	3.20	3,901,199.94	320,078.73
2	132	24	6.76	5.75	11.99	58.90	2.00	5.07	0.35	32.27	28.93	4.04	10.79	22.47	24.55	11.00	2.54	3.12	3,596,236.84	307,133.92
3	144	16	6.52	5.52	12.45	59.45	3.00	4.87	0.34	32.32	19.42	4.00	11.07	22.38	24.54	11.00	2.61	3.18	3,555,304.28	295,539.88
4	103	20	6.68	5.85	12.09	59.55	2.35	5.00	0.42	31.75	24.04	3.68	11.42	18.32	18.88	11.00	2.60	3.06	2,488,436.19	244,617.37
5	161	20	6.97	5.74	11.73	59.80	2.55	5.20	0.27	31.21	24.23	4.15	11.36	25.52	31.23	11.00	2.60	3.28	6,504,886.70	441,004.78
6	159	21	6.68	5.70	11.73	60.00	2.35	5.02	0.39	32.33	24.23	3.97	11.28	26.12	28.88	11.00	2.60	3.18	5,575,364.54	358,607.27
7	109	21	6.42	5.45	11.88	60.00	2.45	4.78	0.27	31.85	24.11	3.87	12.16	17.81	19.27	11.00	2.62	3.19	2,782,627.85	264,549.09
8	237	21	3.96	3.36	11.74	60.00	2.60	2.95	0.21	36.02	24.46	5.69	8.68	26.58	40.40	11.00	4.01	4.84	7,519,486.05	540,116.72
9	265	21	3.17	2.69	11.54	61.40	2.60	2.36	0.16	30.00	24.37	1.48	7.26	17.48	64.67	11.00	7.43	8.84	9,750,607.18	633,647.61
10	125	21	6.21	5.26	15.66	52.10	2.30	4.64	0.32	30.45	24.60	4.04	10.62	22.18	23.63	11.00	2.66	3.24	4,141,053.59	338,604.91
11	98	19	6.59	5.56	12.31	58.80	4.40	4.90	0.33	22.94	21.65	3.84	20.30	15.34	17.14	11.00	2.62	3.22	1,878,976.01	229,737.57
12	71	20	11.74	10.00	12.23	57.30	1.55	8.85	0.60	29.87	24.23	2.36	18.02	20.32	10.09	11.00	1.04	1.38	1,199,008.92	144,604.77

13	120	17	5.60	4.75	23.55	47.00	2.90	4.20	0.27	23.82	23.64	3.91	10.79	20.06	18.50	22.00	2.57	3.16	2,373,401.30	302,208.39
14	57	21	6.98	6.78	12.33	59.25	2.00	5.25	0.75	31.53	24.26	1.52	11.27	11.90	10.47	11.00	2.56	2.62	863,799.56	18,913.22

Table C2. Scenario id and description

Scenario	Description
0	Worst case scenario
1	Base model
2	Awareness = 2
3	Awareness = 4
4	Response-likelihood = 0.5
5	Response-likelihood = 0.3
6	IT-capability = 0.6
7	IT-capability = 0.4
8	Quick scan
9	Deep scan
10	Hierarchical EDR placement
11	Random network topology
12	Deep scan frequency = 2
13	EDR percentage = 0.3
14	Exponential IT response-likelihood

REFERENCES

Alhogail, A., & Alsabih, A. (2021). Applying machine learning and natural language processing to detect phishing email. *Computers & Security*, *110*, 102414.

Anastasopoulou, K., Mari, P., Magkanaraki, A., Spanakis, E. G., Merialdo, M., Sakkalis, V., & Magalini, S. (2020, September). Public and private healthcare organisations: A socio-technical model for identifying cybersecurity aspects. In *Proceedings of the 13th International Conference on Theory and Practice of Electronic Governance* (pp. 168-175).

Anawar, S. Y. A. R. U. L. N. A. Z. I. A. H., Kunasegaran, D. L., Mas'ud, M. Z., & Zakaria, N. A. (2019). Analysis of phishing susceptibility in a workplace: a big-five personality perspectives. *J Eng Sci Technol*, *14*(5), 2865-2882.

Anderson, P. (1999). Perspective: Complexity theory and organization science. *Organization Science*, *10*(3), 216-232.

Appelbaum, S. H. (1997). Socio-technical systems theory: an intervention strategy for organizational development. *Management Decision*.

Aslan, Ö., Aktuğ, S. S., Ozkan-Okay, M., Yilmaz, A. A., & Akin, E. (2023). A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions. *Electronics*, *12*(6), 1333.

Babate, A., Musa, M., Kida, A., & Saidu, M. (2015). State of cyber security: emerging threats landscape. *International Journal of Advanced Research in Computer Science & Technology*, *3*(1), 113-119.

Bada, M., Sasse, M. A., & Nurse, J. R. C. (2019). Cyber security awareness campaigns: Why do they fail to change behaviour?. In G. Pernul, P. Y. A. Ryan, & E. Weippl (Eds.), *Computer Security – ESORICS 2019: 24th European Symposium on Research in Computer Security*, Luxembourg City, Luxembourg, September 23–27, 2019, Proceedings, Part II (pp. 118-136). Springer.

Baxter, G., & Sommerville, I. (2011). Socio-technical systems: From design methods to systems engineering. *Interacting with Computers*, *23(1)*, 4-17.

Bennett, C. C. (2021). Emergent Robotic Personality Traits via Agent-Based Simulation of Abstract Social Environments. *Information*, *12*(3), 103.

Beznosov, K., & Beznosova, O. (2007). On the imbalance of the security problem space and its expected consequences. *Information Management & Computer Security*, *15*(5), 420-431.

Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, *99*(suppl_3), 7280-7287.

Burns, A. J., Posey, C., Courtney, J. F., Roberts, T. L., & Nanayakkara, P. (2017). Organizational information security as a complex adaptive system: insights from three agent-based models. *Information Systems Frontiers*, *19*, 509-524.

Caltagirone, S., Pendergast, A., & Betz, C. (2013). *The diamond model of intrusion analysis*. Center For Cyber Intelligence Analysis and Threat Research Hanover Md.

Chen, Y., Zahedi, F., & Abbasi, A. (2011). Interface design elements for anti-phishing systems. In *Service-Oriented Perspectives in Design Science Research: 6th International Conference, DESRIST 2011, Milwaukee, WI, USA, May 5-6, 2011. Proceedings 6* (pp. 253-265). Springer Berlin Heidelberg.

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.

Choi, T. Y., Dooley, K. J., & Rungtusanatham, M. (2001). Supply networks and complex adaptive systems: control versus emergence. *Journal of Operations Management*, *19*(3), 351-366.

Cobb, M. (2010). Preventing phishing attacks: Enterprise best practices. *SearchSecurity. co.uk*.

Colwill, C. (2009). Human factors in information security: The insider threat–Who can you trust these days?. *Information Security Technical Report*, *14*(4), 186-196.

Continella, A., Guagnelli, A., Zingaro, G., De Pasquale, G., Barenghi, A., Zanero, S., & Maggi, F. (2016, December). Shieldfs: a self-healing, ransomware-aware filesystem. In *Proceedings of the 32nd Annual Conference on Computer Security Applications* (pp. 336-347).

Crooks, A. T., & Heppenstall, A. J. (2011). Introduction to agent-based modelling. In *Agent-based models of geographical systems* (pp. 85-105). Dordrecht: Springer Netherlands.

Dhamija, R., Tygar, J. D., & Hearst, M. (2006, April). Why phishing works. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 581-590).

Eian, I. C., Yong, L. K., Li, M. Y. X., Qi, Y. H., & Fatima, Z. (2020). Cyber attacks in the era of covid-19 and possible solution domains.

Fievet, L., & Sornette, D. (2018). Calibrating emergent phenomena in stock markets with agent based models. *PloS one*, *13*(3), e0193290.

Frank, M., Jaeger, L., & Ranft, L. M. (2022). Contextual drivers of employees' phishing susceptibility: Insights from a field study. *Decision Support Systems*, *160*, 113818.

Frauenstein, E. D. (2019). An investigation into students responses to various phishing emails and other phishing-related behaviours. In *Information Security: 17th International Conference, ISSA 2018, Pretoria, South Africa, August 15–16, 2018, Revised Selected Papers 17* (pp. 44-59). Springer International Publishing.

Frauenstein, E. D., & von Solms, R. (2009). Phishing: How an organization can protect itself. In *ISSA* (pp. 253-268).

Frauenstein, E. D., & Von Solms, R. (2013). An enterprise anti-phishing framework. In *Information Assurance and Security Education and Training: 8th IFIP WG 11.8 World Conference on Information Security Education, WISE 8, Auckland, New Zealand, July 8-10, 2013, Proceedings, WISE 7, Lucerne Switzerland, June 9-10, 2011, and WISE 6, Bento Gonçalves, RS, Brazil, July 27-31, 2009, Revised Selected Papers 8* (pp. 196-203). Springer Berlin Heidelberg.

Garello, R., & Mousavi, S. M. (2021). *A survey on Cybersecurity in 5G.*

Ge, Y., Lu, L., Cui, X., Chen, Z., & Qu, W. (2021). How personal characteristics impact phishing susceptibility: The mediating role of mail processing. *Applied Ergonomics*, *97*, 103526.

Goel, S., Williams, K., & Dincelli, E. (2017). Got phished? Internet security and human vulnerability. *Journal of the Association for Information Systems*, *18*(1), 2.

Gupta, B. B., Tewari, A., Jain, A. K., & Agrawal, D. P. (2017). Fighting against phishing attacks: state of the art and future challenges. *Neural Computing and Applications*, *28*, 3629-3654.

Gupta, S., Gupta, P., Ahamad, M., & Kumaraguru, P. (2015). Abusing phone numbers and cross-application features for crafting targeted attacks. *arXiv preprint arXiv:1512.07330*.

Halevi, T., Memon, N., & 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)*.

Ham, J. V. D. (2021). Toward a better understanding of "Cybersecurity". *Digital Threats: Research and Practice*, *2*(3), 1-3.

Harrison, B., Svetieva, E., & Vishwanath, A. (2016). Individual processing of phishing emails: How attention and elaboration protect against phishing. *Online Information Review*, *40*(2), 265-281.

Harrison, B., Vishwanath, A., Ng, Y. J., & Rao, R. (2015, January). Examining the impact of presence on individual phishing victimization. In *2015 48th Hawaii International Conference on System Sciences* (pp. 3483-3489). IEEE.

Hartono, E., Holsapple, C. W., Kim, K. Y., Na, K. S., & Simpson, J. T. (2014). Measuring perceived security in B2C electronic commerce website usage: A respecification and validation. *Decision support systems*, *62*, 11-21.

Hassan, W. U., Bates, A., & Marino, D. (2020, May). Tactical provenance analysis for endpoint detection and response systems. In *2020 IEEE Symposium on Security and Privacy (SP)* (pp. 1172-1189). IEEE.

Hinson, G. (2003). Human factors in information security. *IsecT Ltd*.

Holland, J. H. (2006). Studying Complex Adaptive Systems. *Journal of systems science and complexity*, *19*, 1-8.

Hong, J. (2012). The state of phishing attacks. *Communications of the ACM*, *55*(1), 74-81.

Ibrahim, A., Thiruvady, D., Schneider, J. G., & Abdelrazek, M. (2020). The challenges of leveraging threat intelligence to stop data breaches. *Frontiers in Computer Science*, *2*, 36.

Jain, A. K., & Gupta, B. B. (2022). A survey of phishing attack techniques, defence mechanisms and open research challenges. *Enterprise Information Systems*, *16*(4), 527-565.

Jakobsson, M. (2007). The human factor in phishing. *Privacy & Security of Consumer Information*, *7*(1), 1-19.

Jakobsson, M., & 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).

Jakobsson, M., Tsow, A., Shah, A., Blevis, E., & Lim, Y. K. (2007). What instills trust? a qualitative study of phishing. In *Financial Cryptography and Data Security: 11th International Conference, FC 2007, and 1st International Workshop on Usable Security, USEC 2007, Scarborough, Trinidad and Tobago, February 12-16, 2007. Revised Selected Papers 11* (pp. 356-361). Springer Berlin Heidelberg.

Kamal, S. U. M., Ali, R. J. A., Alani, H. K., & Abdulmajed, E. S. (2016). Survey and brief history on malware in network security case study: Viruses, worms and bots. *ARPN Journal of Engineering and Applied Sciences*, *11*(1), 683-698.

Kamruzzaman, A., Ismat, S., Brickley, J. C., Liu, A., & Thakur, K. (2022, December). A Comprehensive Review of Endpoint Security: Threats and Defenses. *In 2022 International Conference on Cyber Warfare and Security (ICCWS)* (pp. 1-7). IEEE.

Kang, J. Y., & Aldstadt, J. (2019). Using multiple scale spatio-temporal patterns for validating spatially explicit agent-based models. *International Journal of Geographical Information Science*, *33*(1), 193-213.

Kaniyamattam, K. (2022). 71 Agent-based modeling: A historical perspective and comparison to other modeling techniques. *Journal of Animal Science*, *100*(Supplement_3), 32-33.

Karamagi, R. (2022). A Review of Factors Affecting the Effectiveness of Phishing. *Comput. Inf. Sci*, *15*, 1-20.

Karantzas, G., & Patsakis, C. (2021). An empirical assessment of endpoint detection and response systems against advanced persistent threats attack vectors. *Journal of Cybersecurity and Privacy*, *1*(3), 387-421.

Katterbauer, K., Hassan, S. Y. E. D., & Cleenewerck, L. (2022). Financial cybercrime in the Islamic finance metaverse. *Journal of Metaverse*, *2*(2), 56-61.

Khan, S., Saleh, T., Dorasamy, M., Khan, N., Leng, O. T. S., & Vergara, R. G. (2022). A systematic literature review on cybercrime legislation. *F1000Research*, *11*(971), 971.

Kienzle, D. M., & Elder, M. C. (2003, October). Recent worms: a survey and trends. In *Proceedings of the 2003 ACM workshop on Rapid Malcode* (pp. 1-10).

Kohavi, L. (2021). Keeping ahead of the phishing curve. *Computer Fraud & Security*, *2021*(11), 6-8.

Kumar, S., Rautaray, S. S., & Pandey, M. (2017, November). Malvertising: A case study based on analysis of possible solutions. In *2017 International Conference on Inventive Computing and Informatics (ICICI)* (pp. 288-291). IEEE.

Lam, T., & Kettani, H. (2019, April). PhAttApp: A phishing attack detection application. In *Proceedings of the 2019 3rd International Conference on Information System and Data Mining* (pp. 154-158).

Lee, J. S., Filatova, T., Ligmann-Zielinska, A., Hassani-Mahmooei, B., Stonedahl, F., Lorscheid, I., ... & Parker, D. C. (2015). The complexities of agent-based modeling output analysis. *Journal of Artificial Societies and Social Simulation*, *18*(4).

Liggett, T. (2018). *Evolution of endpoint detection and response platforms* (Doctoral dissertation, Utica College).

Lingenfelter, B., Vakilinia, I., & Sengupta, S. (2020, January). Analyzing variation among IoT botnets using medium interaction honeypots. In *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 0761-0767). IEEE.

Macal, C. M., & North, M. J. (2010). Tutorial on agent-based modelling and simulation. *Journal of Simulation, 4(3),* 151-162.

Madihah, S., & Nazean, J. (2006, November). Knowledge structure on virus for user education. In *2006 International Conference on Computational Intelligence and Security* (Vol. 2, pp. 1515-1518). IEEE.

Malatji, M., Von Solms, S., & Marnewick, A. (2019). Socio-technical systems cybersecurity framework. *Information & Computer Security*.

McAlaney, J., & Benson, V. (2020). Cybersecurity as a social phenomenon. In *Cyber influence and cognitive threats* (pp. 1-8). Academic Press.

Miller, J. H., & Page, S. (2009). Complex adaptive systems. In *Complex Adaptive Systems*. Princeton university press.

Mitnick, K. D., & Simon, W. L. (2003). *The art of deception: Controlling the human element of security*. John Wiley & Sons.

Mohammed, M. A., Gunasekaran, S. S., Mostafa, S. A., Mustafa, A., & Abd Ghani, M. K. (2018, August). Implementing an agent-based multi-natural language anti-spam model. In *2018 International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR)* (pp. 1-5). IEEE.

Mwakalinga, G. J., & Kowalski, S. (2011). Modelling the Enemies of an IT Security System-A Socio-Technical System Security Model.

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), 72-83.

Namanya, A. P., Cullen, A., Awan, I. U., & Disso, J. P. (2018, August). The world of malware: An overview. In *2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud)* (pp. 420-427). IEEE.

Neghina, D. E., & Scarlat, E. (2013). Managing Information Technology Security in the Context of Cyber Crime Trends. *International journal of computers communications & control*, *8*(1), 97-104.

North, M., Macal, C., & Campbell, P. (2005). Oh behave! Agent-based behavioral representations in problem solving environments. *Future Generation Computer Systems*, *21*(7), 1192-1198.

Nurse, J. R. (2021). Cybersecurity awareness. *arXiv preprint arXiv:2103.00474*.

Obolensky, M. N. (2014). *Complex adaptive leadership: Embracing paradox and uncertainty*. Gower Publishing, Ltd.

Olagbemiro, A. (2019). Cyberspace as a complex adaptive system and the policy and operational implications for cyberwarfare. In *National Security: Breakthroughs in Research and Practice* (pp. 250-264). IGI Global.

Østby, G., & Kowalski, S. J. (2021). *A case study of a municipality phishing attack measures-towards a socio-technical incident management framework.* CEUR Workshop Proceedings.

Palomo-Briones, G. A., Siller, M., & Grignard, A. (2022). An Agent-Based Model of the Dual Causality Between Individual and Collective Behaviors in an Epidemic. *Computers in biology and medicine*, *141*, 104995.

Parmar, B. (2012). Protecting against spear-phishing. *Computer Fraud & Security*, *2012*(1), 8-11.

Parrish Jr, J. L., Bailey, J. L., & Courtney, J. F. (2009). A personality based model for determining susceptibility to phishing attacks. *Little Rock: University of Arkansas*, 285-296.

Perozzo, H., Zaghloul, F., & Ravarini, A. (2022). CyberSecurity Readiness: A Model for SMEs based on the Socio-Technical Perspective. *Complex Systems Informatics and Modeling Quarterly*, (33), 53-66.

Pfleeger, S. L., & Caputo, D. D. (2012). Leveraging behavioral science to mitigate cyber security risk. *Computers & security*, *31*(4), 597-611.

Pienta, D., Thatcher, J. B., & Johnston, A. (2020). Protecting a Whale in a Sea of Phish. *Journal of information technology*, *35*(3), 214-231.

Pienta, D., Thatcher, J. B., & Johnston, A. C. (2018). A taxonomy of phishing: Attack types spanning economic, temporal, breadth, and target boundaries.

Podder, P., Mondal, M., Bharati, S., & Paul, P. K. (2021). Review on the security threats of internet of things. *arXiv preprint arXiv:2101.05614*.

Pokrivčáková, S. (2017). The digital age in literary education. In: Teaching Literature for the 21st Century (pp. 11-30).

Ponsard, C., Grandclaudon, J., & Bal, S. (2019). Survey and Lessons Learned on Raising SME Awareness about Cybersecurity. *ICISSP*, 558-563.

Provos, N., Mavrommatis, P., Rajab, M., & Monrose, F. (2008). *All your iframes point to us.*

Puhakainen, P., & Siponen, M. (2010). Improving employees' compliance through information systems security training: an action research study. *MIS quarterly*, 757-778.

Puianu, M., Flangea, R. O., Marinescu, M., & Marinescu, V. (2017, September). Cloud computing for a hybrid system. In *2017 16th RoEduNet Conference: Networking in Education and Research (RoEduNet)* (pp. 1-5). IEEE.

Purkait, S., Kumar De, S., & Suar, D. (2014). An empirical investigation of the factors that influence Internet user's ability to correctly identify a phishing website. *Information Management & Computer Security*, *22*(3), 194-234.

Railsback, S. F., Lytinen, S. L., & Jackson, S. K. (2006). Agent-based simulation platforms: Review and development recommendations. *Simulation*, *82*(9), 609-623.

Rajivan, P., & Gonzalez, C. (2018). Creative persuasion: a study on adversarial behaviors and strategies in phishing attacks. *Frontiers in psychology*, *9*, 135.

Rajivan, P., Janssen, M. A., & Cooke, N. J. (2013, September). Agent-based model of a cyber security defense analyst team. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vol. 57, No. 1, pp. 314-318). Sage CA: Los Angeles, CA: SAGE Publications.

Rekouche, K. (2011). Early phishing. *arXiv preprint arXiv:1106.4692*.

Rodrigues, L. S. (2017). Challenges of digital transformation in higher education institutions: A brief discussion. In *Proceedings of 30th IBIMA Conference*.

Scheponik, T., Sherman, A. T., DeLatte, D., Phatak, D., Oliva, L., Thompson, J., & Herman, G. L. (2016, October). How students reason about cybersecurity concepts. In *2016 IEEE Frontiers in Education Conference (FIE)* (pp. 1-5). IEEE.

Seri, R., & Secchi, D. (2017). How many times should one run a computational simulation?. *Simulating Social Complexity: A Handbook*, 229-251.

Shah, D., Shah, V., Shah, H., & Kanai, P. 2017. Survey on Computer Worms. *International Journal on Recent Innovation Trends in Computing and Communication, 5(8),* 184-194.

Shahbaznezhad, H., Kolini, F., & Rashidirad, M. (2021). Employees' behavior in phishing attacks: what individual, organizational, and technological factors matter?. *Journal of Computer Information Systems*, *61*(6), 539-550.

Shim, W. (2015). Agency problems in information security: theory and application to korean business. *인터넷전자상거래연구*, *15*(5), 1-15.

Sittig, D. F., & Singh, H. (2016). A socio-technical approach to preventing, mitigating, and recovering from ransomware attacks. *Applied clinical informatics*, *7*(02), 624-632.

Sumner, C., Byers, A., & Shearing, M. (2011). Determining personality traits & privacy concerns from facebook activity. *Black Hat Briefings*, *11*(7), 197-221.

Tasmin, S., Sarmin, A. K., Shalehin, M., & Haque, A. B. (2022). Combating the Phishing Attacks: Recent Trends and Future Challenges. *Advanced Practical Approaches to Web Mining Techniques and Application*, 106-137.

Thompson, B., & Morris-King, J. (2018). An agent-based modeling framework for cybersecurity in mobile tactical networks. *The Journal of Defense Modeling and Simulation*, *15*(2), 205-218.

Thomson, K. L., Von Solms, R., & Louw, L. (2006). Cultivating an organizational information security culture. *Computer fraud & security*, *2006*(10), 7-11.

Turner, J. R., Baker, R., & Morris, M. (2018). Complex adaptive systems: Adapting and managing teams and team conflict. *Organizational conflict*, *1*, 65-93.

Twardus, J. (2005). *The use of heuristics in identifying self-propagating malicious mobile code*. West Virginia University.

Valecha, R., Gonzalez, A., Mock, J., Golob, E. J., & Raghav Rao, H. (2020). Investigating phishing susceptibility—an analysis of neural measures.

In *Information Systems and Neuroscience: NeuroIS Retreat 2019* (pp. 111-119). Springer International Publishing.

Verma, R., & Hossain, N. (2014). Semantic feature selection for text with application to phishing email detection. In *Information Security and Cryptology--ICISC 2013: 16th International Conference, Seoul, Korea, November 27-29, 2013, Revised Selected Papers 16* (pp. 455-468). Springer International Publishing.

Vishwanath, A. (2015). Examining the distinct antecedents of e-mail habits and its influence on the outcomes of a phishing attack. *Journal of Computer-Mediated Communication*, *20*(5), 570-584.

Wang, J., Herath, T., Chen, R., Vishwanath, A., & Rao, H. R. (2012). Research article phishing susceptibility: An investigation into the processing of a targeted spear phishing email. *IEEE transactions on professional communication*, *55*(4), 345-362.

Whyte, J., Dagher, G. G., & Hagenah, S. (2023, March). BEACON Labs: Designing Hands-on Lab Modules with Adversarial Thinking for Cybersecurity Education. In *Journal of The Colloquium for Information Systems Security Education* (Vol. 10, No. 1, pp. 6-6).

Wilensky, U. (1999). NetLogo. *http://ccl. northwestern. edu/netlogo/*.

Wilensky, U., & Rand, W. (2015). *An introduction to agent-based modeling: modeling natural, social, and engineered complex systems with NetLogo*. Mit Press.

Yan, R. (2019). *Cybersecurity behavior in organizations: a literature review* (Doctoral dissertation).

Yang, R., Zheng, K., Wu, B., Li, D., Wang, Z., & Wang, X. (2022). Predicting user susceptibility to phishing based on multidimensional features. *Computational Intelligence and Neuroscience*, *2022*.

Yeo, G., & Neal, A. (2008). Subjective cognitive effort: A model of states, traits, and time. *Journal of Applied Psychology*, *93*(3), 617.

Zhang, J. E., Zhao, H., & Chang, E. C. (2012). Equilibrium asset and option pricing under jump diffusion. *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics*, *22*(3), 538-568.