Information Security Risk and Maturity Analysis

Utilising Directed Graphs

by

Kevin Swann

A thesis submitted to the University of East London
for the degree of Professional Doctorate
in Information Security

School of Architecture, Computing & Engineering
University of East London

Abstract

In the current era of cyber-attack proliferation, it is imperative to better understand and mitigate information security risks within an organisation. By their nature, existing cybersecurity frameworks and standards do not model the relationships between cybersecurity elements such as controls, control objectives, threats, vulnerabilities, etc. and how one piece can impact another. This weakness in current frameworks makes it difficult to understand the context and prioritise risk mitigation activities, often resulting in a “box ticking” approach.

This thesis investigates the use of Directed Graphs as an analytical framework to represent, assess, and improve cybersecurity maturity and risk management. Traditional risk assessment models and cybersecurity frameworks often suffer from limitations such as static representations, scalability challenges, and a lack of dynamic adaptability to evolving cyber threats. This research proposes a graph-based approach to address these shortcomings, leveraging the relational power of Directed Graphs to represent assets, controls, threats, and vulnerabilities as interconnected nodes and edges.

The study begins with a comprehensive literature review of existing cybersecurity frameworks and assessment methodologies, identifying key limitations and areas where graph-based models offer improvements. A systematic methodology is presented, detailing the construction of Directed Graph models, including node and edge definitions, calculation formulas for key attributes such as Threat Value (Tv), Vulnerability Value (Viv), Risk Value (Rv), and Likelihood Value (Lv), and their mathematical justifications. The research further explores how Directed Graphs enable dynamic risk propagation analysis, gap identification, and prioritization of mitigation strategies.

A practical case study is conducted to validate the proposed model using a custom developed application called CyConex, which is used to demonstrate the effectiveness in assessing cybersecurity risks and visualizing vulnerabilities across an organizational network. Results from the case study indicate that Directed Graphs provide improved clarity, scalability, and actionable insights compared

to traditional risk management techniques. Evaluation of the results highlights both the strengths and limitations of the approach, offering recommendations for refining the model in future applications.

This thesis contributes to the evolving field of cybersecurity by presenting a scalable, adaptable, and mathematically justified Directed Graph framework for cybersecurity maturity and risk assessment. It bridges theoretical insights with practical applicability, offering a foundation for future research and real-world implementations in complex cybersecurity environments.

Acknowledgements

I want to sincerely thank my supervisor, Dr Ameer Al-Nemrat, who, without his patience, guidance, understanding and experience, this work would not have been completed.

Thanks also to my cohort friends Chris and Selcuk, whose thoughts, views and ideas were instrumental throughout my studies and in shaping the ideas for this research.

Finaly my heartfelt thanks to Claire for her unwavering supporting and understanding though this journey.

Table of Contents

Table of Figures

Table of Tables

Table of Equations

Chapter 1 - Introduction

"Information Security Risk and Maturity Analysis Utilising Directed Graphs"

In today's rapidly evolving digital landscape, organisations face growing challenges in managing and mitigating information security risks. Traditional frameworks for assessing cybersecurity risk and maturity often lack the capacity to illustrate complex interdependencies between various security elements, such as threats, controls, and vulnerabilities, limiting their effectiveness in real-world scenarios. This thesis explores an innovative approach to this problem by employing directed graphs, a mathematical structure known for its ability to represent asymmetric relationships and dependencies in interconnected systems.

Understanding information security risk is paramount for organisations in today's interconnected digital landscape. The potential for harm, damage, or loss resulting from vulnerabilities in information systems, networks, or digital assets necessitates a proactive approach to assessing and managing information security risks. Concurrently, adherence to established information security frameworks is crucial to ensure robust security measures and regulatory compliance. This thesis explores the integration of directed graphs as a comprehensive approach to assessing information security risk and framework compliance.

Organisations need to identify vulnerabilities, analyse potential threats, and evaluate the likelihood and impact of cyber-attacks or security incidents to assess information security risk effectively. This process involves comprehensive risk assessments encompassing various steps, such as system identification, asset inventory, threat identification, vulnerability assessment and risk analysis. Directed graphs offer a robust framework to model and analyse these risk factors, enabling the identification of potential attack paths, impact assessment, vulnerability mapping and risk mitigation strategies.

Adhering to information security frameworks is crucial for organisations to establish and maintain robust security practices, achieve regulatory compliance, and demonstrate their commitment to protecting sensitive information. Information security frameworks provide structured guidelines, best practices, and standards for managing information security risks. They encompass controls, policies, procedures, and technical measures organisations can adopt to enhance their security posture. By mapping framework requirements onto directed graphs and connecting them to organisational components, organisations gain insights into compliance gaps, identify areas of non-compliance, and prioritise mitigation efforts.

The increasing complexity of cybersecurity threats and the interconnected nature of organisational security components necessitate an advanced approach to security risk and maturity assessment. Traditional information security frameworks, while comprehensive, often lack the capability to represent the intricate dependencies between cybersecurity elements such as controls, objectives, and threats. This results in a fragmented view that limits the ability of organisations to effectively prioritise and mitigate risks.

Directed graphs are particularly suited to information security because they allow each element within a security model (e.g., assets, threats, controls) to be represented as nodes, while the relationships between them (e.g., dependencies, control mechanisms) are represented as directed edges. This visual representation helps reveal potential pathways for attacks, key areas of vulnerability, and strengths within the organisation's security posture. The directed graph framework thus serves as a tool for both risk and maturity analysis, enabling a dual assessment of an organisation's security capabilities and current risk exposure.

This research aims to enhance organisations understanding of their information security posture by utilising directed graphs as a visualisation and analysis tool. A directed graph, a digraph, is a mathematical structure composed of nodes and directed edges representing asymmetric relationships

or information flows between nodes. Directed graphs visually represent interconnected components, vulnerabilities, and potential threats by representing complex relationships, dependencies, and information flows within an organisation's cyber ecosystem.

The risk analysis component of this approach uses directed graphs to identify and evaluate security vulnerabilities, assess the impact of potential threats, and model complex risk scenarios. In doing so, the model provides a dynamic view of security risks, which can be continuously updated as new threats emerge or as security controls are enhanced.

The maturity analysis component leverages the same graph-based framework to assess an organisation's cybersecurity capabilities relative to established benchmarks or frameworks. This aspect of the research focuses on evaluating how well security measures are implemented, managed, and maintained across the organisation, offering insights into areas that may require further development or resource allocation.

Through the novel application of directed graphs and associated mathematical schema, this thesis aims to provide a novel visual and analytical framework that enhances decision-making by integrating both cyber risk assessment and maturity evaluation. This combined approach addresses existing gaps in traditional security frameworks, enabling a more holistic and strategic understanding of information security.

This thesis emphasises the integration of directed graphs for a holistic view of information security risk and framework compliance. Combining the visualisation and analysis capabilities of risk graphs and compliance graphs enables organisations to align risk management and compliance activities effectively. This integration facilitates risk-informed compliance, allowing organisations to prioritise efforts based on associated risks. Furthermore, it enhances decision-making by visually representing the relationships between risks, compliance requirements and organisational components.

Furthermore, this integrated approach supports ongoing monitoring and continuous improvement. The combined graph serves as a foundation for tracking changes in risk profiles, identifying emerging risks and evaluating the effectiveness of mitigation measures. Stakeholder engagement and reporting benefit from the graphical representation, enabling clear communication of the organisation's information security posture to executives, board members, auditors, and other stakeholders.

Organisations can proactively identify and address vulnerabilities, enhance security measures and achieve regulatory compliance by employing directed graphs to assess information security risk and framework compliance. This research contributes to the information security risk management field and provides practical insights for organisations aiming to develop a robust security posture. The subsequent chapters will delve into the methodology, analysis, findings, and recommendations, elucidating the value of integrating directed graphs in assessing information security risk and framework compliance.

1.1 Challenges with existing Information Security Risk and Maturity Models

Organisations find it challenging to model cybersecurity risk and measure their cybersecurity maturity due to an array of interconnected factors:

Dynamic Threat Landscape and Cybersecurity Evolution:

The ever-evolving nature of cybersecurity threats and rapid technological changes make risk modelling and maturity assessment challenging. Organisations must constantly update their models and definitions of maturity to keep pace with new malware, attack vectors and technologies such as IoT, cloud computing, AI and blockchain.

Complex IT Environments:

The intricate and interconnected nature of modern IT ecosystems, which include cloud services, on-premises systems, mobile devices, and third-party integrations, adds layers of complexity to risk modelling and understanding the overall cybersecurity posture necessary for maturity assessments.

Data Challenges:

Obtaining accurate and comprehensive data on past incidents, threats, vulnerabilities, controls, and other elements is critical for risk modelling and maturity assessment. Many organisations, especially those facing new threats, find collecting and maintaining the necessary historical data challenging.

Quantification and Qualitative Challenges: Converting cybersecurity risks into quantifiable metrics and quantifying qualitative aspects of maturity, such as organisational culture and governance, are inherently complex. This impedes the development of objective models and maturity assessments.

Human Factors:

The unpredictable nature of human behaviour, from insider threats to errors, impacts both cybersecurity risk and maturity. The human element adds uncertainty to risk models and is crucial in assessing an organisation's cybersecurity maturity.

Resource Constraints:

Both risk modelling and maturity assessments demand significant resources, including skilled personnel, tools, and time. Resource constraints are especially acute for smaller organisations.

Regulatory Complexity and Compliance:

Compliance with a constantly changing set of regulations and standards across different jurisdictions is challenging. While compliance is an aspect of cybersecurity maturity, it does not equate to maturity. Moreover, incorporating these regulatory requirements into risk models is an involved process.

Lack of Standardisation and Benchmarks:

The absence of a universally accepted standard for cybersecurity maturity and the lack of benchmarks for comparing against peers or industry standards complicates selecting frameworks and understanding where an organisation stands.

Communication, Culture and Executive Buy-in:

A culture that does not prioritise cybersecurity, coupled with a lack of effective communication between technical teams and management, hampers risk modelling and maturity assessment. Additionally, with executive support recognising the value of cybersecurity, efforts may be funded and prioritised.

Over-reliance on Tools and Solutions:

Organisations sometimes rely too much on off-the-shelf tools for risk modelling and assessing maturity. These tools may need customisation to fit the organisation's specific circumstances and risks; relying solely on them can lead to inadequate assessments.

Bias, Subjectivity and Decision Making:

The decisions involved in risk modelling and maturity assessments can be influenced by biases and subjectivity. Without an objective, data-driven approach, this can lead to unrealistic perceptions of an organisation's cybersecurity posture.

Scope, Scale and Continuous Monitoring:

Deciding on the scope and scale of risk modelling and maturity assessments is challenging. Cybersecurity maturity is not static but requires continuous monitoring and adaptation. Implementing processes for ongoing measurement and evaluation is a demanding task.

In summary, the intertwined nature of these challenges necessitates a holistic approach to cybersecurity, combining risk modelling with maturity assessments, continuous monitoring, and adaptation to the dynamic cybersecurity landscape.

Why a Visualisation Model is Required

The above stated limitation results in a fragmented view that hampers the ability to prioritise and mitigate risks effectively. A visual representation, such as a directed graph-based model, addresses this gap by enabling organisations to:

Illustrate Complex Relationships: Directed graphs effectively represent the asymmetric relationships and dependencies between security components. This visual clarity is critical for identifying potential attack paths, dependencies, and areas of vulnerability within an organisation's cybersecurity posture.

Enhance Decision-Making: By providing a dynamic and holistic view, the visualisation model aids stakeholders in prioritising risks and aligning resources with high-impact areas. It bridges the communication gap between technical teams and executives by presenting actionable insights in an intuitive format.

Facilitate Dynamic Risk Assessment: Unlike static models, a directed graph can adapt to evolving cyber threats and changing organisational controls, providing real-time updates and analysis.

Integrate Risk and Maturity Assessments: The visual model supports a dual-purpose framework, enabling simultaneous assessment of risks and maturity levels, essential for aligning security strategies with regulatory compliance and industry benchmarks.

Improve Stakeholder Engagement: The graphical representation simplifies the complexity of cybersecurity data, making it accessible to diverse stakeholders, including auditors, board members, and technical teams. Thus, it fosters collaborative decision-making.

1.2 Research Challenges

Based on professional experience and critical evaluation of existing frameworks, noteworthy research challenges exist within the information security maturity assessment domain. These present opportunities for further study in several key research areas:

Development of an Innovative Maturity Assessment Framework:

Research can focus on creating an innovative framework for assessing information security maturity that addresses current limitations, such as question-and-answer approaches to maturity assessment. Ideally, such an innovative framework should acknowledge and assess relationships among elements evaluated for security maturity evaluation.

Modelling Dependencies Between Security-Impacting Elements:

Another crucial aspect is accurately developing models to represent an organisation's interdependencies among security-impacting elements. For instance, change management practices might be effective, yet their performance depends on system reliability - modelling such interdependencies could provide more accurate assessments of security maturity levels.

Comprehensive Assessment of Enterprise Security Elements:

Research can also focus on broadening the scope of control within maturity assessment frameworks like NIST CSF to encompass more elements that impact security within an enterprise environment. Extending this scope ensures that significant elements are assessed accurately, accurately representing security maturity levels.

Establishing a Granular Taxonomy for Maturity Levels:

Existing frameworks typically use a uniform taxonomy for assessing maturity levels across various controls, which might not reflect their varying natures accurately. Research could aim at creating a more granular taxonomy that better reflects information security maturity levels within specific rules.

Integration and Standardisation Across Frameworks: There is an opportunity to research the

development of methodologies that can integrate or bridge different maturity frameworks, providing a way to leverage the strengths of each while achieving a more comprehensive and accurate maturity assessment.

Dynamic Maturity Assessment:

Considering the ever-evolving nature of cybersecurity threats and technologies, research into a dynamic maturity assessment model that adapts to real-time changes would be valuable. This could include the integration of threat intelligence and continuous monitoring.

Effectiveness Metrics and Continuous Improvement:

Research could investigate how to measure the effectiveness of information security controls and processes quantifiable and how this measurement can feed into a continuous improvement cycle for security maturity.

Human Factors and Organisational Culture:

Exploring the impact of human factors and organisational culture on information security maturity and developing models that account for these factors could also be an area of research.

This thesis will create a comprehensive, adaptive, and nuanced approach to assessing information security maturity, considering interdependencies, broadened scope of controls, granular taxonomy, and the dynamic nature of the cybersecurity landscape. The research could contribute to more effective and realistic assessments of information security maturity and help organisations better manage their cybersecurity risks.

1.3 Research Aim

The primary aim of this research is to develop a comprehensive visual framework that supports information security risk analysis and maturity assessment. This framework seeks to enhance

decision-making by enabling stakeholders to visualize complex dependencies among cybersecurity elements, leading to improved prioritization and risk mitigation.

1.4 Research Objectives

The Research objectives briefly set out what the research is aiming to achieve. The goals describe the outcomes the researcher seeks to achieve through the research activity and provide focus to the study.

The objectives for this research are:

Research Objective 1:

How can a directed graph-based framework be designed to offer a more comprehensive and accurate representation of the interactions and dependencies among human factors, policy elements and technological components within an enterprise? What advantages does this approach have over traditional frameworks?

Research Objective 2:

How does a directed graph-based maturity assessment framework enhance the understanding and evaluation of enterprise information and cybersecurity maturity? How can this approach be leveraged to develop more granular taxonomies and metrics and facilitate better-informed decision-making for risk reduction?

Research Objective 3:

What are the challenges and considerations in implementing a directed graph-based maturity assessment framework within an enterprise, and how can they be addressed to ensure the effectiveness and scalability of the model?

Research Objective 4 (Additional):

How can the directed graph-based maturity assessment framework be applied in domains such as cyber insurance, regulatory compliance assessments and organisational risk management, and what value does it bring to these areas?

These research questions aim to explore the development of a directed graph-based framework for information security maturity assessment, critically evaluate its efficacy compared to traditional frameworks and understand its applications and implications in enhancing enterprise information and cybersecurity maturity.

1.5 Research Questions

Based on the initial research, the following revised and elaborated research questions are proposed:

RQ1:

How do traditional information security frameworks address the visualization of dependencies between security elements, and where do they fall short?

RQ2:

How does a directed graph-based model enhance understanding and support a more nuanced analysis of information security maturity?

RQ3:

What challenges arise in implementing a directed graph framework for security maturity assessment, and how can they be addressed?

RQ4:

How does the Cyconex tool, using a directed graph approach, contribute to identifying and addressing organizational security risks in the case study?

These research questions are designed to delve into the efficacy of traditional information security frameworks, explore the merits of employing a directed graph-based framework and evaluate how such a framework can be leveraged to enhance the assessment and understanding of information and cybersecurity maturity within an enterprise.

1.6 Thesis Structure

The thesis is structured to provide a logical flow from foundational concepts to practical implementation and analysis, systematically addressing the research questions while building a cohesive narrative around the use of Directed Graphs for cybersecurity maturity and risk assessment. Below is a chapter-by-chapter overview:

Chapter 1: Introduction

This chapter establishes the foundation for the thesis, providing an overview of the research context, the problem statement, and the rationale for the study. It introduces the research questions and objectives, explaining the motivation for employing Directed Graphs as the primary analytical tool for cybersecurity assessment. The chapter highlights the significance of the research in addressing gaps within traditional cybersecurity frameworks and defines the scope and boundaries of the investigation.

Chapter 2: Literature Review

This chapter critically examines existing cybersecurity frameworks, maturity models, and risk assessment methodologies. It evaluates their strengths, weaknesses, and limitations, identifying the gaps that Directed Graphs can address. The review also explores prior applications of graph theory in cybersecurity, emphasizing their effectiveness and shortcomings. This positions the research within its broader academic and practical context, addressing the key knowledge gaps the thesis seeks to fill.

Chapter 3: Methodology

This chapter details the research design and methodological approach adopted in the study. It provides a rationale for selecting Directed Graphs as the analytical framework and describes the processes

involved in data collection, node and edge definition, graph construction, and attribute calculations. Key mathematical formulas, including those for Threat Value (Tv), Vulnerability Value (Vv), and Risk Value (Rv), are introduced and justified to ensure the model's theoretical robustness and applicability.

Chapter 4: Information Security Risk and Maturity Assessment and Frameworks

This chapter delves into analysing information security risk and maturity assessment frameworks. It highlights the need for an integrated approach, contrasting the limitations of existing frameworks with the potential benefits of Directed Graphs. The chapter explores how Directed Graphs can bridge the gaps in modelling interdependencies among security controls, threats, and vulnerabilities.

Chapter 5: Basic Graph Theory and Application to Information Security

This chapter introduces fundamental concepts of graph theory and their application to information security. It explains how Directed Graphs can represent assets, vulnerabilities, threats, and controls, with edges denoting relationships and dependencies. The chapter also covers advanced graph analysis techniques, such as pathfinding, centrality measures, and clustering, supporting risk and maturity assessments.

Chapter 6: Using Directed Graphs for Assessing Information Security Risk

This chapter focuses on applying Directed Graphs to assess information security risks. It demonstrates how graphs can model attack paths, assess vulnerabilities, and identify cascading risks within an organizational context. The chapter emphasizes the model's dynamic adaptability and ability to prioritize mitigation strategies based on quantitative analysis.

Chapter 7: Using Directed Graphs for Simultaneously Modelling Information Security Risk and Maturity

This chapter explores Directed Graphs' dual-purpose application for risk and maturity assessments. It demonstrates how the integrated framework enables organizations to align risk management with

maturity objectives, effectively identifying gaps and prioritizing improvements in their cybersecurity posture.

Chapter 8: Case Study and Validation

This chapter presents the practical validation of the Directed Graph model through a case study using the CyConex application. It outlines the scenario, implementation process, and data sources, showcasing how the model evaluates risks, identifies vulnerabilities, and supports decision-making. The chapter highlights the model's strengths, limitations, and practical utility.

Chapter 9: Reflection and Appraisal

This chapter reflects on the research findings and their alignment with the research questions. It discusses the implications of the Directed Graph model for cybersecurity frameworks and evaluates the challenges encountered during the study. Recommendations for refining and extending the model are also presented.

Chapter 10: Concluding Remarks and Future Work

The concluding chapter synthesizes the key findings and contributions of the research. It revisits the research questions, summarising how each was addressed throughout the thesis. Future research directions are suggested, emphasizing scalability, broader applications, and further refinement of the Directed Graph model. The chapter concludes by reflecting on the overall impact of the cybersecurity research.

Chapter 2: Literature Review

A systematic literature review relating to cyber risk management frameworks was previously undertaken (Swann, 2020). The selection of keywords for the search were based on and the authors' experience and feedback from the initial searches. During the development of the workflow keywords were identified and subsequently tuned according to the results from the initial search. The original review identified 116 papers which were compared against the general assessment methodology they proposed, novelty of the approach and a more specific assessment of the proposed methodology.

The author has undertaken an updated systematic review if the state-of-the-art literature review which have been published in the past 12 months. This review identified a further 45 papers which meet the original criteria; of these papers the following are of particular relevance.

Subsequently a further literature review relating to information security risk assessments utilising graph models was undertaken. The selection of keywords for the search were based on feedback from the earlier literature reviews. Again, following this review the results and feedback were used to shape the problem domain along with the research questions.

A number of systematic literature reviews relating to information security risk and maturity assessment frameworks have been undertaken.

As the direction and scope of the research has developed the focus of the literature reviews has changed subtly from Information Security Risk Assessment focused to Information Security Maturity Assessment focused.

The original premise for the research was to identify where graphs could be used for assessing information security risk. Initial literature reviews found a body of research work in this area. Whilst there were identified areas for further research that could be pursued a more original research area was identified in the related field of Information Security Maturity.

Consequently, the literature reviews were updated to focus on Information Security Maturity assessments and frameworks and how graphs could be used to improve such assessments. The following diagram illustrates the stages of development of these targeted literature reviews:

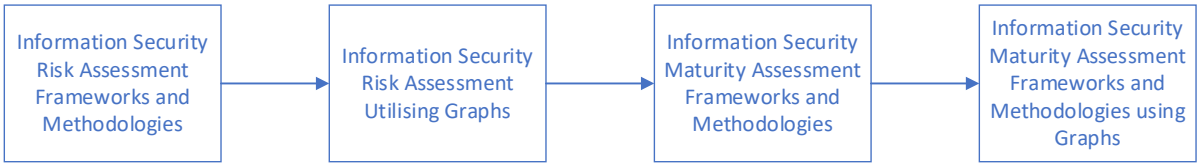

Figure 1-Target Literature Reviews

The selection of keywords for the search were based on feedback from the earlier literature reviews. The results and feedback were again used to further shape and refine the problem domain along with the research questions. The following diagram illustrates the literature review process that has been followed:

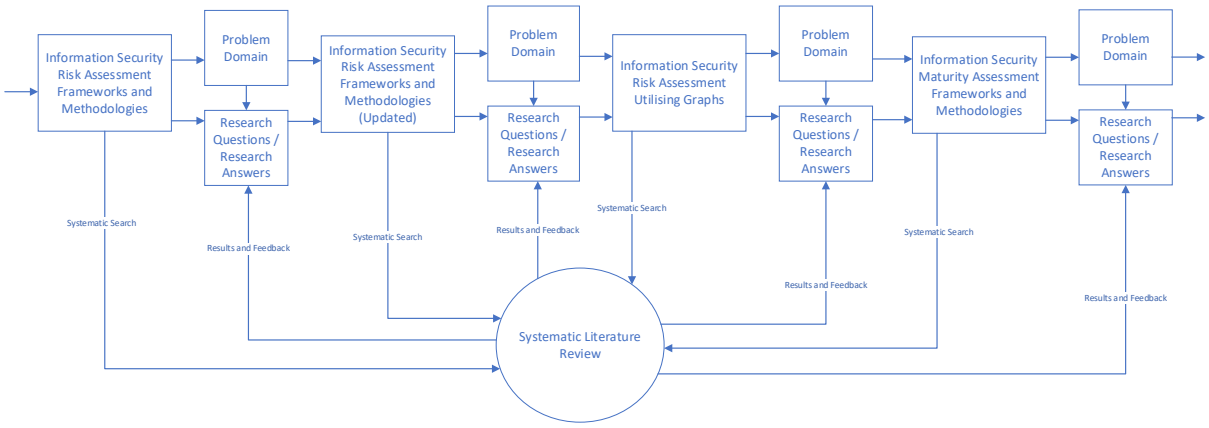

Figure 2-Literature Reviews

2.1 Key Literature

In the scholarly article by Alam, Islam, Hossain, and Hossain (2023), the researchers elaborate on a graph-based cybersecurity risk management model. The central argument revolves around the efficacy of graph theory as a tool for depicting the intricate relationships between assets, threats, and

vulnerabilities within a cybersecurity system. This model can enable the identification of potential attack paths, the evaluation of attack likelihood and impact and the formation of mitigation strategies.

Dehghantanha, Conti and Wang (2023) conducted a comprehensive survey on the several types of graphs applicable to cyber risk management. The authors outline Asset-based, Threat-based, Vulnerability-based and Security control-based as the prevalent graph types for Cyber Risk Management (CRM). The study further explores various graph-based algorithms applicable to diverse CRM tasks like Risk assessment, Risk mitigation, Risk response and Risk visualisation. The authors conclude by underscoring the advantages of graph-based methods for CRM, arguing that their benefits outweigh the challenges.

Lagraa et al. (2023) present an in-depth exploration of how graph-based data representation and analytics can enhance network security monitoring, with a particular focus on detecting botnet activity. The research emphasizes the unique advantage of modelling network traffic and communication patterns as graphs, where nodes represent individual devices or systems, and edges signify communication pathways or data flows between them. By converting network activity logs into graph structures, the study demonstrates how previously hidden or complex relationships, such as those indicative of coordinated botnet activity, can be more effectively identified and analysed. In their methodology, the authors apply advanced graph analytics techniques, including community detection and centrality measures, to pinpoint high-risk nodes and identify abnormal clusters of activity. Graph traversal algorithms are also utilized to trace communication patterns and uncover command-and-control (C2) traffic, which is a common hallmark of botnet behaviour. The findings reveal that graph-based models outperform traditional signature-based intrusion detection systems in detecting low-frequency, stealthy botnet activity.

The study on GraphSPD (*GraphSPD, 2023*) explores the application of Graph Neural Networks (GNNs) for detecting software security patches, a critical component of maintaining secure systems in

a rapidly evolving threat landscape. The research proposes an innovative approach by modelling software source code and patches as Code Property Graphs (CPGs). These graphs unify diverse representations of software, including its syntax, control flow, and data dependencies, into a single, analysable structure. The nodes within these graphs represent essential program elements such as functions, variables, and control points, while the edges capture relationships and data flows between these components. GraphSPD leverages GNNs to analyse these graphs, using trained models to predict whether specific code changes correspond to security-related patches. The approach allows for nuanced insights into software vulnerabilities by capturing both the structural and semantic relationships embedded in the codebase. Through this graph-based analysis, GraphSPD achieves a higher accuracy rate compared to traditional static analysis tools, particularly in reducing false positives that often plague conventional patch detection mechanisms.

(Kumar et al., 2023) The application of graph theory in cryptography and network security has been explored in-depth through the Graph Theory Matrix Approach (GTMA). This research examines how graph-based mathematical models can optimize various aspects of secure communication systems, cryptographic algorithms, and overall network resilience. In this approach, networks and encryption processes are modelled as graphs, with nodes representing cryptographic elements such as encryption keys, algorithms, and access control mechanisms, and edges signifying the relationships or data flows between these components. The study highlights how graph matrices can represent complex encryption workflows, providing a mathematical foundation for analysing vulnerabilities, optimizing key distribution, and ensuring secure communication channels. Graph structures enable the visualization of cryptographic dependencies, which can help identify potential weak points or misconfigurations within an encryption system. One of the most valuable contributions of this research is its ability to provide a visual and analytical foundation for understanding the intricate relationships within cryptographic systems.

The foundational work by Phillips and Swiler in network-vulnerability analysis highlights the power of graph theory in understanding and mitigating cybersecurity risks within complex network environments (Phillips & Swiler, 1998). Their research focuses on using graph-based models to represent network configurations, vulnerabilities, and potential attack paths. In their model, nodes represent assets such as servers, endpoints, or networking equipment, while edges represent connections, dependencies, or relationships between these assets. The study introduces a method for analysing vulnerabilities using graph traversal algorithms, allowing researchers to simulate potential attack pathways and assess the likelihood of exploitation. By applying techniques such as shortest-path algorithms and centrality measures, the authors demonstrate how the most vulnerable assets and critical attack vectors can be identified. Furthermore, Phillips and Swiler emphasize the importance of graph-based visualization for communicating network vulnerabilities to stakeholders. The graphical representation simplifies the complexity of interdependent vulnerabilities, offering a clear and intuitive overview of potential risks and attack scenarios.

Graph-based visual analytics have emerged as a vital tool for understanding and managing complex security incidents (Cybersecurity Springer, 2018). This study explores how threat actor activities, attack campaigns, and incident data can be represented as interconnected graph structures. Nodes within these graphs represent key entities such as compromised assets, threat actors, and vulnerabilities, while edges signify communication patterns or attack pathways. The research demonstrates how visual graph representations allow analysts to trace threat timelines, identify adversary strategies, and uncover hidden relationships between unrelated incidents. This study emphasizes the importance of graph-based visualizations in simplifying large datasets and presenting actionable insights. It underscores their value in collaborative threat analysis, enabling organizations to make real-time decisions and improve their cyber threat resilience.

In an opinion piece by Oppliger (2018), the author critiques conventional information security management approaches, arguing that they provide abstract methodologies and vague principles rather

than practical methods for quantitative risk analysis. Consequently, Oppliger proposes three viable alternatives: baseline requirement, vulnerability management and qualitative risk analysis, asserting their practicality and compatibility with other methods.

The paper "Cybersecurity Assessment Framework: A Systematic Review" (Abassi Haji Juma; Arry Akhmad Arman; Fadhil Hidayat 2003) analyses the weaknesses observed across existing cybersecurity frameworks. It highlights inconsistencies in implementation as one of the primary issues, where organizations interpret and apply frameworks differently, often leading to fragmented security postures. Additionally, limited adaptability to emerging threats is emphasized, as many frameworks are static and struggle to evolve with rapidly changing cyber risk landscapes.

Another significant weakness is the overreliance on static risk assessment models, which fail to capture real-time threat dynamics. Frameworks often lack mechanisms to integrate continuous monitoring or dynamic feedback loops, limiting their effectiveness in addressing modern cyber threats. Moreover, poor scalability remains a critical challenge, especially for large or highly distributed organizations, where frameworks cannot easily accommodate the complexity and scale of operations. The study also discusses the ambiguity in risk assessment metrics, where subjective interpretations of control effectiveness and risk impact lead to inconsistent results. Additionally, resource allocation within frameworks is frequently misaligned, causing inefficiencies in implementing security measures. Smaller organizations face challenges, such as high implementation costs, resource constraints, and skill gaps, which prevent them from fully utilizing cybersecurity frameworks.

The research concludes that current cybersecurity frameworks, while essential, often act as reactive tools rather than proactive systems. To address these shortcomings, the paper calls for frameworks incorporating dynamic risk models, context-aware adaptation mechanisms, and standardized benchmarks for measuring control effectiveness. Integrating automation, artificial intelligence, and

real-time analytics is also recommended to make frameworks more robust, scalable, and aligned with the realities of modern cybersecurity threats.

The paper "Cybersecurity Assessment MethodsWhy Aren't They Used?" (Leszczyna 2024) examines the gap between the development of cybersecurity assessment methods in academic and research domains and their limited adoption in practical, operational environments. The study identifies several key barriers that hinder widespread adoption despite the availability of robust theoretical frameworks and methodologies.

One significant issue is the complexity and impracticality of many academic assessment methods. These methods are often highly detailed, theoretical, and resource-intensive, making them difficult for organizations to implement effectively in real-world scenarios. Additionally, there is a lack of standardization across different frameworks, complicating their integration into existing cybersecurity infrastructures. Organizations struggle to align these methods with their operational needs without clear benchmarks and uniform guidelines.

The study also highlights resource constraints as a critical factor. Many organizations, especially smaller ones, lack the financial resources, skilled personnel, and time to adopt and maintain sophisticated assessment tools. Furthermore, there exists a disconnect between academic priorities and industry requirements, with academic research often focusing on emerging theoretical concepts rather than addressing immediate, practical concerns faced by cybersecurity professionals. This disconnects leads to a perceived lack of relevance of many academic assessment methods in practical security operations. The authors suggest that increased collaboration between academia and industry is essential to bridge this gap. Researchers must focus on creating methods that balance theoretical rigour with practical usability. Standardization, simplified implementation processes, and better alignment with real-world operational needs are key strategies for improving adoption rates.

Wangen, Hallstensen and Snekkenes (2018) surveyed forty-six practitioners of information security risk assessments to understand the alignment between academic research problems and industry experiences. Among the key findings were that practitioners needed to differentiate methods for different organisational tiers; the CISO typically led risk assessments; knowledge of the information asset was deemed crucial; a qualitative approach was most frequently used, and many practitioners found the methods inadequate.

Shamala et al. (2015) reviewed six information security risk assessment models and found that they primarily relied on secondary data. In response to this finding, the researchers developed a new collective information structure model based on primary data collected from a survey of information security professionals in Malaysia. This model aimed to make the risk assessment process more systematic, accurate and complete, improving its effectiveness.

Bhattacharjee, Sengupta, and Mazumdar (2013) acknowledged the challenges of risk assessments in reviewing several current methodologies, concluding that they failed to comprehensively address inter-asset relationships, dependencies among vulnerabilities and the relationships between threats and vulnerabilities. To rectify these shortcomings, the authors proposed an asset-based model, considering all these factors and the core elements of the business information system.

Alhajri et al. (2019) conducted a comprehensive literature review on the different approaches to information security risk assessment, concluding with a set of criteria deemed applicable to all methods. The authors discussed the advantages and limitations of the three main approaches – qualitative, quantitative and hybrid – based on examining several models within each category.

Wangen (2017) applied the Core Unified Risk Framework (CURF), a comprehensive risk identification, estimation, and evaluation framework, to compare three different ISRA methods based on their tasks, applications, and results. The aim was to guide ISRA practitioners in choosing a suitable way.

Fernandez and Garcia (2016) investigated the impact of minor changes in the complexity of a dependency graph on the estimated risks using the MAGERIT methodology. The authors compared the strengths and weaknesses of a complex vs. a more straightforward approach and demonstrated the possibility of using it effectively.

Aksu et al. (2017) developed a new asset and vulnerability-centric quantitative model for IT system risk assessment based on the premise that the existing structured approaches show severe defects. Their research proposes a metrics-focused risk assessment methodology that employs formulas for calculating and aggregating high and low-risk metrics. Using a four-step methodology, the team created three types of ordinal metricsbase, temporal and environmental. Their model effectively defines threat sources on attack graphs, showcasing their location, capability and motivation parameters and classifies them as high or low level.

In his article, Genchev (2020) highlights the need for an approach to tackle the problems associated with collecting and processing information required for risk assessment in Information Security Risk Management (ISRM) systems. He outlines the primary challenges in assessing and managing information risks and suggests solutions for overcoming them. A significant part of the proposed solution involves the development of a software product for collecting and processing data related to the organisation's information security risks. The proposed software can generate a dynamic risk level for each organisation's assets, necessitating constant monitoring and updating for efficacy.

Baras et al. (2014) criticised the need for an apparent structure in current models of information security management that would allow for easy reuse without exhaustive analysis. They proposed a new approach, a unified conceptual metamodel, that organises all the domain's essential properties, specifications, and components to provide a more structured framework for managing information security.

Välja et al. (2015) suggested enhancing the existing attack graph analysis solution, P2CySeMoL, by adding capabilities for analysing aspects of interoperability and availability. This enhancement would improve its accuracy when dealing with unauthorised security attacks within an organisation. The proposed metamodel improves P2CySeMoL as it allows for modelling the communication intent and differentiating attacker capabilities of all authorised system users.

Koch et al. (2000) proposed a role-based access control system using graph transformations. The system uses graph structures to manage user and administrative roles, eliminating the need for a metamodel to describe potential evolutions in the administrator structure. This work aligns with Nyanchama & Osborn (1999), who described their work on role graphs supporting role-based access control.

Sommestad et al. (2009) introduced a model-based assessment framework for analysing cybersecurity provided by different architectural scenarios. The framework utilises Bayesian statistics-based Extended Influence Diagrams to express attack graphs. Furthermore, they demonstrate how this structure can be captured in an abstract model to support analysis based on architectural models.

Sengupta et al. (2013) proposed a graph-based representation of enterprise information systems. Their methodology aids in detecting access anomalies more easily and identifies the policies (or access rights) that cause vulnerabilities.

Cheng & Zhang (2010) introduced the concept of a 'network shell' as a logical system boundary and used directed graphs to ascertain potential attack paths. However, their approach needs to provide a conventional risk assessment.

Keramati (2016) used a Bayesian graph to calculate a risk score in a system that may contain zero-day vulnerabilities. Similarly, Aksu et al. (2017) focused on establishing a risk score for a system comprising known vulnerabilities.

2.2 Review of Current Work on Existing Security Frameworks and Gaps in Knowledge

Existing security frameworks such as NIST CSF, ISO/IEC 27001, FAIR, OCTAVE, and CIS Controls provide structured approaches for managing cybersecurity risks. They focus on identifying assets, vulnerabilities, and controls and establishing compliance standards. For instance:

NIST CSF emphasises five core functions (Identify, Protect, Detect, Respond, Recover) but is flexible and requires customisation.

ISO/IEC 27001 provides a comprehensive Information Security Management System (ISMS) but lacks dynamic adaptability to evolving cyber threats.

FAIR focuses on risk quantification but is less effective in visualising complex relationships among threats, vulnerabilities, and controls.

OCTAVE targets organisational self-assessment but does not sufficiently model the interdependencies between various security components.

2.3 The Gap in Existing Knowledge

Despite their strengths, these frameworks share common limitations that hinder their effectiveness in real-world, dynamic cybersecurity landscapes:

Static Representations: Most frameworks rely on static models that fail to capture the evolving nature of cybersecurity threats and the dynamic interdependencies among organisational assets, controls, and threats.

Limited Visualisation Capabilities: Existing frameworks do not emphasise visual tools for representing relationships between cybersecurity elements, which complicates decision-making and stakeholder engagement.

Fragmented View of Security: Traditional frameworks often treat risk, compliance, and maturity assessments as separate processes, lacking an integrated, holistic approach.

Complex Interdependencies: Current methodologies inadequately model the cause-and-effect relationships between threats and vulnerabilities, limiting understanding of cascading risks.

Scalability Issues: Many frameworks struggle to scale effectively in large, distributed environments with diverse assets and controls.

This gap underscores the need for a model and approach that integrates these fragmented components into a unified framework, dynamically visualises relationships, and supports simultaneous risk and maturity assessments.

2.4 Existing Work on Directed Graphs for Information Security Analysis

The literature review highlights several applications of directed graphs in cybersecurity:

Phillips and Swiler (1998). Introduced graph-based models for network vulnerability analysis, demonstrating how directed edges can represent attack paths and help identify critical assets.

Lagraa et al. (2023). Explored graph-based models for network security monitoring, particularly for detecting botnets and analysing communication patterns.

GraphSPD (2023): Utilised directed graphs like Code Property Graphs (CPGs) to model software vulnerabilities and predict security patches.

Sengupta et al. (2013). Proposed a graph-based representation of enterprise systems for anomaly detection and vulnerability analysis.

Keramati (2016): Applied Bayesian-directed graphs for risk scoring in systems with zero-day vulnerabilities.

2.5 Gaps in Existing Graph-Based Approaches

While directed graphs have demonstrated utility in various cybersecurity contexts, significant gaps remain:

Lack of Integration with Frameworks: Existing graph-based models rarely align with established security frameworks like NIST CSF or ISO/IEC 27001, limiting their adoption in structured risk and maturity assessments.

Limited Use for Maturity Assessments: Most research focuses on risk analysis rather than assessing and improving cybersecurity maturity.

Insufficient Practical Validation: Few studies validate graph-based models through comprehensive, real-world case studies or systematic comparisons with traditional methods.

Absence of Unified Metrics: Existing models lack standardised metrics to quantify the effectiveness of security controls or the maturity of an organisation's cybersecurity posture.

This analysis identifies the gaps in existing security frameworks and the potential of directed graphs to address these shortcomings. However, it also highlights the need for further research to integrate directed graphs into holistic security and maturity assessment frameworks.

2.6 Frameworks in Cybersecurity

Frameworks are structured principles, guidelines, and standards designed to help organisations manage cybersecurity risks and improve their security posture. Examples of widely used frameworks include:

NIST Cybersecurity Framework (CSF): A risk-based approach to managing cybersecurity, emphasising five core functions (Identify, Protect, Detect, Respond, Recover).

ISO/IEC 27001: A comprehensive standard for establishing and maintaining an Information Security Management System (ISMS).

FAIR and OCTAVE: Risk management frameworks focused on assessing and quantifying cybersecurity risks.

Frameworks serve as roadmaps for implementing security practices and achieving regulatory compliance. However, they are often static, requiring organisations to adapt them to evolving threats and operational complexities. Key limitations of frameworks include:

Inability to Model Dynamic Relationships: Frameworks cannot represent interdependencies between security elements (e.g., how vulnerabilities affect risks or controls mitigate threats).

Limited Visual Representation: Frameworks typically rely on textual or tabular formats, which do not facilitate clear understanding or communication of complex security relationships.

Fragmentation of Risk and Maturity Assessments: Frameworks often treat risk management and maturity evaluation as separate processes, leading to inefficiencies in aligning priorities.

2.7 Models in Cybersecurity

On the other hand, models are analytical tools or systems designed to simulate, analyse, or represent specific aspects of cybersecurity. In this thesis, a graph-based model is proposed, using Directed Graphs to:

Represent assets, vulnerabilities, threats, and controls as interconnected nodes and edges.

Analyse dynamic relationships and cascading effects within an organisation's cybersecurity ecosystem.

Provide a visual and mathematical basis for risk assessment and maturity evaluation.

Unlike frameworks, models focus on analytical and computational representation of cybersecurity elements, offering dynamic adaptability and deeper insights into relationships.

2.8 The Focus of This Thesis is a Graph-Based Model

This thesis aims not to replace existing frameworks but to enhance their utility by introducing a complementary graph-based model. The distinction and intent are as follows:

Frameworks Provide Structure: NIST CSF and ISO/IEC 27001 offer a structured approach to implementing and managing cybersecurity controls and policies.

The Model Provides Analysis and Visualisation: The proposed Directed Graph model complements these frameworks by visualising complex relationships and enabling dynamic risk and maturity assessments.

2.9 Related Work on Graph-Based Models

Graph theory has been explored in cybersecurity to some extent, with research highlighting its potential for modelling attack paths, identifying vulnerabilities, and analysing dependencies:

> Phillips and Swiler (1998) introduced graph-based models for network vulnerability analysis.
>
> Lagraa et al. (2023) emphasised using graphs to detect botnets and analyse communication patterns.
>
> GraphSPD (2023) applied graph neural networks for software vulnerability detection using Code Property Graphs.
>
> Sengupta et al. (2013) demonstrated how graph-based enterprise system representations aid anomaly detection.

Despite these advancements, existing graph-based models often focus on specific applications, such as vulnerability detection or risk scoring. They rarely integrate with established frameworks or support dual-purpose risk and maturity assessments. This thesis addresses these gaps by developing a unified graph-based model that aligns with frameworks like NIST CSF and ISO/IEC 27001 while providing a visual and analytical tool for holistic cybersecurity management.

Chapter 3: Methodology

This chapter outlines the methodology employed to address the challenges associated with information security risk and maturity assessment, emphasizing the central role of Directed Graphs as the chosen representation tool. The research investigates the capacity of directed graphs to model the complex dependencies and relationships inherent in information security frameworks, controls, assets, and threats. Furthermore, this chapter provides a comparative analysis between directed graphs and other commonly used representation tools, such as Semantic Networks and Bayesian Networks, to establish the rationale for adopting directed graphs as the methodological foundation for this research.

Adopting directed graphs is not incidental but the result of a deliberate evaluation process. This research recognizes the need for a representation method that encapsulates structural complexity and causal relationships within cybersecurity domains. Directed graphs offer a robust mathematical structure that enables the representation of asymmetric relationships where one element impacts another unidirectionally an essential characteristic of cybersecurity dependencies. This chapter also details the methodological steps taken to construct, validate, and apply the directed graph-based model, culminating in its deployment and validation using the CyConex application.

3.1 Research Methodology Overview

The methodological approach adopted in this thesis can be broadly divided into three interconnected phases: data collection and graph schema design, graph construction and analysis, and validation through case studies. In the initial phase, cybersecurity data was gathered from established frameworks such as NIST CSF, ISO/IEC 27001, and FAIR to ensure a comprehensive representation of controls, vulnerabilities, threats, and assets. This phase involved defining the key elements to be represented as nodes and identifying their relationships, which were captured as edges.

These nodes and edges were formalized into a directed graph schema in the second phase. Nodes represented security entities such as assets, controls, objectives, and threats, while edges illustrated the

relationships, dependencies, and influence flows between these entities. Advanced graph analysis techniques, including pathfinding algorithms, centrality measures, and risk propagation models, were applied to extract meaningful insights from the graph structure.

The final phase focused on validation, where the directed graph model was implemented within CyConex, a bespoke software developed as part of this research. Through real-world case studies, the graph-based model's efficacy in assessing risk and maturity was examined, and its performance was compared against traditional assessment methodologies.

3.2 Why Directed Graphs?

Directed graphs are mathematical structures composed of nodes and directed edges that define asymmetric relationships between nodes. This directionality is critical in information security, where relationships are often inherently unidirectional. For example, a control might mitigate a vulnerability, or an attack might exploit a weakness in an asset. Directed graphs capture these cause-and-effect relationships, making them particularly suitable for representing cybersecurity dependencies.

The structured graphs align closely with cybersecurity risk and maturity assessment requirements. Security controls, objectives, vulnerabilities, threats, and assets form a network of interdependencies that cannot be fully captured through traditional static models. Directed graphs provide the flexibility to represent these relationships dynamically while preserving their underlying asymmetry. Furthermore, directed graphs support advanced analytical techniques, such as shortest-path algorithms for identifying critical attack pathways, centrality analysis for determining the most influential nodes, and flow analysis for assessing cascading impacts of vulnerabilities or controls.

A key strength of directed graphs lies in their adaptability. As cybersecurity threats evolve and organizational controls are updated, the directed graph model can dynamically incorporate these changes without requiring a complete structural overhaul. This adaptability ensures that the methodology remains relevant in rapidly changing cybersecurity landscapes. Directed graphs also

allow for dual-purpose analysis, where the same graph structure can simultaneously represent risk dependencies and maturity alignment, offering an integrated view of an organization's cybersecurity posture.

3.3 Comparison with Alternative Representation Tools

While directed graphs have been selected as the primary methodology for this research, alternative representation tools must be considered to highlight their relative strengths and limitations. Semantic Networks, Bayesian Networks, Fault Trees, and Markov Chains are the most common alternatives.

Semantic Networks, for instance, are often used to represent conceptual relationships in cybersecurity frameworks. They excel in illustrating hierarchical or associative relationships between entities. However, they do not represent dynamic dependencies and cannot effectively capture quantitative risk propagation or causal relationships. Directed graphs, in contrast, offer a more granular representation of these dependencies, mainly when modelling asymmetric cause-and-effect relationships.

Bayesian Networks are another potential tool often employed for probabilistic risk assessment. They allow organizations to evaluate certain likelihoods and the probabilistic dependencies between risk factors. While Bayesian Networks are highly effective for statistical modelling, they are often computationally intensive and require extensive historical data for accurate probability estimations. Directed graphs, by contrast, offer a more accessible approach to dependency modelling without requiring exhaustive probability datasets.

Fault Trees, commonly used for root-cause analysis, are effective at tracing the origins of vulnerabilities or control failures. However, they lack the flexibility to dynamically represent evolving dependencies, making them unsuitable for scenarios where relationships between controls, assets, and threats continuously shift. Directed graphs, however, can adapt to such changes, enabling real-time representation and analysis of security states.

Finally, Markov Chains are used to model state transitions within a system. While they are valuable in specific contexts, such as sequential event modelling, they are limited in representing complex dependencies or multilateral interactions that extend beyond state transitions. Directed graphs are inherently better equipped to handle such multivariate relationships.

In summary, while these tools have strengths, they lack the versatility, scalability, and analytical robustness of directed graphs. Directed graphs emerge as the most appropriate choice for modelling information security dependencies, offering qualitative visualization and quantitative analytical capabilities.

3.4 Methodological Implementation

The methodological implementation of directed graphs within this research followed a structured process. First, data was collected from cybersecurity frameworks, standards, and organizational documentation to represent controls, objectives, threats, and assets accurately. This data was then mapped onto a graph schema, where each component was defined as a specific node type, and relationships between these components were represented as directed edges.

Once the graph schema was established, it was populated with real-world data using graph databases and visualized through specialized graph visualization tools. Analysis techniques such as pathfinding, centrality analysis, and network clustering were employed to identify critical vulnerabilities, control dependencies, and attack pathways.

The final step involved validating the directed graph model using the CyConex software application. The model's performance was evaluated against traditional risk and maturity assessment frameworks through empirical testing and case studies, confirming its effectiveness in offering dynamic and holistic insights into an organization's cybersecurity posture.

3.5 Strengths and Limitations of the Methodology

The use of directed graphs in this research offers several advantages. They provide a holistic view of information security dependencies, enabling stakeholders to visualize complex relationships and prioritize risk mitigation efforts. Their dynamic adaptability ensures that changes in threat intelligence or control measures are easily integrated into the model. Additionally, directed graphs support advanced analytical techniques, allowing organizations to derive actionable insights from their cybersecurity data.

However, this methodology also presents certain limitations. The initial setup and schema design of directed graphs require significant domain expertise, and the model's accuracy depends heavily on the quality of input data. Furthermore, large graph datasets can introduce computational challenges, mainly when performing advanced analyses such as risk propagation modelling.

Chapter 4 - Information Security Risk and Maturity Assessment and Frameworks

This research primarily focuses on Information Security, which, per the Oxford English Dictionary, is described as "the condition of being safeguarded against unauthorised access to information, especially electronic data, or the steps implemented to realise this protection." On the other hand, a formal definition for Cybersecurity is currently absent; however, the term 'cyber' pertains to "electronic communication networks and virtual reality."

The terms 'cybersecurity' and 'information security' are frequently used interchangeably in scholarly and industry discourses (Solms & Niekerk, 2013). Solms and Niekerk maintain that while substantial convergence exists between these two concepts, they are not "entirely synonymous". They propose that cybersecurity transcends the confines of information security, encompassing information assets and other resources.

Chang et al. (1999) further elaborates on the broad scope of cybersecurity, encompassing computer security, application and operating system security and data classification and encryption. Conversely, Chang's definition of information security extends to physical security, operations security, security policy and awareness, investigation, people security and business continuity planning.

Multinational technology company IBM presents cybersecurity as safeguarding critical systems and sensitive information from digital threats (IBM, n.d.), while the UK's National Centre Security Centre (NCSC, n.d.) defines it as the approach through which individuals or organisations mitigate the risk of cyber-attacks.

From the professional perspective of the author, cybersecurity can be perceived as a subset of information security, with its scope confined to the protection of digital information and information technology systems. In contrast, Information Security protects all forms of information, whether

digital or non-digital. Consequently, within an enterprise environment where information exists both in digital and non-digital formats, it is crucial to adopt a holistic approach to information security that safeguards all information assets.

4.1 Information Security Risk

Cybersecurity risk embodies the potential for adverse consequences, such as harm, damage, or loss, precipitated by exploiting vulnerabilities inherent in information systems, networks, or digital assets. It constitutes the probability of a cybersecurity incident or attack and the potential repercussions such events can have on an organisation's functioning, reputation, and stakeholders.

The genesis of cybersecurity risks is multifaceted, originating from a spectrum of sources. External threats encompass malicious entities such as hackers, cybercriminals and actors sponsored by adversarial states. On the other hand, internal risks emerge from accidental data breaches, threats posed by insiders, or suboptimal security practices that do not align with industry standards or best practices.

A cybersecurity risk assessment entails a comprehensive procedure typically involving several crucial steps. Initially, it requires identifying the systems and data to be assessed and creating an inventory of information assets. These assets are then classified based on their value and criticality to the organisation's operations.

Subsequent steps involve the identification of potential threats, both external and internal, that could exploit the vulnerabilities inherent in the systems or networks. These threats are then mapped against the vulnerabilities and weaknesses identified by assessing systems, networks, and applications.

Evaluating the likelihood and potential impact of threats exploiting the vulnerabilities allows for a more nuanced understanding of the risk landscape. The final step involves a comprehensive risk analysis based on the identified risks' likelihood, impact, and criticality. This analysis aids in the

prioritisation of mitigation efforts, ensuring that the most substantial risks are addressed promptly and efficiently.

4.2 Information security Risk Assessment Approaches

Risk assessment is vital in identifying, analysing, and prioritising risks associated with information systems and digital assets in cybersecurity. The approaches to cybersecurity risk assessment are varied and nuanced, each offering distinct methodologies and benefits:

Quantitative Risk Assessment:

This approach embraces mathematical models and statistical techniques to quantify the likelihood and potential impact of cyber threats. The quantitative nature of this method allows for a more precise evaluation of risk, facilitating the prioritisation of security efforts and the allocation of resources. It aids in creating a cost-effective balance between the potential loss from a cyber-attack and the investment required to implement adequate security measures.

Qualitative Risk Assessment:

This approach relies on non-numerical methods, such as expert judgment, risk workshops and structured interviews, to assess cybersecurity threats. It leans heavily on the experience and insights of cybersecurity experts to identify potential vulnerabilities, threats, and their potential impact. While it may lack the precision of quantitative methods, it provides a broader, more contextual understanding of risk. It is often used with a quantitative risk assessment to achieve a comprehensive risk landscape.

Simulation-Based Risk Assessment:

This method employs computer simulations to model the potential impact of cyber threats, often encapsulating the complexity of real-world systems and interactions. By simulating different scenarios, the effectiveness of various security controls and mitigation strategies can be tested and evaluated, thereby providing insights into their practicality and efficiency before real-world implementation.

Data-Driven Risk Assessment:

This approach leverages historical data from past cyber incidents to predict future threats' likelihood and potential impact. By utilising machine learning and data analytics, emerging threats and patterns can be identified, contributing to a proactive approach to cybersecurity. This method can be instrumental in prioritising security efforts and ensuring that resources are allocated to areas with the highest potential risk.

Each approach has its strengths and weaknesses and choosing a particular method, or a combination of methods will depend on an organisation's specific needs, resources, and risk tolerance. A well-rounded cybersecurity risk assessment strategy should balance these approaches, thus comprehensively understanding the organisation's cyber risk landscape.

4.3 Information security Frameworks

A cybersecurity framework is an organised set of principles, best practices and industry standards that guide organisations to bolster their cybersecurity posture. These frameworks offer a systematic and strategic roadmap to successfully identify, safeguard, detect, respond to, and recuperate from cybersecurity threats and incidents.

A comprehensive assortment of controls, policies, procedures, and technical measures typically constitutes such frameworks. Organisations can leverage these resources to inaugurate and sustain a resilient security program. In addition, these frameworks act as a blueprint for organisations to evaluate their security proficiencies, pinpoint deficiencies and implement appropriate security controls tuned to their specific requirements and risk profile.

Adopting a cybersecurity framework offers many advantages for organisations in managing their security posture and systematically identifying and mitigating cybersecurity risks. The structured format of these frameworks simplifies assessing vulnerabilities, discerning potential threats, and executing the appropriate security controls to reduce risks effectively.

By adhering to such a framework, organisations can optimally prioritise their endeavours and allocate resources, ensuring that the most critical risks are addressed first. Furthermore, this streamlined risk management process enables organisations to respond swiftly to threats and prevent potential breaches proactively.

The practical implementation of a cybersecurity framework also helps instil a culture of security within the organisation, fostering awareness and understanding of cyber risks among employees, and promoting responsible behaviours. This holistic approach to cybersecurity management allows organisations to anticipate, prevent and respond to cyber threats, enhancing their resilience and trustworthiness in an increasingly digital and interconnected world more effectively. A cybersecurity framework is a meticulously organised set of principles, best practices and industry standards that guide organisations to bolster their cybersecurity posture. These frameworks offer a systematic and strategic roadmap to successfully identify, safeguard, detect, respond to, and recuperate from cybersecurity threats and incidents.

A comprehensive assortment of controls, policies, procedures, and technical measures typically constitutes such frameworks. Organisations can leverage these resources to inaugurate and sustain a resilient security program. In addition, these frameworks act as a blueprint for organisations to evaluate their security proficiencies, pinpoint deficiencies and implement appropriate security controls tuned to their specific requirements and risk profile.

Adopting a cybersecurity framework offers many advantages for organisations in managing their security posture and systematically identifying and mitigating cybersecurity risks. The structured format of these frameworks simplifies assessing vulnerabilities, discerning potential threats, and executing the appropriate security controls to reduce risks effectively.

By adhering to such a framework, organisations can optimally prioritise their endeavours and allocate resources, ensuring that the most critical risks are addressed first. Furthermore, this streamlined risk

management process enables organisations to respond swiftly to threats and prevent potential breaches proactively.

The practical implementation of a cybersecurity framework also helps instil a culture of security within the organisation, fostering awareness, understanding of cyber risks among employees, and promoting responsible behaviours. This holistic approach to cybersecurity management allows organisations to anticipate, prevent and respond to cyber threats, enhancing their resilience and trustworthiness in an increasingly digital and interconnected world more effectively.

4.4 The Difference Between a Maturity Assessment and a Risk Assessment

Cybersecurity maturity and risk assessment are two distinct, but related processes organisations can undertake to improve their cybersecurity posture. Here is the difference between the two:

Cybersecurity Maturity Assessment:

A cybersecurity maturity assessment evaluates the overall maturity and effectiveness of an organisation's cybersecurity program. It assesses the organisation's capabilities, practices, and controls to prevent, detect, respond to, and recover from cyber threats. A maturity assessment aims to identify strengths, weaknesses, and areas for improvement in the organisation's cybersecurity practices. It benchmarks the organisation's cybersecurity maturity level and helps develop a roadmap for enhancing cybersecurity capabilities.

Key features of a cybersecurity maturity assessment include:

Evaluating the effectiveness of cybersecurity controls and processes already implemented.

Assessing the organisation's adherence to industry best practices and cybersecurity frameworks.

Examining the organisation's governance and management of cybersecurity.

Assessing the organisation's cybersecurity culture, awareness, and training programs.

Identifying gaps and opportunities for improvement in cybersecurity practices.

Providing recommendations and a roadmap for enhancing cybersecurity maturity over time.

Cybersecurity Risk Assessment:

A cybersecurity risk assessment focuses on identifying and evaluating an organisation's risks and vulnerabilities. It involves assessing the potential impact and likelihood of various threats to the organisation's information assets and systems. A risk assessment aims to effectively identify, prioritise, and mitigate cybersecurity risks. It helps organisations make informed decisions about allocating resources and implementing appropriate controls to manage and reduce risks.

Key features of a cybersecurity risk assessment include:

Identifying and cataloguing assets, including information systems, data, and infrastructure.

Assessing potential threats and vulnerabilities that could exploit those assets.

Analysing the potential impact and likelihood of each risk scenario.

Prioritising risks based on their significance and potential consequences.

Developing risk mitigation strategies, including implementing controls, safeguards, and countermeasures.

Monitoring and regularly reassessing risks as the threat landscape evolves.

A cybersecurity maturity assessment evaluates the overall effectiveness and maturity of an organisation's cybersecurity program, while a cybersecurity risk assessment focuses on identifying and managing specific risks and vulnerabilities. Both reviews are essential components of a comprehensive cybersecurity program, helping organisations understand their current state, prioritise improvements and proactively manage risks.

4.5 Common Security Maturity Frameworks

Information and information security frameworks go back several decades. They are rooted in large enterprises or governmental organisations, such as the 'Information Security Evaluations Maturity Model' from City Group in 2000 or the 'System Security Engineering Capability Maturity Model' from the National Security Agency in 2001.

Such frameworks have continued to be developed to reflect the ongoing changes in technology and information security and have evolved into a few common frameworks and a more comprehensive number of niche frameworks focused on specific industries or organisational types.

Organisations use several common cybersecurity maturity assessment frameworks to evaluate and improve their cybersecurity posture. Some of the most widely recognised frameworks include:

NIST Cybersecurity Framework (CSF):

Developed by the National Institute of Standards and Technology (NIST), the CSF provides a risk-based approach to managing cybersecurity risks. It consists of five core functions: Identity, Protect, Detect, Respond and Recover.

ISO/IEC 27001:

This international standard provides a comprehensive framework for establishing, implementing, maintaining, and continually improving an information security management system (ISMS) within an organisation.

CIS Controls:

The Centre for Internet Security (CIS) Controls offers a set of best practices designed to help organisations protect their critical systems and data from cyber threats. It provides a prioritised list of twenty controls that organisations should implement.

COBIT (Control Objectives for Information and Related Technologies):

COBIT is an IT governance framework that helps organisations align their IT activities with business goals. It includes a comprehensive set of controls and management practices related to cybersecurity.

SANS Critical Security Controls (CSC):

The SANS CSC is a set of twenty specific security controls organisations can adopt to improve cybersecurity defences. It provides actionable and measurable steps to enhance security posture.

Cybersecurity Capability Maturity Model (CMMC):

CMMC is a framework developed by the U.S. Department of Défense (DoD) to assess and enhance the cybersecurity capabilities of defence contractors. It consists of five levels, each representing an increasing cybersecurity maturity level.

ITIL (Information Technology Infrastructure Library):

While primarily focused on IT service management, ITIL also includes guidance on managing security incidents, vulnerabilities, and risks. It provides a structured approach to managing IT services and aligning them with business needs.

These frameworks provide organisations with a structured approach to assess their cybersecurity maturity, identify gaps, and prioritise improvements. Organisations may adopt one or multiple frameworks to enhance their cybersecurity posture depending on the specific needs and regulatory requirements.

4.5.1 NIST CSF

In 2013, then US President Obama issued an Executive Order for Improving Critical Infrastructure Cybersecurity, which required the development of a voluntary risk-based information security framework that provided a “prioritised, flexible, repeatable, performance-based and cost-effective approach” to managing information security risk for critical infrastructure services. Consequently, a

framework was developed in the US, led by the National Institute of Standards and Technology (NIST).

The NIST CSF was designed to allow organisations to assess their information security business risks and to guide their use of the framework cost-effectively and pragmatically.

The framework is divided into three parts:

First, the Framework Core is a set of activities, outcomes and references that describe approaches to elements of information security consisting of five functions, subdivided into twenty-two categories of security outcomes and ninety-eight security controls.

Secondly, the framework an organisation can use Implementation Tiers to clarify its assessment of information security risk and the degree of sophistication of its management approach.

The Framework Profile is a list of outcomes an organisation has chosen from the categories and subcategories based on its business needs and individual risk assessments.

The Framework core functions are by far the most widely implemented part of the NIST CSF and consist of the following five functional areas:

Identify – Develop the organisational understanding to manage information security risk to systems, assets, data, and capabilities.

Protect – Develop and implement the appropriate safeguards to ensure the delivery of critical infrastructure services.

Detect – Develop and implement the appropriate activities to identify the occurrence of an information security event.

Respond – Develop and implement the appropriate activities to act regarding a detected information security event.

Recover – Develop and implement the appropriate activities to maintain resilience plans and restore any capabilities or services impaired by an information security event.

Each functional area is divided into categories of information security outcomes related to activities, such as ‘Asset Management’ and ‘Access Control.’ Subcategories further divide a class into specific security controls. Example security controls include ‘External information systems are catalogued’ and ‘Data-at-rest is protected.’

The NIST CSF does provide a structured approach for organisations to manage and improve their cybersecurity posture. It addresses the complex interactions and dependencies among controls applied to people, policy, and technology within an enterprise setting. The CSF is designed to be flexible and adaptable, making it a valuable tool for understanding and managing these relationships. However, there are still some areas where the framework may fall short in addressing these dependencies:

People, Policy, and Technology: The NIST CSF explicitly recognises the importance of addressing cybersecurity from the perspective of people, processes (policy) and technology. It includes categories like "Protect," "Detect," and "Respond," which encompass various controls related to these elements.

Interdependencies:

The framework acknowledges the interdependencies between controls through its structured approach. For example, it encourages organisations to identify how specific controls (e.g., access controls or security awareness training) support and interact with each other to achieve cybersecurity objectives.

Shortcomings of NIST CSF:

Specific Control Implementation: While the NIST CSF outlines categories and subcategories of controls, it does not provide detailed, prescriptive guidance on how to implement each control. Organisations may need additional NIST publications (such as NIST Special Publication 800-53) or industry-specific standards to get more granular implementation details.

Customisation and Adaptation: The flexibility of the CSF, while advantageous, can also be a challenge. Organisations must determine which controls and practices are most relevant to their context. This customisation can be complex and requires a deep understanding of the organisation's risks and dependencies.

Maturity Assessment: The framework provides a structure for assessing cybersecurity maturity but does not offer specific criteria or metrics to measure the maturity of controls related to people, policy, and technology. Organisations may need to develop their assessment criteria or refer to other sources.

Dynamic Nature of Dependencies: The cybersecurity landscape constantly evolves and the dependencies between controls, people, policy, and technology can change rapidly. The CSF does not provide real-time or continuous monitoring guidance for assessing how changes in one area might impact others.

In summary, the NIST CSF addresses the complex interactions and dependencies among controls applied to people, policy, and technology within an enterprise setting. Its strength lies in its structured approach and recognition of these interdependencies. However, organisations must be prepared to customise and adapt the framework to their needs and context. They may need to supplement it with more detailed implementation guidance and assessment criteria to address the relationships between controls and their interdependencies fully.

4.5.2 Cybersecurity Capability Maturity Model (C2M2)

The Cybersecurity Capability Maturity Model (C2M2) was developed in 2012 by the U.S. energy sector and the Department of Energy (DOE) and is intended for evaluating and improving organisations' cybersecurity. The C2M2 is managed by the DOE's Office of Cybersecurity, Energy Security and Emergency Response (CESER) Cybersecurity for Energy Delivery Systems (CEDS) division.

The C2M2 assessment is intended to provide a manageable and detailed description of an organisation's information security. It assesses the maturity of the organisation's information security in ten categories or domains and identifies areas that may be improved. Also, the outputs of a C2M2 assessment offer valuable baseline information if organisations consider adopting one of the formal information security standards such as ISO 27001.

The ten domains within C2M2 are:

Risk Management

Asset, Change and Configuration Management

Identity and Access Management

Threat and Vulnerability Management

Situational Awareness

Information Sharing and Communications

Event and Incident Response, Continuity of Operations

Supply Chain and External Dependencies Management

Workforce Management

Cybersecurity Programme Management

C2M2 assesses around three hundred controls across the ten domains where each control has a Maturity Indicator Level, or MIL, which measures the control’s implementation within the organisation. Each control is scored with one of four classifications:

Not Implemented: No evidence of the control being implemented exists.

Partially Implemented: There is some evidence of relevant activity, usually on an ad-hoc basis.

Largely Implemented: Clear evidence exists that controls are in place and used by many staff.

Fully Implemented: Strong controls are fully embedded within the organisation's day-to-day operation.

Like the NIST Cybersecurity Framework, C2M2 addresses the complex interactions and dependencies among controls applied to people, policy, and technology within the energy sector. Here is an analysis of how C2M2 addresses these aspects and where it may fall short:

Coverage:

People, Policy, and Technology: C2M2 explicitly recognises the importance of addressing cybersecurity from a holistic perspective. It divides cybersecurity into domains: Governance, Risk Management, Resilience, Access Control and Account Management. These domains encompass controls related to people, policy, and technology.

Interdependencies:

The framework acknowledges the interdependencies between controls within and across domains. It emphasises the need for organisations to consider how rules and practices in one field may impact or support those in another.

Shortcomings:

Specific Control Implementation: Like the NIST CSF, C2M2 does not provide detailed, step-by-step implementation guidance for individual controls. Organisations may need to refer to other resources or standards for specific implementation details.

Customisation and Adaptation: C2M2, while comprehensive, may need customisation to fit an organisation's specific needs. It is primarily tailored to the energy sector, so organisations in other industries may find some controls outside their context.

Maturity Assessment: C2M2 offers a maturity model for organisations to assess their cybersecurity practices, but it does not provide specific criteria or metrics for measuring the maturity of controls in detail. Organisations may need to develop their metrics for assessing control effectiveness and maturity.

Dynamic Nature of Dependencies: The framework does not inherently provide real-time monitoring or continuous assessment capabilities. Organisations must implement processes to adapt to changing cybersecurity risks and dependencies over time.

Limited Applicability: C2M2 is primarily designed for the energy sector, so it may not fully address the needs or nuances of cybersecurity in other industries.

In summary, the Cybersecurity Capability Maturity Model (C2M2) offers a structured approach to assessing and improving cybersecurity practices within the energy sector. It does address the complex interactions and dependencies among controls applied to people, policy, and technology to a significant extent. However, organisations using C2M2 should be prepared to customise it to their specific context, supplement it with detailed implementation guidance and establish metrics for assessing control maturity. Additionally, they should consider that C2M2 is industry-specific and may not be directly applicable outside the energy sector.

4.5.3 ISO/IEC 27001

ISO/IEC 27001 is an international information security management system (ISMS) standard. It provides a systematic approach for organisations to establish, implement, maintain, and continually improve their information security practices. The standard specifies the requirements for creating an effective ISMS that protects information confidentiality, integrity, and availability.

The standard provides a list of controls to address various aspects of information security, including access control, physical security, human resources security, cryptography, incident management,

business continuity and compliance. These controls are customisable based on an organisation's needs and risk assessments.

ISO/IEC 27001 requires organisations to develop and maintain specific documentation to support their ISMS. This includes policies, procedures, guidelines, and records demonstrating information security controls' implementation and effectiveness.

Organisations can undergo a formal certification process to demonstrate compliance with ISO/IEC 27001. Certification involves an independent audit by a certification body to assess the organisation's ISMS against the standard's requirements. Achieving certification indicates a commitment to information security best practices and can enhance an organisation's reputation.

ISO/IEC 27001 is an international information security management system (ISMS) standard. It, too, addresses the complex interactions and dependencies among controls applied to people, policy, and technology within an enterprise setting. Here is an analysis of how ISO/IEC 27001 handles these aspects and where it falls short:

Coverage:

People, Policy, and Technology:

ISO/IEC 27001 recognises the importance of addressing information security from people, processes (policy) and technology perspectives. It includes requirements for establishing, implementing, monitoring, reviewing, maintaining, and improving an ISMS.

Interdependencies:

The standard implicitly acknowledges the interdependencies between controls, primarily through risk assessment and management processes. Organisations are encouraged to identify and assess risks, determine rules to mitigate them and evaluate the effectiveness of these controls.

Shortcomings:

Detailed Control Implementation: ISO/IEC 27001, like the NIST CSF, does not provide detailed, step-by-step implementation guidance for specific controls. It defines what needs to be accomplished (e.g., risk assessment, access control policies) but leaves the specifics of how to achieve it to the organisation. This can be challenging for organisations needing more expertise in information security.

Customisation and Adaptation: ISO/IEC 27001 offers a framework but does not specify which controls or policies an organisation should implement. It is up to the organisation to tailor the standard to its specific needs and context. This customisation can be complex and require significant effort.

Lack of Maturity Assessment: ISO/IEC 27001 does not provide explicit maturity assessment criteria for people, policy, and technology controls. Organisations often must develop their maturity models or refer to other standards.

Continuous Improvement: While ISO/IEC 27001 encourages a culture of continuous improvement through the PDCA (Plan-Do-Check-Act) cycle, it does not prescribe specific methodologies or tools for ongoing assessment of control effectiveness or real-time monitoring of dependencies.

Dynamic Nature of Dependencies: Like the NIST CSF, ISO/IEC 27001 may need help to keep up with the rapid changes in the cybersecurity landscape. It does not provide real-time guidance for assessing how evolving threats and technologies impact the interdependencies between controls.

In summary, ISO/IEC 27001 addresses the complex interactions and dependencies among controls applied to people, policy and technology within an enterprise setting to some extent. It provides a structured framework for managing information security risks but leaves many implementation and customisation details to the organisation. Organisations may need to supplement ISO/IEC 27001 with additional guidance, best practices, and maturity models to effectively manage these relationships and dependencies.

4.5.4 CIS Controls

The CIS Controls, formerly the SANS Critical Security Controls (CSC), is a set of best practices designed to help organisations improve their cybersecurity posture. Developed by the Centre for Internet Security (CIS), the CIS Controls provide a prioritised and actionable list of security measures that organisations can implement to enhance their defences against cyber threats.

Some key aspects of the CIS Controls:

Risk-Based Approach: The CIS Controls are based on a risk management approach, focusing on the most common and impactful cyber threats organisations face. The controls are organised into three implementation levels: Basic, Foundational and Organisational. This allows organisations to prioritise and implement controls based on specific risks and available resources.

Actionable Guidance: The CIS Controls provide specific, actionable guidance for each control. They offer clear recommendations on what organisations should do to implement the control effectively. This includes technical configurations, system hardening guidelines, security policies and procedures.

Consensus-Driven: The CIS Controls are developed through a consensus-driven process involving a broad community of cybersecurity experts, practitioners, and organisations. This collaborative approach ensures that the controls represent the collective knowledge and experience of the cybersecurity community.

Continuous Monitoring: The CIS Controls emphasise the importance of continuous monitoring and assessment. Organisations are encouraged to regularly measure the effectiveness of implemented controls, identify gaps, and adjust their security measures accordingly. This helps organisations stay proactive and respond to emerging threats effectively.

Integration with Other Frameworks: The CIS Controls are designed to complement and align with other cybersecurity frameworks and standards, such as the NIST Cybersecurity Framework (CSF),

ISO/IEC 27001 and COBIT. This allows organisations to integrate CIS Controls into their cybersecurity programs and initiatives.

Security Automation: The CIS Controls recognise the value of security automation in improving efficiency and effectiveness. The controls include guidance on leveraging automated tools and technologies to implement, monitor and enforce security measures. Automation helps organisations streamline security processes and reduce human error.

Continuous Improvement: The CIS Controls framework promotes a culture of continuous improvement. Organisations are encouraged to reassess their security posture regularly; update controls as new threats emerge and stay informed about the latest cybersecurity trends and technologies.

The benefits of implementing the CIS Controls include:

Improved Defence: By implementing the prioritised controls, organisations can strengthen their defences against common and significant cyber threats, reducing the risk of successful attacks.

Practical Guidance: The CIS Controls provide actionable recommendations that organisations can readily implement to enhance their security posture, even with limited resources.

Community Support: The CIS Controls benefit from a community-driven approach, leveraging the expertise and experience of a broad range of cybersecurity professionals and organisations.

Compliance Alignment: The CIS Controls align with various regulatory requirements and frameworks, making it easier for organisations to demonstrate compliance with industry standards and regulations.

Continuous Adaptation: The CIS Controls encourage organisations to continuously monitor, assess and adapt their security measures to evolving threats, helping them stay resilient.

The Center for Internet Security (CIS) Controls, like the NIST CSF, the CIS Controls address complex interactions and dependencies among controls applied to people, policy, and technology within an enterprise setting, but they have some unique characteristics:

Coverage:

People, Policy, and Technology: The CIS Controls are organised into three implementation groups, with Group 1 focusing on basic cybersecurity hygiene, Group 2 on foundational security and Group 3 on advanced security practices. These groups encompass controls related to people, policy, and technology.

Interdependencies:

The CIS Controls recognise the interdependencies among controls and guide prioritising and implementing them based on an organisation's risk profile. They emphasise that controls should be implemented in a prioritised manner to build a strong security foundation.

Shortcomings:

Specific Control Implementation: Like the NIST CSF, the CIS Controls provide high-level descriptions of controls but do not offer detailed, step-by-step implementation guidance. Organisations may need to refer to additional resources or standards for specific implementation details.

Customisation and Adaptation: While the CIS Controls provide a prioritised list of controls, organisations must still customise them to their specific needs. The controls are not one-size-fits-all and adaptation is necessary to address an organisation's unique risks and dependencies.

Maturity Assessment: The CIS Controls offer a framework for assessing an organisation's cybersecurity maturity. However, they do not provide specific metrics or criteria for measuring the maturity of controls related to people, policy, and technology. Organisations may need to develop their assessment criteria.

Dynamic Nature of Dependencies: The cybersecurity landscape constantly evolves and the interdependencies between controls and their effectiveness can change over time. The CIS Controls do not provide continuous monitoring or real-time assessment guidance.

In summary, CIS Controls are a valuable framework for addressing complex interactions and dependencies among controls related to people, policy, and technology within an enterprise setting. They provide a prioritised list of controls and recognise the need for customisation based on an organisation's specific context. However, like the NIST CSF, organisations may need to supplement the CIS Controls with more detailed implementation guidance, assessment criteria and real-time monitoring capabilities to fully address the relationships between controls and their dependencies.

4.6 Common Security Risk Frameworks

Organisations use several common cybersecurity risk frameworks to assess, manage and mitigate cyber risks. Some of the most widely recognised frameworks include:

NIST Cybersecurity Framework (CSF): The NIST CSF provides a comprehensive approach to managing and reducing cybersecurity risks. It consists of five core functions: Identity, Protect, Detect, Respond and Recover. The framework helps organisations align their cybersecurity efforts with business objectives and prioritise risk mitigation measures.

ISO/IEC 27005: This international standard focuses on information security risk management. It provides a systematic and structured approach to identifying, assessing, and treating information security risks. ISO/IEC 27005 guides organisations in establishing an effective risk management process and integrating it into their cybersecurity program.

FAIR (Factor Analysis of Information Risk): FAIR is a quantitative risk assessment framework that aims to provide organisations with a more accurate and defensible understanding of their cybersecurity risks. It uses a structured approach to quantify and prioritise risks based on likelihood, impact, and vulnerability.

OCTAVE (Operationally Critical Threat, Asset and Vulnerability Evaluation): OCTAVE is a risk assessment methodology developed by Carnegie Mellon University. It helps organisations identify and prioritise information security risks based on their critical assets and operational objectives. OCTAVE focuses on understanding an organisation's risk environment and developing risk mitigation strategies.

COBIT (Control Objectives for Information and Related Technologies): While primarily an IT governance framework, COBIT also includes guidance on managing cybersecurity risks. It provides a comprehensive set of controls and management practices that organisations can implement to mitigate risks related to information and technology.

COSO (Committee of Sponsoring Organisations of the Treadway Commission) ERM (Enterprise Risk Management) Framework: Although not explicitly focused on cybersecurity, the COSO ERM Framework provides a broader perspective on risk management. It helps organisations assess and manage risks holistically, considering internal and external factors impacting business objectives, including cybersecurity risks.

ITIL (Information Technology Infrastructure Library): ITIL includes guidance on managing various aspects of IT service management, including risk management. While not solely focused on cybersecurity, ITIL provides a framework for identifying and addressing risks associated with IT services and their impact on business operations.

These frameworks offer structured approaches to identify, assess and manage cybersecurity risks. Organisations often adopt one or a combination of these frameworks based on their needs, industry requirements and regulatory obligations.

4.6.1 ISO/IEC 27005

ISO/IEC 27005 is an international standard that guides information security risk management. It is part of the ISO/IEC 27000 series, which includes various standards and guidelines for information security management systems (ISMS).

Some key aspects of ISO/IEC 27005:

Risk Management Process: ISO/IEC 27005 outlines a systematic and structured information security risk management approach. It provides a framework for organisations to establish and maintain an effective risk management process within their overall information security program.

Risk Assessment: The standard emphasises the importance of conducting risk assessments to identify and analyse information security risks. It guides organisations' methods, techniques, and tools to assess risks associated with their assets, vulnerabilities, threats, and impacts.

Risk Treatment: ISO/IEC 27005 helps organisations develop risk treatment plans based on the identified risks. It guides organisations in selecting appropriate risk mitigation measures and controls to manage and reduce risks to an acceptable level. The standard also addresses residual risk and needs ongoing monitoring and reassessment.

Integration with ISO/IEC 27001: ISO/IEC 27005 aligns with ISO/IEC 27001, the standard for information security management systems. It guides how to integrate risk management practices into the overall ISMS framework ISO/IEC 27001 established. By combining both standards, organisations can develop a comprehensive and risk-based approach to information security management.

Risk Communication: ISO/IEC 27005 emphasises the importance of effective communication and stakeholder involvement in risk management. It guides how to communicate risks to relevant stakeholders, including senior management, decision-makers, and other interested parties, to ensure a shared understanding of risks and their potential impacts.

Documentation and Reporting: The standard addresses risk management activities' documentation and reporting requirements. It emphasises maintaining records of risk assessments, treatment plans and risk-related decisions. Clear and concise reporting helps organisations track risk management efforts, demonstrate compliance and support informed decision-making.

Continuous Improvement: ISO/IEC 27005 encourages organisations to establish a culture of continuous improvement in their risk management practices. It emphasises the need for regular monitoring, review, and reassessment of risks to adapt to changing threats, vulnerabilities, and business environments.

By adopting ISO/IEC 27005, organisations can establish a consistent and structured approach to information security risk management. The standard helps organisations identify, assess, treat, and monitor information security risks. It enables them to make informed decisions and allocate resources effectively to protect their assets and meet their information security objectives.

ISO/IEC 27005 is an international standard that guides information security risk management. It is part of the ISO/IEC 27000 series, which includes various standards and guidelines for information security management systems (ISMS).

ISO/IEC 27005 is an international standard on information security risk management. It provides guidelines and a systematic approach for identifying, assessing, and managing information security risks within an organisation. Like the analysis of the NIST CSF, here is an assessment of how well ISO/IEC 27005 addresses the complex interactions and dependencies among controls applied to people, policy, and technology within an enterprise setting, as well as its potential shortcomings:

Coverage:

People, Policy, and Technology: ISO/IEC 27005 considers the importance of addressing risk management from the perspective of people, policy (such as security policies and procedures) and technology. It recognises that risks can arise from vulnerabilities in any of these areas.

Risk Interdependencies: The standard emphasises the need to consider interdependencies between various aspects of an organisation, including its business processes, technology infrastructure, human resources, and external factors. It encourages organisations to identify how changes in one area may affect others and lead to new risks.

Shortcomings:

Specific Control Implementation: ISO/IEC 27005 is primarily a risk management framework and does not provide detailed, prescriptive guidance on implementing specific security controls. Organisations may need to refer to other ISO/IEC standards (e.g., ISO/IEC 27001) or industry-specific standards for control implementation details.

Customisation Required: Like NIST CSF, ISO/IEC 27005 requires significant customisation to fit an organisation's specific context and needs. While it provides a structured risk management process, organisations must adapt it to their unique risk landscape, which can be complex and challenging.

Lack of Maturity Assessment: ISO/IEC 27005 does not include specific maturity assessment criteria for people, policy, and technology controls. It focuses more on risk assessment and management rather than control maturity.

Continuous Monitoring: The standard does not offer explicit guidance on continuous monitoring of risk and control effectiveness. Organisations may need to integrate other monitoring and assessment practices to ensure that controls remain effective.

Integration with Other Standards: ISO/IEC 27005 is often used in conjunction with other ISO/IEC standards, such as ISO/IEC 27001 (Information et al.) and ISO/IEC 27002 (Code of Practice for Information Security Controls), to provide a more comprehensive approach to information security. This integration may be necessary to address the full range of controls and their interdependencies.

In summary, ISO/IEC 27005 can be considered a valuable standard for information security risk management and recognises the importance of addressing risks related to people, policy, and technology within an organisation. However, it primarily focuses on risk assessment and management and does not provide detailed guidance on control implementation or maturity assessment. Organisations often use ISO/IEC 27005 with other ISO/IEC standards and frameworks to create a comprehensive approach to information security that addresses the complex relationships between controls and their dependencies.

4.6.2 FAIR (Factor Analysis of Information Risk):

FAIR, which stands for Factor Analysis of Information Risk, is a quantitative risk assessment framework that provides a structured approach to evaluating and measuring information security risks. Developed by the FAIR Institute, FAIR aims to enhance risk management practices by applying a standard methodology for analysing and communicating cyber risks.

Here are some critical aspects of FAIR:

Quantitative Risk Assessment: FAIR focuses on quantitative analysis, which means it assigns numerical values to various factors related to risks, such as likelihood, impact, and vulnerability. This allows for more precise measurement and comparison of risks, facilitating informed decision-making.

Factors and Variables: FAIR breaks down the risk assessment into different factors and variables. Factors include threat event frequency, vulnerability, and potential impact. Variables further refine these factors, providing more specific parameters for risk analysis.

Risk Scenario Analysis: FAIR employs risk scenario analysis, which involves identifying and evaluating specific risk scenarios. Each risk scenario considers different threat sources, potential events, and impacts. By examining a range of scenarios, organisations gain a more comprehensive understanding of their risk landscape.

Calibration: FAIR incorporates calibration, aligning risk estimations with available data and expert judgment. This helps organisations make risk assessments grounded in real-world data and experienced insights, reducing subjectivity, and enhancing accuracy.

Risk Measurement: FAIR enables the measurement of risk in terms of probable loss and frequency of occurrence. It employs various techniques, such as Monte Carlo simulation, to model and quantify risk factors and derive meaningful risk metrics.

Risk Communication: FAIR emphasises effective risk communication by providing a standardised vocabulary and structure for discussing and reporting risks. This facilitates clear and consistent communication between stakeholders, allowing for better risk understanding and decision-making.

Continuous Improvement: FAIR encourages continuous improvement by emphasising ongoing monitoring, assessment, and refinement of risk analysis. By updating risk models and incorporating new data, organisations can adapt to evolving threats and maintain accurate risk assessments.

Benefits of using FAIR include:

Improved Decision-making: FAIR provides a quantitative basis for evaluating risks, enabling organisations to prioritise resources and make data-driven decisions. It helps stakeholders understand the potential impact of risks and assess the cost-effectiveness of risk mitigation efforts.

Enhanced Risk Communication: FAIR's standardised approach to risk communication improves understanding and facilitates meaningful discussions among stakeholders. This leads to better risk awareness and more effective collaboration in managing risks.

Scalability: FAIR can be applied to various risks across multiple industries. Its flexibility allows organisations to adapt and tailor the framework to their needs and risk profiles.

Consistency and Reproducibility: FAIR promotes consistency in risk assessments by providing a standardised methodology. This allows for reproducibility, making comparing and tracking risks over time and across different parts of an organisation more manageable.

Integration with Other Frameworks: FAIR can be integrated with other risk management frameworks and standards, providing a quantitative analysis component to enhance existing risk management practices.

Factor Analysis of Information Risk (FAIR) is a framework for analysing and managing information security risks. It takes a quantitative approach to risk assessment, aiming to provide more precise and structured insights into risk factors and their interactions. Here is an analysis of how FAIR addresses complex interactions and dependencies among risk factors related to people, policy, and technology within an enterprise setting:

Coverage:

People, Policy, and Technology: FAIR recognises that information security risks can result from various factors, including human behaviours (people), organisational policies and procedures (policy) and technology vulnerabilities and assets (technology). It explicitly considers these dimensions when assessing risk.

Interdependencies: FAIR's strength lies in its ability to model and quantify the interdependencies between various risk factors. It provides a structured framework for understanding how changes or vulnerabilities in one area (e.g., technology) can impact the overall risk posture.

Shortcomings:

Complexity: The quantitative nature of FAIR can be both a strength and a weakness. The framework can be complex and require significant expertise to implement effectively, making it less accessible for smaller organisations or those with limited resources.

Data Requirements: FAIR relies on data to perform quantitative risk assessments. Gathering and maintaining the necessary data can be challenging for some organisations, especially when measuring and quantifying human factors.

Integration Challenges: While FAIR provides a robust methodology for risk analysis, it may not offer explicit guidance on integrating risk management into broader business processes or policy development. Organisations may need to complement FAIR with other frameworks or practices.

In summary, FAIR is a comprehensive framework that addresses complex interactions and dependencies among risk factors related to people, policy, and technology within an enterprise setting. Its quantitative approach and focus on interdependencies make it a valuable tool for organisations seeking a more precise understanding of their information security risks. However, it may require significant resources and expertise to implement effectively, and organisations should be prepared to customise it to their specific needs and context. Additionally, it is important to consider its data requirements and integration challenges when adopting.

4.6.3 OCTAVE (Operationally Critical Threat, Asset and Vulnerability Evaluation)

OCTAVE, which stands for Operationally Critical Threat, Asset and Vulnerability Evaluation, is a risk assessment methodology developed by Carnegie Mellon University's Software Engineering Institute (SEI). It is designed to help organisations identify and prioritise information security risks and develop risk mitigation strategies based on their specific operational objectives and critical assets.

Coverage:

Risk Assessment Approach: OCTAVE takes a holistic approach to risk assessment, focusing on the organisation's critical assets and operational objectives. It combines asset-based, threat-based, and vulnerability-based risk assessments to understand risks comprehensively.

Structured Methodology: OCTAVE follows a structured methodology consisting of three phases: Phase 1 (Build Asset-Based Threat Profiles), Phase 2 (Identify Infrastructure Vulnerabilities) and

Phase 3 (Characterise Risks). Each phase involves specific activities and techniques to systematically identify, analyse and prioritise risks.

Asset-Based Approach: OCTAVE emphasises identifying and understanding critical assets and their importance to the organisation's mission or business objectives. Organisations can prioritise risk mitigation efforts and allocate resources effectively by focusing on assets.

Threat Profiles: OCTAVE uses threat profiles to identify and analyse potential threats and threat actors relevant to the organisation. Threat profiles help understand the motivations, capabilities and potential actions of attackers that may pose risks to critical assets.

In the second phase, OCTAVE aims to identify vulnerabilities in the organisation's infrastructure that threats could exploit. This includes examining technical vulnerabilities, procedural weaknesses and human factors that may contribute to risks.

Risk Characterisation: In the final phase, OCTAVE characterises risks by combining the knowledge of critical assets, threats, and vulnerabilities. This step involves analysing the potential impact of risks and assigning risk ratings or prioritisation scores to guide risk mitigation efforts.

Risk Mitigation Strategies: OCTAVE helps organisations develop risk mitigation strategies and plans based on the identified risks. This may involve implementing controls, improving security measures, enhancing policies and procedures, or addressing process weaknesses. The goal is to develop a risk management roadmap tailored to the organisation's needs.

Shortcomings

Specific Control Implementation: While OCTAVE offers a structured risk assessment and management approach, it does not offer detailed, step-by-step guidance on implementing individual controls. Organisations may need supplementary resources or industry-specific standards to obtain more granular implementation details.

Customisation and Contextualisation: The adaptability of OCTAVE, though advantageous, can present challenges. Organisations must discern which controls and practices are most pertinent to their unique circumstances. This customisation can be intricate and necessitates a profound comprehension of the organisation's risks and dependencies.

Maturity Assessment: The framework outlines a methodology for assessing cybersecurity maturity, but it does not furnish specific criteria or metrics to gauge the maturity of controls related to people, policy, and technology. Organisations might need to devise their assessment criteria or refer to external sources for guidance.

Dynamic Nature of Dependencies: The cybersecurity landscape is in perpetual flux and the dependencies among controls, individuals, policy, and technology can change rapidly. OCTAVE does not provide real-time or continuous monitoring recommendations for evaluating how alterations in one area might affect others.

The OCTAVE framework addresses the intricate interactions and dependencies among controls applied to people, policy, and technology within an enterprise context. Its strength lies in its structured methodology and acknowledgement of these interdependencies. Nevertheless, organisations must be prepared to tailor and adjust the framework to suit their specific requirements and context. They may need to supplement it with more detailed implementation guidance and assessment criteria to address the relationships between controls and their interdependencies comprehensively.

4.7 Assessing the Impact on Risk from the Implementation of Frameworks

Evaluating the repercussions of implementing a cybersecurity framework on cybersecurity risk entails a comprehensive examination of how the framework enhances an organisation's capacity to manage and mitigate risks. This evaluation process is multifaceted and includes several critical steps.

Framework Mapping involves aligning the requirements and suggestions of the chosen cybersecurity framework with the identified risks and vulnerabilities. This mapping process aids in visualising how the framework addresses the specific risks that the organisation faces.

Gap Analysis is conducted to pinpoint any discrepancies or areas where the organisation's current security measures or practices do not meet the standards set by the framework. Determining the size and criticality of these gaps is vital to understanding their potential impact on cybersecurity risk.

The Risk Treatment Assessment phase evaluates how implementing the framework's recommended controls, practices and procedures will influence the identified risks. This phase is crucial to tailoring the framework's recommendations to the organisation's risk landscape.

Mitigation Effectiveness is then assessed to measure how the framework's implementation reduces the identified risks and vulnerabilities. This is a critical step in validating the effectiveness of the security measures.

Compliance Enhancement assessment considers the influence of the framework on the organisation's compliance posture. This is particularly important in industries where regulatory compliance is mandatory.

Post-implementation Risk Assessment compares the risk landscape after implementing the framework with the baseline risk assessment. Evaluating the reduction in identified risks, enhancements in security posture and the overall impact on cybersecurity risk provides a comprehensive view of the framework's effectiveness.

Given the complexity and scope of these tasks, it is evident that an enormous investment of time, effort and resources is essential for this approach to be practical. Organisations must devote dedicated personnel, including cybersecurity professionals or consultants and allocate sufficient time and budget to each stage. Furthermore, it is essential to note that cybersecurity is an ongoing process. Therefore,

continuous monitoring, reviewing and improvement efforts are vital for maintaining a robust security posture in the long term.

Chapter 5 – Basic Graph Theory and Application to Information Security

Graph theory is an integral branch of mathematics that studies the properties and behaviour of structures known as graphs. A graph is an abstract representation of two or more objects where links connect them; these objects are called nodes and edges (or arcs) join them. Graph theory models relationships and processes in various disciplines, such as computer science, physics, sociology, and biology.

At its core, graph G is usually denoted by G = (V, E), where V is a non-empty set of vertices and E is an edge set with either one or two endpoints for every edge. An edge thus connects its endpoints.

Graphs can be divided into two main groups: directed or undirected. An undirected graph has edges without an associated direction; they may be traversed in either direction. On the other hand, edges in a directed digraph have an assigned direction and each edge forms an ordered pair of vertices.

Vertex degrees, which indicate how many edges connect to each vertex, are a core concept in graph theory. When discussing directed graphs, we distinguish between their in-degree (number of incoming edges) and out-degree (number of outgoing edges).

Graphs can be further defined based on their properties. For instance, they can be described as connected if there exists an uninterrupted path between any two vertices of a connected graph; similarly, a cycle occurs if its path starts and ends at the same vertex, while connected graphs with no cycles present define trees.

The application of graph theory extends into numerous fields. For instance, in computer science, algorithms, such as Dijkstra's shortest path algorithm, have been developed to solve problems modelled with graphs. In biology, graphs are used to model structures of molecules in chemistry and the spread of diseases in epidemiology.

5.1 Directed Graphs

Directed graphs, often called digraphs in mathematical terminology, are fundamental constructs representing relationships between entities within a particular set. These entities, termed vertices, or nodes are interconnected by directed edges, lines or arcs with intrinsic directionality. This characteristic distinguishes directed graphs from their undirected counterparts, which depict symmetric relationships. Conversely, directed graphs illuminate asymmetric relationships, signifying the flow of information, influence, or any other unidirectional interaction between nodes.

Every edge within a directed graph, in terms of its inherent direction, originates from a specific node (designated as the source) and terminates at a different node (identified as the target or destination). This directed edge visually embodies the concept of an arrow or a directed line segment, thereby encapsulating notions such as cause-and-effect, dependencies, or sequential relationships between nodes. These directed edges capture the essence of unidirectional interactions or influences between various nodes within the directed graph.

The applicability of directed graphs spans a plethora of real-world scenarios. For instance, they can model information flow in communication networks, where data travels from one node to another in a specific direction. They can also represent network topology in computer networks, demonstrating how nodes are interconnected and how data packets navigate the network. Furthermore, directed graphs can depict dependencies in various contexts, like project management, where specific tasks must precede others, or in software libraries, where some functions rely on others. Finally, they can even illustrate influence or hierarchical relationships, such as social influence in social networks or predator-prey relationships in ecological systems.

A diverse set of graph algorithms facilitates the analysis and optimisation of directed graphs. Traversal algorithms, like depth-first search (DFS) or breadth-first search (BFS), are employed to systematically visit all the vertices in a graph, unveiling structural properties or discovering specific nodes. Path-

finding algorithms like Dijkstra's or Bellman-Ford's algorithms are utilised for more complex scenarios, where the shortest or most efficient path between nodes needs to be determined. These powerful computational tools enable the examination, analysis, and optimisation of directed graphs, making them suitable for various applications, from network routing and social network analysis to biological pathway discovery and machine learning.

5.2 Entities, Nodes, Relationships and Edges

Nodes in a graph represent individual entities or objects, while edges depict relationships among them. Each node, in turn, represents one of these entities directly: each node often corresponds with precisely one node within the graph.

Entities: Entities can refer to any objects, elements, or entities you want to represent and examine within a system. Regarding cybersecurity risk assessment, entities might include servers, network devices, users, applications databases, or any other relevant system component.

Nodes: In graph theory, nodes represent entities as individual units or points. Each node corresponds to an entity within a system. For instance, if you were looking at cybersecurity risk in network infrastructure analysis, each device (router, switch) would be represented as nodes in your graph.

Edges: In a graph, edges represent relationships among entities. They connect nodes and show how these connections exist between entities. They can define data flow, dependencies, control relationships, trust or user access relationships depending on their purpose and context.

By visualising entities as nodes and their relationships as edges in a graph, one can visualise and analyse the complex interconnections and dependencies within a system. This helps in understanding its overall structure and identifying vulnerabilities or potential impacts of risks within it.

5.3 Nodes

In the graph, nodes represent elements that assess cybersecurity risk or maturity. There are five main classes of nodes:

Threat Actor: Represents an entity or an individual that threatens the cybersecurity system.

Attack: Represents a specific type of attack that a threat actor can carry out.

Vulnerability: Represents a weakness or flaw in the system that an attacker can exploit.

Asset: Represents a valuable component or resource within the system that needs to be protected.

Control: Represents a security control or measure that can be implemented to mitigate risks and protect assets.

Each node in these classes has attributes that describe it in a real-world context. The attributes fall into three main types:

Metadata: These attributes are primarily used by the underlying application to describe the structure and properties of the node.

Content: These attributes provide real-world information about what the node represents. They help in understanding the specific details or characteristics of the node.

Values: These attributes are used for the graph's risk and maturity calculations. They may represent quantitative values or measures associated with the node.

Additionally, there are three supporting classes of nodes:

Objective: Represents a specific objective or goal related to cybersecurity risk management or maturity.

Group: Represents a collection or category of related nodes, often used for organisational or classification purposes.

Information: Represents additional information or data that can be associated with a node to provide more context or details.

5.4 Edges

In the graph, edges represent relationships between nodes. Each edge has attributes that describe the relationship in a real-world context. The attributes can be categorised into three main types:

Metadata: These attributes are primarily used by the underlying application to describe the structure and properties of the edge.

Content: These attributes provide real-world information about what the edge represents. They help understand the specific details or characteristics of the relationship between the connected nodes.

Values: These attributes are used for the graph's risk and maturity calculations. They may represent quantitative values or measures associated with the relationship.

The edges connect nodes, indicating how they relate to cybersecurity risk or maturity. The specific type of relationship represented by an edge depends on the classes of connected nodes and the context of the graph.

By representing nodes and edges in this way, the graph allows for the modelling and analysis of cybersecurity risks, relationships, and dependencies, enabling the assessment of risk levels, identification of vulnerabilities and the evaluation of security controls and measures.

5.4.1 Edge Strength Value

The Edge Strength Value (Ev) modifies how node output values (e.g., Threat Actor Mitigated Value, Attack Mitigated Value, etc.) are passed to destination nodes.

5.5 Graph Schema

In graph theory, a critical concept is the graph schema, an abstract representation serving as a meta-model of a graph-based data model. This framework or blueprint encapsulates the expected structure of a graph, specifying the different types of nodes or vertices, edges or relationships and associated properties which can be found within the graph. It helps provide consistency in interpreting and manipulating the graph data across various analyses or applications.

Types of Nodes or Vertices: The graph schema elucidates the variety of nodes present. These nodes usually denote diverse entities. As an example, within a graph designed to illustrate a social network, nodes might be categorised as "Person", "Company" and "Event". These node types encapsulate various entities that are fundamental to the modelled system.

Types of Edges or Relationships: The graph schema also articulates the different types of edges that can be established between nodes. These edges typically signify a spectrum of relationships between the nodes. Continuing with the social network example, relationships might be classified as "Friend", "Employer", or "Attended". These edge types serve to signify distinct relational structures within the model.

Properties: The nodes and edges within a graph may possess properties contributing additional details about the nodes and edges. For instance, within a "Person" node, properties such as "Name", "Age" and "Location" might be presented. Conversely, a "Friend" edge might bear properties such as "Since", indicating the year the friendship commenced.

The schema contributes a cohesive, structured overview of what the data within the graph might comprise, facilitating effective querying and manipulation of the graph data. This becomes indispensable within fields such as databases or knowledge graphs, where the data structure necessitates comprehension before it can be interacted with effectively.

A similar concept known as an ontology is utilised in closely related fields such as Semantic Web and the Resource Description Framework (RDF) data model. Ontologies provide more sophisticated techniques to model the relationships between varying types of entities and their properties, offering additional depth and complexity in modelling the data.

5.6 The Role of Frameworks in Supporting the Development of Models and Their Contribution to Research

Frameworks play a foundational role in shaping cybersecurity research's structure, methodology, and analytical approach, mainly when using Directed Graphs for risk assessment, maturity evaluation, and compliance validation. Widely recognized frameworks such as ISO/IEC 27001, NIST Cybersecurity Framework (CSF), and CIS Controls provide established standards, guidelines, and methodologies for identifying, managing, and mitigating cybersecurity risks. These frameworks serve not only as reference points for best practices but also as structured datasets from which models can be built, validated, and refined.

5.6.1 Standardization and Structure for Model Development

Cybersecurity frameworks offer standardized structures that define key components such as controls, control objectives, risks, and mitigation strategies. These components act as predefined nodes in a Directed Graph model, where:

Controls represent specific measures implemented to mitigate risks.

Control Objectives define desired security outcomes or compliance requirements.

Risks represent potential threats or vulnerabilities impacting assets.

Dependencies and Relationships between these nodes form the graph's edges, indicating causality, influence, or control flow.

By leveraging these standardized elements, the research can map complex cybersecurity environments into graph structures that are both logically consistent and scalable. This structured foundation ensures that the resulting models are aligned with recognized best practices and are not arbitrarily constructed.

5.6.2 Consistency Across Domains

Frameworks also ensure consistency across different cybersecurity domains. For example:

ISO/IEC 27001 focuses on a comprehensive, risk-based approach to managing information security.

NIST CSF emphasizes a lifecycle approach through core functions like Identify, Protect, Detect, Respond, and Recover.

CIS Controls highlight prioritized, actionable measures to prevent and respond to cyber threats.

This consistency enables the research to build models that transcend organizational boundaries and remain relevant across different industries or regulatory environments. The research benefits from this universality, as the graph-based model can be applied to multiple compliance scenarios without requiring fundamental structural changes.

5.6.3 Providing Measurable Parameters

Most frameworks introduce quantifiable parameters for evaluating controls, threats, and vulnerabilities. Framework guidelines often embed metrics such as Control Effectiveness, Likelihood of Exploitation, Residual Risk Impact, and Asset Criticality. These parameters directly inform the mathematical calculations used in the Directed Graph model, such as:

Vulnerability Value (Vv)

Likelihood Value (Lv)

Risk Value (Rv)

By using framework-defined metrics, the research ensures that calculations are grounded in established methodologies and can be transparently justified. This enhances the model's credibility and facilitates validation against real-world cybersecurity scenarios.

5.6.4 Facilitating Model Validation and Benchmarking

Frameworks provide reference benchmarks against which the model's outputs can be validated. For example:

A Directed Graph model can compare residual risk scores against target benchmarks defined by ISO 27005.

NIST CSF maturity levels (e.g., Tier 1: Partial, Tier 4: Adaptive) can be represented as graph node attributes, enabling dynamic analysis of compliance maturity across different domains.

CIS Controls offer clear mappings between control activities and threat mitigation objectives, enabling precise graph traversal analysis for identifying risk propagation pathways.

These benchmarks allow the research to verify whether the Directed Graph model aligns with real-world compliance and risk management expectations, adding reliability and reproducibility to the research findings.

5.6.5 Supporting Comprehensive Gap Analysis

Frameworks inherently highlight expected states for controls, assets, and security objectives. Directed Graph models can overlay real-world security data onto these expected states to visualize and quantify compliance or risk management gaps. For example:

Nodes representing critical controls can be evaluated for missing edges or broken dependencies.

Vulnerability pathways can be traced back to unfulfilled framework requirements.

Residual risk nodes can be compared against predefined acceptable risk thresholds.

This structured analysis supports gap identification at both granular and systemic levels, directly aligning with the research's compliance and audit goals.

5.6.6 Enhancing Research Scalability and Adaptability

Frameworks are designed to be scalable across organizations of different sizes and levels of complexity. Similarly, Directed Graph models benefit from this scalability by:

Adapting to small-scale assessments (e.g., focusing on specific controls or asset groups).

Scaling up to enterprise-wide risk assessments involving thousands of nodes and edges.

Moreover, frameworks evolve over time to address emerging cybersecurity threats, regulatory changes, and technological advancements. When anchored to framework principles, graph-based models inherit this adaptability, ensuring they remain relevant and future-proof.

5.6.7 Contribution to Research Objectives

From a research perspective, frameworks provide:

Empirical Data Sources: Control mappings, maturity scores, risk parameters, and other predefined metrics serve as empirical inputs for model testing.

Validation Mechanisms: Results derived from Directed Graph models can be validated against compliance audits or certification results based on the chosen framework.

Theoretical Justification: Frameworks offer a theoretical foundation for justifying model choices, attribute weightings, and analytical pathways.

This synergy supports the research by ensuring that findings are rooted in established cybersecurity principles, thereby enhancing the work's overall academic rigour and practical relevance.

Frameworks such as ISO/IEC 27001, NIST CSF, and CIS Controls are critical in developing, validating, and implementing Directed Graph models for cybersecurity assessments. By offering standardization, consistency, quantifiable parameters, and validation benchmarks, frameworks provide both the theoretical foundation and practical structure required for building robust models.

In this research, integrating these frameworks ensures that the Directed Graph methodology is scalable, adaptable, and aligned with recognized cybersecurity best practices. This alignment enhances the findings' transparency, reliability, and applicability, contributing to a more structured and data-driven approach to managing cybersecurity risk and compliance.

5.7 How Does an Information Security Maturity Model Work?

Typically, an information security maturity model delineates a spectrum of elements that significantly influence an organisation's capability to manage information and information security proficiently. These elements encapsulate diverse areas, encompassing leadership and governance, risk management procedures and technical safeguards.

Each of these elements is accompanied by a description of the practices one would anticipate finding within an organisation at varying maturity levels. This detailed explication facilitates a comprehensive understanding of what each level of maturity entails, and the requisite practices needed to achieve it.

When an organisation assesses its overarching security maturity, it conducts a comparative analysis between its existing practices and those delineated across the levels of each element within the maturity model. This comparison enables the organisation to gauge its current level of maturity, identify gaps or areas of weakness and gain insight into the practices it needs to adopt or enhance to elevate its level of information security maturity.

This process helps organisations benchmark their current state and provides a roadmap for continuous improvement in their information security management. The maturity model is a strategic guide,

enabling organisations to systematically assess and improve their practices, progressively advancing their information security maturity over time.

An information security maturity model is a valuable tool that aids organisations in the effective management of their information security practices. A structured approach to assessing and improving these practices facilitates the organisation's journey towards achieving a robust and mature information security posture.

5.8 Use of Directed Graphs for Assessing Framework Compliance

Directed graphs are an extremely useful tool for assessing an organisation's compliance with a cybersecurity framework. Their capacity to model intricate relationships and dependencies makes them invaluable in visualising and interpreting an organisation's alignment with prescribed cybersecurity standards.

In the context of mapping cybersecurity framework requirements, each stipulation within the framework can be encapsulated as a distinct node within the graph. In contrast, the relationships, dependencies, or interactions between these requirements can be represented as directed edges. This structured graphical model aids in establishing a clear and comprehensive understanding of the compliance assessment structure, facilitating the systematic analysis of framework adherence.

When mapping the various components within an organisation, such as assets, applications, infrastructure, policies and procedures, each element can be represented as individual nodes within the graph. This illustrated embodiment of organisational components furnishes a detailed overview of the organisational elements contributing to cybersecurity and their interrelations, thereby providing a holistic view of the organisation's security posture.

A critical application of directed graphs in assessing framework compliance lies in establishing the connections, or directed edges, between the organisational components and the corresponding framework requirements. These edges symbolise the alignment between the organisation's practices

and the cybersecurity framework's expectations, indicating the extent to which specific components fulfil the required standards.

Analysing the directed graph enables a robust compliance assessment with the cybersecurity framework requirements. The degree of compliance can be determined by evaluating the presence or absence of directed edges between organisational components and framework requirements. The absence of an edge might signal potential gaps or areas of non-compliance, while the presence of an edge implies alignment and adherence to the framework.

Lastly, directed graphs extend to visualisation and reporting. The graphical representation of the compliance status can be leveraged to present a clear, intuitive depiction of the compliance assessment findings. This visual portrayal not only simplifies the interpretation of the compliance status but also facilitates effective communication of the findings to various stakeholders, enabling informed decision-making in enhancing the organisation's cybersecurity posture.

Achieving and demonstrating compliance requires a structured, repeatable approach that can adapt to evolving risks, organizational changes, and audit requirements. Directed graphs offer an analytical and visualization tool for managing this process, providing clarity and precision in capturing relationships, dependencies, and gaps across controls, assets, and risks.

When applied to framework compliance assessments, directed graphs support organizations across five key stages: Planning, Data Gathering, Assessment, Gap Analysis and Compliance Reporting.

5.8.1 Planning

In the planning stage of a cybersecurity maturity review, the use of directed graphs can assist in several ways:

Defining Scope and Objectives: Directed graphs can help visualise the organisation's cybersecurity landscape, including its assets, systems, networks, and relationships. By representing these

components as nodes and their connections as directed edges, the organisation can gain a comprehensive overview of its scope and define the specific areas to be included in the maturity review. The directed graph can help identify the boundaries of the assessment and set clear objectives based on the nodes and edges to be evaluated.

Identifying Data Sources: Directed graphs can assist in identifying and mapping the data sources that need to be collected for the maturity review. Each node in the graph represents a data source, such as documentation, policies, procedures, or technical configurations. By visualising the connections between nodes, the organisation can determine the relevant data sources for the assessment and ensure comprehensive data gathering.

Understanding Interdependencies: Directed graphs can depict the interdependencies between different components of the organisation's cybersecurity practices. The directed edges in the graph represent the relationships and flows of information or influence between nodes. By analysing these interdependencies, the organisation can identify critical paths, dependencies, or potential risks that may impact its cybersecurity maturity. This understanding helps prioritise the assessment and focus on areas where improvements or controls are needed.

5.8.2 Data Gathering

In the data-gathering stage of a cybersecurity maturity review, the use of directed graphs can assist in several ways:

Organising Data Sources: Directed graphs can serve as a visual framework for organising and categorising the various data sources that need to be collected for the review. Each node in the graph represents a specific data source, such as documentation, policies, procedures, or technical configurations. By structuring the graph based on the types of data sources, the organisation can clearly represent the information it needs to gather.

Identifying Relationships: Directed graphs can help identify the relationships between data sources and their interdependencies. The directed edges in the graph represent these relationships and connections between nodes. By analysing the graph, the organisation can understand how different data sources are linked and how changes or updates in one data source may impact others. This understanding aids in capturing a comprehensive view of the organisation's cybersecurity practices.

Assessing Completeness: Directed graphs can be utilised to assess the completeness of data gathering. The organisation can identify any missing or incomplete data sources by comparing the nodes in the graph with the intended scope of the maturity review. This assessment ensures that all relevant information is captured, minimising the risk of overlooking critical aspects of cybersecurity practices.

Visualising Data Flow: Directed graphs can depict the flow of data or information within the organisation's cybersecurity practices. The directed edges in the graph represent the flow of data from one node (data source) to another. This visualisation helps understand how information is processed, shared, or transmitted across different components of the organisation's cybersecurity infrastructure. It enables the organisation to identify potential vulnerabilities or areas where data protection measures must be strengthened.

5.8.3 Assessment

In the assessment stage of a cybersecurity maturity review, the use of directed graphs can assist in several ways:

Visualisation of Maturity Levels: Directed graphs can be leveraged to visually represent the maturity levels defined in the cybersecurity framework. Each node in the graph can be labelled with the corresponding maturity level, ranging from low to high. By assigning maturity levels to nodes, the organisation can easily visualise its current maturity status across different components of its cybersecurity practices. This graphical representation provides an intuitive view of the organisation's maturity levels, enabling easier analysis and assessment.

Comparative Analysis: Directed graphs facilitate a comparative analysis of the organisation's current practices against the desired practices outlined in the cybersecurity framework. By comparing the maturity levels of different nodes, the organisation can identify gaps, discrepancies, or areas where improvement is required. The directed edges in the graph can represent the expected relationships or connections between nodes at different maturity levels, enabling a comprehensive assessment of the organisation's maturity status.

Identifying Areas of Strength and Weakness: The graphical representation of maturity levels in directed graphs allows for identifying areas of strength and weakness in the organisation's cybersecurity practices. Nodes with higher maturity levels indicate areas of strength where the organisation's practices align well with the desired standards. Conversely, nodes with lower maturity levels represent areas of weakness that require improvement. By analysing the distribution of maturity levels across the graph, the organisation can prioritise its efforts and allocate resources effectively to enhance areas of weakness.

Understanding Interdependencies: Directed graphs can depict the interdependencies and relationships between different components of the organisation's cybersecurity practices. The directed edges in the graph represent these relationships, indicating the expected connections between nodes. By analysing the graph, the organisation can gain insights into how the maturity levels of different components impact each other. This understanding helps assess the overall maturity of the organisation's cybersecurity practices and identify areas where improvements in one component may have cascading effects on others.

Assessment Scoring or Rating: Directed graphs can be utilised to assign scores or ratings to the maturity levels of different nodes. The organisation can define a scoring system or rating scale and apply it to assess the maturity levels represented in the graph. This scoring or rating process helps quantify the organisation's maturity and enables the comparison of different components or areas. The

directed graph provides a visual reference for the scoring or rating process, ensuring consistency and accuracy in the assessment.

5.8.4 Gap Analysis

In the gap analysis stage of a cybersecurity maturity review, the use of directed graphs can assist in several ways:

Identifying and Visualising Gaps: Directed graphs can effectively depict the gaps between the organisation's current and desired maturity levels defined in the cybersecurity framework. The gaps become visually apparent by representing the current maturity levels as nodes and the desired maturity levels as reference points or labels in the graph. The directed edges can represent the gaps' extent, or the steps required to bridge those gaps. This visualisation helps stakeholders easily understand and identify the areas where improvements are needed to align with the desired maturity levels.

Prioritising Improvement Efforts: Directed graphs can assist in prioritising improvement efforts by highlighting the gaps that impact the organisation's cybersecurity posture. The graph enables a clear visualisation of the magnitude and significance of each gap. Stakeholders can focus on addressing the gaps that have the most critical implications for the organisation's security and allocate resources accordingly. The directed graph provides a visual representation that aids decision-making and strategic planning for improvement initiatives.

Identifying Root Causes: Directed graphs can help identify the root causes of gaps in the organisation's cybersecurity maturity. By examining the relationships between nodes and the directed edges in the graph, stakeholders can trace the causes and dependencies that contribute to the identified gaps. This analysis enables a deeper understanding of the underlying factors leading to the gaps, allowing the organisation to address the root causes rather than solely focusing on superficial symptoms. Understanding the root causes is crucial for developing effective and sustainable solutions.

Facilitating Remediation Planning: Directed graphs assist in developing a structured plan for closing the identified gaps. Each gap can be associated with specific actions, controls, policies, or procedures needed to bridge the gap and achieve the desired maturity level. By extending the graph with nodes representing the required actions and connecting them with the relevant nodes representing the current maturity levels, stakeholders can develop a roadmap for remediation. The directed graph provides a visual framework that aids in organising and sequencing the remediation efforts, ensuring a systematic approach to addressing the identified gaps.

Tracking Progress: Directed graphs can be utilised to track the progress of improvement initiatives and monitor the closure of gaps over time. The directed graph can be updated accordingly as the organisation implements remediation actions, updates the maturity levels, and closes the gaps. This enables stakeholders to visually track the progress and assess the effectiveness of the implemented measures. The graph serves as a visual representation of the organisation's improvement journey, helping stakeholders monitor the closure of gaps and evaluate the overall progress in enhancing cybersecurity maturity.

5.8.5 Compliance Reporting

Reporting is crucial in communicating an organisation's compliance status to stakeholders, including management, auditors, regulatory bodies, and business partners. Directed graphs can enhance compliance reporting in several ways:

Visual Representation: Directed graphs visually represent compliance status, making it easier for stakeholders to understand complex information. The graph visually depicts the relationships between the organisation's components and the requirements of the cybersecurity framework. By utilising shapes, colours, labels, or other visual cues, the graph can indicate the compliance status of each component, such as compliant, non-compliant, or partially compliant. The visual nature of the graph

simplifies the communication of compliance information, ensuring that stakeholders can quickly grasp the organisation's overall compliance posture.

Comprehensive Overview: Directed graphs offer a comprehensive overview of the organisation's compliance status with the cybersecurity framework. The graph encompasses all relevant components, requirements, and relationships, providing a holistic representation of compliance achievements and gaps. Stakeholders can easily identify which requirements are fully met, which have partial compliance and where non-compliance exists. The graph enables stakeholders to see the bigger picture, enabling better decision-making and prioritisation of resources for compliance improvement.

Drill-Down Capability: Directed graphs allow stakeholders to drill down into specific components or requirements for further analysis. Interacting with the graph allows stakeholders to access more detailed information through nodes and edges. They can investigate specific compliance gaps, review associated documentation, or understand the underlying factors contributing to non-compliance. The ability to drill down into the graph ensures stakeholders can obtain the necessary context and depth of information for effective decision-making.

Trend Analysis: Directed graphs can track compliance trends over time. By creating multiple versions of the graph representing compliance status at different points in time, stakeholders can compare and analyse changes in compliance posture. Trends can be visualised by highlighting differences in the graph, such as changes in the presence or absence of directed edges or alterations in the compliance status of specific components. This trend analysis helps stakeholders understand the organisation's progress, identify improvement areas, and assess the effectiveness of compliance initiatives.

Remediation Roadmap: Directed graphs can serve as a roadmap for compliance remediation efforts. Based on the identified gaps and non-compliant components in the graph, stakeholders can develop a plan of action to address the deficiencies. The graph can be extended with additional nodes representing the required actions, controls, or policies to achieve compliance. The directed edges then

connect the remediation actions to the non-compliant components, illustrating the steps needed for compliance improvement. This roadmap provides stakeholders with a clear and structured approach for remediating compliance gaps, facilitating resource allocation and tracking progress.

Supporting Documentation: Directed graphs in compliance reporting can be accompanied by supporting documentation. The graph can be a visual summary or executive-level overview, while the supporting documents provide detailed explanations, evidence, and justifications for compliance findings. Stakeholders can refer to the directed graph as a navigational tool to access specific documentation related to compliance status, assessment results, remediation plans and evidence of compliance. Combining the directed graph and supporting documentation ensures a comprehensive and well-supported compliance reporting process.

5.9 Graph Schema

The following diagram illustrates the basic graph schema for assessing framework compliance:

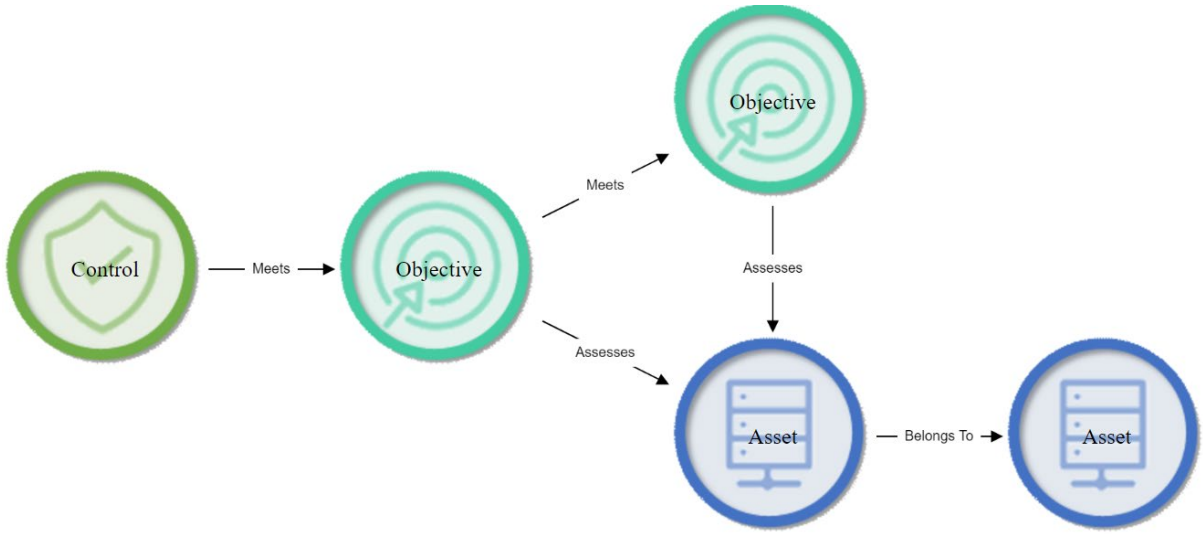

Figure 3 - Graph Schema

5.10 Implementation of Directed Graphs for Assessing Framework Compliance

5.10.1 Asset Nodes

An asset is a valuable resource owned or controlled by an organisation and used in its operations. Assets can be tangible or intangible and protecting these assets is a fundamental objective of cybersecurity.

There are different types of assets that can be relevant in cybersecurity:

Hardware Assets:

Computers and laptops

Servers

Network devices such as routers, switches, and firewalls

Mobile devices like smartphones and tablets

Security cameras and access control systems

Software Assets:

Operating systems

Database management systems

Business applications and tools

Security software such as antivirus programs and encryption tools

Information Assets:

Customer data, such as names, addresses and payment information.

Intellectual property, such as patents, trademarks, and proprietary algorithms

Employee records

Financial information, such as revenues, expenses, and forecasts

Health records in a healthcare setting

Network Assets:

Local Area Networks (LAN)

Wide Area Networks (WAN)

Virtual Private Networks (VPN)

Cloud services and resources

Human Assets:

Employees and their expertise

Teams and departments

Contractors and third-party partners

Reputational Assets:

Brand image

Customer trust

Partnerships and alliances

When considering the context of cybersecurity, assets have a minimum of three attributes: confidentiality, integrity, and availability. This common triad has been augmented with a fourth attribute of accountability.

Asset Node Attributes

Confidentiality (Ac)

Confidentiality refers to the principle that information in an asset should only be accessible to authorised individuals or systems and must be protected against unauthorised access and disclosure.

The following are examples of assets that can be impacted by a breach of confidentiality:

Personal Data	This includes names, addresses, social security numbers, and financial information. Unauthorised access can lead to identity theft and fraud.
Corporate Secrets	Trade secrets, proprietary algorithms, unpublished financial information, and other sensitive business data.
Communication	Emails, text messages, phone calls and other types of private communication.
Authentication Information	Usernames, passwords, PINs, and other credentials.
Health Records	Personal health information, medical histories, diagnoses, etc.
Financial Data	Bank account details, credit card numbers, transaction histories and more.
Legal Documents	Contracts, intellectual property rights, litigation-related documents, etc.
Government Data	Classified information, intelligence reports, military tactics and more.
Research Data	Data from academic, scientific, or corporate research that has not been published.
Digital Identities	Certificates, private keys, and other cryptographic materials.

Configuration Information	System configurations, network topologies, firewall rules, etc., can be exploited if exposed.
Operational Procedures	Information about how a company operates can be valuable for competitors or malicious actors.
Employee Data	Personal and professional data about employees, such as salaries, performance reviews, etc.
Consumer Data	Data collected about consumers, such as buying habits, preferences, and histories.
Software Source Code	Particularly for proprietary software, which can be exploited if exposed.
Educational Records	Grades, transcripts, student personal information, etc.
CCTV Footage	Videos from security cameras that monitor sensitive areas.
Biometric Data	Fingerprints, retinal scans, voice recognition data, etc.

Table 1 - Impacted by a breach of confidentiality.

Integrity (Ai)

Integrity refers to the accuracy and reliability of data and information systems. It ensures that the data is protected from unauthorised modifications and that it remains intact and unaltered from its original state when it is required.

The following are examples of assets that can be impacted by a breach of integrity:

Software and Applications	Ensuring that the software being used or delivered has not been tampered with by malicious actors.
Databases	Guaranteeing that the data stored in databases remains accurate and free from unauthorised alterations.

Communication	Ensuring that the messages sent over a network, like emails or texts, are received as they were sent without any unauthorised changes.
Digital Signatures	They verify the integrity of data. If data integrity is compromised, the digital signature will not match.
Transaction Records	For financial institutions, ensuring that transactions are accurately recorded and not tampered with is crucial.
Logs	System and event logs that track system activities. Tampering can hide unauthorised activities.
Backup Data	Ensuring backups are accurate replicas of the original data.
Web Content	Ensuring that the content displayed on websites has not been altered by unauthorised entities.
Operational & Configuration Data	Ensuring that system configurations, operational procedures and other setup data remain consistent and trustworthy.
Code Repositories	Ensuring that the source code remains unaltered from its original form.
Update & Patch Management	Ensuring that updates/patches to the software are genuine and not injected with malicious code.
Authentication Systems	Ensuring that authentication data (like password hashes) remain unaltered.
Cryptography	Cryptographic hashes are used to verify data integrity. Any alteration in the data will result in a different hash value.
Network Traffic	Monitoring and ensuring that the data packets transmitted over a network are not tampered with during transit.
Audit Trails	Records that show who accessed what data and when. Compromised integrity can mean false trails.

Digital Media	Ensuring that digital media files, such as videos or audio recordings, remain in their original and unaltered form.
Sensor Data	In IoT (Internet of Things) devices, ensuring that data from sensors has not been tampered with before it is processed or acted upon.

Table 2 - Impacted by a breach of Integrity.

Availability (Aa)

Availability refers to the assurance that data, information systems and resources are accessible and usable when needed by authorised users. It means that the information should be available whenever it is required. Availability is crucial for the proper functioning of any organisation. When systems are down, or data is inaccessible, productivity suffers and, in some cases, it can even have severe consequences such as loss of revenue, customer trust, or, in extreme cases, human lives (e.g., in healthcare or critical infrastructure environments).

The following are examples of assets that can be impacted by a breach of availability:

Servers	Ensuring that servers are up and running to provide services and data to users.
Network Infrastructure	Routers, switches, and other networking equipment should be operational to facilitate data transmission.
Databases	Must be available for reading/writing operations whenever required.
Websites	Ensuring that websites are accessible to users and not down.
Cloud Services	Services provided by cloud providers should be available as per the service level agreement (SLA).
Backup Systems	They need to be available, especially during a primary system failure or data corruption.

Authentication Systems	Systems that authenticate users should be up and running to grant access.
Firewalls and Security Devices	Ensuring that they are operational to protect internal networks while allowing legitimate traffic.
Storage Devices	Data storage devices, both primary and backup, should be operational and accessible.
Power Supplies	Ensuring continuous power supply to critical infrastructure.
Communication Channels	Phone lines, email servers, chat systems, etc., should be available for communication.
Remote Access	For businesses with remote workers or multiple sites, ensuring remote access systems are functional.
Data Centres	Their environment (cooling, power) needs to be maintained for the continuous operation of hosted systems.
Software Applications	Crucial software applications, especially those used in businesses, should be running without interruptions.
Disaster Recovery Sites	In case of major disruptions, DR sites should be ready to take over.
Internet Connectivity	Ensuring uninterrupted internet access.
Content Delivery Networks (CDNs)	For faster web content delivery, CDNs should be operational.
Load Balancers	They distribute incoming traffic across multiple servers to ensure no single server is overwhelmed.
Mobile Apps	It should be functional and accessible to users.
IoT Devices	Devices connected to the Internet of Things should remain operational and connected.

Table 3 - Impacted by a breach of Integrity

Accountability (Aac)

While not traditionally part of the classic CIA triad (Confidentiality, Integrity, Availability), it is an important aspect that has been increasingly recognised in cybersecurity. It is sometimes referred to as part of an extended model known as the CIAA quartet. Accountability refers to the principle that actions and activities within an information system should be traceable to a specific entity, such as a user or a process. This means that individuals or systems are held responsible for their actions, especially if these actions have an impact on the security or operation of the system.

The following are examples of assets that can be impacted by a breach of accountability:

User Activity Logs	Records of user activities, which can be audited to determine who did what and when.
Authentication Mechanisms	Systems that ensure only authorised users can access specific resources, tying actions to identities.
Audit Trails	Detailed records that track changes made to a system or data, can be reviewed for compliance and forensic purposes.
Access Control Lists (ACLs)	Lists that define who can access what resources, ensuring only authorised individuals have access.
Digital Signatures	They validate the authenticity and integrity of a message or document and can trace it back to a signer.
Data Ownership	Assigning responsibility for specific data sets to specific entities or departments.
Chain of Custody	Documenting the sequence of custody and control of evidence, ensuring its integrity from collection to presentation in court.
Contractual Agreements	Ensuring vendors, partners and third parties adhere to agreed-upon security standards.

Data Breach Notification Laws	Regulations that mandate the reporting of data breaches to affected individuals and/or regulatory bodies.
Regulatory Compliance	Meeting security standards set by regulations (e.g., GDPR, HIPAA) which often requires accountability mechanisms.
Security Incident and Event Management (SIEM)	Systems that provide real-time analysis of security alerts and can trace incidents to their source.

Table 4 - Impacted by a breach of Accountability.

Asset Node Value (Av)

The value of an Asset Node is calculated based on the importance of its attributes:

$$CapA_v = \left(\frac{A_c + A_i + A_a + A_{ac}}{4}\right)$$

Equation 1 - Asset Node

The calculation for the value of an Asset Node (CapAv) as the average of its attributes is justified as follows:

Holistic Assessment of Importance:

These attributes represent fundamental dimensions of information security and operational importance:

Confidentiality ensures that sensitive data is protected from unauthorised access.

Integrity safeguards the accuracy and reliability of the data.

Availability guarantees that the asset is accessible when needed for organisational processes.

Accountability/Accessibility supports traceability and usability, essential for effective risk management and compliance.

By incorporating all four attributes, the calculation captures a comprehensive view of the asset's criticality in the system.

Uniform Weighting:

Averaging the attributes assigns equal importance to each dimension, reflecting the principle that no single attribute dominates unless explicitly stated by the organisation. This approach ensures a balanced evaluation, which is particularly useful when no prior bias exists regarding the relative importance of these factors.

Simplified Scoring:

The averaging method is straightforward, promoting transparency and consistency in assessing asset value across diverse types. This simplicity supports scalability when applying the model to multiple Asset Nodes within a directed graph structure.

Alignment with Risk Management Principles:

The formula aligns with established risk management frameworks like ISO/IEC 27005, which emphasise a balanced consideration of asset properties when assessing potential risks' impacts.

Foundation for Comparative Analysis:

By normalising the value (using the average), the CapAv metric becomes comparable across assets, enabling the model to identify priority areas for resource allocation, control implementation, and risk mitigation.

5.10.2 Objective Nodes

An objective refers to a specific goal or target that an organisation aims to achieve to improve its cybersecurity posture. These objectives often comprise a broader cybersecurity maturity model that helps organisations measure, assess, and enhance their cybersecurity capabilities over time. The

concept of cybersecurity maturity recognises that cybersecurity is not a one-time effort but an ongoing process that evolves as threats, technologies, and business needs change.

Target Objective Strength is a concept that is often used in risk assessments and cybersecurity maturity models. It refers to the desired level of effectiveness or strength that an organisation aims to achieve for a particular security control or set of controls. It is a benchmark or goal that guides the implementation and improvement of security measures.

Objective Node Attributes

Sum of Maximum Control Strengths	The "Sum of Maximum Control Strengths" refers to the aggregated strength of all the controls at their maximum effectiveness.
Sum of Maximum Control Strengths (Edge Impacted)	The "Sum of Maximum Control Strengths (Edge Impacted" refers to the aggregated strength of all the controls at their maximum effectiveness modified by the strength of the related Edge between the Control Node and Objective Node.
Sum of Actual Control Strengths (Edge Impacted)	The "Sum of Actual Control Strengths (Edge Impacted)" in cybersecurity risk assessment and management refers to the aggregated strength of all implemented security controls in their current state modified by the strength of the related Edge between the Control Node and Objective Node. This measurement reflects the actual effectiveness of the controls as they are configured and deployed within the organisation's environment.
Manually Set	"Manually Set" refers to the process of setting or configuring a parameter, control, value, or setting by human intervention, as opposed to having it automatically set by a system or tool.

	When assessing control strengths or implementing security measures, there may be instances where the values or configurations need to be manually adjusted by a security analyst, system administrator, or another responsible individual.

Table 5 - Objective node attributes

5.10.3 Control Nodes

A control is a measure or mechanism to reduce the risk of a security threat by mitigating vulnerabilities or protecting against potential attacks. Controls help ensure the confidentiality, integrity and availability of information systems and data. They are used to prevent unauthorised access, maintain data accuracy, and ensure the proper functioning of IT systems.

Controls can be categorised into three main types:

Physical Controls:

These measures protect an organisation's assets and data. Examples include:

Security Guards: Employed to monitor and protect the physical premises.

Surveillance Cameras: Used to monitor areas and record activity.

Locks and Access Cards: To control and restrict access to sensitive areas.

Fire Suppression Systems: To protect equipment from fire damage.

Technical (or Logical) Controls:

These are hardware or software measures to protect systems and data from unauthorised access and other cyber threats. Examples include:

Firewalls: Used to block unauthorised access to a network.

Encryption: Used to protect the confidentiality and integrity of data.

Antivirus Software: To protect systems from malware.

Access Control Lists (ACLs): To define who can access a particular system or resource.

Intrusion Detection Systems (IDS): Monitor networks or systems for malicious activity.

Administrative Controls (or Procedural Controls):

These are policies, procedures and regulations implemented by an organisation to manage day-to-day operations and compliance with security policies. Examples include:

Security Policies and Procedures: Documents that define the organisation's stance on security.

Security Awareness Training: Regular training for employees on security best practices.

Incident Response Plan: A plan to follow in case of a security incident.

Regular Audits and Assessments: To ensure compliance with policies and regulations.

Background Checks: Performed before hiring employees with sensitive information access.

In many frameworks, controls are also classified based on their function concerning the risk:

Preventive Controls: Designed to prevent an incident from occurring (e.g., firewalls, training, locks).

Detective Controls: Designed to detect and alert when an incident occurs (e.g., intrusion detection systems, surveillance cameras).

Corrective Controls: Designed to limit an incident's impact or restore systems to normal operation after an incident (e.g., backup and restore procedures, incident response plans).

Compensating Controls: Designed to provide alternative security measures when standard controls are not feasible (e.g., using additional monitoring when a system cannot meet the required patch level due to compatibility issues).

Control Node Attributes

Control Nodes are comprised of two attributes: Control Strength and Control Implementation.

Control Strength (Cs)

Control strength refers to the effectiveness of a security control in mitigating or preventing a specific threat or vulnerability. It measures how well a control can safeguard assets against identified risks. The strength of a control is an important aspect to consider when performing a risk assessment and when planning for risk management.

Design and Implementation: How well the control is designed and implemented will play a significant role in its strength. A poorly designed or improperly implemented control might not be effective in mitigating risks.

The following are examples that can impact a control strength:

Complexity	Overly complex controls can be difficult to manage and may introduce additional vulnerabilities or be prone to misconfiguration.
Technology	The underlying technology upon which a control is based can affect its strength. Outdated technologies may have known vulnerabilities that can be exploited.
Human Element	If a control relies heavily on human intervention or action, it may be more prone to error. For instance, a policy that requires manual review might be less inherently strong than an automated system.
Configurability	Controls that can be finely tuned or configured to a specific environment or use case might have better inherent strength than one-size-fits-all solutions.

Adaptability	How well the control can adapt to changing threats and environments. Controls that are static and cannot be updated or adjusted may lose effectiveness over time.
Coverage	The scope and coverage of the control. A control that only addresses a subset of systems or data might have limited inherent strength compared to a control with broader coverage.
Depth of Defence	Controls that operate at a deeper level within a system (e.g., kernel-level protection) might be more inherently strong than those operating at a surface level.
Maturity	Newly developed or untested controls might not have proven their effectiveness, whereas mature controls with a proven track record might be considered stronger.
Dependencies	If a control's operation depends on other systems or factors, it may be less inherently strong. For example, a network intrusion detection system that relies on timely threat intelligence feeds might be compromised if those feeds are delayed or inaccurate.
Vendor Reputation	The reputation and track record of the vendor providing a security solution can influence the perceived and actual strength of the control.
Vulnerabilities	Any known vulnerabilities in the control itself can weaken its strength.
Lifecycle Management	How well the control can be updated, patched, or replaced can impact its long-term strength.
Feedback Mechanisms	Controls that provide feedback or alerts about their operation or potential breaches can enhance their strength by enabling rapid response.

Table 6 - Impacts on control strength

Implementation (Ci)

Implementation strength refers to how effectively and efficiently a control is deployed, configured, and integrated into an organisation's environment. Unlike control strength, which measures the inherent effectiveness of a control, implementation strength focuses on the practical aspects of how the control is applied in the real world.

The following are examples that can impact a control implementation:

Configuration	How a control is configured can affect its strength. Misconfigurations can render a potentially strong control ineffective or introduce new vulnerabilities.
Integration with Other Systems	If a control is part of a larger system or suite of controls, how well it is integrated can influence its strength. Poor integration can lead to gaps or overlaps in protection.
Maintenance and Updates	Over time, controls may require updates or patches to remain effective. If these are not applied in a timely manner, the control's implemented strength can diminish.
Operational Practices	The procedures and practices surrounding the control's operation can impact its strength. For instance, if alerts generated by a control are routinely ignored, its effectiveness is reduced.
Training and Awareness	The knowledge and awareness of the personnel operating or interacting with the control can influence its strength. Untrained staff might misuse or bypass the control, reducing its effectiveness.

Environmental Factors	The specific environment in which a control is deployed can impact its effectiveness. For example, a control designed for a corporate network might not be as effective in a cloud environment without adjustments.
Monitoring and Response	The mechanisms in place to monitor the control's operation and respond to incidents can affect its strength. Without proper monitoring, breaches might go unnoticed.
Redundancy and Failover	If a control fails, are there backup systems or processes in place to take over? The absence of redundancy can reduce the control's implemented strength.
Physical Environment	Physical factors, such as the location of servers or access controls to data centres, can impact the strength of some controls.
Compatibility	If a control is not fully compatible with the systems it is meant to protect or the infrastructure it is deployed on, its effectiveness can be compromised.
Performance Impact	If implementing a control significantly affects system performance, users might try to bypass or disable it, reducing its strength.
Feedback Loops	How feedback from the control is processed and acted upon can influence its effectiveness. For example, if false positives from an intrusion detection system are not addressed, they can lead to alert fatigue.
External Dependencies	If a control relies on external services or third parties (e.g., threat intelligence feeds or cloud services), the reliability and quality of these can impact the control's strength.

Table 7 - Impact on control implementation

Control Node Value (Cv)

The value of a Control Node is calculated based on the values of its attributes:

$$C_v = \left(\frac{C_s\ C_i}{100}\right)$$

Equation 2 - Control Node

The calculation of the Control Node Value (`Cv`) using the formula can be justified as follows:

Reflecting Control Effectiveness:

Control Strength (`Cs`): This attribute quantifies the control's inherent robustness or efficacy in mitigating threats (e.g., its ability to reduce vulnerabilities or neutralise risks).

Control Implementation (`Ci`): This attribute assesses how well the control is operationalised in the system (e.g., its deployment, configuration, or adherence to policies and practices).

Combining these two attributes, the formula represents a control's practical effectiveness, acknowledging that a robust control is valuable only if well implemented.

Multiplicative Relationship:

The multiplication between `Cs` and `Ci` emphasises the interdependence of these two attributes. A high score in one dimension cannot fully compensate for a low score in the other:

A control with high strength but poor implementation could be more effective.

Similarly, a well-implemented but inherently weak control cannot adequately mitigate risks.

This interdependent relationship ensures that the calculated `Cv` captures the actual operational value of the control.

Normalisation by 100:

Dividing by 100 normalises the score to a meaningful scale, simplifying comparisons between controls:

If `Cs` and `Ci` are both percentages (ranging from 0 to 100), their product will naturally result in values between 0 and 10,000. Dividing by 100 scales this back to a practical range (0–100), which is easier to interpret and integrate into broader models.

Alignment with Risk and Compliance Models:

This formula aligns with cybersecurity frameworks and maturity models, emphasising strength and implementation when evaluating control effectiveness. For instance:

NIST CSF highlights the need for solid, well-implemented controls to achieve protection objectives.

ISO/IEC 27001 requires organisations to assess controls' design and operational effectiveness.

Quantitative Decision-Making:

The calculated `Cv` provides a quantifiable measure of a control's value, enabling:

Prioritisation: Controls with low `Cv` scores can be flagged for review or enhancement.

Comparison: Multiple controls can be evaluated against one another to allocate resources effectively.

Support for Continuous Improvement:

The formula supports dynamic recalculation as controls are improved (e.g., strengthening a control or enhancing its implementation), providing a feedback loop for ongoing optimisation of the security posture.

Compensating Control Value (CCv)

The Compensating Control Value (CCv) results from the highest combination of the Control Value (Cv) and the Edge Strength Value (Ev) of the associated Edge between the Control Node and a target Node.

The Compensating Control Value (CCv) is calculated as follows:

$$CC_v = max(C_v\ E_v)$$

Equation 3 - Compensating Control Value

The calculation of the Compensating Control Value (CCv) using the formula can be justified as follows:

Purpose of Compensating Controls

In cybersecurity risk and maturity models, compensating controls serve as alternative or supplementary measures when primary controls are either insufficient, infeasible, or cost-prohibitive. These controls are not intended to replicate the exact functionality of primary controls but to mitigate residual risk effectively through alternative pathways.

The calculation of CCv ensures that the most impactful risk mitigation pathway is recognised and prioritised, whether it arises from the inherent strength of the control itself (Cv) or the robustness of its influence on a specific relationship (Ev). This aligns with the principle of pragmatic risk management, where emphasis is placed on measurable outcomes rather than rigid adherence to predefined control structures.

Dual Contribution: Control Value (Cv) and Edge Strength (Ev)

The Control Value (Cv) represents the control's inherent strength, effectiveness, and implementation status. This includes attributes like:

Effectiveness: How well the control reduces or mitigates risk.

Coverage: The breadth of the control's impact across assets, vulnerabilities, or threats.

Implementation Status: Degree of deployment and operationalisation of the control.

On the other hand, the Edge Strength Value (Ev) quantifies the quality and significance of the relationship between the Control Node and the target node. This value captures:

Dependency Weight: How strongly the control influences the connected asset, threat, or vulnerability.

Risk Mitigation Impact: The degree to which the edge reduces or alters risk propagation.

Directional Flow Strength: The clarity and effectiveness of the control's interaction with the target node.

By evaluating both Cv and Ev, the model acknowledges that a control's effectiveness is not solely intrinsic but heavily influenced by how it interacts with other nodes within the graph.

Justification for the Maximum Function (max)

Using the maximum function (max) in the calculation reflects a risk-aware prioritisation strategy, where the highest contributing factor (either Cv or Ev) dictates the compensating control's overall value. This approach is justified for the following reasons:

Prioritisation of the Strongest Contribution

In cybersecurity risk management, the strongest available mitigation pathway should be prioritised, whether from the control's inherent attributes (Cv) or its relational influence (Ev). By selecting the maximum value, the calculation ensures that the most significant risk-reducing factor is emphasised without diminishing its contribution.

Context-Driven Flexibility

Specific controls may have lower inherent strength (Cv) in real-world scenarios but disproportionately high influence on specific assets or vulnerabilities due to **strong relational dependencies (Ev). For example:

If it is located on a critical network choke point, a moderately effective firewall control (Cv) might have a high edge strength (Ev).

Conversely, a highly effective encryption control (Cv) may have a weak relationship (Ev) with an isolated system node.

The maximum function dynamically adjusts based on these contextual factors, ensuring neither dimension is undervalued.

Simplified Decision-Making

Using the maximum value simplifies the comparative evaluation of compensating controls across complex graph structures. Decision-makers can quickly identify and prioritise the most effective compensating measures without the cognitive burden of interpreting nuanced weight distributions.

Alignment with Cybersecurity Best Practices

The calculation aligns with established cybersecurity frameworks, including ISO/IEC 27005, NIST CSF, and FAIR, which emphasise:

Effectiveness of Risk Mitigation Measures: Prioritising the most effective available control pathways.

Dynamic Adaptation: Recognising both inherent control strength and relational dependencies.

Operational Practicality: Promoting decision-making strategies that balance analytical robustness with real-world applicability.

The model adheres to these principles by adopting the maximum function, emphasising effectiveness and practicality in identifying compensating control value.

Comparative Analysis Across Controls

The Compensating Control Value (CCv) calculation standardises the evaluation of controls across diverse graph structures. This standardisation allows cybersecurity analysts to:

Compare Controls Fairly: Analysts can rank compensating controls effectively by focusing on the highest contributing factor.

Identify Critical Pathways: Controls with high CCv scores can be flagged as critical mitigation points in the cybersecurity graph.

Optimise Resource Allocation: Investment and monitoring resources can be directed towards controls with the highest compensating potential.

This comparability ensures that compensating controls are not evaluated in isolation but within the broader context of their dependencies and relationships within the cybersecurity ecosystem.

Adaptability to Evolving Cybersecurity Landscapes

Cybersecurity threats are dynamic, and control effectiveness (Cv) or relationship strength (Ev) may fluctuate over time. The maximum function inherently supports adaptability by allowing the dominant factor to shift dynamically based on changes in:

Threat Environment: Increased threat severity may enhance the dependency weight (Ev).

Control Improvement: Upgraded or better-implemented controls may increase their inherent effectiveness (Cv).

This flexibility ensures that the CCv remains context-aware and reflective of real-time security dynamics.

Transparency and Interpretability

The max(Cv, Ev) calculation is mathematically straightforward and transparent. It minimises ambiguity in interpreting results, making it accessible to technical analysts and non-technical stakeholders. This clarity is particularly valuable when cybersecurity assessments are communicated to executive boards, auditors, or regulatory bodies.

The calculation of the Compensating Control Value (CCv) as the maximum value between the Control Value (Cv) and the Edge Strength Value (Ev) is both theoretically robust and effective. It ensures that the most significant contributing factor inherent control strength or relational dependency is appropriately emphasised.

Use of the Maximum Function:

The maximum value between the inherent effectiveness of the control and the strength of the relationship or dependency influenced by the control) ensures that:

It dominates the compensating control value if the control is solid and well-implemented (CCv is high).

If the control is weaker but the relationship it affects has a high inherent edge value (Ev), this relationship compensates for the lack of control strength.

This approach reflects the highest available security contribution to the system.

Realistic Representation of Risk Mitigation:

In many scenarios, a control's effectiveness is influenced by its context or impact on related system elements. By incorporating Ev , the formula accounts for the real-world interdependencies that can enhance or diminish the effectiveness of compensating controls. For example, a control (Cv) may have limited standalone value, but its ability to strengthen a vital edge (Ev) compensates for this weakness.

Alignment with Risk-Based Decision-Making:

By focusing on the maximum contribution, the formula supports risk prioritisation as it ensures that the compensating control value reflects the best available protection or mitigation mechanism, enabling better resource allocation. This approach aligns with frameworks like NIST CSF and ISO/IEC 27001, emphasising adaptive and contextual risk management.

Flexibility in Implementation:

The formula accommodates variations in system design or operational priorities by recognising their importance for highly critical edges (Ev) even if the associated control is weaker. The edge value becomes less critical for strong controls, reflecting the control's inherent robustness.

Promotes Strategic Improvements:

The CCv value provides actionable insights for continuous improvement as a low CCv value highlights areas where neither the control nor the edge is sufficiently strong, guiding enhancements in both areas.

By tracking changes to CCV or Ev, organisations can evaluate the impact of security investments over time.

Support for Graph-Based Analysis:

This formula considers intrinsic control value and contextual influence within a graph-based methodology, where nodes and edges represent system elements and their relationships. This dual focus enhances the model's accuracy in assessing compensating controls.

5.10.4 Node Relationships

Objective and Asset Nodes

Objective and asset nodes are used to model the relationship between security objectives and assets within an organisation's infrastructure in a structured and visual manner. Relationships between objective nodes and asset nodes can be represented in a directed graph:

Asset Nodes: In the directed graph, asset nodes represent the various assets of the organisation that need protection. These can include hardware (servers, routers), software (applications, databases), data (customer information, intellectual property) and other valuable resources.

Objective Nodes: The objective nodes represent the security objectives associated with the assets. These objectives typically include but are not limited to confidentiality, integrity, availability, and accountability.

Directed Edge: An edge from an objective node to an asset node can indicate that the asset has a specific security objective. Edge strengths indicate the degree of relevance or importance of a particular security objective for an asset.

Multiple Associations: A single control node may be associated with multiple objective nodes and vice versa. For example, a firewall might be associated with both the integrity and availability objectives, as it can help protect against unauthorised changes (integrity) and help ensure that services remain available (availability).

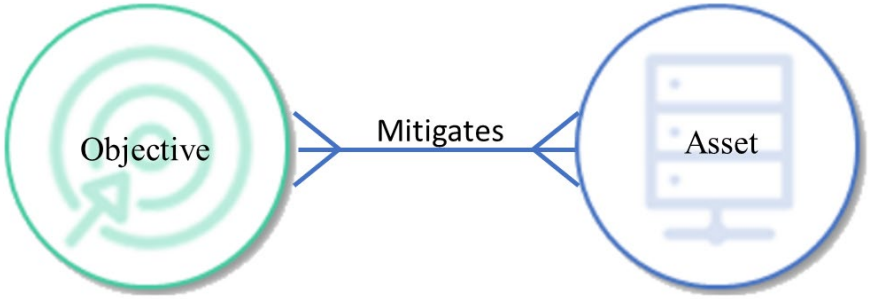

Figure 4 - Objective to asset relationship

Control and Objective Nodes

Control nodes and objective nodes are used to model the relationships between security controls and the objectives they aim to achieve. This helps in understanding and visualising how various security measures contribute to the security objectives of an organisation.

Control nodes and objective nodes are used to model the relationships between security controls and the objectives they aim to achieve. This helps in understanding and visualising how various security measures contribute to the security objectives of an organisation.

Control Nodes: Control nodes in the directed graph represent specific security controls that an organisation plans to implement. These controls can be technical (e.g., firewalls, encryption), administrative (e.g., policies, training), or physical (e.g., locks, security cameras) in nature.

Objective Nodes: Objective nodes represent the security objectives that the organisation aims to achieve.

Directed Edge: Edges in the directed graph connect control nodes to objective nodes, indicating which controls are contributing to which objectives. An edge from a control node to an accurate node indicates that the control is meant to support or enforce the particular security objective. Edge strengths indicate the degree of relevance or importance of a specific objective of security for an asset.

Multiple Associations: A single control node may be associated with multiple objective nodes and vice versa. For example, a firewall might be associated with both the integrity and availability objectives, as it can help protect against unauthorised changes (integrity) and help ensure that services remain available (availability).

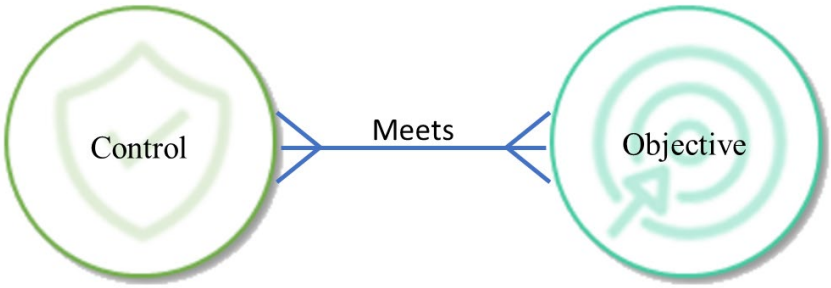

Figure 5 - Control to objective relationship

Objective and Objective Nodes

Objective nodes can have relationships with other objective nodes. This means that the achievement of one security objective might be dependent on or contribute to the achievement of another security objective. Directed edges between objective nodes in the graph can represent these relationships.

Interdependent Objectives: Some security objectives can be interdependent, meaning that achieving one objective could either positively or negatively affect the achievement of another. For example, in certain cases, maximising confidentiality through strong encryption could impact data availability by introducing latency.

Supporting Objectives: One objective directly supports the achievement of another. For example, proper authentication (an objective in itself) might be necessary to ensure data integrity by ensuring only authorised users can make changes.

Compensating Objectives: Sometimes, when one objective cannot be fully achieved due to various constraints, another objective can be emphasised to compensate. For example, if data cannot be kept

as confidential as desired due to regulatory requirements for sharing, an emphasis on the integrity and authenticity of the data can compensate by ensuring that the data remains trustworthy.

Directed Edge: An edge from one objective node to another can signify a dependency or relationship where the first objective supports or impacts the second. Edge strengths indicate the degree of relevance or importance of a particular security objective for an asset.

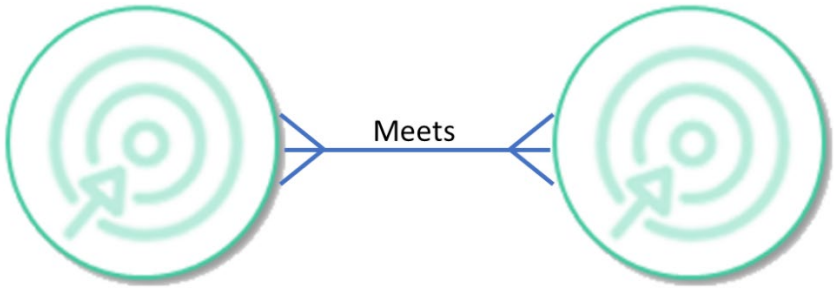

Figure 6 - Objective to objective relationship

Worked Example 1 – Single Control, Single Objective

The following simple example demonstrates how Objective Compliance is calculated:

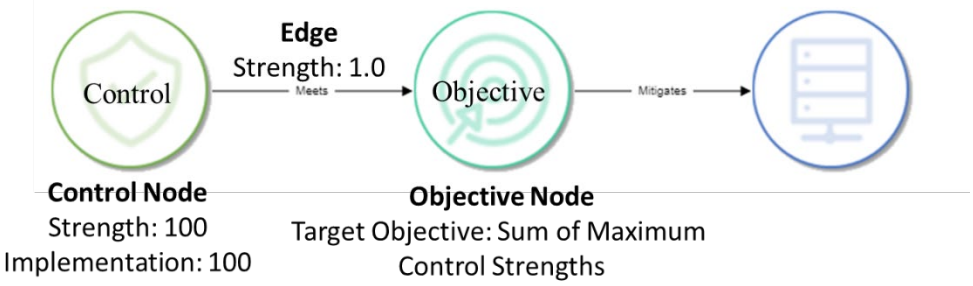

Figure 7 - Worked Example 1 – Single Control, Single Objective

The Target Objective Strength (Os) is calculated as follows when using the “Sum of Maximum Control Strengths”:

$$O_S = \sum_{i=1}^{n} 100$$

$$O_S = 100$$

The Control Value (Cv) is calculated as follows:

$$C_v = \left(\frac{C_s\, C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{100 \times 100}{100}\right)$$

Result:

$$C_v = 100$$

Objective Compliance (Oc) is calculated as

$$O_c = \left(\frac{\sum_{i=1}^{n} C_v E_v}{O_s}\right) \times 100$$

Substitution:

$$O_c = \left(\frac{(100 \times 1.0)}{100}\right) \times 100$$

Result:

$$O_c = 100$$

Equation 4 - Worked Example 1 – Single Control, Single Objective

Worked Example 2 – Single Control, Single Objective

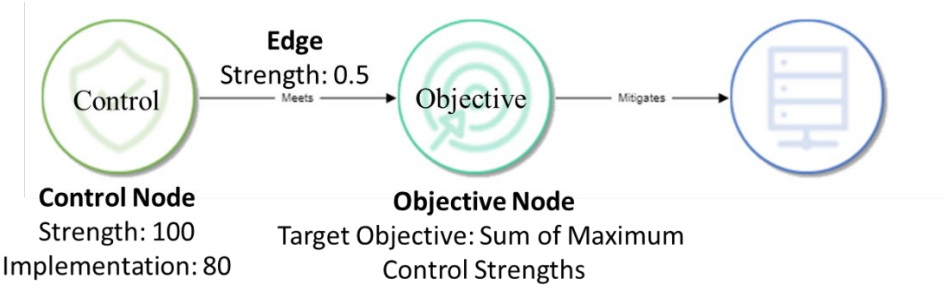

Figure 8 - Worked Example 2 – Single Control, Single Objective

The Target Objective Strength (Os) is calculated as follows when using the "Sum of Maximum Control Strengths":

$$O_S = \sum_{i=1}^{n} 100$$

$$O_S = 100$$

The Control Value (Cv) is calculated as follows:

$$C_v = \left(\frac{C_s\ C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{100 \times 80}{100}\right)$$

Result:

$$C_v = 80$$

Objective Compliance (Oc) is therefore calculated as

$$O_c = \left(\sum_{i=1}^{n} \left(\frac{c_v E_v}{O_s}\right) i\right) \times 100$$

Substitution:

$$O_c = \left(\frac{80 \times 0.5}{100}\right) \times 100$$

Result:

$$O_c = 40$$

Equation 5 - Worked Example 2 – Single Control, Single Objective

Worked Example 3 – Multiple Control, Single Objective

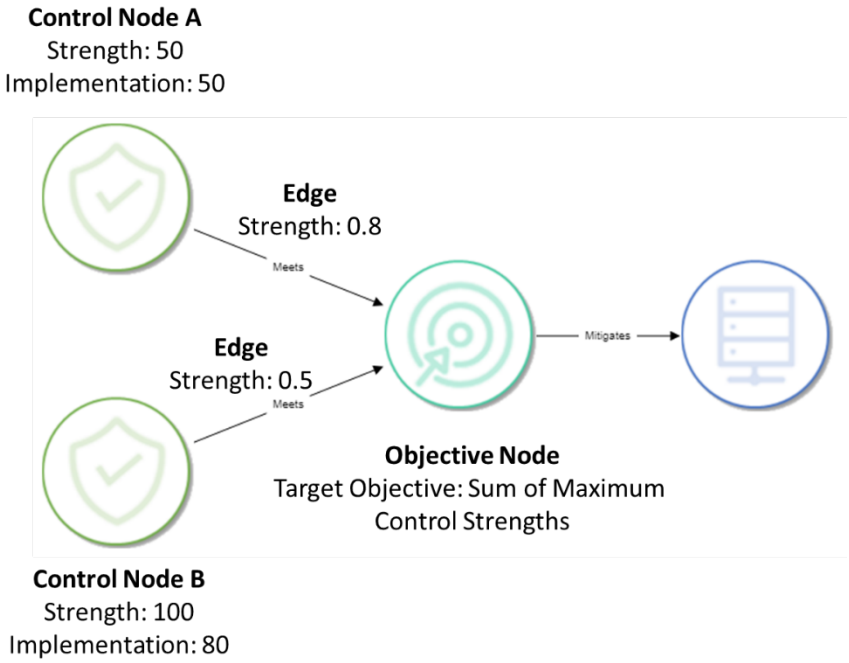

Figure 9 - Worked Example 3 – Multiple Control, Single Objective

The Target Objective Strength (Os) is calculated as follows when using the “Sum of Maximum Control Strengths”:

$$O_S = \sum_{i=1}^{n} 100$$

Substitution:

$$O_S = 100 + 100$$

Result:

$$O_S = 200$$

Control Node A, Control Value (Cv) is calculated as follows:

$$C_v = \left(\frac{C_s\ C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{50 \times 50}{100}\right)$$

Result:

$$C_v = 25$$

Control Node B, Control Value (Cv) is calculated as follows:

$$C_v = \left(\frac{C_s\ C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{100 \times 80}{100}\right)$$

Result:

$$C_v = 80$$

Objective Compliance (Oc) is calculated as

$$O_c = \left(\frac{\sum_{i=1}^{n} C_v E_v}{O_s}\right) \times 100$$

Substitution:

$$O_c = \left(\frac{(25 \times 0.8) + (80 \times 0.5)}{200}\right) \times 100$$

Results:

$$O_c = 30$$

Equation 6 - Worked Example 3 – Multiple Control, Single Objective

Chapter 6 - Using Directed Graphs for Assessing Information Security Risk

Directed graphs serve as potent analytical tools for assessing cybersecurity risk, lending themselves to many applications within this domain. Their capability to model and interpret complex, interrelated systems provides an effective avenue for understanding potential attack vectors, impact scenarios, vulnerability distributions and mitigation strategies in the cybersecurity landscape.

In cybersecurity analysis, directed graphs comprehensively represent an organisation's digital infrastructure. Systems, networks, and assets can be encapsulated as distinct nodes, while the interactions, data flow, or dependencies between them can be portrayed as directed edges. This comprehensive graphical representation of the organisational infrastructure can elucidate potential attack paths, providing an avenue for proactive threat identification and management.

Regarding impact assessment, directed graphs offer a unique lens to envision the potential repercussions of a security incident. By interpreting the directed edges as channels of potential harm, organisations can foresee the possible cascading effects of a cyber-attack. This ability to predict sequential or domino effects is invaluable in understanding the full extent of potential damage, thereby aiding in devising more effective incident response plans.

Directed graphs also serve as an ideal framework for vulnerability mapping within an organisational infrastructure. Each node, representing a system, network, or asset, can be linked with its corresponding vulnerabilities, providing a clear picture of weak points within the infrastructure. This mapping enables targeted vulnerability management and helps prioritise patching or fortification efforts.

Directed graphs play a critical role in developing risk mitigation strategies. They enable the identification of critical nodes and high-risk pathways within the graph, facilitating the formulation of

targeted measures to bolster security controls. By fortifying these critical nodes and pathways, organisations can effectively minimise the potential impact of cyber threats.

Finally, the utility of directed graphs extends to visualisation and communication. They offer a visually intuitive portrayal of cybersecurity risk, simplifying the task of conveying intricate risk scenarios to various stakeholders. The graphical nature of the graphs not only illuminates the interdependencies and flows of information within the system but also underscores potential vulnerabilities, thereby enhancing overall understanding and decision-making in the face of cybersecurity threats.

Information security and risk analysis consider elements essential components that make up an entire structure or process; together, they compose the full risk evaluation framework.

Risk evaluation in information security includes several elements. They could include:

Asset identification and threat analysis: As well as vulnerability assessment to determine any weaknesses within systems.

Impact Analysis: Assessing the potential consequences of each threat. Likelihood Assessment: Evaluating the likelihood that an attacker could exploit vulnerabilities. Risk Rating: Determining an acceptable level of risk based on potential impact and likelihood.

Controls Evaluation: Examining the efficacy of existing security measures.

Regulators and Compliance Requirements: Examining legal and industry standards as requirements.

Each element plays an integral part in the risk assessment process, helping identify, assess, and mitigate potential security threats.

6.1 Use of Directed Graphs for Assessing Information Security Risk

Directed graphs are a useful tool for evaluating information security risks within an organisation. Their inherent ability to represent complex relationships and interactions renders them essential in understanding and representing an organisation's security risk landscape.

In the domain of security risk assessment, every identified risk can be symbolised as a unique node within the directed graph. In parallel, the potential causal factors, consequences, or correlations among these risks can be depicted as directed edges connecting these nodes. This structured representation facilitates a systematic approach to understanding the intricate web of risks and their interdependencies.

Mapping Security Threats and Vulnerabilities: When detailing an organisation's various threats and vulnerabilities, each element can be represented as individual nodes within the graph. This visual representation offers a consolidated view of all security issues, enabling stakeholders to understand better and address potential security breaches.

Understanding Consequence Paths: Directed graphs illuminate the paths that threats might exploit to cause damage. By representing potential attack vectors or sequences as directed edges, the graph clarifies how a threat can progress, affecting multiple nodes (risks) and eventually leading to a significant security incident.

Risk Prioritisation: Analysing the directed graph aids in prioritising risks based on their interconnectedness and potential impact. Risks with more incoming or outgoing edges might be considered more critical, as they influence or are influenced by many other risks. This aids in determining which threats require immediate attention and which can be addressed in subsequent phases.

Risk Mitigation Strategy Visualisation: Directed graphs can be further enhanced by integrating potential mitigation strategies as nodes. The directed edges in such cases would represent the efficacy

of a particular strategy in mitigating or reducing a specific risk. This makes decision-making more data-driven, ensuring resources are allocated to the most effective countermeasures.

Stakeholder Communication: The visual representation offered by directed graphs facilitates transparent communication with stakeholders. Presenting a graph allows for a more intuitive understanding of the risk landscape, helping stakeholders, even without deep technical knowledge, to grasp the nuances of the organisation's security challenges.

Dynamic Risk Assessment: One of the most significant advantages of using directed graphs is their dynamic nature. As new risks emerge or existing risks evolve, they can be seamlessly incorporated into the graph. This ensures the risk assessment remains relevant and up to date, adapting to the ever-changing cybersecurity environment.

6.1.1 Asset Identification

Asset Identification is the first step in the risk assessment process. In information security, an asset is any data, device, or other component of the environment that supports information-related activities. Assets should be adequately protected to ensure business continuity legal compliance and to prevent damage to an organisation's reputation.

Examples of Assets include:

Hardware: This includes servers, workstations, laptops, mobile devices, routers, switches, firewalls, and any other physical device that is part of your IT infrastructure.

Software: This includes operating systems, databases, applications, and any other software that is used in your organisation. It is also important to understand the dependencies between different pieces of software.

Data: This includes customer data, employee data, intellectual property, and any other type of data that your organisation collects, processes, or stores. Different types of data may have different levels of sensitivity and thus require different levels of protection.

Services: Services provided by the organisation or to the organisation, such as cloud services, email services, network services, etc.

People: People are also considered as assets. This includes employees, contractors, customers, and anyone else who might have access to your organisation's data or systems.

Processes: Business and IT processes that use, transmit, or store data should also be identified as assets.

Physical locations: This includes data centres, office buildings, employee home office.

6.1.2 Threat Analysis

Threat analysis, also known as threat modelling or threat assessment, is the process of identifying, documenting, and understanding threats to a system. It is a critical part of the risk assessment process in information security. Here is an expanded explanation of the main components:

Threat Identification: This is the process of identifying all potential threats to your assets. A threat can be defined as anything that has the potential to cause serious harm to a system. They can be broadly categorised into:

Natural Threats: These include natural disasters such as floods, earthquakes, fires, etc., that could disrupt or damage your systems.

Human Threats: These can be intentional (like hackers or insiders intentionally trying to cause harm) or unintentional (like an employee accidentally deleting important files).

Environmental Threats: These include things like power failures, chemical spills, or other incidents that could disrupt your systems.

Threat Categorisation: Once threats have been identified, they should be categorised. This could be based on the type of threat (e.g., natural, human, environmental), the potential impact of the threat, or the level of sophistication of the threat.

Threat Modelling: This involves creating scenarios to understand how each threat could potentially impact your organisation. For example, what would happen if a hacker were able to exploit a vulnerability in your system? Or what would happen if a key server were to fail? Threat modelling helps you understand the potential paths that an attacker could take, which can inform your defences.

Threat Evaluation: This involves assessing each threat to determine its potential impact and the likelihood of it occurring. This information can be used to prioritise threats and determine where to focus your security efforts.

Threat Intelligence: This involves staying up to date on the latest threats and threat actors. This can include subscribing to threat intelligence feeds, participating in industry forums, or working with a security consultant. Threat intelligence can provide valuable information about the tactics, techniques, and procedures (TTPs) that threat actors use, which can inform your defences.

6.1.3 Vulnerability Assessment

Vulnerability Assessment is a critical step in the risk assessment process. It involves identifying, quantifying, and prioritising (or ranking) the vulnerabilities in a system. Here is an expanded explanation of its main components:

Vulnerability Identification: This involves identifying the weaknesses in your system that could be exploited by a threat. Vulnerabilities can exist in many forms, such as software bugs, misconfigurations, weak passwords, lack of data encryption, outdated software, or lack of proper

access controls. Tools like vulnerability scanners can be used to automate the process of finding known vulnerabilities.

Vulnerability Analysis: Once vulnerabilities have been identified, the next step is to analyse them to understand the potential impact if they were to be exploited. This involves understanding what data or system functionality could be affected and the potential consequences of a breach.

Vulnerability Prioritisation: Not all vulnerabilities present the same level of risk. Some may pose a significant risk to your organisation, while others may be less critical. Prioritisation involves ranking vulnerabilities based on factors such as the potential impact of a breach, the ease of exploitation and the value of the affected asset. This helps in determining which vulnerabilities to address first.

Remediation Planning: For each identified and prioritised vulnerability, a plan should be made to mitigate it. This might involve patching software, changing configurations, improving access controls, or other actions. In some cases, if the risk is low and the cost of remediation is high, an organisation might choose to accept the risk rather than mitigate it.

Continuous Monitoring and Reassessment: Vulnerability assessment is not a one-time activity but a continuous process. New vulnerabilities can be introduced over time and old vulnerabilities can become more severe as new exploitation techniques are developed. Regular scanning, monitoring and reassessment are necessary to stay ahead of potential threats.

6.1.4 Impact Analysis

Impact Analysis, in the context of information security risk assessment, is the process of determining the potential consequences or the business impact that could occur if the identified threats exploit the vulnerabilities in the system. It is a crucial step that helps organisations understand the severity of the risk they could face.

Key components of Impact Analysis include:

Data Loss or Corruption: This involves assessing the impact of losing or corrupting critical data. Depending on the type of data, the impact could range from minimal to catastrophic. For instance, loss of sensitive customer data could lead to regulatory fines, lawsuits, and damage to the company's reputation.

Service Disruption: This includes assessing the impact of disruption to your services or operations. If a threat were to bring down your website or disrupt your network, for instance, it could lead to a loss of business and customer trust.

Financial Impact: This involves quantifying the potential financial loss that could result from a security breach. This could include direct costs (such as regulatory fines or the cost of remediation) and indirect costs (such as loss of business or damage to your reputation).

Regulatory and Legal Impact: If a breach results in the loss of sensitive data, your organisation could face regulatory penalties, lawsuits, or other legal consequences.

Reputational Impact: A data breach can damage your organisation's reputation, leading to loss of customers, partners, or investors. This is often difficult to quantify, but it is a crucial factor to consider.

Operational Impact: The impact on the daily operations of the business is also an essential factor. If systems are compromised, the time taken to recover and the resources required for the same are part of the operational impact.

6.1.5 Likelihood Assessment

Likelihood analysis in information security risk evaluation involves estimating the probability that an attacker exploits a specific vulnerability. It is an essential component to understanding risk levels and prioritising mitigation efforts.

Some key components of a likelihood assessment:

Threat Capability: This refers to the ability of threat actors (for instance, hackers) to exploit vulnerabilities successfully. For instance, more experienced hackers could exploit complex vulnerabilities more successfully than less capable hackers could.

Threat Motivations: Threat actors require some sort of motivation to exploit vulnerabilities - this could include financial gain, disruption goals or ideological reasons, among others.

Existence of Vulnerabilities: Vulnerabilities exploitable by threat actors are an integral component of risk. A system could contain numerous vulnerabilities; however, if protective mechanisms are put into place that deter threat actors from exploiting those vulnerabilities, then their likelihood may decrease significantly.

Effectiveness of Current Controls: The existence and effectiveness of current security controls can have a profound impact on the likelihood of threats to our systems, such as strong firewalls and up-to-date antivirus software, which can drastically lower the odds of successful cyber-attacks.

External Factors: External factors, including the general security environment, value of information being protected and frequency of similar attacks targeting similar targets may all influence its likelihood.

Once the likelihood is estimated, it can be combined with the impact assessment to calculate the overall risk. Typically, organisations will use a scale (like low, medium, or high) to rate likelihood. For example, a threat with a high likelihood and high impact would be considered a high risk that needs immediate attention.

6.1.6 Risk Rating

Risk Rating, in the context of information security risk assessment, involves assigning a level of risk to each identified threat-vulnerability pair. It is a critical step in prioritising mitigation efforts and determining where to allocate resources.

The risk rating is typically calculated based on the results of the impact analysis and the likelihood assessment. The higher the potential impact and the higher the likelihood, the higher the risk rating. Common considerations are:

Risk Calculation: Risk is usually calculated as a function of likelihood and impact. This can be as simple as multiplying the likelihood score by the impact score, or it can involve more complex formulas.

Risk Scale: The risk rating is typically assigned based on a defined scale. For instance, you might use a scale of 1-5, where 1 represents a low risk and 5 represents a high risk. Or you might use a color-coded scale, where green represents low risk, yellow represents medium risk and red represents high risk.

Risk Categorisation: Depending on the risk rating, risks can be categorised as low, medium, or high. High risks are those that could have a significant impact on the organisation and have a high likelihood of occurrence. These should be addressed as a priority.

Risk Tolerance: The organisation's risk tolerance or risk appetite (how much risk the organisation is willing to accept) should be considered when determining the risk rating. For instance, an organisation with a low-risk tolerance might rate a given risk as high, while an organisation with a high-risk tolerance might rate the same risk as a medium.

6.1.7 Controls Assessment

Control analysis is an essential element of information security risk evaluation. This step involves evaluating the effectiveness of current security controls to manage identified risks. Controls can include policies, procedures, hardware/software solutions or actions designed to manage them.

Control Identification: This step involves identifying all current controls that help mitigate identified risks and mitigate risks through:

Preventive Controls: Intended to stop an incident from taking place. Examples include firewalls, user training courses and strong access controls.

Detective Controls: Used to detect and alert of an event once it takes place - for example intrusion detection systems (IDSs) and regular audits.

Corrective Controls: Corrective controls aim at mitigating incidents during or after their occurrence to minimise their impact, such as disaster recovery plans and backup systems.

Control Analysis: This step involves evaluating each control's ability to reduce risks. Aspects to consider include its capacity to prevent threats from exploiting vulnerabilities or reduce the impact should a breach take place.

Control Gaps Identification: When existing controls have proven insufficient or ineffective, any "control gaps" identified indicate areas in which additional or improved measures might be necessary to adequately manage risk.

Recommendations for Improvement: Based on an assessment, recommendations are made to enhance the control environment by either installing new controls, upgrading existing ones or changing how certain ones are used. This could involve anything from new controls being introduced or enhanced to shifting how certain ones are employed or changing their use altogether.

6.2 Graph Schema

The following diagram illustrates the basic graph schema for assessing information security risk:

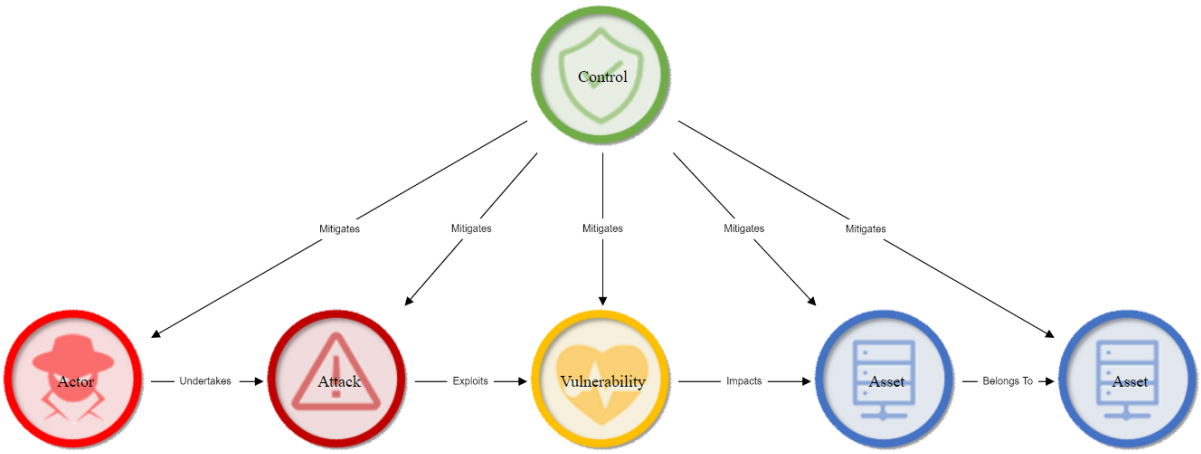

Figure 10 - Graph schema

6.3 Implementation of Directed Graphs for Modelling Information Security Risk

6.3.1 Threat Actor Node

A Threat Actor (TA) is a person or entity that has the ability or intent to undertake some form of Attack (AT) against an Asset (A).

Threat Actor Attributes

The Threat Actor Value (TAv) is used as the basis for the Threat Actor calculations. The Threat Actor Value (TAv) is calculated from four attributes, Access, Capability, Resources and Motivation.

The following describes each of these attributes:

Access (TAa)

There are different types of access that a threat actor can have once they gain unauthorised entry into a system or network, for example:

User-level access	This type of access allows the attacker to access resources and perform actions that are permitted for a regular user on the system or network. This may include reading or modifying files, accessing applications, or running commands.

Administrative access	This type of access provides the Threat Actor with full control over the system or network. It allows them to perform all actions that an administrator or superuser can do, such as creating new accounts, installing software, changing system configurations, and accessing sensitive data.
Network-level access	This type of access allows the Threat Actor to access other systems or devices on the network, which can potentially lead to a wider attack surface and more damage.
Remote access	This type of access allows the Threat Actor to control the system or network from a remote location, using tools like remote desktop software or backdoor malware.
Persistent access	This type of access enables the Threat Actor to maintain their presence on the system or network even after they have been detected or removed. They may use techniques like hiding their presence, creating backdoors, or establishing persistence mechanisms to maintain access for an extended period.
Physical access	This gives the Threat Actor physical access to a computer system or device, usually by gaining entry to a building or a restricted area where the system is located. Physical access can be particularly dangerous because it allows the attacker to bypass many of the security measures that are in place to protect the system, such as firewalls, intrusion detection systems and authentication controls.

Table 8 - Threat actor access

Capability (TAc)

Capability from a Threat Actor perspective refers to the skills, knowledge, resources, and tools that a Threat Actor has at their disposal to carry out an attack. Threat actors may have different levels of capability, ranging from low-skilled script kiddies who use off-the-shelf hacking tools to sophisticated state-sponsored hackers with advanced technical skills and extensive resources. Their capability may depend on various factors, such as their motivation, funding, access to technology and level of sophistication.

There are several types of common Threat Actors; here are some examples:

Cybercriminals	These are individuals or groups who use illegal means to make money through cybercrime. Cybercriminals may engage in activities such as hacking, identity theft, fraud, and ransomware attacks.
Hacktivists	These are individuals or groups who use hacking to advance political or social causes. Hacktivists may deface websites, leak confidential information, or disrupt online services to draw attention to their causes.
Nation-state actors	These are government-sponsored groups that use cyberattacks to achieve political, economic, or military objectives. Nation-state actors may conduct espionage, sabotage, or cyberwarfare against other countries or organisations.
Insiders	These are individuals who have authorised access to a system or network but use that access for malicious purposes. Insiders may steal sensitive information, plant malware, or cause other types of damage to the system.
Script kiddies	These are low-skilled hackers who use off-the-shelf hacking tools to carry out attacks without much knowledge or experience. Script kiddies may engage in activities such as website defacement or DDoS attacks.

Advanced persistent threats (APTs)	These are sophisticated, well-funded groups that use advanced techniques to carry out targeted attacks over an extended period. APTs may use social engineering, zero-day exploits, or custom malware to infiltrate a system and steal sensitive information.

Table 9 - Threat actor types

Resources (TAr)

Threat Actor resources refer to the assets or capabilities that a Threat Actor has at their disposal to carry out an attack. These resources can include various tools, techniques, and funding that the attacker can use to achieve their objectives.

Examples of resources that a Threat Actor may have include:

Technical expertise	Advanced technical knowledge of computer systems, networking, programming languages and malware development can be a significant resource for a threat actor.
Exploits	Threat actors may have access to zero-day vulnerabilities, malware and other exploits that can be used to bypass security controls and gain access to systems or networks.
Botnets	Botnets are networks of compromised computers that are controlled by a single attacker. Botnets can be used for DDoS attacks, spam campaigns, or other malicious activities.
Funding	Some threat actors, such as nation-state actors or cybercriminal organisations, may have significant funding at their disposal to acquire the resources they need to carry out sophisticated attacks.
Human resources	A threat actor may have a team of skilled hackers or other personnel who can carry out various aspects of the attack.

Social engineering tactics	Threat actors may use social engineering tactics such as phishing, pretexting, or baiting to trick users into divulging sensitive information or granting access to systems or networks.

Table 10 - Threat actor resources

Motivation (TAm)

Threat Actor motivation refers to the reasons or goals that drive a Threat Actor to carry out a cyber-attack. Understanding the motivations of a Threat Actor can help organisations and individuals develop effective cybersecurity strategies and defences.

Motivations of a threat actor can vary widely, and some common motivations include:

Financial gain	Cybercriminals may carry out attacks to make money by stealing sensitive information, conducting fraud, or launching ransomware attacks.
Espionage	Nation-state actors or other entities may carry cyberattacks to obtain sensitive information, trade secrets, or intellectual property from other countries or organisations.
Ideology	Hacktivists may launch attacks to advance a political or social cause, such as activism, free speech, or government transparency.
Sabotage	Some attackers may carry out attacks to cause disruption, damage, or harm to organisations or individuals.
Personal gain or revenge	Insiders or disgruntled employees may launch attacks to seek personal gain or revenge against their employers or colleagues.
Thrill-seeking	Some attackers may carry out attacks for the thrill or challenge of breaking into systems or networks.

Table 11 - Threat actor motivations

Theat Actor Value (TAv)

The value (TAv) of a Threat Node is calculated based on the values of its attributes:

Access (TAa): How easy is the Threat Actor to gain and maintain access to the target? A higher value from 0 to 100 indicates easier access.

Capability (Tac): How capable is the Threat Actor in undertaking the attack? Greater capabilities are indicated in a higher value of 0 to 100.

Resources (TAr): What level of resources does the Threat Actor have to bring to bear in the attack? Higher levels of resources are indicated by a higher value in the range of 0 to 100.

Motivation (TAm): How motivated is the Threat Actor in achieving the attack? Higher levels of motivation are indicated by a higher value in the range of 0 to 100.

The Threat Actor Value calculation is:

$$TA_v = \left(\frac{TA_a + TA_c + TA_r + TA_m}{4}\right)$$

Equation 7 - Threat Actor Value

The calculation of the Threat Actor Value (TAv) using the formula can be justified as follows:

Holistic Representation of Threat Actor Potential:

The formula evaluates the Threat Actor's potential by considering four critical attributes:

Access (TAa): Reflects how easily the Threat Actor can penetrate and sustain access to the target. Easier access indicates a higher likelihood of successful exploitation.

Capability (TAc): Measures the Threat Actor's skills, expertise, and ability to execute the attack effectively.

Resources (TAr): Represents the availability of material, financial, or technological resources that support the attack.

Motivation (TAm): Reflects the Threat Actor's determination or drive to achieve their objective.

Combining these dimensions, (TAv) provides a comprehensive measure of the Threat Actor's overall threat level.

Equal Weighting for Objectivity:

Averaging the four attributes assumes that each attribute contributes equally to the Threat Actor's value. This is a reasonable assumption in the absence of evidence suggesting the dominance of any one factor. For example, a highly motivated attacker with low capability or resources is as dangerous as a well-equipped attacker with moderate motivation.

Normalisation and Comparability:

Dividing by 4 normalises the value to the same scale as the inputs (0–100), ensuring consistency across assessments. This allows direct comparisons of Threat Actor Values across different nodes or scenarios within the graph-based model.

Alignment with Risk-Based Decision-Making:

Threat Actor Value (TAv) directly feeds into risk assessments by quantifying the threat level of different actors. Higher values indicate a more significant potential to cause harm.

The formula aligns with frameworks like FAIR and NIST CSF, which prioritise understanding threat sources as part of risk evaluation.

Simplicity and Scalability:

The calculation is straightforward and can be easily applied across multiple Threat Nodes in the graph, facilitating scalability within complex systems. Simplicity in the calculation enhances transparency, aiding stakeholders in understanding the results.

Practical Insights for Mitigation:

The individual components provide granular insights into the threat actor's specific strengths or vulnerabilities; for example, tightening access controls may mitigate a Threat Actor with high motivation (TAm) but low access (TAa).

Dynamic Recalculation for Evolving Threats:

As the threat landscape evolves (e.g., changes in resources or motivation), (TAv) can be recalculated, ensuring the graph remains up-to-date and reflects current risks.

Support for Graph-Based Models:

In graph-based methodologies, nodes and edges represent entities and their relationships. The (TAv) value ensures that Threat Nodes are quantitatively integrated into the analysis, enabling nuanced risk assessments.

Threat Actor Mitigated Value (TAmv)

The Threat Actor Mitigated Value (TAmv) results from the Threat Actor Value (TAv) of the Node reduced by the highest Compensating Control Value (CCv) to this Node. The Threat Actor Mitigated Value (TAmv) is used as the basis for further graph calculations.

The Threat Actor Mitigated Value (TAmv) is calculated as follows:

$$TA_{mv} = TA_v - \max\left(CC_v\right)$$

Equation 8 - Threat Actor Mitigated Value

The calculation of the Threat Actor Mitigated Value (TAmv) using the formula can be justified as follows:

Representation of Residual Threat:

After applying the most effective compensating control (maxCCv), the formula quantifies the residual threat level. This directly reflects the Threat Actor's remaining risk, forming the basis for further analysis in the graph model.

Focus on Maximum Mitigation:

By subtracting (maxCCv), the formula ensures that the most robust available compensating control, representing the best-case mitigation scenario, is considered. This prioritisation emphasises using the most effective control to reduce the threat actor's impact.

Dynamic Risk Management:

This approach provides a dynamic evaluation of residual risks, allowing recalculation as new compensating controls are introduced or existing ones are improved. It ensures that the threat mitigation process evolves with changes in the control environment.

Alignment with Risk Management Frameworks:

TAmv aligns with established principles in frameworks like ISO/IEC 27005 and NIST CSF, which emphasise identifying and addressing residual risks after mitigation. This ensures that no threats are overlooked, even when controls are applied.

Encouraging Stronger Controls:

The formula incentivises implementing and enhancing strong controls by focusing on the maximum compensating control value. It highlights the direct relationship between control strength and residual threat reduction.

Support for Prioritisation:

Higher TAmv values indicate significant residual risks, enabling the prioritisation of additional mitigation measures. Conversely, lower TAma values demonstrate effective mitigation, assuring control sufficiency.

Simplified Integration into Graph Models:

The calculation is straightforward and easily integrated into graph-based methodologies. The TAmv value becomes a node attribute that can influence the graph's other nodes (e.g., assets or vulnerabilities) and edges.

Risk-Based Resource Allocation:

Decision-makers can leverage TAmv to allocate resources effectively, focusing on high residual risk areas that may require additional controls or measures to mitigate.

Practical Implications for Mitigation:

The formula provides actionable insights:

- Existing controls are adequate if TAmv is low.
- If TAmv is high, it signals the need for enhanced controls or other risk treatments.

Ensures Comprehensive Risk Analysis:

By reducing the Threat Actor Value (TAv) by the most effective compensating control, TAmv ensures that residual risks are systematically accounted for, supporting robust and evidence-based decision-making in cybersecurity risk management.

6.3.2 Attack Node

An Attack (AT) is a set of actions performed by a Threat Actor (TA) to a Vulnerability (V) to negatively impact the confidentiality, integrity, availability, or accountability of an Asset (A).

Attack Node Attributes

Attack Complexity (ATc)

Attack complexity represents the degree of sophistication or the level of expertise, resources, and specialised tools required to successfully execute a cyber-attack. This concept encapsulates the inherent difficulty of exploiting a particular system vulnerability.

Scope of Target Systems	Single Systems, such as attacking an individual system or user, multiple Systems, or entire networks.
Technical Sophistication	Basic Attacks, such as simple phishing emails, password guessing, etc., are carried out through Advanced Persistent Threats (APTs), highly sophisticated and targeted attacks often sponsored by nation-states, and zero-day exploits, where vulnerabilities unknown to software vendors are used.
Required Infrastructure	Simple Infrastructure where only a single system is needed to launch an attack, through to botnets using a network of compromised systems to launch distributed attacks or multi-stage infrastructure using several layers of command-and-control servers to obfuscate the attack's origin.
Duration	Short-term attacks like DDoS happen over a brief period and long-term where attacks are slow and stealthy and might go unnoticed for months or even years.
Evasion Techniques	This includes simple proxy usage, rootkits, traffic obfuscation, polymorphic malware, and other techniques to avoid detection.

Table 12 - Attack complexity

Attack Proliferation (ATp)

Attack proliferation represents the dispersion and frequency of a specific type of cyber-attack. This is contingent upon several factors, such as the accessibility of exploit tools, the number of systems

susceptible to the attack, the potential rewards for successful exploits and the inherent difficulty or risk involved in executing the attack.

Exploit Vector	Wormable exploits allow malware to spread automatically without human interaction versus manual propagation, where an attack requires manual steps for propagation, such as spear-phishing emails sent to specific targets.
Attack Automation	Such as botnets and malware kits, allowing simplified attack launching versus custom-coded attacks.
Vulnerability Prevalence	Suppose a vulnerability exists in popular or widely used software/hardware. In that case, its exploitation can lead to rapid proliferation (e.g., a vulnerability in a popular operating system). In contrast, niche vulnerabilities targeting fewer common systems may have a slower or more limited spread.
Highly Connected Systems	Interconnected systems (e.g., IoT devices) can facilitate faster spread compared to isolated systems that operate in silos or have limited connectivity might resist rapid proliferation.

Table 13 - Attack proliferation

Attack Value (ATv)

The value (ATv) of an Attack Node is calculated based on the values of its attributes:

Attack Complexity (ATc): How complex is the attack to undertake successfully. A higher value in the range of 0 to 100 indicates a lower complexity attack.

Attack Proliferation (ATp): How prolific is the attack? A greater proliferation of the attack is indicated by a higher value in the range of 0 to 100.

The Attack Value is calculated as follows:

$$AT_v = \left(\frac{AT_c\ AT_p}{2}\right)$$

Equation 9 - Attack Value

The calculation of the Attack Node Value (ATv) using the formula can be justified as follows:

Balanced Assessment of Attack Characteristics:

The formula combines two key attributes:

Attack Complexity (ATc): Represents how difficult the attack is to execute. A lower complexity (ATc) indicates that the attack is more straightforward, posing a greater risk.

Attack Proliferation (ATp): This number reflects how widespread or prevalent the attack is. A higher ATp suggests a more commonly used or available attack, increasing its likelihood and impact.

By incorporating both factors, the formula provides a balanced evaluation of the threat level posed by the attack.

Equal Weighting for Objectivity:

Averaging the sum of ATc and ATp assumes both attributes are equally important in determining the overall attack value. This ensures an unbiased approach unless specific evidence suggests one factor should dominate.

Normalisation and Comparability:

The division by 2 normalises the combined value to the same scale as the inputs (0–100). This ensures the attack value is easily comparable across different node nodes in the graph-based model.

Alignment with Risk Management Practices:

Risk management frameworks like NIST CSF and ISO/IEC 27005 emphasise understanding attack characteristics to assess risk. The formula aligns with these principles by quantitatively evaluating attacks' difficulty and prevalence.

Encouragement for Countermeasures:

The formula highlights the importance of addressing efficiently executed attacks (low ATc) and widely proliferated attacks (high ATp). This encourages the development of countermeasures tailored to specific attack characteristics.

Support for Threat Prioritisation:

Higher ATv values indicate a greater attack potential, prompting an immediate focus on mitigation measures. Conversely, lower ATv values signal fewer critical threats, enabling resource prioritisation.

Dynamic Integration into Graph-Based Models:

ATv is input for downstream graph calculations, influencing attacks' interaction with other nodes (e.g., vulnerabilities or assets). This integration ensures a comprehensive view of the attack's impact within the system.

Simplified yet Insightful Calculation:

The formula is simple and interpretable, ensuring stakeholders can quickly implement and understand it while providing actionable insights into attack characteristics.

Actionable Insights:

A high ATc (low complexity) and ATp (high proliferation) suggest that:

The attack is easily executable and commonly available.

Mitigations should focus on restricting access to resources or reducing the attack's applicability.

Comprehensive Threat Evaluation:

By capturing both the ease of execution and the prevalence of the attack, ATv provides a holistic view of attack potential, which is crucial for assessing and managing risks effectively.

Attack Mitigated Value (ATmv)

The Attack Mitigated Value (ATmv) represents the residual risk or remaining severity of an attack node after applying the most effective compensating control (CCv) associated with it. This value serves as a refined metric for assessing the impact of attack pathways within a cybersecurity-directed graph model. Mathematically, the calculation is expressed as:

$$AT_{mv} = AT_v - \max(CC_v)$$

Equation 10 - Attack Mitigated Value

At its core, this calculation seeks to quantify how much risk remains after applying the strongest available compensating control, enabling the model to propagate and prioritise residual risks throughout the graph accurately.

The calculation of the Attack Mitigated Value (ATmv) using the formula can be justified as follows:

Purpose of Attack Mitigated Value (ATmv)

In cybersecurity, controls rarely eliminate risk entirely, they reduce or mitigate it to an acceptable level. The ATmv captures this reduced risk state as an essential parameter for ongoing risk analysis, prioritisation, and mitigation planning. Attacks in a directed graph represent pathways or events through which vulnerabilities are exploited, and controls represent interventions designed to disrupt or minimise those pathways. Without accounting for the impact of the most muscular control, the model risks overstating the severity of an attack node and failing to prioritise mitigation efforts effectively.

By explicitly reducing the Attack Value (ATv) using the highest Compensating Control Value (CCv), the calculation ensures that the most impactful mitigation effort is prioritised, whether it arises from the control's intrinsic properties or its relational effectiveness with the target attack node. This approach mirrors real-world cybersecurity strategies, where the goal is not merely to apply controls but to prioritise and maximise their overall impact.

Relationship Between ATv and CCv

The Attack Value (ATv) quantifies an attack's inherent severity, incorporating factors such as complexity, proliferation potential, damage potential, and accessibility. It represents the raw risk the attack node poses before any mitigating measures are considered.

On the other hand, the Compensating Control Value (CCv) reflects the cumulative mitigating power of the most effective control associated with that attack node. This value accounts for both the control's inherent effectiveness (Cv) and its strength of interaction with the attack (Ev). Controls are rarely uniformly effective across all threats, and their mitigation power often depends on the specific dependencies and pathways they address.

Using the maximum compensating control value (max(CCv)), the calculation emphasises the best mitigation pathway rather than diluting the result by averaging or summing multiple weaker controls. This ensures that the residual risk assessment accurately reflects the most potent control, aligning with the cybersecurity principle of risk-aware prioritisation.

Justification for Subtraction (ATv - max(CCv))

The subtraction operation captures the net effect of the most muscular control on the attack's inherent risk. In cybersecurity, risk is not merely mitigated conceptually it is quantitatively reduced through adequate controls. Subtracting the highest control value directly from the attack value reflects this reduction in precise mathematical terms.

This approach ensures transparency and interpretability. Analysts, decision-makers, and auditors can clearly understand how applying a specific control reduces the risk a given attack poses. The result, ATmv, becomes an immediately interpretable metric representing the residual risk state after accounting for control intervention.

Furthermore, this subtraction approach avoids the complexity of nonlinear modelling or probabilistic risk propagation, keeping the calculation both computationally efficient and scalable. As cybersecurity

assessments often involve vast and interconnected datasets, this method's simplicity ensures it remains practical for real-world implementation without sacrificing analytical robustness.

Residual Risk as a Basis for Further Analysis

The Attack Mitigated Value (ATmv) is not an endpoint but a transitional metric. Its primary purpose is to serve as a foundation for further graph-based calculations, such as propagating residual risks to downstream nodes, identifying critical attack paths, or evaluating systemic vulnerabilities. For example, an attack node with a high residual value despite significant control intervention signals a structural weakness or inadequacy in the existing control mechanisms.

In scenarios where multiple attack nodes intersect or cascade their impacts onto shared assets or vulnerabilities, ATmv becomes a critical input for assessing compound or cumulative risks. This capability allows the graph to provide dynamic insights into risk propagation pathways, guiding more strategic prioritisation of resources and mitigation strategies.

Alignment with Risk Management Frameworks

The calculation of ATmv aligns with established cybersecurity and risk management frameworks, including ISO/IEC 27005, NIST CSF, and FAIR (Factor Analysis of Information Risk). Each framework emphasises the importance of accounting for residual risk after applying risk treatments.

ISO/IEC 27005, for instance, emphasises the need for continuous risk assessment and the evaluation of control effectiveness to determine residual risk levels. Similarly, NIST CSF highlights risk mitigation as an iterative process where control performance directly influences the post-mitigation risk posture. The ATmv calculation captures this principle mathematically, ensuring consistency with internationally accepted standards and practices.

Adaptability and Real-Time Responsiveness

One of the key strengths of the ATmv calculation lies in its adaptability to evolving threat environments. If an attack's inherent severity (ATv) increases due to new vulnerabilities or emerging threats, the ATmv will automatically reflect this change upon recalculation. Similarly, if the performance of a compensating control (CCv) improves or degrades over time, the residual risk value adjusts dynamically.

This adaptability supports continuous monitoring and real-time updates, ensuring that the graph-based cybersecurity model remains responsive to changes without requiring significant recalibration.

Decision-Making and Resource Allocation

From an operational standpoint, the ATmv serves as a decision-support metric for cybersecurity teams and leadership. Nodes with high residual attack values become focal points for further investigation, resource allocation, and mitigation planning. For instance:

Nodes with persistently high ATmv scores may require enhanced compensating controls or alternative mitigation strategies.

Nodes with effectively reduced ATmv values indicate well-functioning controls, offering confidence in the security posture.

Organisations can optimise their cybersecurity investments by prioritising risk reduction efforts based on ATmv and ensuring that resources are directed toward the most critical areas.

Transparency and Interpretability

The mathematical simplicity of the formula (ATv - max(CCv)) ensures that the results remain transparent and easily interpretable across technical and non-technical stakeholders. Analysts can confidently communicate residual risk levels, while decision-makers can trust the results as a basis for strategic planning.

Threat Value (Tv)

The Threat Value (Tv) represents the residual risk associated with a Threat Node after considering both the mitigating effects of associated Threat Actors (TAmv) and the Attack Mitigated Value (ATmv) from related attacks. The formula incorporates the highest Threat Actor Mitigated Value (TAmv), adjusted by the Edge Strength Value (Ev), alongside the Attack Mitigated Value (ATmv) to derive a comprehensive measure of the threat's overall impact on the cybersecurity graph. Mathematically, this is expressed as:

$$T_v = \max\left(TA_{mv}\ E_v\right)AT_{mv}$$

Equation 11 - Threat Value

The calculation of the Attack Mitigated Value (ATmv) using the formula can be justified as follows:

Purpose of Threat Value (Tv)

In a cybersecurity graph, Threat Nodes represent potential compromise or harm arising from malicious intent, vulnerability exploitation, or systemic weaknesses. The Threat Value (Tv) is a quantifiable measure of the remaining threat severity after accounting for threat actor influences and attack mitigations.

The calculation aims to combine these dimensions into a single metric, allowing for the propagation of threat risk across the graph and supporting subsequent analysis, such as identifying critical pathways, prioritizing mitigation efforts, and assessing systemic vulnerabilities.

At its core, the Threat Value (Tv) reflects both the external influence of threat actors and the residual risk from mitigated attacks, balancing these factors based on their relative contributions to overall threat severity.

Components of the Calculation

Threat Actor Mitigated Value (TAmv)

The Threat Actor Mitigated Value (TAmv) represents the residual influence of a Threat Actor Node on the threat node in question. Threat actors, whether human attackers, malicious insiders, or automated systems, influence vulnerabilities and assets differently. This influence is often characterized by:

Capability: The technical skills or resources available to the actor.

Motivation: The actor's intent and persistence in pursuing exploitation.

Access: The actor's level of access to assets or systems.

The Edge Strength Value (Ev) refines this relationship by quantifying how strongly the threat actor node interacts with the threat node. A high Ev indicates a stronger, more influential relationship, while a lower Ev suggests limited direct impact.

Incorporating both TAmv and Ev into the formula ensures that the most influential relationship is emphasized when evaluating the threat node. Using the maximum value between TAmv and Ev, the calculation prioritizes the most substantial contributing factor, whether it stems from the actor's residual impact or the direct interaction strength.

Attack Mitigated Value (ATmv)

The Attack Mitigated Value (ATmv) quantifies the remaining impact of an attack on this threat node after considering the most effective compensating control. This value reflects the direct residual risk of attack vectors exploiting vulnerabilities or weaknesses.

While TAmv accounts for the influence of external actors, ATmv captures the internal state of residual risk at the attack level. These two dimensions provide a dual perspective one focusing on external actor dynamics and the other on intrinsic node-level risk.

Justification for the Maximum Function (max)

Using the maximum function (max) in the calculation reflects a risk-aware prioritization principle, where the strongest contributing factor, whether TAmv or Ev, dictates the primary driver of the threat's severity.

This approach is justified because not all threat nodes are influenced equally by their actors or edges. For example, a highly capable threat actor with significant resources may dominate the risk assessment despite weaker edge interactions. Conversely, a weak actor with a disproportionately strong edge connection to the threat node might still pose significant risk due to the pathway's criticality.

Using the maximum value ensures that the most significant influence is prioritized, preserving analytical clarity and preventing weaker factors from diluting the risk assessment.

Justification for the Multiplication with Attack Mitigated Value (ATmv)

After identifying the most influential value between TAmv and Ev, the calculation multiplies this value by the Attack Mitigated Value (ATmv). This multiplication serves two critical purposes:

First, it creates a dependency relationship. The residual attack risk (ATmv) is a baseline severity measure for the threat node, anchoring the calculation in tangible risk metrics. Without this baseline, the influence of threat actors and edge relationships would lack contextual grounding.

Second, it integrates external and internal perspectives. While TAmv/Ev captures external influences, ATmv represents the node's internal risk state. Combining these factors ensures that the resulting Threat Value (Tv) reflects both dimensions, avoiding bias toward internal or external risk factors.

This multiplicative relationship is also mathematically consistent with risk propagation principles, where downstream impacts depend on initial severity (ATmv) and amplifying or mitigating factors (TAmv/Ev).

Basis for Further Risk Calculations

The Threat Value (Tv) is a foundational downstream graph analysis metric. It enables risk propagation to adjacent nodes, supports vulnerability prioritization, and provides insights into systemic risk patterns.

For example, nodes with high Tv scores may serve as focal points for further risk mitigation, while low Tv scores suggest effective control and actor management. This value also supports dynamic graph queries, such as:

Identifying nodes with high residual risk despite multiple mitigating controls.

Prioritizing pathways where actor influence disproportionately amplifies attack severity.

Analysing systemic risk concentrations across interconnected nodes.

Without an accurate calculation of Tv, subsequent graph-based risk calculations would lack the granularity and precision required for meaningful analysis.

Alignment with Risk Management Frameworks

The calculation of Threat Value (Tv) aligns with established cybersecurity frameworks, including:

ISO/IEC 27005: Emphasizes risk evaluation through the combined actor, asset, and control dimensions.

NIST Cybersecurity Framework (CSF) encourages dynamic evaluation of threat severity based on evolving actors and attack conditions.

FAIR (Factor Analysis of Information Risk) Highlights the interplay between threat actors, attack pathways, and control influences.

These frameworks collectively advocate for holistic threat assessment, balancing external and internal dimensions, a principle inherently captured in the TV calculation.

Adaptability to Dynamic Environments

The formula supports dynamic recalculation in response to changing conditions. If an actor's capabilities improve, the TAmv increases, and the Tv reflects this change. Similarly, if compensating controls reduce ATmv, the overall Tv is correspondingly lowered. This adaptability ensures that the cybersecurity graph remains responsive to evolving risks, emerging threats, and updated controls.

6.3.3 Vulnerability Node

A Vulnerability (V) is any weakness in a system, component, or process that can be negatively impacted by an Attack (AT) and subsequently exposes an Asset (A). Vulnerabilities are not introduced; rather they are inherent within the system, component, or process.

Vulnerability Value (Vv)

The Vulnerability Value (Vv) is used as the basis for the Vulnerability calculations. The Vulnerability Value (Vv) is calculated from five attributes: Ease of Exploitation, Expose to Attack, Privileges Required, Interaction Required and Exposes Additional Scope.

The following describes each of these attributes:

Ease of Exploitation (Vee)

The easier a vulnerability is to exploit, the higher the risk associated with it, since it is more likely to be targeted by attackers. Several things contribute to the ease of exploitation of a vulnerability:

Technical Knowledge Required	Some vulnerabilities may require an attacker to have a deep understanding of specific technologies, protocols, or programming languages. The more specialised knowledge required, the less easy a vulnerability is to exploit.

Availability of Exploit Tools/Code	If exploit code or automated tools are publicly available for a vulnerability, it drastically lowers the barrier to exploiting that vulnerability. When exploit code is published, even less skilled attackers could potentially exploit the vulnerability.
Attack Vector	Vulnerabilities that can be exploited remotely over the network are easier to exploit than those requiring physical access or user interaction.
Environment Specificity	Some vulnerabilities only exist in specific configurations or under certain conditions. The more specific the conditions required to exploit a vulnerability, the harder it becomes to find suitable targets and exploit them.
Payload Constraints	For some vulnerabilities, there might be significant constraints on the malicious payload, such as size or format. Constraints can make it more challenging to develop a working exploit, particularly one that achieves a desired malicious outcome.
Detection and Visibility	Some vulnerabilities, when exploited, can cause disruptions or noticeable changes, drawing attention. If exploiting a vulnerability is likely to trigger alarms or be detected quickly, it may deter some attackers or require them to be stealthier, making the exploitation process more complex.

Table 14 - Vulnerability easy of exploitation

Exposure to Attack (Vea)

Exposure to attack refers to the extent and way a vulnerability is exposed or accessible to potential attackers. It is a crucial component of risk assessment, since the risk associated with a particular vulnerability is significantly affected by the degree of exposure.

Attack Surface Area:	This refers to the total sum of points in a software environment where an unauthorised user can attempt to enter data or extract data. A larger attack surface typically means a higher exposure, as there are more points for an attacker to target.
System Accessibility	Systems that are publicly accessible (e.g., web servers) are inherently more exposed than systems behind firewalls or in private networks. Systems with public-facing interfaces are more exposed to potential attacks from a wider range of threat actors.
Network Architecture and Segmentation	A well-segmented network can contain breaches and limit movement. Poorly segmented networks can allow attackers to move laterally more easily, increasing the exposure of multiple systems following a single breach.
Physical Security Measures	The physical security of hardware and data centres can impact exposure to threats like theft, tampering, or sabotage. Inadequate physical security can expose systems to additional risks, especially from insider threats.

Table 15 - Vulnerability exposure to attack

Privileges Required (Ver)

Privileges Required refers to the level of access or user privileges an attacker would need to successfully exploit a given vulnerability. It is an essential aspect of assessing the risk posed by potential vulnerabilities. User privileges can range from the rights of a normal user to those of a system administrator. Privileges can allow actions such as reading, writing, or deleting files, executing tasks, or changing system settings. The level of privileges required to exploit a vulnerability has a direct impact on its severity:

No Privileges Required	If a vulnerability can be exploited without any special privileges, the risk is inherently high. This means that any user, or even an unauthorised individual, could potentially exploit the vulnerability. These are typically prioritised for remediation.
User Privileges Required	In this case, the attacker must have access to a user account on the system to exploit the vulnerability. The risk is less than if no privileges were required, but it is still considerable. If a user's account can be compromised (e.g., through phishing, weak passwords, etc.), the attacker can then use this access to exploit the vulnerability.
Admin Privileges Required	Here, the attacker must have administrative-level access to exploit the vulnerability. While this might seem like a high barrier, it is worth noting that if an attacker gains administrative access, they can potentially cause significant harm. Therefore, even though the likelihood is reduced, the impact of such a vulnerability can be severe.

Interaction Required (Vir):

Interaction Required refers to whether a vulnerability can be exploited solely by the attacker or if the exploit also requires action from a user, such as clicking a link or opening a file. This factor can significantly affect a vulnerability's risk rating because it directly impacts the likelihood of successful exploitation.

No User Interaction Required	If a vulnerability can be exploited without any user interaction, it is typically considered to have a high risk. Automated attacks, worms, or network-borne attacks typically fall into this category. For example, a server-side vulnerability in a widely used protocol like HTTP could be exploited by simply sending maliciously crafted network traffic to the server, requiring no user interaction.
User Interaction Required	If a vulnerability requires some form of user interaction to be successfully exploited, the risk is usually considered to be lower (though still significant). This is because the attacker needs to rely on a user to perform a specific action. For example, a client-side vulnerability in a web browser might require a user to visit a malicious website or click on a specific link. Phishing attacks are a common example of this type of vulnerability.

Table 16 - Vulnerability interaction required

Exposes Additional Scope (Ves):

When a vulnerability "exposes additional scope," it means that exploiting this vulnerability allows an attacker to impact or gain access to resources, systems, or network segments beyond the initial target

or boundary of the compromised system. This concept is vital for understanding the potential reach and severity of a vulnerability.

"Scope" in this context refers to the range or boundary within which a system operates or the extent of its authority. In a well-segmented and managed network, different systems and network segments have their own scopes, defined by elements like network configurations, access controls and security policies.

Vulnerability Value (Vv)

The Vulnerability Value (Vv) is calculated as follows:

Ease of Exploitation (Vee): How easy is it for an attack to exploit this vulnerability. An easier-to-exploit vulnerability is indicated by a higher value in the range of 0 to 100.

Exposure to Attack (Vea): How exposed is the vulnerability to a potential attack. A greater exposure is indicated by a higher value in the range of 0 to 100.

Privileges Required (Ver): Does the vulnerability require special system privileges on the system, such as Admin, to be effective. The lower the privileges required is indicated by a higher value in the range of 0 to 100.

Interaction Required (Vir): Does the vulnerability require the user to undertake some type of action to be effective. The lower the user interaction required is indicated by a higher value in the range of 0 to 100.

Exposes Additional Scope (Ves): Does this vulnerability expose other aspects of the Asset that could be further compromised. The greater additional scope exposed is indicated by a higher value in the range of 0 to 100.

The Vulnerability Value (Vv) is calculated as follows:

$$V_v = \left(\frac{V_{ee} + V_{ea} + V_{pr} + V_{ir} + V_{es}}{5}\right)$$

Equation 12 - Vulnerability Value

The following provides a justification for including these attributes, the rationale behind the equal weighting approach, and the calculation's overall transparency and utility in supporting downstream risk analysis.

Holistic Assessment of Vulnerability Characteristics

Vulnerabilities are inherently multifaceted and cannot be assessed accurately through a single dimension. Each attribute included in the formula captures a distinct aspect of a vulnerability's characteristics, ensuring a comprehensive evaluation:

Ease of Exploitation (Vee): This attribute measures how straightforward an attacker can exploit the vulnerability. A vulnerability requiring minimal technical knowledge or no specialized tools will have a higher score, as it poses a greater risk of exploitation.

Exposure to Attack (Vea): This factor assesses how exposed the vulnerability is to potential attacks. Vulnerabilities on externally accessible systems or unpatched software are considered highly exposed and, therefore, more likely to be targeted.

Privileges Required (Vpr): Privilege requirements evaluate whether an attacker needs elevated access (e.g., root, admin) to exploit the vulnerability. Vulnerabilities that require no special privileges to execute an exploit are inherently more dangerous and score higher.

Interaction Required (Vir): User interaction is a critical factor in exploitability. Vulnerabilities that do not require a user to click on a link, download a file, or perform an action are easier to exploit and pose a higher risk.

Exposes Additional Scope (Ves): This attribute measures whether the vulnerability's exploitation opens pathways to compromise additional systems, assets, or services. Vulnerabilities with cascading effects or chain-exploitation potential have a higher score.

By incorporating these five dimensions, the calculation reflects a balanced and comprehensive evaluation of a vulnerability's overall risk potential. Omitting any of these factors would create an incomplete representation of the true risk posed by the vulnerability.

Equal Weighting Across Attributes

The calculation averages the five attributes, assigning equal importance to each. This approach ensures that no single attribute dominates the vulnerability's overall assessment unless explicitly defined by an organizational risk strategy.

Equal weighting is justified for several reasons:

Lack of Universal Hierarchy: In many scenarios, the relative importance of these attributes can vary depending on the specific system, asset, or threat landscape. Assigning equal weight avoids introducing biases based on assumptions that might not apply universally.

Simplicity and Clarity: Averaging the five attributes creates a clear, interpretable metric without adding unnecessary complexity to the assessment process.

Baseline Comparison: Equal weighting provides a baseline vulnerability score, allowing organizations to layer on context-specific weightings if needed without compromising the assessment's initial clarity.

Unless an organization has explicit reasons to prioritize one attribute over others (e.g., critical systems exposed to public networks may prioritize Vea), equal weighting ensures a neutral and fair evaluation across diverse vulnerability types.

Transparency and Interpretability

The averaging method simplifies the calculation, making it transparent and easy to understand for technical cybersecurity analysts and non-technical stakeholders. The numerical result, expressed as a single value between 0 and 100, allows for:

Quick Comparison Across Vulnerabilities: Higher scores immediately signal greater risk, enabling prioritization of patching or mitigation efforts.

Visualization in Risk Dashboards: The single numeric value can be easily visualized in graphs, dashboards, and reports.

Consistent Benchmarking: Vulnerability values can be consistently compared across systems, environments, and timeframes.

This transparency reduces ambiguity in communicating risk metrics, particularly in decision-making forums involving executives, auditors, or regulators.

Alignment with Cybersecurity Frameworks and Standards

The attributes chosen for the Vulnerability Value (Vv) calculation align with established cybersecurity risk assessment frameworks, such as:

Common Vulnerability Scoring System (CVSS): CVSS evaluates vulnerabilities based on exploitability, access requirements, and impact, mirroring the attributes included in this formula.

NIST Cybersecurity Framework (CSF): NIST emphasizes vulnerability management as a critical pillar of risk assessment and encourages multi-dimensional evaluation of vulnerabilities.

ISO/IEC 27005: The standard advocates for structured vulnerability assessment methodologies incorporating exploitability and systemic impact.

By incorporating attributes aligned with these frameworks, the calculation ensures compatibility with industry standards and enhances credibility in risk assessments.

Supports Strategic Decision-Making and Prioritization

The Vulnerability Value (Vv) is a foundational metric for strategic decision-making in cybersecurity operations. Nodes with higher vulnerability values indicate areas requiring immediate attention, whether through:

Patch Management: Prioritizing updates for systems with highly exploitable vulnerabilities.

Compensating Controls: Implementing alternative mitigation measures when patches are unavailable.

Access Control Adjustments: Restricting access pathways to reduce exposure.

The calculation's standardized nature also supports comparative analysis across different systems, helping cybersecurity teams focus on vulnerabilities that represent the highest overall risk.

Dynamic Adaptability

The Vulnerability Value (Vv) calculation is inherently adaptable to changing threat landscapes.

If a patch is applied, the Exposure to Attack (Vea) score decreases, lowering the overall vulnerability value.

If an attacker develops a new exploit tool, the Ease of Exploitation (Vee) score may increase the value.

This adaptability ensures that the Vv remains context-aware and responsive to emerging risks.

Foundation for Further Graph Calculations

The Vulnerability Value (Vv) serves as a key input for downstream graph calculations, such as:

Risk Propagation: Understanding how high-risk vulnerabilities propagate risks to connected nodes (e.g., assets or controls).

Attack Path Analysis: Identifying critical vulnerability pathways that attackers most likely exploit.

Control Optimization: Evaluating whether controls effectively reduce vulnerability risk.

Without an accurate representation of Vv, subsequent calculations risk being inaccurate, undermining the integrity of the entire graph model.

Vulnerability Mitigated Value (TAmv)

The Vulnerability Mitigated Value (Vmv) results from the Vulnerability Value (Vv) of the Node reduced by the highest compensating Control Value (Cv) to this Node. The Vulnerability Mitigated Value (Vmv) is used as the basis for further graph calculations.

The Vulnerability Mitigated Value (Vmv) is calculated as follows:

$$V_{mv} = V_v - \max\left(CC_v\right)$$

Equation 13 - Vulnerability Mitigated Value

This section provides a detailed justification for including both Vulnerability Value (Vv) and Compensating Control Value (CCv) in the formula, the rationale for the subtraction operation, and the significance of this calculation for subsequent graph-based cybersecurity risk analysis.

Purpose of Vulnerability Mitigated Value (Vmv)

Vulnerabilities represent potential weaknesses or flaws in a system that attackers can exploit to gain unauthorized access, disrupt operations, or compromise data integrity. However, not all vulnerabilities pose equal risks after controls are applied. Compensating controls, whether technical (e.g., firewalls, patching), administrative (e.g., policies, training), or procedural (e.g., multi-factor authentication), can reduce the effective severity and exploitability of vulnerabilities.

The Vulnerability Mitigated Value (Vmv) serves as a residual risk metric, quantifying the vulnerability's remaining threat after applying the most effective control. This metric ensures that cybersecurity assessments are not overly conservative by assuming vulnerabilities remain fully exploitable despite strong controls. Conversely, it prevents an overly optimistic bias by ensuring that only the strongest control's impact is accounted for.

Vmv bridges the gap between raw vulnerability risk and the mitigated state achieved through control implementation.

Components of the Calculation

The Vulnerability Value (Vv) quantifies the intrinsic risk associated with a vulnerability. It captures critical factors such as:

Ease of Exploitation (Vee): How easily an attacker can exploit the vulnerability.

Exposure to Attack (Vea): How visible and accessible the vulnerability is to potential attackers.

Privileges Required (Vpr): The level of access needed to exploit the vulnerability.

Interaction Required (Vir): The degree of user interaction required to trigger the vulnerability.

Exposes Additional Scope (Ves): Whether exploiting the vulnerability can compromise additional assets or systems.

This comprehensive score provides the baseline risk potential of a vulnerability before any compensating measures are applied.

The Compensating Control Value (CCv) represents the strength of the most effective control in mitigating vulnerability. This value accounts for both:

Control Effectiveness (Cv): The intrinsic strength and performance of the control in addressing the vulnerability.

Edge Strength (Ev): The quality and intensity of the control's relationship with the vulnerability node.

The calculation prioritises the most effective mitigation mechanism by taking the maximum value across all controls associated with the vulnerability, ensuring that weaker or peripheral controls do not dilute the risk assessment.

Justification for the Subtraction Operation (Vv- max(CCv))

The subtraction operation directly represents the net reduction in vulnerability risk achieved by the strongest control. This approach is mathematically sound and conceptually aligns with risk management principles.

Direct Representation of Risk Reduction

Subtraction provides an unambiguous measure of residual risk by quantifying how much of the vulnerability's original risk (Vv) has been reduced by the best available control (CCv). The result, Vmv, reflects the remaining vulnerability state after accounting for the most effective mitigation measure.

Emphasis on the Strongest Control

The calculation ensures that the most impactful control dominates the residual risk assessment by using the maximum Compensating Control Value (CCv) rather than an average or cumulative score. This prioritization prevents weaker controls from disproportionately influencing the result, ensuring an accurate representation of the vulnerability's mitigated state.

Clarity and Interpretability

The subtraction method is simple, transparent, and interpretable. It allows technical analysts, decision-makers, and stakeholders to understand how applying control reduces a vulnerability's risk. The residual risk score can be directly compared across multiple vulnerabilities, supporting efficient prioritization.

Alignment with Risk Management Frameworks

The calculation of Vulnerability Mitigated Value (Vmv) aligns with internationally recognized risk management frameworks, including:

ISO/IEC 27005: Emphasizes assessing residual risk after implementing controls to ensure vulnerabilities are mitigated to acceptable levels.

NIST Cybersecurity Framework (CSF): This framework highlights the importance of risk mitigation and ongoing evaluation of residual risks in cybersecurity operations.

FAIR (Factor Analysis of Information Risk): Advocates for quantifiable metrics to represent risk reduction resulting from implemented controls.

Each framework underscores the necessity of evaluating post-control residual risk, a principle inherently captured in the Vmv calculation.

Residual Vulnerability as a Basis for Further Calculations

The Vulnerability Mitigated Value (Vmv) is not an isolated metric but a foundational input for downstream graph-based calculations. It directly informs:

Risk Propagation Analysis: Understanding how residual vulnerability risk affects connected nodes, including assets and threats.

Pathway Analysis: Identifying critical paths where residual vulnerabilities pose significant systemic risks.

Control Effectiveness Evaluation: Assessing whether controls effectively reduce vulnerability impact across connected assets.

Nodes with high Vmv values after mitigation indicate areas where vulnerabilities remain significantly exploitable despite control efforts, signalling a need for additional interventions or alternative strategies.

Supports Strategic Decision-Making

The Vmv metric facilitates strategic decision-making by providing:

Clear Prioritization: Nodes with high residual vulnerability values become priority targets for patching, enhanced controls, or architectural changes.

Resource Optimization: Cybersecurity resources can be allocated more effectively to address the most significant residual risks.

Compliance Alignment: Demonstrating quantified residual risk supports regulatory compliance and audit transparency.

Adaptability to Dynamic Threat Environments

The Vmv calculation is inherently adaptable to changes in the vulnerability's risk profile or the effectiveness of applied controls.

Patching Vulnerabilities: Reduces the Vv, thereby lowering the residual risk.

Control Improvements: An increase in CCv directly decreases the Vmv value.

Emerging Exploits: If a vulnerability becomes easier to exploit, the Vv increases, and the residual risk reflects this heightened threat.

This adaptability ensures that the cybersecurity model remains reflective of real-world risk dynamics.

The calculation of Vulnerability Mitigated Value (Vmv) as the difference between the Vulnerability Value (Vv) and the maximum Compensating Control Value (CCv) is a mathematically robust and conceptually sound approach to representing residual vulnerability risk.

By emphasizing the strongest available control and providing a transparent risk reduction measure, the formula aligns with industry standards, supports dynamic adaptability, and serves as a key input for downstream graph analysis and decision-making processes.

Likelihood Value

A Likelihood Value (Lv) results from the combined effects of the highest Threat Value (Tv) to this Node and the Vulnerability Mitigated Value (Vmv). The Threat Value (Tv) is impacted by the Edge Strength Value (Ev) of the Edge between the Attack Node and this Node. The Likelihood Value is used as the basis of further risk calculations on the graph.

The Likelihood Value (Lv) from this node is calculated as follows:

$$L_v = \max\left(T_v\ E_v\right)V_{mv}$$

Equation 14 - Likelihood Value

The following justifies including these attributes, the rationale behind the combination of maximum and multiplication operations, and the calculation's overall transparency and utility in supporting downstream risk analysis. The Likelihood Value (Lv) quantifies the probability of successfully exploiting a node in a cybersecurity graph, incorporating both external threat factors and internal vulnerability characteristics. It combines the Threat Value (Tv) representing the influence of threats and attack mechanisms with the Vulnerability Mitigated Value (Vmv), which captures the residual vulnerability risk. Additionally, the Edge Strength Value (Ev) adjusts the threat value by factoring in the quality and significance of the connection between nodes.

The Likelihood Value (Lv) quantifies the probability or chance of a successful exploit occurring at a given node within the cybersecurity-directed graph. This value combines the Threat Value (Tv), which

encapsulates the severity and influence of threats, and the Vulnerability Mitigated Value (Vmv), which reflects the residual risk of a vulnerability after the control application. Additionally, the Edge Strength Value (Ev) adjusts the threat value based on the quality and strength of the connection between the attack node and the target node.

Purpose of Likelihood Value (Lv)

In cybersecurity risk analysis, likelihood represents the probability that a specific vulnerability will be successfully exploited when exposed to a threat. Unlike static probability models, directed graph representations allow for a dynamic and contextual evaluation of likelihood by considering intrinsic node attributes (e.g., residual vulnerability risk) and external influences (e.g., threat actor behaviour and attack pathways).

The Likelihood Value (Lv) serves as a critical input for downstream calculations, such as risk propagation, attack path analysis, and prioritization of mitigation efforts. By incorporating Threat Value (Tv) and Vulnerability Mitigated Value (Vmv), along with the relational impact captured by Edge Strength (Ev), the calculation ensures that the most significant contributing factors are appropriately emphasized.

Components of the Calculation

The Threat Value (Tv) represents the severity and influence of threat actors and attack mechanisms impacting the node. It considers:

Threat Actor Influence: The capability, motivation, and access of attackers.

Edge Strength (Ev): The strength and clarity of the relationship between the attack and target nodes.

Attack Mitigated Value (ATmv): Residual attack risk after applying controls.

Threat Value captures the external pressure and risk potential introduced by malicious actors and their strategies.

The Edge Strength Value (Ev) modifies the influence of the threat node on the current node. If the edge has a high strength value, it indicates a strong, well-defined pathway for threat propagation. Conversely, a weaker edge suggests limited or indirect influence.

Using the maximum value (max(Tv, Ev)) ensures that the most impactful contributor dominates the likelihood calculation. For example:

> A highly motivated and resourceful threat actor (high Tv) may dominate the node risk, regardless of a weak edge connection.
>
> Conversely, a weaker threat actor (low Tv) may exert significant risk if the edge connection (high Ev) provides direct and vulnerable access.

This prioritization reflects the principle of risk-aware prioritization, ensuring that the strongest driver of likelihood is emphasized.

The Vulnerability Mitigated Value (Vmv) represents the residual risk associated with the vulnerability at the node after applying compensating controls. The baseline likelihood multiplier anchoring the threat influences (Tv, Ev) to the node's inherent vulnerability state.

Nodes with higher Vmv indicate significant vulnerabilities despite existing controls, amplifying the likelihood of a successful exploit when paired with an active threat or strong edge influence.

Justification for the Maximum Function (max(Tv, Ev))

The maximum function (max) ensures that the calculation is dominated by the strongest contributor between Threat Value (Tv) and Edge Strength Value (Ev).

If a threat actor is highly capable and persistent (high Tv), their influence will drive the likelihood value.

If the edge connecting the attack node and target node is exceptionally strong (high Ev), it indicates a direct and effective attack pathway, amplifying the likelihood of exploitation.

This approach avoids diluting critical risk factors and ensures that the calculation reflects the most impactful contributor. In cybersecurity contexts, prioritizing the most significant risk driver aligns with risk management best practices.

Justification for Multiplication with Vulnerability Mitigated Value (Vmv)

After determining the most significant external risk driver using max(Tv, Ev), the calculation multiplies this value by the Vulnerability Mitigated Value (Vmv). This multiplication serves two key purposes:

Grounding External Risk in Internal Node Context:

External risk drivers (threat or edge influence) must interact with internal node characteristics (vulnerability risk) to create an exploitable scenario. Multiplying by Vmv ensures that likelihood is contextualized to the node's residual vulnerability risk.

Reflecting Combined Risk Dynamics:

Multiplication captures the amplifying effect of external threats and internal vulnerabilities. A high Threat Value (Tv) will not pose a significant risk if the Vulnerability Mitigated Value (Vmv) is very low, and vice versa. This ensures a balanced representation of combined risk factors.

Mathematically, this approach reflects established risk propagation models where external pressures and internal weaknesses combine to determine overall likelihood.

Alignment with Cybersecurity Risk Management Frameworks

The calculation of Likelihood Value (Lv) aligns with established cybersecurity risk frameworks:

ISO/IEC 27005 emphasizes assessing the probability of successful exploitation based on threat activity and vulnerability state.

NIST Cybersecurity Framework (CSF): Highlights the importance of combining threat actor influence, attack pathways, and residual vulnerabilities in risk assessments.

FAIR (Factor Analysis of Information Risk) Advocates for risk probability assessments that integrate external threats and internal vulnerability states.

The formula adheres to these principles, providing a robust, industry-aligned approach to likelihood assessment.

Dynamic Adaptability to Threat Landscape Changes

The Likelihood Value (Lv) is inherently adaptive to changing conditions:

An increase in Threat Value (Tv) due to emerging attacker tactics or resources will dynamically elevate Lv.

If edge relationships (Ev) strengthen or weaken, the likelihood value adjusts accordingly.

Improvements in compensating controls that reduce Vmv will proportionally decrease Lv.

This adaptability ensures that the likelihood metric reflects real-world conditions and supports continuous risk monitoring.

Strategic Decision-Making and Prioritization

The Likelihood Value (Lv) serves as a key decision-support metric:

Nodes with high Lv values indicate urgent areas for intervention.

Nodes with low Lv values suggest reduced immediate risk, enabling resource allocation to higher-priority vulnerabilities.

This clarity supports targeted mitigation efforts, optimized control deployments, and effective stakeholder communication.

The calculation of the Likelihood Value (Lv) as the product of the maximum Threat Value (Tv) or Edge Strength Value (Ev) and the Vulnerability Mitigated Value (Vmv) is a mathematically sound and conceptually robust approach to assessing the probability of successful exploitation at a node.

It balances external threat dynamics and internal vulnerability risk, ensuring a comprehensive representation of exploitation likelihood. Aligned with cybersecurity best practices and frameworks, the calculation supports dynamic risk adaptation, strategic decision-making, and effective prioritization in complex cybersecurity environments.

Risk Value

A Risk Value (Rv) results from the combined effects of the highest Likelihood Value (Lv) to this Node and the Asset Mitigated Value (Amv). The Likelihood Value (Lv) is impacted by the Edge Strength Value (Ev) of the Edge between the Vulnerability Node and this Node. The Risk Value (Rv) is used as the basis of further risk calculations on the graph.

The Risk Value (Rv) for this node is calculated as follows:

$$R_v = \max\left(L_v\ E_v\right)A_{mv}$$

Equation 15 - Risk Value

The following justifies including these attributes, the rationale behind the combination of maximum and multiplication operations, and the calculation's overall transparency and utility in supporting downstream risk analysis. The Risk Value (Rv) represents the quantitative measure of potential harm or impact at a specific node within a cybersecurity-directed graph. It combines the Likelihood Value

(Lv), which expresses the probability of a successful exploit occurring, with the Asset Mitigated Value (Amv), which represents the asset's residual value after accounting for controls and mitigations. The Edge Strength Value (Ev) also adjusts the likelihood value to account for the relationship strength between the vulnerability node and the asset node.

Inclusion of Key Attributes

The formula integrates three key elements to provide a holistic assessment of risk:

Likelihood Value (Lv): This value reflects the probability of a vulnerability being successfully exploited, incorporating the interplay between threat actors, vulnerability mitigation, and edge relationships.

Edge Strength Value (Ev): This value captures the strength and quality of the relationship between the vulnerability node and the asset node. Strong edges indicate a clearer and more direct pathway for risk propagation.

Asset Mitigated Value (Amv): Represents the residual criticality or importance of the asset after considering compensating controls. Higher asset values indicate more significant potential harm if the risk materializes.

By incorporating these dimensions, the calculation captures both external drivers of risk (via Lv and Ev) and internal asset significance (via Amv), resulting in a balanced, context-aware representation of risk.

Rationale for the Maximum Operation (max(Lv,Ev))

The maximum function (max) ensures that the most impactful factor between Likelihood Value (Lv) and Edge Strength Value (Ev) dominates the calculation.

If Lv dominates: It indicates that the primary driver of risk is the probability of exploitation due to highly active threat actors, exploitable vulnerabilities, or inadequate mitigation.

If Ev dominates: It signifies that the pathway or connection strength between the vulnerability node and the asset node creates an elevated risk potential, even if the inherent likelihood value is moderate.

Using the maximum value prevents less significant contributors from diluting the most critical factor. This prioritization aligns with the principle of risk-aware prioritization, which emphasizes addressing the strongest risk drivers first.

Multiplication with Asset Mitigated Value (Amv)

Once the dominant driver of risk is established via the maximum operation, it is multiplied by the Asset Mitigated Value (Amv). This multiplication serves two essential purposes:

Contextualizing Risk to Asset Significance: Even if a threat or pathway is highly likely, the ultimate risk depends on the value and importance of the impacted asset. Multiplying by Amv ensures that the risk is anchored to the asset's residual value, reflecting the potential harm or loss.

Capturing Combined Dynamics: Multiplication reflects the amplifying effect of asset criticality on external and pathway risks. A highly significant asset (high Amv) paired with a high likelihood of a strong edge pathway leads to elevated risk. Conversely, an asset with low residual value will result in comparatively lower overall risk, even if external factors are high.

This approach mirrors widely accepted risk assessment methodologies, such as those found in ISO/IEC 27005 and the FAIR (Factor Analysis of Information Risk) model, which emphasize the combination of likelihood and asset value in quantifying risk.

Transparency and Interpretability

The calculation of Risk Value (Rv) is both mathematically transparent and conceptually intuitive:

The maximum function (max(Lv,Ev)) is straightforward, emphasizing the strongest contributor.

Multiplication by Asset Mitigated Value (Amv) ties risk to asset significance.

The resulting Risk Value (Rv) is a single, interpretable metric that allows stakeholders to compare risks across nodes, prioritize mitigation actions, and communicate findings effectively.

This clarity ensures the results are easily understandable for technical analysts, executives, and regulatory auditors.

Alignment with Risk Management Frameworks

The calculation aligns with recognized cybersecurity and risk management standards:

ISO/IEC 27005: Emphasizes assessing risk as a combination of likelihood and asset impact.

NIST Cybersecurity Framework (CSF): This framework highlights the importance of dynamic risk evaluation based on asset criticality and threat likelihood.

FAIR Model: Advocates for quantitative risk analysis grounded in the probability of exploitation and magnitude of asset loss.

By adhering to these frameworks, the calculation ensures credibility, consistency, and practical applicability in real-world cybersecurity assessments.

Utility in Downstream Risk Analysis

The Risk Value (Rv) serves as a critical input for downstream graph-based calculations and decision-making processes:

Risk Propagation: Evaluating how risk values propagate across interconnected nodes and pathways.

Attack Path Analysis: Identifying nodes where elevated risk values form critical points of compromise.

Resource Allocation: Prioritizing high-risk nodes for additional controls, monitoring, or resource allocation.

Strategic Mitigation Planning: Informing long-term strategies for reducing systemic risks within the cybersecurity ecosystem.

Without an accurate representation of Rv, subsequent calculations risk becoming inconsistent, leading to suboptimal risk management strategies.

Adaptability to Changing Risk Dynamics

The calculation supports dynamic updates in response to changes in threat conditions, asset values, or pathway strengths:

If the Likelihood Value (Lv) increases due to new threat intelligence, the Risk Value (Rv) adjusts proportionally.

If compensating controls reduce the Asset Mitigated Value (Amv), the risk level decreases accordingly.

Changes in Edge Strength (Ev) dynamically influence the dominant contributor to risk.

This adaptability ensures that the Rv metric remains contextually accurate and responsive to evolving cybersecurity landscapes.

Strategic Decision-Making and Prioritization

The Risk Value (Rv) provides organizations with a quantifiable and comparable measure of risk across all nodes:

Nodes with high Rv values become immediate priorities for intervention.

Nodes with lower Rv values can be deprioritized, allowing resources to focus on higher-impact areas.

Comparative analysis across nodes enables organizations to identify systemic vulnerabilities and optimize control deployments.

This clarity empowers strategic decision-makers to focus efforts where they will have the most significant impact on risk reduction and resilience enhancement.

The calculation of Risk Value (Rv) as the product of the maximum Likelihood Value (Lv) or Edge Strength Value (Ev) and the Asset Mitigated Value (Amv) provides a mathematically sound, conceptually robust, and actionable measure of cybersecurity risk.

The calculation captures a balanced and dynamic representation of risk by integrating external likelihood factors, pathway influences, and internal asset significance. It aligns with internationally accepted cybersecurity frameworks, supports strategic decision-making, and serves as a key input for downstream risk analysis and resource optimization.

6.4 Node Relationships

6.4.1 Threat Actor and Attack Nodes

A threat actor and an attack are closely related, but they refer to different aspects of a cybersecurity incident. A threat actor (also known as a malicious actor) refers to an individual or group that is responsible for an attack on a system. Threat actors can have various motivations, such as financial gain, political beliefs, business competition, or just causing chaos and disruption. They can range from individual hackers to organised crime groups, state-sponsored groups, or even disgruntled employees within an organisation.

An attack is the actual malicious action carried out by the threat actor. It can take many forms, such as malware infection, denial-of-service attack, phishing attempt, ransomware attack, or exploiting a vulnerability in a system. The attack is the method by which the threat actor attempts to achieve their goal, which can be data theft, system disruption, monetary extortion, etc.

The threat actor is the "who" and the attack is the "how" in a cybersecurity incident. Understanding both aspects is crucial for effective cybersecurity management and incident response. Knowing the tactics, techniques, and procedures (TTPs) used by different types of threat actors can help organisations prepare for and defend against attacks.

Within the graph schema, Threat Actors and Attacks have a directed, many-to-many relationship; for example, one or more Threat Actors can undertake one or more Attacks.

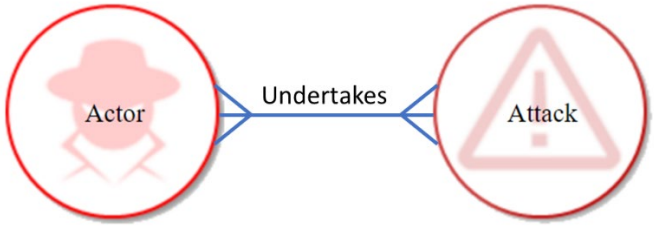

Figure 11 - Threat actor to attack relationship

6.4.2 Attack and Vulnerability Nodes

An attack is an act that exploits a vulnerability in a system. A vulnerability is a weakness or flaw in a system that can be exploited to compromise its integrity, confidentiality, availability, or accountability. A vulnerability is a condition that can potentially allow an attack to occur. When a threat actor identifies a vulnerability, they can exploit it via an attack using various methods such as malware, phishing, SQL injection, etc., to attack the system and potentially compromise, damage, steal information, disrupt services, or perform other malicious activities.

Within the graph schema, Attacks and Vulnerabilities have directed, many-to-many relationships; for example, one or more Attacks can exploit one or more Vulnerabilities.

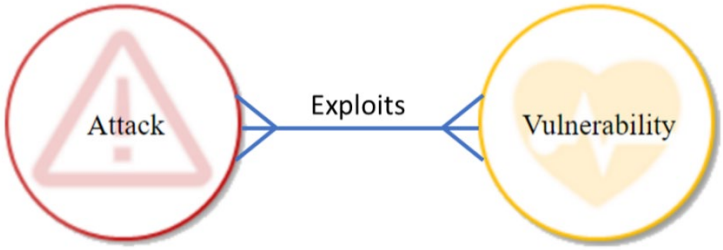

Figure 12 - Attack to vulnerability relationship

6.4.3 Vulnerability and Asset Nodes

The relationship between a vulnerability and an asset is crucial in cybersecurity. A vulnerability poses a threat to an asset. If a vulnerability exists and is exploited, the asset can be compromised, potentially leading to a loss of confidentiality, integrity, or availability of the asset.

Within the graph schema, Vulnerabilities and Assets have directed many to many relationships; for example, one or more Vulnerabilities Impact one or more Assets.

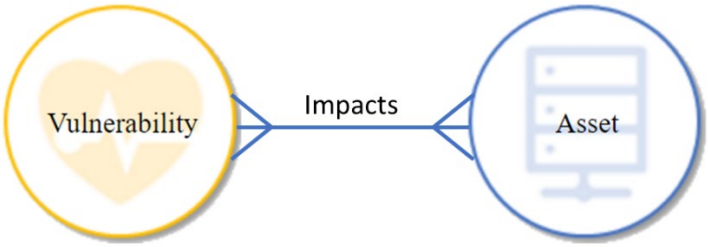

Figure 13 - Vulnerability to asset relationship

6.4.4 Control and Threat Actor Nodes

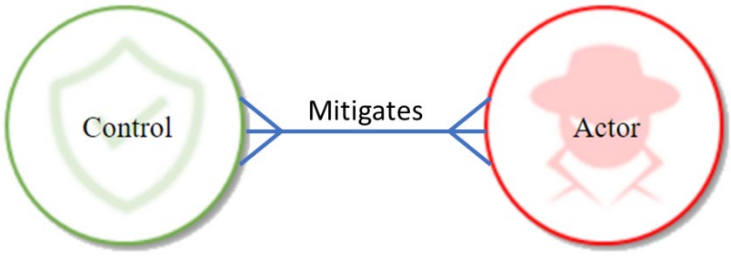

Figure 14 - Control to actor relationship

A control refers to a safeguard or countermeasure implemented to mitigate risks posed by threat actors. Controls are designed to minimise the likelihood or impact of a security breach or compromise. Control nodes can help mitigate the risk associated with Threat Actors by:

Access Controls: By implementing strong access controls, such as authentication mechanisms (e.g., passwords, and multi-factor authentication), organisations can limit unauthorised access to systems and sensitive information. This reduces the risk of threat actors gaining unauthorised entry into critical resources.

Intrusion Detection/Prevention Systems (IDS/IPS): These systems monitor network traffic and detect potential intrusions or malicious activities. By deploying IDS/IPS solutions, organisations can identify and block suspicious network traffic, mitigating the risk of threat actors exploiting vulnerabilities or launching attacks.

Firewalls: Firewalls act as a barrier between an organisation's internal network and external networks (such as the Internet). They inspect and filter incoming and outgoing traffic based on predefined security rules. Firewalls help prevent unauthorised access and protect against various types of threats, including those initiated by threat actors.

Encryption: Encryption is the process of encoding information to make it unreadable without the appropriate decryption key. By encrypting sensitive data, organisations can protect it from unauthorised access even if it falls into the hands of threat actors. Encryption helps mitigate the risk of data breaches and unauthorised disclosure.

Security Awareness and Training: Educating employees about cybersecurity best practices and raising awareness about potential threats can significantly reduce the risk posed by threat actors. Training programs help employees identify and report suspicious activities, avoid social engineering attacks and follow secure practices, thereby mitigating the likelihood of successful attacks.

Incident Response Planning: Having a well-defined incident response plan enables organisations to effectively respond to and mitigate the impact of a security incident caused by threat actors. This includes promptly detecting and containing the breach, minimising data loss or compromise and restoring normal operations. A structured response plan can limit the potential damage caused by threat actors.

Within the graph schema, Controls and Threat Actors have directed, many-to-many relationships; for example, one or more Controls mitigate one or more Threat Actors.

6.4.5 Control and Attack Nodes

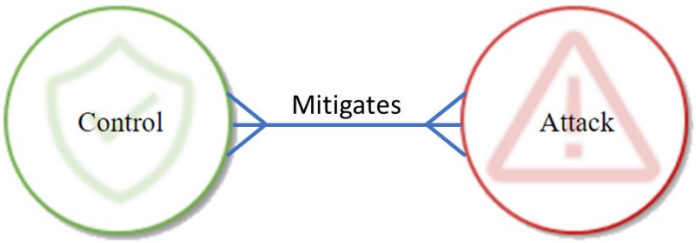

Figure 15 - Control to attack relationship

Controls can help mitigate the risk associated with Attacks by providing:

Multi-Factor Authentication (MFA): This is a preventive control that enhances the security of user logins. In addition to a password, it requires another factor, like a token from a phone app or a fingerprint before a user can access a system. This helps to prevent unauthorised access, even if a password is compromised, as the attacker would also need the second factor.

Firewalls: Firewalls act as a barrier between a trusted internal network and untrusted external networks. They can block traffic based on IP addresses, port numbers, or protocols. This can prevent attacks that rely on specific network connections or vulnerabilities.

Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS): IDS and IPS monitor network traffic for suspicious activity. If they detect an attempted attack, such as a brute force attack or an SQL injection, they can alert administrators or even automatically block the traffic.

Encryption: Data encryption transforms data into an unreadable format using an algorithm and an encryption key. Even if an attacker manages to steal the data, they will not be able to read or use it without the encryption key.

Security Information and Event Management (SIEM) Systems: SIEM systems aggregate and analyse log data from various systems in real time. They can detect suspicious activities that could indicate an ongoing attack, such as multiple failed login attempts and raise the alarm for immediate action.

Regular Patching and Updates: Keeping software and systems updated is a fundamental control for cybersecurity. Many attacks exploit known vulnerabilities in outdated software. Regular patching ensures these vulnerabilities are fixed.

Backup and Recovery Systems: These are corrective controls that ensure the organisation can recover from a data loss event, such as a ransomware attack, where an attacker encrypts your data and demands a ransom for the decryption key. Regular backups allow the organisation to restore their data without paying the ransom.

Security Awareness Training: This preventive control aims to educate employees about common cyber threats and how to identify and respond to them. Many attacks, such as phishing, rely on tricking users into giving away their login credentials or other sensitive information.

Within the graph schema, Controls and Attacks have a directed, many-to-many relationship; for example, one or more Controls mitigates one or more Threat Actors.

Control and Vulnerability Nodes

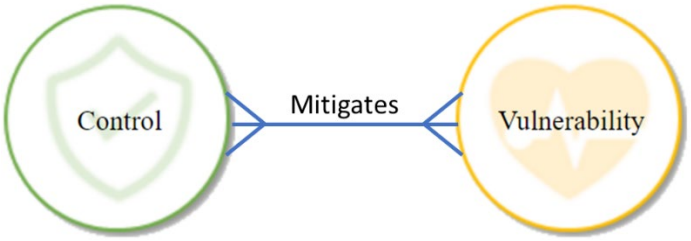

Figure 16 - Control to vulnerability relationship

By mitigating vulnerabilities, controls protect information and systems from unauthorised access, use, disclosure, disruption, modification, or destruction:

Patch Management: Regularly applying security patches to applications and operating systems is a control that can mitigate vulnerabilities. By keeping software up to date, known vulnerabilities are fixed, reducing the chances of exploitation.

Access Controls: Implementing robust access control ensures that only authorised individuals can access sensitive data and systems. Techniques like role-based access control (RBAC) or attribute-based access control (ABAC) can grant privileges based on the principle of least privilege, where users are granted only the permissions necessary to perform their job functions.

Firewalls and Intrusion Detection Systems (IDS): Network firewalls and IDSs are controls that can protect against unauthorised access and malicious activities. Firewalls can be configured to block traffic from known malicious IP addresses and IDSs can monitor for unusual patterns, alerting administrators when suspicious activity is detected.

Encryption: Encrypting data in transit and at rest is a control that can mitigate the risk associated with data breaches. Even if an attacker gains access to sensitive data, encryption can render the data unreadable without the proper decryption keys.

Antivirus and Anti-malware Software: These controls can help identify and block malware, including viruses, worms, and Trojan horses, which may exploit vulnerabilities to gain unauthorised system access.

Security Policies and Procedures: Administrative controls in the form of policies and procedures can also help mitigate vulnerabilities. For example, having a security policy that mandates regular password changes and strong passwords can mitigate the risk of password guessing or brute force attacks.

Security Awareness Training: Educating employees and users about cybersecurity best practices can be an effective control. This can mitigate vulnerabilities that stem from human error or social engineering attacks, such as phishing.

Network Segmentation: Dividing a network into separate segments, especially for sensitive data, can reduce the attack surface and contain breaches. Restricting the access and movement between segments, this control can mitigate the potential damage of an exploited vulnerability.

Regular Audits and Vulnerability Scanning: Regular audits and scanning for vulnerabilities can help identify and address weaknesses before they can be exploited. This proactive control helps organisations stay ahead of emerging threats.

Multi-Factor Authentication (MFA): Implementing MFA is an effective control to secure user accounts. By requiring more than one form of verification to authenticate, this control reduces the risk of account compromise due to stolen or weak passwords.

Application Allowlisting: Only allowing approved applications to run on systems prevents the execution of unauthorised or malicious software. This control can be particularly effective against unknown malware or zero-day exploits.

6.4.6 Control and Asset Nodes

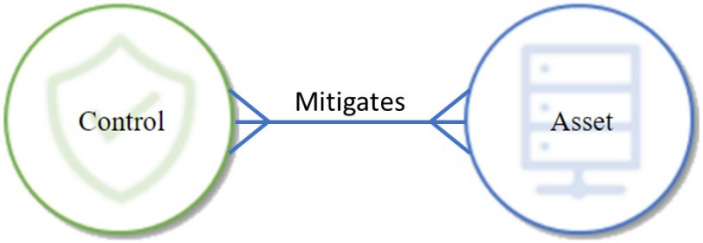

Figure 17 - Control to asset relationship

Examples of ways a Control can mitigate risk to an Asset Node include:

Access Control: By implementing access controls like Role-Based Access Control (RBAC) or Attribute-Based Access Control (ABAC), you can ensure that only authorised users can access specific assets. This helps mitigate the risk of data leakage or unauthorised access to critical systems.

Data Encryption: Encrypting data at rest and in transit can protect it from unauthorised access. If an asset such as a storage device is compromised, the encrypted data would still be secure unless the attacker also has the decryption keys.

Antivirus and Anti-malware Software: These controls protect assets such as computers and servers from malware infections. Keeping antivirus software updated ensures protection against the latest threats.

Firewalls and Network Segmentation: Implementing firewalls and segmenting networks can protect networked assets. By isolating critical assets in separate network segments and implementing firewalls, you can prevent the spread of network intrusion and minimise the risk of compromised assets.

Backup and Recovery Procedures: Regularly backing up data and having a robust recovery process is a control that mitigates the risk of data loss due to hardware failures, ransomware, or other disastrous events. This ensures that assets in the form of data can be restored quickly.

Patch Management: Regularly updating software and firmware on assets reduces the risk of known vulnerabilities being exploited. Keeping assets patched is essential for maintaining security.

Security Awareness Training: Training employees to recognise phishing emails and to follow best security practices helps protect assets from compromises due to human error or social engineering.

Physical Security Controls: Implementing physical security measures like access cards, biometric access and surveillance cameras helps to protect physical assets such as servers, workstations and network equipment from theft or tampering.

Multi-Factor Authentication (MFA): MFA adds a layer of security to protect assets by ensuring that a compromised password alone is not enough for an attacker to gain access.

Security Monitoring and Incident Response: Implementing security monitoring solutions helps detect unauthorised access or anomalies involving assets. Having an incident response plan ensures that when a security incident occurs, there is a predefined process for handling it, minimising the impact on assets.

Data Classification and Handling Policies: By classifying data based on its sensitivity and applying appropriate handling policies, an organisation can ensure that more sensitive assets receive higher protection.

Configuration Management: Ensuring that the configurations of systems and applications are secure and compliant with best practices helps reduce vulnerabilities and protect assets from exploitation.

6.4.7 Asset to Asset Node

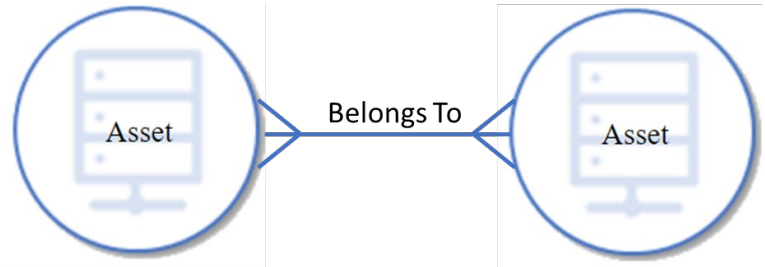

Figure 18 - Asset to asset relationship

In cybersecurity, assets are considered any data, device, or other component of the environment that supports information-related activities. Assets in a cybersecurity context can relate to one another in several ways:

Dependency: Some assets depend on the functioning of others. For example, a web application (asset) might depend on a database (another asset) to function correctly.

Hierarchy: Assets can exist within a hierarchy. For instance, a server can be an asset, but the individual applications and data stored on that server are also assets.

Communication: Assets such as servers and devices might communicate. This communication creates a relationship; if one asset is compromised, it can pose risks to any asset it communicates with.

Ownership: Assets might be owned or managed by the same entity, creating an organisational relationship. This can be important for determining responsibility or assessing risk based on the owner's or manager's security posture.

Physical Location: Assets located in the exact physical location might share risks related to physical threats such as theft, fire, or natural disasters.

Shared Vulnerabilities: If two assets are vulnerable to the same threat, they relate in the context of shared risk. For example, two different systems running the same outdated software might be vulnerable to the same exploit.

In Chapter 4, we introduced the concept of Asset Nodes and identified their four principal attributes: Confidentiality, Integrity, Availability and Accountability. It is important to note that while individual assets will possess distinct values for these attributes, the interconnectedness between assets implies that any impact experienced by one can influence others within its relational web.

For instance, envision a scenario wherein an individual system is represented as an Asset Node. If this system Asset Node is embedded within a larger organisational Asset Node, any disruptions or impacts felt by the system could resonate more broadly, engendering indirect consequences for the overarching organisational Asset. This interrelatedness underscores the importance of meticulously examining asset relationships. Two salient aspects demand our attention:

Direction and Strength of Relationship: It is essential to comprehend the directional flow and the intensity of connections between Asset Nodes.

Nature of Impact: How changes or disturbances in one Asset Node might manifest in another.

To offer a more structured perspective, impacts can be categorised into four distinct types:

Sum: Impacts derived from source Asset Nodes are aggregated. The cumulative score is then integrated with the inherent impact score of the target node.

High Water Mark: This method captures the most pronounced impact from all source Asset Nodes. This peak value is then added to the target Node's impact score.

Low Water Mark: Unlike the previous method, this approach zeroes in on the most muted or least significant impact from the source Asset Nodes. This minimal value is then added to the target Node's inherent impact score.

Average: This method employs a mean value approach. An average impact score is computed from all the contributing source Asset Nodes and then added to the target Node's impact score.

6.4.8 Worked Example 4 – Single Actor, Single Attack, Single Vulnerability, Single Asset

The following simple example demonstrates how Information Security Risk can be calculated:

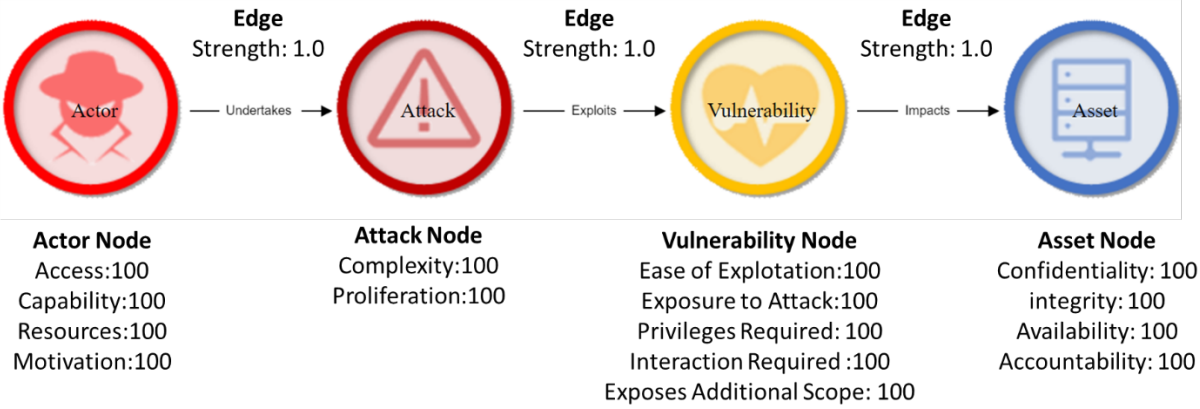

Figure 19 - Worked Example 4 – Single Actor, Single Attack, Single Vulnerability, Single Asset

The Threat Actor Value (TAv) is calculated as follows:

$$TA_v = \left(\frac{TA_a + TA_c + TA_r + TA_m}{4}\right)$$

Substitution:

$$TA_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

$$TA_v = 100$$

Result:

$$TA_v = 100$$

The Attack Value (ATv) is calculated as follows:

$$AT_v = \left(\frac{AT_c\ TA_p}{2}\right)$$

Substitution:

$$AT_v = \left(\frac{100 \times 100}{2}\right)$$

Result:

$$AT_v = 100$$

The Threat Value (Tv) from this node is calculated as follows:

$$T_v = \left(\frac{TA_v \times E_v \times AT_V}{100}\right)$$

Substitution:

$$T_v = \left(\frac{100 \times 1.0 \times 100}{100}\right)$$

Result:

$$T_v = 100$$

The Vulnerability Value (Vv) is calculated as follows:

$$V_v = \left(\frac{V_{ee} + V_{ea} + V_{pr} + V_{ir} + V_{es}}{5}\right)$$

Substitution:

$$V_v = \left(\frac{100 + 100 + 100 + 100 + 100}{5}\right)$$

Result:

$$V_v = 100$$

The Likelihood Value (Lv) from this node is calculated as follows:

$$L_v = \left(\frac{T_v\ E_v\ Vv}{100}\right)$$

Substitution:

$$L_v = \left(\frac{100 \times 1.0 \times 100}{100}\right)$$

Result:

$$L_v = 100$$

The Asset Value (Av) is calculated as follows:

$$A_v = \left(\frac{A_c + A_i + A_a + A_{ac}}{4}\right)$$

Substitution:

$$A_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

Result:

$$A_v = 100$$

The Risk Value (Rv) is calculated as follows:

$$R_v = \left(\frac{L_v\, E_v\, Av}{100}\right)$$

Substitution:

$$R_v = \left(\frac{100 \times 1.0 \times 100}{100}\right)$$

Result:

$$R_v = 100$$

Equation 16 - Worked Example 4 – Single Actor, Single Attack, Single Vulnerability, Single Asset

6.4.9 Worked Example 5 – Single Actor, Single Attack, Single Vulnerability, Single Asset with Multiple Controls

The following simple example demonstrates how Information Security Risk can be calculated when the mitigating effect of Controls are considered.

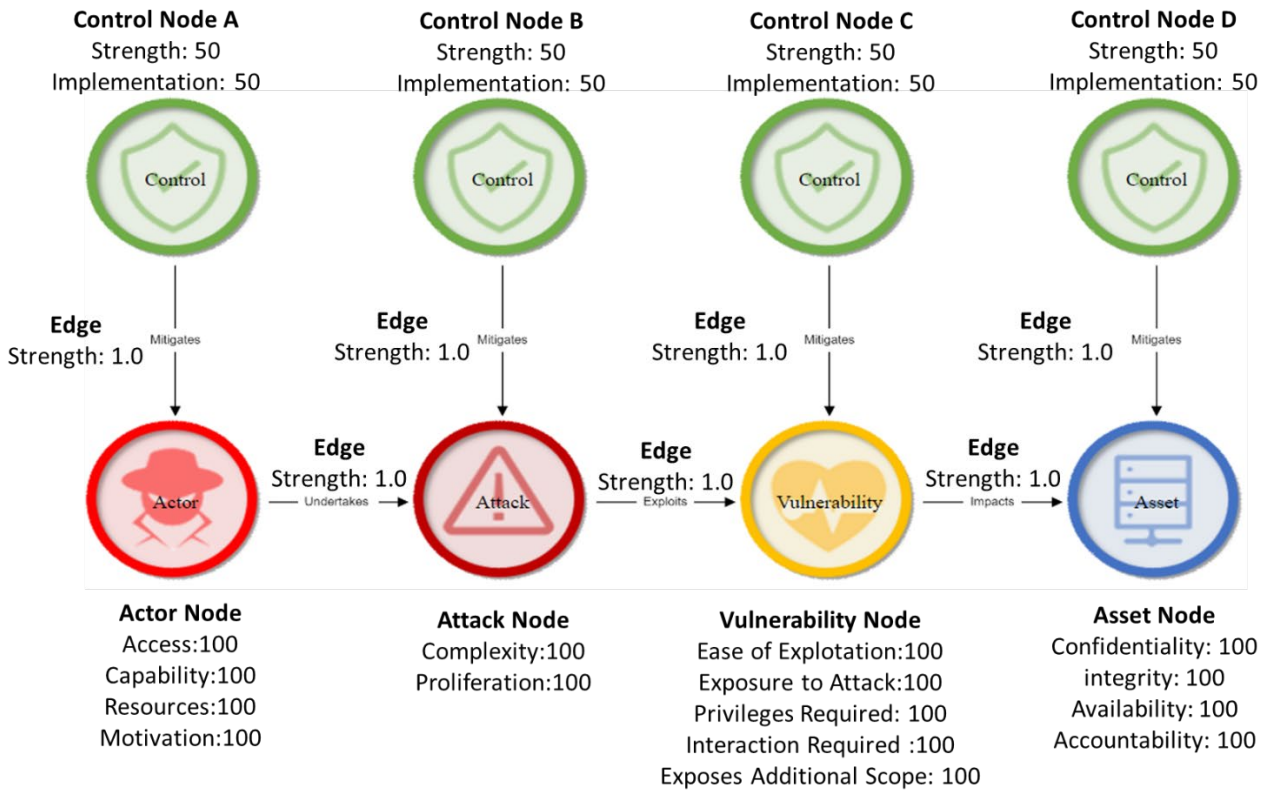

Figure 20 - Worked Example 5 – Single Actor, Single Attack, Single Vulnerability, Single Asset with Multiple Controls

The Control Node Values (Cv) are calculated as follows:

$$C_v = \left(\frac{C_s \ C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{50 \times 50}{100}\right)$$

Result:

$$C_v = 25$$

The Threat Actor Value (TAv) is calculated as follows:

$$TA_v = \left(\frac{TA_a + TA_c + TA_r + TA_m}{4}\right)$$

Substitution:

$$TA_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

Result:

$$TA_v = 100$$

The Compensating Control Value (CCv) is calculated as follows:

$$CC_v = max(C_v\ E_v)$$

Substitution:

$$CC_v = 25\ \times 1.0$$

Result:

$$CC_v = 25$$

The Threat Actor Mitigated Value (TAmv) is calculated as follows:

$$TA_{mv} = TAv - CC_v$$

Substitution:

$$TA_{mv} = 100 -\ 25$$

Result:

$$TAm_v = 75$$

The Attack Value (ATv) is calculated as follows:

$$AT_v = \left(\frac{AT_c\ TA_p}{2}\right)$$

Substitution:

$$AT_v = \left(\frac{100 \times 100}{2}\right)$$

Result:

$$AT_v = 100$$

The Compensating Control Value (CCv) is calculated as follows:

$$CC_v = max(C_s\ E_v)$$

Substitution:

$$CC_v = 25\ \times 1.0$$

Result:

$$CC_v = 25$$

The Attack Mitigated Value (ATmv) is calculated as follows:

$$AT_{mv} = ATv - CC_v$$

Substitution:

$$AT_{mv} = 100 - 25$$

Result:

$$ATm_v = 75$$

The Threat Value (Tv) from this node is calculated as follows:

$$T_v = \left(\frac{TA_{mv} + E_v + AT_{mV}}{100}\right)$$

Substitution:

$$T_v = \left(\frac{75 \times 1.0 \times 75}{100}\right)$$

Result:

$$T_v = 56.25$$

The Vulnerability Value (Vv) is calculated as follows:

$$V_v = \left(\frac{V_{ee} + V_{ea} + V_{pr} + V_{ir} + V_{es}}{5}\right)$$

Substitution:

$$V_v = \left(\frac{100 + 100 + 100 + 100 + 100}{5}\right)$$

Result:

$$V_v = 100$$

The Compensating Control Value (CCv) is calculated as follows:

$$CC_v = max(C_s\ E_v)$$

Substitution:

$$CC_v = 25\ \times 1.0$$

Result:

$$CC_v = 25$$

The Vulnerability Mitigated Value (Vmv) is calculated as follows:

$$V_{mv} = Vv - CC_v$$

Substitution:

$$V_{mv} = 100 -\ 25$$

Result:

$$Vm_v = 75$$

The Likelihood Value (Lv) from this node is calculated as follows:

$$L_v = \left(\frac{T_v\ E_v\ V_{mv}}{100}\right)$$

Substitution:

$$L_v = \left(\frac{56.25 \times 1.0 \times 75}{100}\right)$$

Result:

$$L_v = 42.19$$

The Asset Value (Av) is calculated as follows:

$$A_v = \left(\frac{A_c + A_i + A_a + A_{ac}}{4}\right)$$

Substitution:

$$A_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

Result:

$$A_v = 100$$

The Compensating Control Value (CCv) is calculated as follows:

$$CC_v = max(C_s\ E_v)$$

Substitution:

$$CC_v = 25\ \times 1.0$$

Result:

$$CC_v = 25$$

The Asset Mitigated Value (Amv) is calculated as follows:

$$A_{mv} = Av - CC_v$$

Substitution:

$$A_{mv} = 100 - 25$$

Result:

$$Am_v = 75$$

The Risk Value (Rv) is calculated as follows:

$$R_v = \left(\frac{L_v\ E_v\ A_{mv}}{100}\right)$$

Substitution:

$$R_v = \left(\frac{42.19 \times 1.0 \times 75}{100}\right)$$

Result:

$$R_v = 31.64$$

Equation 17 - Worked Example 5 – Single Actor, Single Attack, Single Vulnerability, Single Asset with Multiple Controls

Chapter 7 - Using Directed Graphs for Simultaneously Modelling Information Security Risk and Maturity

In chapter 5 of this thesis, we explored the application of directed graphs for assessing cybersecurity maturity. Following this, directed graphs were introduced as an innovative approach to depict and assess cybersecurity maturity. We defined key terms such as vertices, edges and weights and discussed the properties of directed graphs that are particularly pertinent to cybersecurity. Through various

examples, it was demonstrated how the components of cybersecurity, including policies, procedures, tools, and human factors, could be represented as vertices and their relationships as edges in a directed graph. We also discussed how these graphs could be used to visually depict the evolution of an organisation’s cybersecurity maturity over time by effectively showcasing how different components interact and evolve.

The merger of a cybersecurity risk graph and a cybersecurity framework compliance graph can offer an organisation many invaluable insights and advantages, fostering a more strategic and risk-focused approach to cybersecurity.

First and foremost, such integration furnishes a comprehensive perspective of an organisation's cybersecurity landscape. The risk graph symbolically encapsulates the organisation's vulnerabilities, threats, and potential impacts, while the compliance graph delineates the organisation's alignment with the cybersecurity framework's requirements. This amalgamation provides a more holistic understanding of the threat landscape and the level of compliance across diverse facets of the organisation.

Next, the unification of these graphs facilitates risk-informed compliance activities. Organisations can prioritise compliance efforts based on the associated risk levels by correlating compliance gaps with high-risk areas identified in the risk graph. This approach ensures a targeted allocation of resources to address compliance and risk mitigation, harmonising compliance activities with risk management strategies.

Furthermore, the combined graph aids in the prioritisation of risk mitigation efforts. By highlighting high-risk profiles in the risk graph and identifying areas of non-compliance associated with them, organisations can concurrently align their risk management strategies to address compliance gaps and high-priority threats. This alignment optimises resource allocation and risk management strategies, effectively addressing the organisation's most pressing cybersecurity challenges.

The integration of these graphs also enhances decision-making processes. By visually representing the relationships between risks, compliance requirements and organisational components, stakeholders can better comprehend the organisation's cybersecurity landscape dynamics. This understanding facilitates informed decisions about risk mitigation strategies, resource allocation and compliance enhancement initiatives. Moreover, it fosters effective communication across teams and departments, fostering a shared understanding of the organisation's cybersecurity posture.

Continuous Improvement The integrated graph serves as a foundational model for ongoing monitoring efforts, enabling organisations to track changes in risk profiles, identify emerging threats and monitor compliance status over time. Regular updates help organisations evaluate the effectiveness of risk mitigation measures, track the progress of compliance initiatives, and foster continuous improvement within their cybersecurity program.

Finally, the combined graph serves as an effective stakeholder engagement and reporting tool. It provides a holistic overview of cybersecurity risks, compliance status and ongoing efforts to address both aspects within a visually intuitive model. This visual representation allows for clear communication with executives, board members, auditors, and other stakeholders to promote transparency, accountability, and support of cybersecurity initiatives.

Integrating both the cybersecurity risk graph and compliance graph provides an effective, holistic, and risk-focused way of managing cybersecurity in any organisation. This approach supports the prioritisation of mitigation efforts, improved decision-making processes, and continuous improvement initiatives and facilitates effective communications with stakeholders to strengthen the organisation's overall cybersecurity posture.

7.1 Use of Directed Graphs for Simultaneously Assessing Information Security Risk & Maturity

Directed graphs can be used to simultaneously assess the information security risk and the maturity level of an organisation. Their versatility in mapping intricate relationships and visualising various stages of progression makes them indispensable in providing a holistic view of an organisation's security stance.

Holistic Risk and Maturity Visualisation: Within the structure of a directed graph, nodes can represent identified security risks and varying levels of security maturity. Directed edges signify the cause-and-effect relationships or transitions between risks and maturity stages. Such a representation ensures that stakeholders can visually interpret the existing vulnerabilities and the maturity of the organisation's security measures.

Interlinking Risks and Maturity: The interconnected nature of directed graphs clearly explains how specific risks might impact the organisation's security maturity. For instance, a severe risk might prevent the transition to a higher maturity level, highlighting areas that require immediate attention.

Strategic Planning and Prioritisation: By mapping risks and maturity levels, organisations can make informed decisions about which risks to tackle first and which maturity stages are achievable in the short and long term. Risks that hinder the progression to higher maturity levels can be prioritised.

Maturity Enhancement Pathways: Directed edges can signify potential improvement paths or strategies to help an organisation elevate its security maturity. This provides clear roadmaps for organisations to follow for enhanced security protocols.

Stakeholder Engagement and Communication: A single graph's dual representation of risk and maturity facilitates transparent and efficient communication with stakeholders. With an apparent

visual aid, even those without a technical background can understand the interplay between risks and maturity and the implications for the organisation.

Adaptive Framework for Evolving Threats: The dynamic nature of directed graphs ensures that as security threats evolve or the organisation advances in its maturity journey, these changes can be promptly and seamlessly integrated into the graph. This keeps the assessment tool relevant, reflecting the ever-evolving landscape of information security.

7.2 Leveraging Cybersecurity Controls for Dual-Purposed Assessment of Risk and Measurement of Maturity

In the current shifts within the cybersecurity domain, the imperative of adopting an all-encompassing strategy for risk evaluation and maturity gauging has never been clearer. At the heart of this dual-faceted strategy is the practice of harnessing a singular cybersecurity control for both undertakings.:

Consistency in Evaluation: Using the same control ensures a consistent framework when assessing risk and measuring maturity. This consistency minimises discrepancies, biases, or gaps from utilising disparate tools or methodologies for the two functions.

Resource Efficiency: By leveraging a singular control for both purposes, organisations can achieve a more streamlined process, conserving time, and financial resources. This efficiency can be crucial, especially for organisations with constrained resources.

Holistic Understanding: Employing the same control provides a comprehensive picture of the cybersecurity landscape. It allows stakeholders to see how risks relate to maturity and vice versa, ensuring a clear understanding of the interplay between vulnerabilities and the organisation's ability to address them.

Facilitated Communication: Unified controls simplify the communication process among stakeholders. By referencing a single, consistent control, it becomes easier for teams and departments to discuss findings, strategies, and improvements without the confusion of differing methodologies.

Continuous Improvement Feedback Loop: The dual use of control creates a feedback mechanism where insights gained from risk assessments can inform maturity progression, and insights from maturity measurements can highlight potential risk areas. This feedback loop ensures that the organisation is always in a state of learning and adaptation.

Enhanced Scalability: As organisations grow and evolve, so do their security needs. A unified control framework allows for more effortless scalability, ensuring that risk and maturity assessments can be adapted to fit changing organisational structures and goals.

Reduction in Training Needs: Using the same control means that staff need only be trained on one system or methodology, leading to faster onboarding, less confusion, and a more cohesive approach to cybersecurity across the organisation.

7.2.1 Worked Example 6 – Worked Example 5 – Single Actor, Single Attack, Single Vulnerability, Single Asset with Multiple Controls

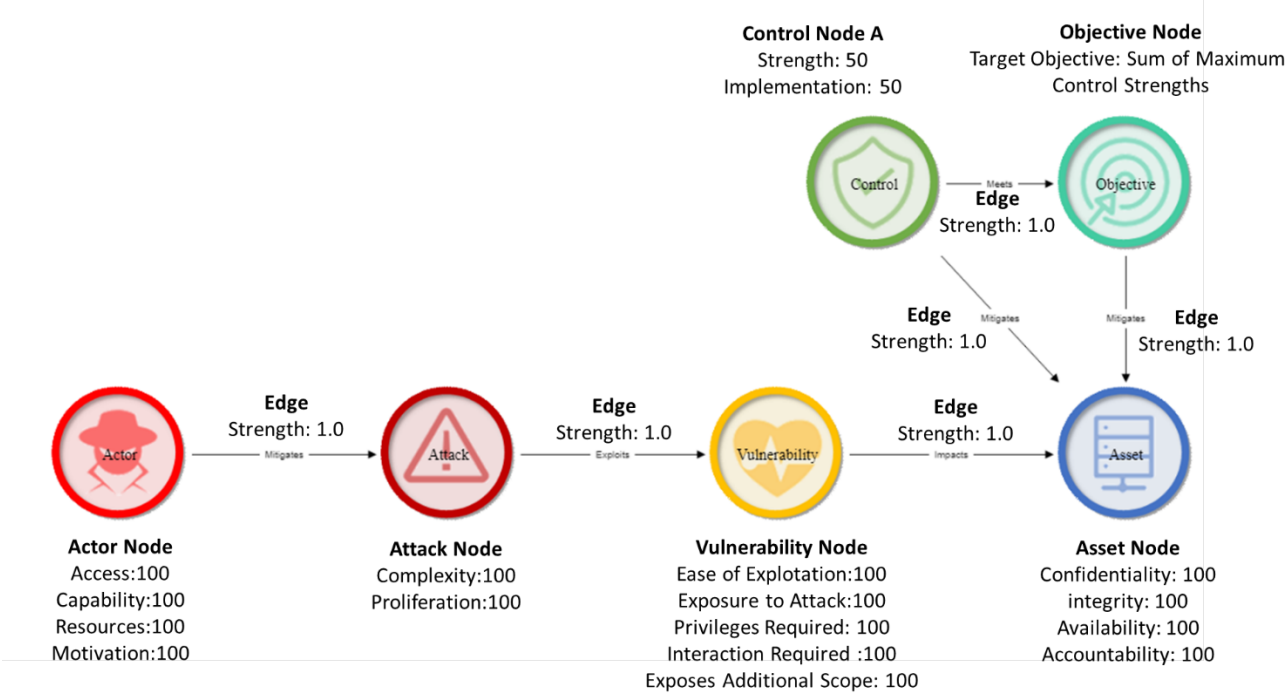

Figure 21 - Worked Example 6 – Single Actor, Single Attack, Single Vulnerability, Single Asset, Single Control, Single Objective

The Threat Actor Value (TAv) is calculated as follows:

$$TA_v = \left(\frac{TA_a + TA_c + TA_r + TA_m}{4}\right)$$

Substitution:

$$TA_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

Result:

$$TA_v = 100$$

The Attack Value (ATv) is calculated as follows:

$$AT_v = \left(\frac{AT_c\ TA_p}{2}\right)$$

Substitution:

$$AT_v = \left(\frac{100 \times 100}{2}\right)$$

Result:

$$AT_v = 100$$

The Threat Value (Tv) from this node is calculated as follows:

$$T_v = \left(\frac{TA_v \times E_v \times AT_V}{100}\right)$$

Substitution:

$$T_v = \left(\frac{100 \times 1.0 \times 100}{100}\right)$$

Result:

$$T_v = 100$$

The Vulnerability Value (Vv) is calculated as follows:

$$V_v = \left(\frac{V_{ee} + V_{ea} + V_{pr} + V_{ir} + V_{es}}{5}\right)$$

Substitution:

$$V_v = \left(\frac{100 + 100 + 100 + 100 + 100}{5}\right)$$

Result:

$$V_v = 100$$

The Likelihood Value (Lv) from this node is calculated as follows:

$$L_v = \left(\frac{T_v\ E_v\ Vv}{100}\right)$$

Substitution:

$$L_v = \left(\frac{100 \times 1.0 \times 100}{100}\right)$$

Result:

$$L_v = 100$$

The Asset Value (Av) is calculated as follows:

$$A_v = \left(\frac{A_c + A_i + A_a + A_{ac}}{4}\right)$$

Substitution:

$$A_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

$$A_v = 100$$

The Risk Value (Rv) is calculated as follows:

$$R_v = \left(\frac{L_v\ E_v\ Av}{100}\right)$$

Substitution:

$$R_v = \left(\frac{100 \times 1.0 \times 100}{100}\right)$$

Result:

$$R_v = 100$$

The Control Value (Cv) is calculated as follows:

$$C_v = \left(\frac{C_s\ C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{50\ \times\ 50}{100}\right)$$

Result:

$$C_v = 25$$

Objective Compliance (Oc) is therefore calculated as:

$$\mathrm{O}_c = \left(\frac{\sum_{i=1}^{n} \mathrm{C}_v \mathrm{E}_v}{\mathrm{O}_s}\right) \times 100$$

Substitution:

$$O_c = \left(\frac{(25 \times 1.0)}{100}\right) \times 100$$

$$O_c = 25$$

The Compensating Control Value (CCv) is calculated as follows:

$$CC_v = max(C_v\ E_v)$$

Substitution:

$$CC_v = 25\ \times 1.0$$

Result:

$$CC_v = 25$$

The Asset Mitigated Value (Amv) is calculated as follows:

$$A_{mv} = Av - CC_v$$

Substitution:

$$A_{mv} = 100 -\ 25$$

Result:

$$Am_v = 75$$

The Target Objective Strength (Os) is calculated as follows when using the "Sum of Maximum Control Strengths":

$$O_S = \sum_{i=1}^{n} a_i$$

Substitution:

$$\mathrm{Os} = 100 \text{ x } 1 = 100$$

Result:

$$\mathrm{O}_S = 100$$

Objective Compliance (Oc) is therefore calculated as

$$\mathrm{O}_c = \left(\sum_{i=1}^{n} \left(\frac{c_v E_v}{\mathrm{O}_s}\right) i\right) \times 100$$

Substitution:

$$\mathrm{O}_c = \left(\frac{25 \times 1.0}{100}\right) \times 100$$

Result:

$$\mathrm{O}_c = 25$$

Equation 18 - Worked Example 5 – Single Actor, Single Attack, Single Vulnerability, Single Asset with Multiple Controls

7.2.2 Worked Example 7 – Single Actor, Single Attack, Single Vulnerability, Single Asset, Multiple Controls, Single Objective

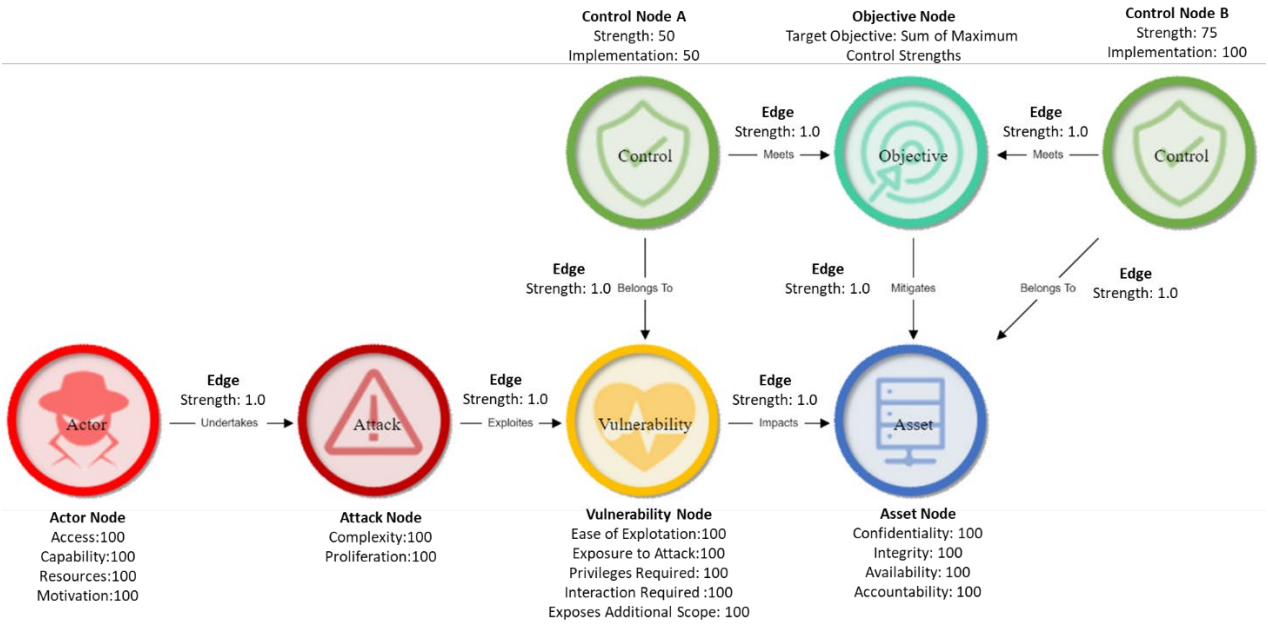

Figure 22 - Worked Example 7 – Single Actor, Single Attack, Single Vulnerability, Single Asset, Multiple Controls, Single Objective

The Threat Actor Value (TAv) is calculated as follows:

$$TA_v = \left(\frac{TA_a + TA_c + TA_r + TA_m}{4}\right)$$

Substitution:

$$TA_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

Result:

$$TA_v = 100$$

The Attack Value (ATv) is calculated as follows:

$$AT_v = \left(\frac{AT_c\ TA_p}{2}\right)$$

Substitution:

$$AT_v = \left(\frac{100 \times 100}{2}\right)$$

Result:

$$AT_v = 100$$

The Threat Value (Tv) from this node is calculated as follows:

$$T_v = \left(\frac{TA_v \times E_v \times AT_V}{100}\right)$$

Substitution:

$$T_v = \left(\frac{100 \times 1.0 \times 100}{100}\right)$$

Result:

$$T_v = 100$$

The Control A Value (Cv) is calculated as follows:

$$C_v = \left(\frac{C_s\ C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{50 \times 50}{100}\right)$$

Result:

$$C_v = 25$$

The Compensating Control A Value (CCv) is calculated as follows:

$$CC_v = max(C_v\ E_v)$$

Substitution:

$$CC_v = 25 \times 1.0$$

Result:

$$CC_v = 25$$

The Vulnerability Value (Vv) is calculated as follows:

$$V_v = \left(\frac{V_{ee} + V_{ea} + V_{pr} + V_{ir} + V_{es}}{5}\right)$$

Substitution:

$$V_v = \left(\frac{100 + 100 + 100 + 100 + 100}{5}\right)$$

$$V_v = 100$$

The Vulnerability Mitigated Value (Vmv) is calculated as follows:

$$V_{mv} = Vv - CC_v$$

Substitution:

$$V_{mv} = 100 - \ 25$$

Result:

$$Vm_v = 75$$

The Likelihood Value (Lv) from this node is calculated as follows:

$$L_v = \left(\frac{T_v \ E_v \ Vmv}{100}\right)$$

Substitution:

$$L_v = \left(\frac{100 \times 1.0 \times 75}{100}\right)$$

Result:

$$L_v = 75$$

The Control B Value (Cv) is calculated as follows:

$$C_v = \left(\frac{C_s \ C_i}{100}\right)$$

Substitution:

$$C_v = \left(\frac{75 \ \times \ 100}{100}\right)$$

Result:

$$C_v = 75$$

The Compensating Control B Value (CCv) is calculated as follows:

$$CC_v = max(C_v\ E_v)$$

Substitution:

$$CC_v = 75\ \times 1.0$$

Result:

$$CC_v = 75$$

The Asset Value (Av) is calculated as follows:

$$A_v = \left(\frac{A_c + A_i + A_a + A_{ac}}{4}\right)$$

Substitution:

$$A_v = \left(\frac{100 + 100 + 100 + 100}{4}\right)$$

$$A_v = 100$$

The Asset Mitigated Value (Amv) is calculated as follows:

$$A_{mv} = Av - CC_v$$

Substitution:

$$A_{mv} = 100 - 25$$

Result:

$$Am_v = 75$$

The Target Objective Strength (Os) is calculated as follows when using the "Sum of Maximum Control Strengths":

$$\mathrm{O}_S = \sum_{i=1}^{n} a_i$$

Substitution:

$$\mathrm{Os} = 100 \text{ x } 2 = 100$$

Result:

$$\mathrm{O}_S = 200$$

Objective Compliance (Oc) is calculated as

$$\mathrm{O}_c = \left(\frac{\sum_{i=1}^{n} \mathrm{C}_v \mathrm{E}_v}{\mathrm{O}_s} \right) \times 100$$

Substitution:

$$\mathrm{O}_c = \left(\frac{(25 \times 1.0) + (75 \times 1.0)}{200} \right) \times 100$$

Result:

236

$$O_c = 50$$

The Risk Value (Rv) is calculated as follows:

$$R_v = \left(\frac{L_v\, E_v\, Av}{100}\right)$$

Substitution:

$$R_v = \left(\frac{75 \times 1.0 \times 100}{100}\right)$$

Result:

$$R_v = 75$$

Chapter 8 – Case Study and Validation

At the inception of this research project, the intricate nature of cyber risk modelling made it evident that a specialised graph modelling mechanism would be pivotal. Several graph modelling methodologies were appraised within the prevalent academic and industry spectrum. However, although proficient in their domain, each method is needed to comprehensively address the complexities of modelling information security risk and compliance, as delineated in this doctoral study's objectives.

This observed gap in existing tools prompted a methodological pivot: the conceptualisation and subsequent creation of a bespoke software solution. The outcome of this endeavour was 'CyConex', a software application developed over approximately 2,500 hours within the Microsoft Visual Studio environment.

CyConex, tailored for the Microsoft Windows platform, is architecturally anchored by several key components:

User Interface: Developed as a Windows x64 desktop application, this interface ensures intuitive and efficient interactions with intricate graph models.

Graph Rendering Mechanism: Employing 'CefSharp', an integration of the Chromium web browser, the application provides a canvas for graph renderings, further augmented by integrating Jscript libraries.

Jscript Libraries: Foundational to CyConex's visualisation capabilities, these libraries, firmly rooted in the paradigms of cytoscape.js and fortified by community-endorsed functional augmentations, facilitate the nuanced graphical depictions central to this research.

C# Libraries: A collection of libraries developed in C# providing the core functionality of the CyConex application.

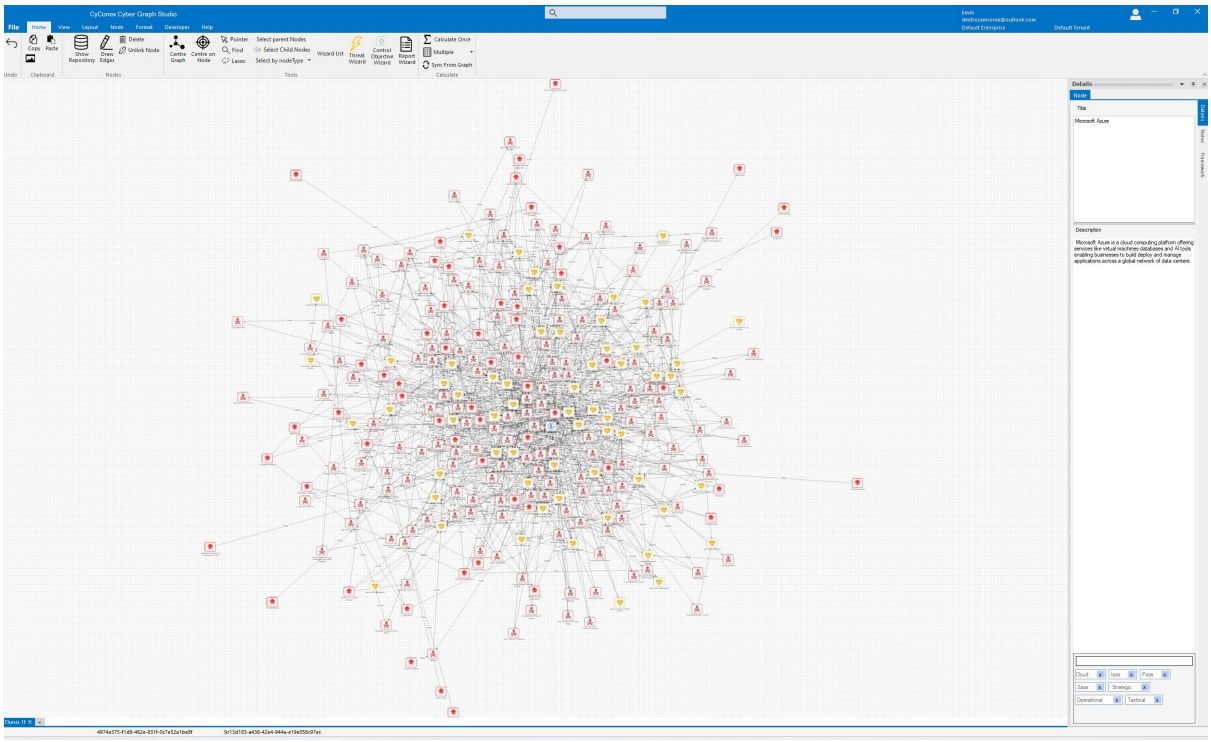

Figure 23 – Screenshot of the CyConex application

The CyConex application offers a sophisticated platform for users to architect, visualise and compute cybersecurity-oriented graphs. Central to its functionality is the capacity to introduce nodes representing various facets of cybersecurity, encompassing Actors, Attacks, Vulnerabilities and Assets. Crucially, the application bestows upon the user granular control, permitting the configuration of individual node attributes, thereby facilitating nuanced representation.

Furthermore, CyConex supports the integration of directed edges amongst nodes. Beyond mere representation, this facilitates embedding specific parameters, encapsulating the intricate dynamics and relationships inherent in cybersecurity constructs.

Once the user has constructed a graph, CyConex undertakes the required calculations. These calculations analyse each node and inter-node relationship, providing the user with an incisive analysis of cybersecurity risks and compliance encapsulated within the graph.
The computational sequence adopted by the application can be described at a high level as follows:

Node Parameter Evaluation: Derive the preliminary scores (e.g., Attack Value) for every node in the graph, contingent on their user-defined configurations.

Pathway Identification: Ascertain all valid paths within the graph utilising a breadth-first methodology.

Pathway Prioritisation: Sequentially process each pathway, commencing with the longest path and culminating with the shortest.

Node Impact Computation: Calculate each node's resultant value, such as the 'Attack Mitigated Value'.

8.1 Case Study

This case study aims to validate the proposed directed graph framework for analysing information security risks. The study evaluates the framework’s practical applicability, scalability, and effectiveness in a real-world organisational setting, specifically within the government sector. This project intends to bridge the gap between theoretical models and their real-world applicability, ensuring that the framework addresses dynamic, evolving threats faced by defence agencies.

8.2 Case Study Scenario

Scenario Overview

Organisation: A national UK government department responsible for safeguarding classified information and coordinating military operations. The organisation handles distributed operations across a national and international estate with tens of thousands of employees and contractors.

Challenges: The organisation must balance financial and operational constraints with operational needs. It has adopted a public cloud for low-classification workloads. Highly capable threat actors will target the organisation's information and services within the public cloud.

8.2 Case Study Objectives

The objectives of the case study were to:

Model the organisation’s cybersecurity risk-directed graphs.

Identify high-risk areas and prioritise mitigation strategies.

Enhance situational awareness and streamline decision-making for both tactical and strategic security operations.

8.3 Approach

The case study took the following approach:

Asset and Scope Definition

This initial stage is critical for establishing a successful cyber risk assessment. It ensures that all stakeholders have a shared understanding of the objectives, scope, and approach. Without a well-defined framework and boundaries, the assessment can lack focus, leading to inefficiencies or gaps in identifying and managing risks.

Identify Potential Vulnerabilities Impacting the Asset

Assessing vulnerabilities is foundational in understanding where an organisation is most susceptible to cyber-attacks. This process involves systematically identifying, analysing, and prioritising weaknesses in systems, applications, methods, and configurations. By understanding vulnerabilities comprehensively, the organisation can implement targeted mitigation strategies to reduce risk.

Identify Potential Attacks Exploiting Vulnerabilities

Identifying potential attacks that exploit vulnerabilities is critical in the cyber risk assessment. This involves analysing known weaknesses within the organisation’s systems, processes, or infrastructure and determining how threat actors can leverage them to execute attacks. By connecting vulnerabilities to potential attack methods, organisations can better understand their exposure and prioritise mitigation strategies.

Identify Potential Threat Actors

Understanding the profile and motivations of potential threat actors is a critical component of a comprehensive cyber risk assessment. Threat actors are individuals, groups, or entities capable of exploiting vulnerabilities to achieve malicious objectives. Identifying and analysing these actors helps organisations anticipate and mitigate potential attacks.

Pre-Mitigation Risk Assessment

A pre-mitigation risk assessment focuses on evaluating risks before implementing controls or mitigations. This stage establishes a baseline understanding of the organisation's inherent risk exposure and identifies critical vulnerabilities, attacks, and potential impacts.

The process involves the following key steps:

- Identify Inherent Risks
- Estimate Likelihood and Impact
- Map Attacks to Vulnerabilities
- Calculate Initial Risk Levels

Identifying Controls to Mitigate Attacks

Mitigating attacks requires implementing technical, administrative, and physical controls designed to prevent, detect, respond to, or recover from security incidents. Identifying the appropriate controls involves aligning them with the vulnerabilities, threats, and organisational context uncovered during the cyber risk assessment. These controls are categorized based on their objectives and functionalities.

In any cybersecurity assessment, it is essential to identify and categorise the primary assets that will be the focus of the evaluation. The assessment focused on the Microsoft Azure public cloud service in this case study.

8.4 Asset and Scope Definition

The assessment's Scope, or Target of Evaluation, was Microsoft Azure. Microsoft Azure is a cloud computing platform and service created by Microsoft. It offers a wide range of computing, analytics, storage, and networking services. Users can choose and configure these services to meet their needs, whether building new applications or running existing ones in the cloud.

Azure provides solutions for various industries and uses cases, such as AI and machine learning, app development, data transformation, and cloud migration. It also supports hybrid and multi-cloud environments, allowing businesses to integrate their on-premises infrastructure with Azure.

A graph node representing Microsoft Azure was added to the graph.

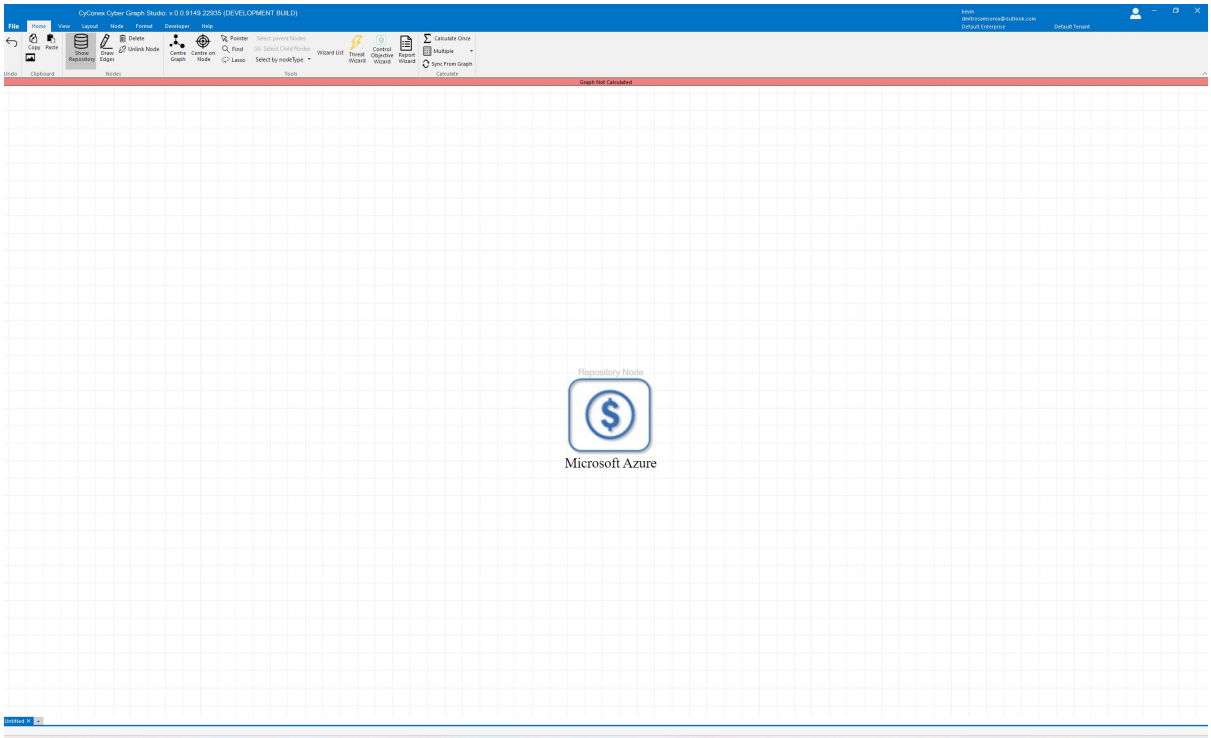

Figure 24 - Case Study - Asset Node Added to Graph

8.5 Identify Potential Vulnerabilities Impacting the Asset

The next stage of the assessment process was to identify potential vulnerabilities that may impact the Microsoft Azure asset.

Identifying Vulnerabilities Impacting Microsoft Azure as Part of a Cyber Risk Assessment

When conducting a cyber risk assessment for environments leveraging Microsoft Azure, it is critical to identify vulnerabilities specific to the platform. As a cloud service provider, Microsoft Azure operates under a shared responsibility model, where Azure is responsible for the security of the cloud infrastructure, and customers are responsible for securing their data, applications, and configurations within the cloud.

A thorough understanding of the Azure environment and architecture is critical for identifying vulnerabilities and assessing risks. This step ensures that all assets, services, and configurations within the Azure ecosystem are accounted for, enabling a focused and effective assessment of potential security gaps.

The activity involved reviewing Azure-specific threat intelligence to identify and mitigate vulnerabilities and threats that could impact an organisation’s cloud environment. This involves leveraging various intelligence sources to gain insights into known vulnerabilities, attack trends, and emerging threats specific to Microsoft Azure services.

Based on these activities, a list of 61 potential vulnerabilities was identified, as detailed in the following table (Full descriptions of the vulnerability can be found in the appendices A1):

API Key Exposure	Inadequate Scalability
API Misconfiguration	Inadequate Secrets Management

Backup Failures	Inadequate Security Group Rules
Broken Authentication and Session Management	Inadequate VLAN Segmentation
Broken Function Level Authorization	Incomplete Visibility into Cloud Usage
Bypassed URL Filtering	Insecure Access Control Policies
Certificate Validation Flaws	Insecure API
Cloud Security Misconfigurations	Insecure API Access
Container Image Vulnerabilities	Insecure API Endpoints
Container Security Flaws	Insecure API Exposure
Cross-Account Access Misconfigurations	Insecure API Gateways
Cross-Region Data Replication Risks	Insecure API Management
Cryptographic Flaws	Insecure Default Settings
Data Leakage through Misconfigured Storage	Insecure DevOps Practices
Data Loss from Accidental Deletion	Insecure Handling of User Data
Default Credentials	Insecure Key Management
Environment Variable Manipulation	Insecure Key Management Practices
Excessive Privileges	Insecure OAuth Implementations
Exposed Secrets and Keys	Insecure Permissions
Failure to Implement Secure Coding Practices	Insecure Remote Management Access
Improper Authentication	Insecure REST API Configurations
Improper Identity and Access Management	Insecure Storage Configurations
Inadequate Data Backup and Recovery	Insecure Third-Party Components

Inadequate Encryption of Data at Rest	Insecure Third-Party Libraries
Inadequate Encryption Strength	Insecure Transport Layer Security
Inadequate IAM Policies	Insecure VM Migration
Inadequate Input Validation	Insufficient Audit Trail
Inadequate Monitoring and Alerting	Insufficient Authorization
Inadequate Protection Against Insider Threats	Insufficient Controls for Infrastructure as a Service (IaaS) Security
Inadequate Resource Isolation	Insufficient DDoS Protection
	Insufficient Incident Response Procedures

Table 17 – Case Study Potential Azure Vulnerabilities

Each of the 61 vulnerabilities was added to the graph as a Vulnerability Node, and a relationship was created from the Vulnerability to the Asset.

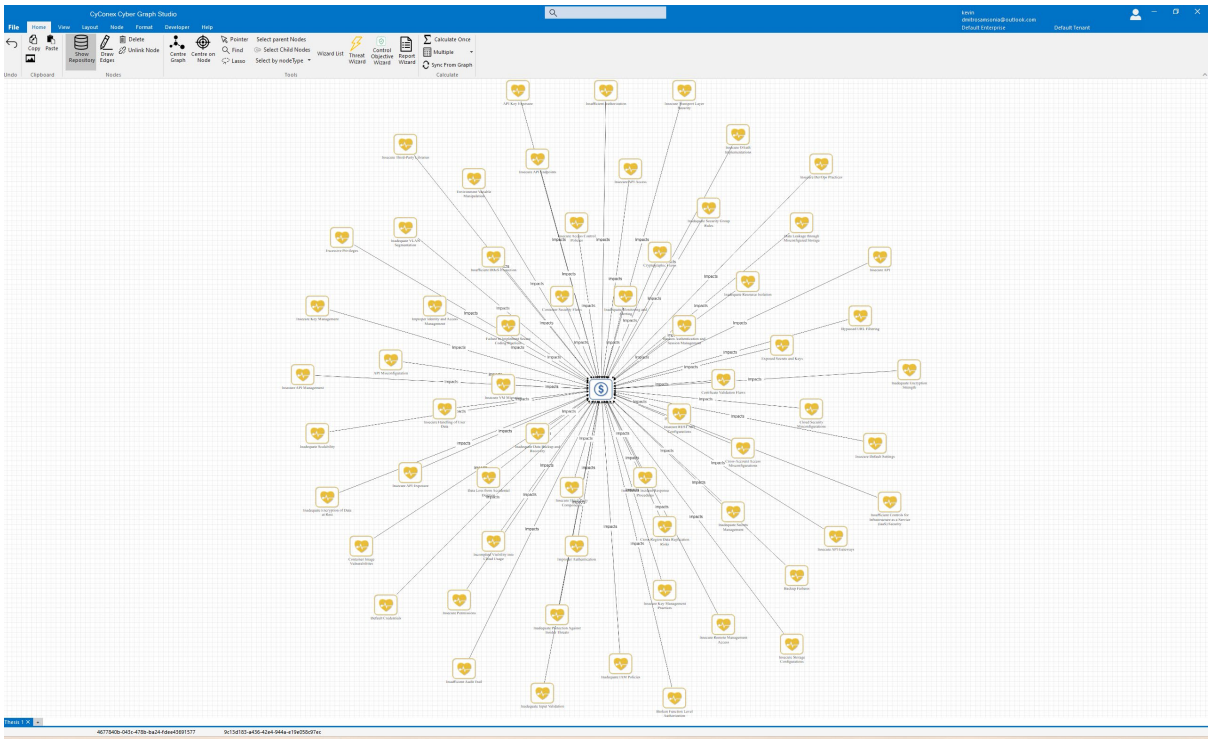

Figure 25 – Case Study Vulnerability Nodes Added to Graph

For each Vulnerability, a distribution range was assigned to each of the Node attributes of:

Ease of Exploitation

Interaction Required

Exposes Additional Scope

Privileges Required

The distribution value was within the range of 0 to 100 and reflected the specific characteristics of the Vulnerability.

8.6 Identify Potential Attacks Exploiting Vulnerabilities

The subsequent stage of the assessment process focused on identifying potential attacks that could exploit vulnerabilities within the Microsoft Azure environment. This critical step builds on the earlier identification of vulnerabilities, aiming to analyse how threat actors could leverage these weaknesses to compromise the organization's assets, disrupt operations, or exfiltrate sensitive data.

When conducting a cyber risk assessment for Microsoft Azure, it is essential to understand the shared responsibility model. Under this model:

Microsoft Azure secures the cloud infrastructure, including physical data centres, networking, and foundational services.

The Customer is responsible for securing their data, applications, identity management, and configurations within the cloud.

This stage emphasises how vulnerabilities in the customer-managed components of the Azure environment could be exploited through specific attack methods. By applying the intelligence gathered, potential attack scenarios exploiting identified vulnerabilities were modelled to assess their feasibility and impact:

Attack Pathways:

Identifying how threat actors could gain access to Azure resources, such as exploiting weak access controls or misconfigured permissions. Evaluating how vulnerabilities could be chained together, such as exploiting a misconfigured storage account to access sensitive data, which could then be used for privilege escalation.

Common Exploitation Methods:

Credential Theft: Exploiting weak passwords or phishing attacks to gain access to Azure Active Directory accounts.

Privilege Escalation: Leveraging misconfigured roles or policies to gain higher-level access.

Misconfigured Storage: Gaining unauthorised access to sensitive data due to improper access controls on Azure Storage Blobs or Files.

Denial-of-Service Attacks: Exploiting vulnerabilities in exposed services to disrupt availability.

Advanced Scenarios:

Assessing how an attacker might use compromised Azure Functions or Logic Apps to execute malicious payloads or automate attacks.

Cloud Platforms: This encompasses the foundational services and infrastructure offered by the CSP. The security of this layer, including the physical hardware, network infrastructure and core software, is typically the responsibility of the CSP.

Cloud Workloads refer to the applications, data and services deployed by the cloud consumer on the CSP's platform. Ensuring these workloads' security, compliance, and management falls on the cloud consumer.

To aptly represent the security dynamics and maturity, we developed a model that:

Distinctively Separates: The cloud platforms depict the foundational services and infrastructures. The cloud workloads represent consumer-driven applications, data, and services.

Mapping Relationships: Each element (both platform and workload) is directly mapped to its specific cloud instance. This precise mapping ensures that each instance's roles, responsibilities, and potential vulnerabilities are identifiable and addressable.

Based on this assessment criteria the following 161 potential attacks were identified (Full descriptions of the vulnerabilities can be found in the appendices A2):

Abuse Elevation Control Mechanism	Exploitation For Privilege Escalation
Account Access Removal	Exploitation Of Remote Services
Account Discovery	Exploitation Of Remote Services- Ssh Hijacking
Account Manipulation	External Remote Services
Activation Of Payloads	Extra Window Memory Injection
Appcert Dlls	Fallback Channels
Applescript	File And Directory Discovery
Application Access Token	File And Directory Permissions Modification
Application Layer Protocol	File Deletion
Application Layer Protocol- Dns	Firmware Corruption
Application Layer Protocol- HttpS	Forced Authentication
Application Layer Protocol- Websocket	Hardware Additions
Application Window Discovery	Hide Artifacts
Automated Collection	Hijack Execution Flow

Automated Collection- Input Capture	Hijack Execution Flow- Dll Search Order Hijacking
Automated Exfiltration	Hijack Execution Flow- Dll Side-Loading
Bash History	Implant Container Image
Boot Or Logon Autostart Execution	Indicator Blocking
Boot Or Logon Initialization Scripts	Indicator Removal on Host
Browser Extensions	Inhibit System Recovery
Brute Force	Input Capture
Clipboard Data	Input Prompt
Cloud Infrastructure Discovery	Internal Spearphishing
Cloud Service Dashboard	Inter-Process Communication
Cloud Service Discovery	Kerberoasting
Cloud Storage Object Discovery	Keychain
Command And Scripting Interpreter	Lateral Tool Transfer
Create Account	Lsass Driver
Create Or Modify System Process	Man In the Browser
Credentials In Files	Man-In-The-Browser
Credentials In Registry	Manipulate Network Traffic
Data Compressed	Manipulation Of Insecure Content
Data Destruction	Masquerading
Data Encoding	Multi-Hop Proxy
Data Encrypted	Multi-Stage Channels
Data Encrypted For Impact	Native Api
Data From Cloud Storage Object	Network Denial of Service

Data From Cloud Storage Object- Automated Collection	Network Service Discovery
Data From Information Repositories	Network Service Scanning
Data From Information Repositories- Automated Collection	Network Share Discovery
Data From Local System	Network Sniffing
Data From Network Shared Drive	Non-Application Layer Protocol
Data From Network Shared Drive- Automated Collection	Non-Standard Port
Data Manipulation	Non-Standard Port- Tcp/Udp
Data Manipulation- Stored Data Manipulation	Os Credential Dumping
Data Obfuscation	Password Spraying
Data Staged	Permission Groups Discovery
Data Transfer Size Limits	Phishing
Data Transfer Size Limits- Email Collection	Port Knocking
Deactivate Security Software	Portable Executable Injection
Deobfuscate/Decode Files Or Information	PowerShell
Digital Certificate Validation	Private Keys
Direct Volume Access	Protocol Tunnelling
Distributed Component Object Model	Remote Access Software
Domain Trust Discovery	Remote Desktop Protocol
Drive-By Compromise	Remote Service Session Hijacking
Dynamic Resolution	Remote Services

Dynamic Resolution- Domain Generation Algorithms	Remote Services- Ssh
Dynamic Resolution- Fast Flux Dns	Remote System Discovery
Email Collection	Resource Hijacking
Encrypted Channel	Screen Capture
Endpoint Denial Of Service	Security Software Discovery
Endpoint Denial of Service- Resource Hijacking	Server Software Component
Endpoint Denial of Service- Service Exhaustion Flood	Shared Modules
Event Triggered Execution	Signed Binary Proxy Execution
Event Triggered Execution- Application Shimming	Spearphishing Link
Event Triggered Execution- Component Object Model Hijacking	Steal Application Access Token
Event Triggered Execution- Screensaver	System Binary Proxy Execution
Execution Guardrails	System Information Discovery
Execution Through Api	System Owner User Discovery
Exfiltration Over Alternative Protocol	System Services
Exfiltration Over Alternative Protocol- Domain Name System	System Shutdown Reboot
Exfiltration Over Alternative Protocol- Secure Shell	Taint Shared Content
Exfiltration Over C2 Channel	Transfer Data to Cloud Account
Exfiltration Over Physical Medium	Trusted Relationship
Exfiltration Over Usb	Unencrypted Communication
Exfiltration Over Web Service	User Execution

Exploit Public-Facing Application	User Execution- Malicious File
Exploitation For Client Execution	User Execution- Malicious Link
Exploitation For Credential Access	Valid Accounts
	Virtualization Sandbox Evasion

Table 18 – Case Study - Identified Attacks

An algorithm was applied to the attack nodes on the graph to proportionally size the node relative to the value of the attack of the node. This allows simple identification of the nodes with the greatest value, whilst also providing immediate feedback on the number of and scale of attacks present on the graph.

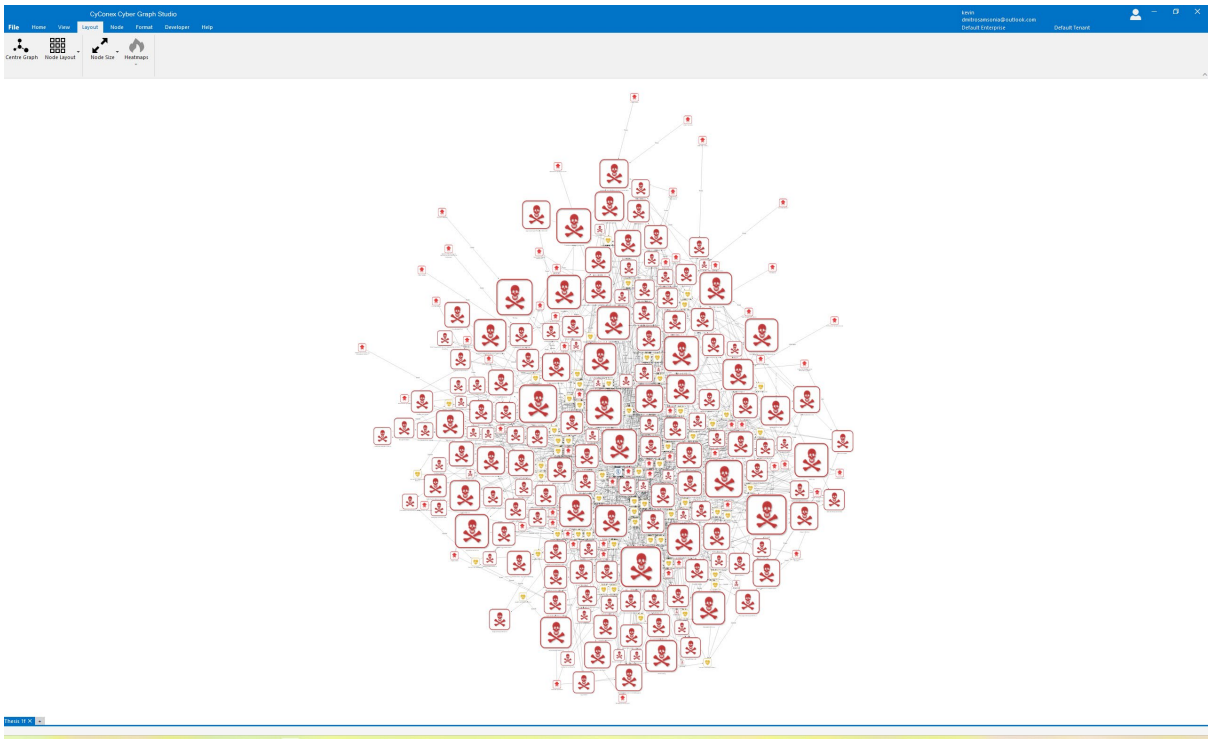

Figure 26 – case Study - Nodes Scales for Attack

8.7 Pre-Mitigation Risk Assessment

The next stage of the assessment was to undertake a Pre-Mitigation Risk Assessment, which is used to determine the risk faced by the target of evaluation asset.

A pre-mitigation risk assessment focuses on evaluating risks before implementing controls or mitigations. This stage establishes a baseline understanding of the organisation's inherent risk exposure and identifies critical vulnerabilities, attacks, and potential impacts.

The process involves the following key steps:

Identify Inherent Risks:

> Analyse the vulnerabilities and attacks associated with assets in their current state without considering any controls. This step helps determine the raw risk level posed by existing weaknesses.

Estimate Likelihood and Impact:

> Assess the likelihood of potential attacks exploiting identified vulnerabilities and the associated impacts on confidentiality, integrity, and availability. This estimation provides insight into the severity of risks.

Map Attacks to Vulnerabilities:

> Establish a clear connection between identified vulnerabilities and threat actors' methods to exploit them. This mapping helps visualise the organisation's exposure and identify high-priority risks.

Calculate Initial Risk Levels:

> Assign risk levels based on likelihood and impact. These levels represent the organisation's risk profile in the absence of mitigations.

Monte Carlo Analysis

As part of the Pre-Mitigation Risk Assessment calculations a Monte Carlo analysis is used. A Monte Carlo analysis is a computational technique to understand the impact of uncertainty, variability, or randomness in a system, model, or process. It relies on running many simulations using randomly generated inputs within specified ranges to estimate the possible outcomes and their probabilities.

Key Elements of Monte Carlo Analysis:

Random Sampling:

- Inputs to the model are represented as probability distributions (e.g., normal, uniform, or exponential) instead of fixed values.
- Random samples are drawn from these distributions for each simulation run.

Simulation:

The model or system is executed multiple times (often hundreds or thousands of iterations) with different sets of random inputs.

Results Aggregation:

- The outputs from all simulations are collected and analysed to provide a distribution of outcomes.

The results help stakeholders understand the range of scenarios, assess risks, and make informed decisions under uncertainty.

In this assessment, a Monte Carlo simulation consisting of 100 iterations was performed to analyse the behaviour and outcomes of the graph-based model. During each iteration, the attributes of nodes and edges within the graph were randomly selected from a predefined statistical distribution. These

distributions could represent characteristics such as probabilities, weights, or other metrics relevant to the nodes and edges, ensuring that the simulation captured a range of scenarios.

The randomly assigned attributes were then used to initialize or "seed" the graph for computation, allowing the simulation to reflect the stochastic nature of real-world conditions. For each seeded graph, computations were carried out to evaluate various metrics or values of interest derived from the graph structure.

On completion of each iteration, the results, whether node values, edge weights, or other graph properties, were aggregated into a cumulative distribution. This aggregate distribution provided a comprehensive view of the range and likelihood of different outcomes across the simulation. From this final distribution, the mode (the most frequently occurring value) of any required metric was selected, offering a representative value for decision-making or analysis. This approach enabled robust insights by accounting for variability and uncertainty inherent in the input parameters and graph dynamics.

Likelihood

A fundamental component of a risk calculation is Likelihood, which defines the probability of the risk materialising. In this graph model, Likelihood is a result of the attributes of the Threat Actor, Attack and Vulnerability.

Based on the calculations from the graph model, the following table lists the Likelihood of each vulnerability impacting the target of evaluation Asset. The Likelihood value is a range of between 0 and 100, with 0 meaning the vulnerability will never impact the asset, to 100 meaning the vulnerability will impact the asset.

The following table shows the top 10 Vulnerability Likelihood, (A complete table can be found in the Appendices A3):

Vulnerability	Asset	Likelihood
Default Credentials	Microsoft Azure	60
Broken Authentication and Session Management	Microsoft Azure	53
Cross-Account Access Misconfigurations	Microsoft Azure	50
Insecure Remote Management Access	Microsoft Azure	50
Exposed Secrets and Keys	Microsoft Azure	49
Improper Identity and Access Management	Microsoft Azure	49
Insecure Access Control Policies	Microsoft Azure	48
Insecure API Access	Microsoft Azure	48
Inadequate IAM Policies	Microsoft Azure	47
API Key Exposure	Microsoft Azure	46
Inadequate Secrets Management	Microsoft Azure	45

Table 19 – Vulnerability Likelihood

An algorithm was applied to the vulnerability nodes on the graph to proportionally size the node relative to the likelihood impact of the node to the asset. This allows simple identification of the nodes with the greatest impact, whilst also providing immediate feedback on the number of and scale of likelihood present on the graph.

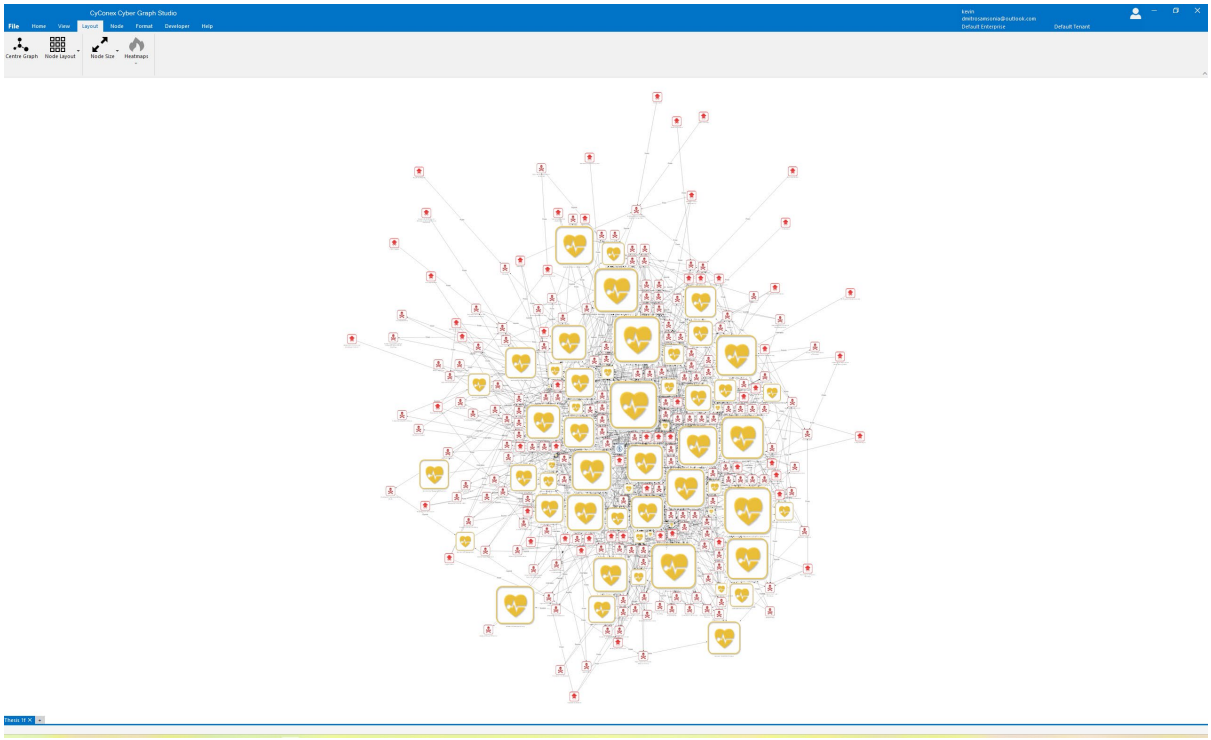

Figure 27 – Case Study Nodes Scaled for Likelihood

Pre-Mitigated Risk

The final pre-mitigation risk calculation is a straightforward calculation of (Likelihood x Asset Impact) /100.

Based on the calculations from the graph model, the following table lists the pre-mitigation risk of each vulnerability impacting the target of evaluation Asset. The risk value is a range of between 0 and 100; with 0 meaning the vulnerability does not impact the asset in any way, to 100 meaning the vulnerability completely impacts the asset.

Confidentiality Risk

The following table lists the top 10 pre-mitigation risks to the Confidentiality of the Asset. The full list of Confidentiality Risks can be found in the appendices.

Vulnerability	Asset	Likelihood	Impact	Risk Value

Default Credentials	Microsoft Azure	41	47	19
Insufficient Authorization	Microsoft Azure	34	47	16
Insecure Remote Management Access	Microsoft Azure	33	47	16
Inadequate Secrets Management	Microsoft Azure	32	47	15
Insecure Access Control Policies	Microsoft Azure	31	47	15
Inadequate IAM Policies	Microsoft Azure	31	47	15
Broken Authentication and Session Management	Microsoft Azure	31	47	15
Improper Identity and Access Management	Microsoft Azure	30	47	14
Cross-Account Access Misconfigurations	Microsoft Azure	30	47	14
API Key Exposure	Microsoft Azure	26	47	12

Table 20 - Pre-Mitigated Risk to Confidentiality

Integrity Risk

The following table lists the top 10 pre-mitigation risks to the Integrity of the Asset. The full table of Integrity Risks can be found in the appendices.

Vulnerability	Asset	Likelihood	Impact	Risk Value
Default Credentials	Microsoft Azure	37	48	18

Cross-Account Access Misconfigurations	Microsoft Azure	27	48	13
Insecure Remote Management Access	Microsoft Azure	26	48	12
Improper Identity and Access Management	Microsoft Azure	25	48	12
Inadequate IAM Policies	Microsoft Azure	25	48	12
Inadequate Secrets Management	Microsoft Azure	24	48	12
Insecure Access Control Policies	Microsoft Azure	23	48	11
Insufficient Authorization	Microsoft Azure	22	48	11
Broken Authentication and Session Management	Microsoft Azure	22	48	11
Insecure DevOps Practices	Microsoft Azure	21	48	10

Table 21 - Pre-Mitigated Risk to Integrity

Availability Risk

The following table lists the top 10 pre-mitigation risks to the Availability of the Asset. The full table of Availability Risks can be found in the appetencies.

Vulnerability	Asset	Likelihood	Impact	Risk Value
Inadequate Data Backup and Recovery	Microsoft Azure	26	40	10

Insufficient DDoS Protection	Microsoft Azure	25	40	10
Insufficient Incident Response Procedures	Microsoft Azure	22	40	9
Insecure Remote Management Access	Microsoft Azure	19	40	8
Improper Identity and Access Management	Microsoft Azure	18	40	7
Insecure Access Control Policies	Microsoft Azure	17	40	7
API Key Exposure	Microsoft Azure	16	40	6
Inadequate IAM Policies	Microsoft Azure	16	40	6
Backup Failures	Microsoft Azure	16	40	6
Broken Authentication and Session Management	Microsoft Azure	16	40	6

Table 22 - Pre-Mitigated Risk to Availability

Accountability Risk

The following table lists the top 10 pre-mitigation risks to the Accountability of the Asset. The full table of Accountability risks can be found in the appendices.

Vulnerability	Asset	Likelihood	Impact	Risk Value
Default Credentials	Microsoft Azure	41	36	15
Insufficient Authorization	Microsoft Azure	34	36	12

Insecure Remote Management Access	Microsoft Azure	33	36	12
Inadequate Secrets Management	Microsoft Azure	32	36	12
Insecure Access Control Policies	Microsoft Azure	31	36	11
Inadequate IAM Policies	Microsoft Azure	31	36	11
Broken Authentication and Session Management	Microsoft Azure	31	36	11
Improper Identity and Access Management	Microsoft Azure	30	36	11
Cross-Account Access Misconfigurations	Microsoft Azure	30	36	11
API Key Exposure	Microsoft Azure	26	36	9

Table 23 - Pre-Mitigated Risk to Accountability

The following image illustrates the distribution of Impact and Likelihood of risk present with the graph:

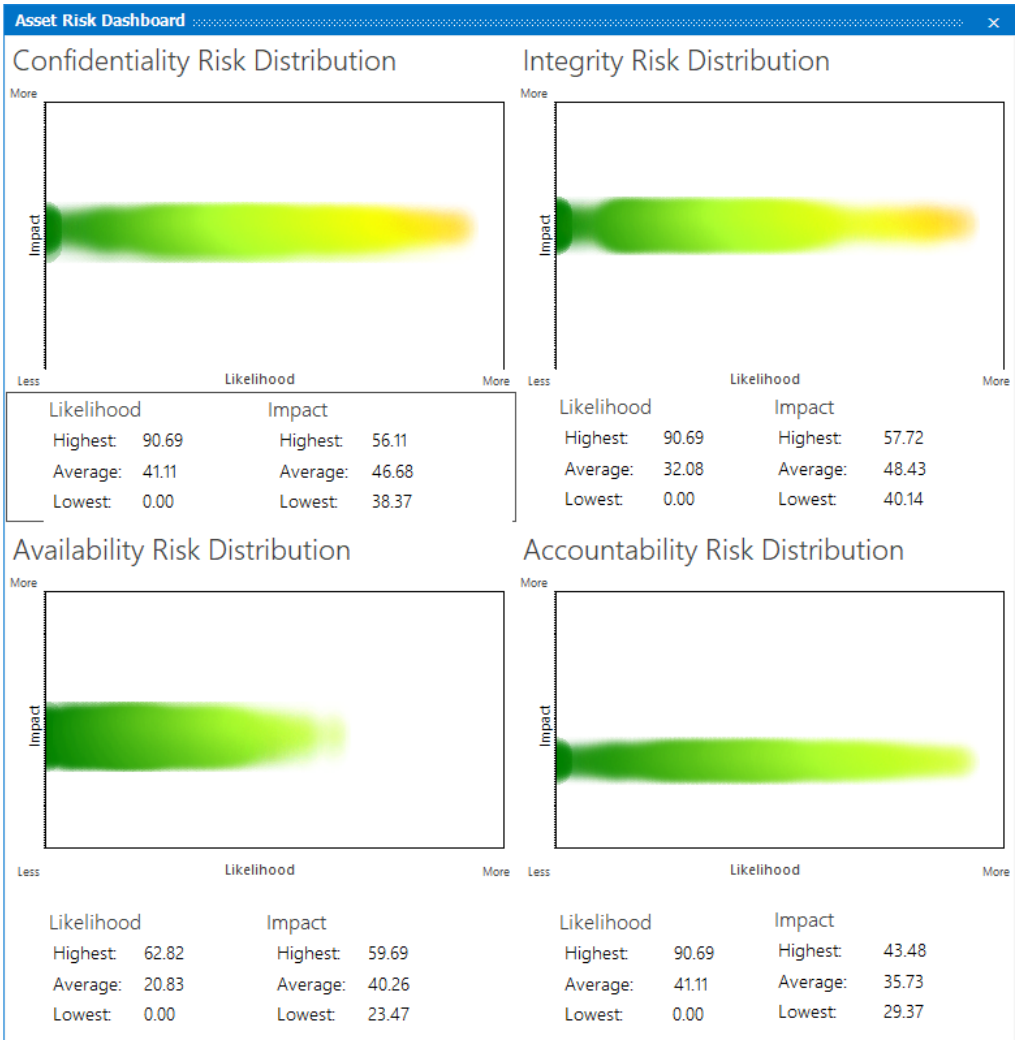

Figure 28 - case Study - Pre-Mitigation Risk Distribution

8.8 Identifying Controls to Mitigate Attacks

The subsequent stage of the cybersecurity risk and maturity assessment process focuses on identifying controls to mitigate attacks by addressing vulnerabilities and threats to the evaluation asset's target.

This stage builds upon the earlier identification of vulnerabilities and threat actors, aiming to map specific security controls to effectively mitigate potential attack methods. The controls are evaluated for their ability to reduce risk exposure, enhance resilience, and ensure compliance with established security frameworks.

Understanding the Role of Controls in Mitigation

Preventive Controls:

Designed to reduce the likelihood of an attack by eliminating vulnerabilities or deterring threat actors. Examples include implementing strong access controls, multi-factor authentication (MFA), and encryption.

Detective Controls:

Focus on identifying attacks in progress or after they have occurred, using tools such as intrusion detection systems (IDS) and security information and event management (SIEM) solutions.

Corrective Controls:

Restore systems or data to a secure state to minimise the impact of successful attacks. Examples include incident response plans and automated remediation.

This stage involves systematically mapping security controls to potential attack methods identified in the earlier phases of the assessment. The mapping process ensures a targeted and prioritised approach to implementing controls based on risk severity.

Based on this assessment criteria the following 169 potential attacks were identified.

80.X Authentication	Control '80.X Authentication' ensures that only authorized users can access systems by verifying their identity through secure methods, such as passwords, biometrics, or multi-factor authentication.
AAA (Authentication)	AAA (Authentication) verifies user identities before granting access to systems or data, ensuring only authorized individuals can access resources, thereby

	protecting against unauthorized access and potential security breaches.
Acceptable Use Policy	An Acceptable Use Policy outlines the permissible and prohibited activities for users accessing an organization's IT resources, ensuring compliance with security protocols and protecting the organization's data and systems.
Access Advisor	Access Advisor is a tool that analyses AWS IAM permissions, helping identify unused permissions to optimize security by recommending removal of unnecessary access, thereby reducing potential attack surfaces.
Access Analyzer	Access Analyzer is an AWS tool that identifies and analyses resource access policies, helping ensure resources are not unintentionally shared publicly or with unauthorized entities, enhancing security and compliance.
Access Anomaly Detection	Access Anomaly Detection identifies unusual access patterns or behaviours, such as atypical login times or locations, to detect potential unauthorized access or compromised accounts, enhancing security by alerting administrators to investigate.
Access Control	Access Control is a security measure that restricts unauthorized users from accessing systems, data, or resources, ensuring only authorized individuals can perform specific actions based on predefined permissions.

Access Control for Boot Configuration	"Access Control for Boot Configuration" restricts unauthorized changes to system boot settings, ensuring only authorized personnel can modify boot parameters, preventing unauthorized access or tampering during system startup.
Access Control for Code Repositories	Access Control for Code Repositories ensures only authorized users can access, modify, or manage code repositories, protecting intellectual property and preventing unauthorized changes or data breaches.
Access Control for Cryptographic Keys	Implement strict access controls for cryptographic keys, ensuring only authorized personnel can access, manage, or use them, to prevent unauthorized decryption or encryption and maintain data confidentiality and integrity.
Access Control for DNS Servers	Implement strict access controls for DNS servers to prevent unauthorized modifications, ensure only authorized personnel can manage configurations, and use logging to monitor and audit access activities for security compliance.
Access Control for Firmware Updates	This control ensures only authorized personnel can initiate or approve firmware updates, preventing unauthorized modifications and maintaining the integrity and security of the system's firmware.
Access Control for Patch Management Tools	Restrict access to patch management tools to authorized personnel only, ensuring secure authentication and role-

	based permissions to prevent unauthorized modifications and maintain system integrity and security.
Access Control Lists (ACL) Review	An Access Control Lists (ACL) Review involves regularly examining and updating permissions to ensure only authorized users have access to specific resources, minimizing unauthorized access and potential security breaches.
Access Controls for Backup Data	Implement strict access controls for backup data by using authentication, authorization, and encryption to ensure only authorized personnel can access, modify, or restore data, protecting against unauthorized access and data breaches.
Access Controls for Certificate Authorities	Implement strict access controls for Certificate Authorities to ensure only authorized personnel can manage, issue, or revoke certificates, protecting against unauthorized access and potential compromise of digital certificates.
Access Logging	Access Logging involves recording details of user access to systems and data, including timestamps, user IDs, and accessed resources, to monitor, detect, and investigate unauthorized or suspicious activities.
Access Review and Recertification	Access Review and Recertification involves regularly evaluating user access rights to ensure they align with current job roles, removing unnecessary privileges to minimize security risks and maintain compliance.

Access Token Revocation	Access Token Revocation is a security control that invalidates tokens, preventing unauthorized access by ensuring that compromised or expired tokens cannot be used to access systems or data.
Access Token Scoping	Access Token Scoping limits the permissions and access rights granted to a token, ensuring it only allows actions necessary for specific tasks, thereby minimizing potential security risks and unauthorized access.
Account Monitoring	Account Monitoring involves tracking user account activities to detect unauthorized access, anomalies, or policy violations, ensuring timely alerts and responses to potential security threats and maintaining system integrity.
Account Provisioning and Deprovisioning	Account Provisioning and Deprovisioning ensures timely creation, modification, and removal of user accounts, minimizing unauthorized access and maintaining security by aligning user access with current roles and responsibilities.
Account Review	Account Review involves regularly examining user accounts and access permissions to ensure they are appropriate, identifying anomalies, and removing or updating access to prevent unauthorized access and reduce security risks.
Account Takeover Protection	Account Takeover Protection detects and prevents unauthorized access by monitoring login attempts,

	identifying suspicious activities, and implementing multi-factor authentication to safeguard user accounts from being compromised.
Active Directory Hardening	Active Directory Hardening involves implementing security measures to protect AD infrastructure, including strong authentication, access controls, regular audits, patching, and monitoring to prevent unauthorized access and mitigate potential threats.
Active Directory Monitoring	Active Directory Monitoring involves continuously tracking and analysing AD activities to detect unauthorized access, changes, or anomalies, ensuring the integrity, security, and compliance of the organization's identity management system.
Activity Monitoring and Logging	Activity Monitoring and Logging involves tracking and recording user and system activities to detect, investigate, and respond to security incidents, ensuring accountability and compliance with security policies.
Advanced Persistent Threat (APT) Monitoring	APT Monitoring involves continuous surveillance and analysis of network activity to detect, respond to, and mitigate sophisticated, long-term cyber threats targeting specific entities, ensuring timely identification and neutralization of malicious activities.
Advanced Threat Protection	Advanced Threat Protection (ATP) is a security solution designed to detect, prevent, and respond to sophisticated cyber threats, including zero-day exploits and advanced

	malware, using behavioural analysis, machine learning, and threat intelligence.
Adversarial Testing	Adversarial Testing involves simulating real-world cyber attacks to identify vulnerabilities and weaknesses in systems, enabling organizations to enhance their security posture by proactively addressing potential threats and exploits.
Alerting and Monitoring	"Alerting and Monitoring" involves continuously observing systems and networks to detect suspicious activities, generating real-time alerts for potential threats, and enabling prompt incident response to mitigate security risks.
Anomalous Activity Detection	Anomalous Activity Detection involves monitoring systems for unusual patterns or behaviours that deviate from the norm, indicating potential security threats or breaches, and triggering alerts for further investigation.
Anti-DDoS Services	Anti-DDoS Services protect against Distributed Denial of Service attacks by monitoring traffic, filtering malicious requests, and ensuring network availability and performance through automated response mechanisms and traffic rerouting.
Anti-Phishing Measures	Anti-Phishing Measures involve implementing tools and practices to detect, block, and educate users about phishing attempts, reducing the risk of credential theft and unauthorized access to sensitive information.

Anti-Tamper Mechanisms	Anti-Tamper Mechanisms protect systems and data from unauthorized alterations by detecting, deterring, or delaying tampering attempts, ensuring integrity and authenticity through physical, software, or cryptographic measures.
API Monitoring	API Monitoring involves tracking and analysing API interactions to detect anomalies, ensure performance, and identify security threats, helping to prevent unauthorized access and data breaches in real-time.
application	The "application" control involves implementing security measures within software applications to prevent unauthorized access, data breaches, and vulnerabilities, ensuring secure coding practices, regular updates, and robust authentication mechanisms.
Application Layer Protocol Inspection	Application Layer Protocol Inspection involves analysing and validating protocol-specific traffic at the application layer to detect and block malicious activities, ensuring compliance with expected protocol behaviour and enhancing security.
Application Programming Interface (API) Security	API Security involves protecting APIs from threats by implementing authentication, authorization, input validation, and monitoring to prevent unauthorized access, data breaches, and ensure secure data exchange between applications.

Application Security Testing	Application Security Testing involves evaluating software applications for vulnerabilities, weaknesses, and security flaws to ensure they are secure against threats, protecting data integrity, confidentiality, and availability throughout their lifecycle.
Apply Configuration Management	"Apply Configuration Management" involves establishing and maintaining secure configurations for systems and software, ensuring consistency, reducing vulnerabilities, and managing changes to prevent unauthorized alterations and maintain system integrity.
Apply Data Encryption at Rest	"Apply Data Encryption at Rest" involves encrypting stored data to protect it from unauthorized access, ensuring confidentiality and integrity, even if physical storage devices are compromised or stolen.
Apply Encrypted Mobile App Communication	This control ensures that all data transmitted between mobile applications and servers is encrypted, protecting sensitive information from interception and unauthorized access during communication over networks.
Apply Firmware Security Updates	"Apply Firmware Security Updates" involves regularly updating device firmware to patch vulnerabilities, enhance security features, and protect against exploits, ensuring the integrity and reliability of hardware components.
Apply MFA to Privileged Accounts	Implementing Multi-Factor Authentication (MFA) for privileged accounts enhances security by requiring

	additional verification steps, reducing the risk of unauthorized access and protecting sensitive systems and data from potential breaches.
Apply Patches and Updates	Regularly install software patches and updates to fix vulnerabilities, enhance security, and ensure systems are protected against known threats and exploits, minimizing the risk of unauthorized access or data breaches.
Apply Patches Promptly	"Apply Patches Promptly" involves regularly updating software and systems with the latest security patches to fix vulnerabilities, reduce exposure to threats, and maintain system integrity and protection against exploits.
Apply Transport Layer Security for IoT Devices	Implement Transport Layer Security (TLS) on IoT devices to encrypt data in transit, ensuring confidentiality, integrity, and authentication between devices and servers, mitigating risks of eavesdropping and data tampering.
Attachment Sandboxing	Attachment Sandboxing involves executing email attachments in a controlled, isolated environment to detect and analyse malicious behaviour, preventing threats from reaching the user's system and ensuring network security.
Attack Surface Analysis	Attack Surface Analysis involves identifying and evaluating all potential entry points in a system or network to minimize vulnerabilities and reduce the risk of unauthorized access or exploitation.

Audit Logs for Data Destruction	This control ensures detailed records of data destruction activities, capturing who performed the action, when, and what data was destroyed, to verify compliance and support forensic investigations.
Authentication and Authorization	Authentication verifies user identity, while authorization determines access levels. Together, they ensure only authenticated users can access resources they're permitted to, enhancing security by preventing unauthorized access.
Authentication Mechanisms	Authentication mechanisms verify user identities before granting access to systems or data, using methods like passwords, biometrics, or multi-factor authentication to ensure only authorized users can access sensitive resources.
Automated Backup Processes	Automated Backup Processes ensure regular, scheduled backups of critical data, minimizing human error and reducing data loss risk by securely storing copies offsite or in the cloud for rapid recovery.
Automated Traffic Analysis	Automated Traffic Analysis involves using software tools to monitor and analyse network traffic in real-time, identifying anomalies, potential threats, and ensuring compliance with security policies to protect against cyberattacks.
AWS Shield	AWS Shield is a managed Distributed Denial of Service (DDoS) protection service that safeguards applications

	running on AWS by minimizing downtime and latency during DDoS attacks.
Azure Bastion	Azure Bastion is a managed service providing secure and seamless RDP/SSH access to virtual machines directly through the Azure portal, without exposing them to public internet, enhancing security.
Azure Key Vault	Azure Key Vault is a cloud service that securely stores and manages cryptographic keys, secrets, and certificates, enabling enhanced data protection and streamlined access management for applications and services.
Azure Sentinel	Azure Sentinel is a cloud-native SIEM and SOAR solution that provides intelligent security analytics and threat intelligence for detecting, investigating, and responding to security incidents across enterprise environments.
Backup and Recovery	Backup and Recovery' involves regularly creating copies of data and systems to ensure availability and integrity, enabling restoration after data loss, corruption, or cyber incidents, minimizing downtime and operational impact.
BGP Route Filtering	BGP Route Filtering involves controlling the advertisement and acceptance of BGP routes to prevent route leaks, hijacks, and ensure network stability by allowing only legitimate routes based on predefined policies.

Binary Authorization	Binary Authorization is a security control that ensures only trusted and verified code or binaries are deployed in production environments, preventing unauthorized or malicious software from being executed.
Boot Guard	Boot Guard is a hardware-based security feature that ensures a trusted boot process by verifying the integrity of the system's firmware and bootloader, preventing unauthorized code execution at startup.
Boot Process Logging	Boot Process Logging involves recording all events and activities during a system's startup sequence, enabling detection of unauthorized changes or anomalies, and aiding in forensic analysis and troubleshooting.
Bootloader Update Management	Bootloader Update Management involves securing, validating, and managing updates to the bootloader to prevent unauthorized modifications, ensuring system integrity and preventing malicious code execution during the boot process.
Brute Force Attack Protection	Brute Force Attack Protection involves implementing measures like account lockouts, CAPTCHA, rate limiting, and multi-factor authentication to prevent unauthorized access by systematically guessing passwords or credentials.
Bucket Policies	Bucket Policies are access management configurations for cloud storage buckets, defining permissions and

	conditions for users and services to ensure secure data access and prevent unauthorized actions.
Certificate Authorities	Certificate Authorities (CAs) are trusted entities that issue digital certificates to verify the identity of websites and users, enabling secure, encrypted communications over networks by establishing trust in public key infrastructure.
Certificate Chain Validation	Certificate Chain Validation ensures that a digital certificate is authentic and trustworthy by verifying its chain of trust, from the end-entity certificate up to a trusted root certificate authority.
Certificate Lifecycle Management	Certificate Lifecycle Management involves overseeing the issuance, renewal, revocation, and expiration of digital certificates to ensure secure communication, authentication, and data integrity across networks and systems.
Change Default Credentials	"Change Default Credentials" involves replacing factory-set usernames and passwords with unique, strong credentials to prevent unauthorized access and reduce the risk of exploitation by attackers using default settings.
Clipboard Management Software	Clipboard Management Software controls data copied to the clipboard, preventing unauthorized access or data leakage by monitoring, restricting, and logging clipboard activities, ensuring sensitive information is not inadvertently shared or exposed.

Code Integrity	Code Integrity ensures that software code has not been altered or tampered with, verifying its authenticity and trustworthiness, thereby protecting systems from malicious code and unauthorized modifications.
Conduct Forensic Log Analysis	Conduct Forensic Log Analysis involves systematically examining log files to identify, understand, and mitigate security incidents, ensuring accurate detection of anomalies, unauthorized access, and potential breaches for effective incident response.
Conduct Penetration Testing	Conduct Penetration Testing involves simulating cyberattacks on a system to identify vulnerabilities, assess security posture, and ensure defences are effective, enabling timely remediation of identified weaknesses.
Conduct Security Awareness Training	Conduct Security Awareness Training involves educating employees on cybersecurity best practices, potential threats, and response protocols to enhance organizational security posture and reduce the risk of human-related security breaches.
Configure Secure Authentication	"Configure Secure Authentication" involves implementing strong authentication mechanisms, such as multi-factor authentication (MFA), to verify user identities, protect against unauthorized access, and enhance overall system security.
Configure Secure Protocols Only	"Configure Secure Protocols Only" ensures that systems and applications use encrypted communication protocols,

	such as HTTPS, SSH, and TLS, to protect data integrity and confidentiality during transmission, mitigating eavesdropping and interception risks.
Container Image Hardening	Container Image Hardening involves securing container images by minimizing vulnerabilities, removing unnecessary components, applying security patches, and ensuring configurations adhere to best practices to reduce attack surfaces and enhance security.
Container Image Scanning	Container Image Scanning involves analysing container images for vulnerabilities, malware, and configuration issues before deployment, ensuring security compliance and reducing risks in containerized environments.
Content Integrity Checking	Content Integrity Checking involves verifying the authenticity and accuracy of data by using cryptographic hashes or checksums to detect unauthorized modifications, ensuring data remains unaltered during storage or transmission.
Continuous Monitoring	Continuous Monitoring involves the real-time or near-real-time observation of an organization's IT environment to detect vulnerabilities, threats, and compliance issues, enabling timely responses to mitigate potential risks.
Continuous Security Education and Reminders	Continuous Security Education and Reminders involve regularly updating employees on cybersecurity best practices and threats through training sessions,

	newsletters, and alerts to reinforce awareness and promote a security-conscious culture.
Control Flow Integrity	Control Flow Integrity (CFI) is a security mechanism that prevents attackers from altering the control flow of a program, ensuring execution follows the intended sequence to mitigate exploits like code injection.
Credential Management	Credential Management involves securely storing, handling, and transmitting user credentials, such as passwords and tokens, to prevent unauthorized access and ensure only authenticated users can access sensitive systems and data.
Credential Vaulting	Credential Vaulting securely stores sensitive authentication data, like passwords and API keys, in an encrypted vault, restricting access to authorized users and applications to prevent unauthorized access and data breaches.
Data Integrity	Data Integrity ensures that information is accurate, consistent, and unaltered during storage, processing, or transmission, protecting against unauthorized modifications and ensuring data reliability and trustworthiness.
Database Permissions	Database Permissions control involves setting and managing access rights to database resources, ensuring only authorized users can perform specific actions,

	thereby protecting data integrity, confidentiality, and preventing unauthorized access.
Data-in-Transit Encryption	Data-in-Transit Encryption protects data being transmitted across networks by encrypting it, ensuring confidentiality and integrity, and preventing unauthorized access or interception during transmission.
DDoS Attack Detection	DDoS Attack Detection involves monitoring network traffic to identify unusual patterns or spikes indicative of Distributed Denial of Service attacks, enabling timely alerts and mitigation to maintain service availability.
Default Password Change Requirement	The "Default Password Change Requirement" control mandates changing default passwords on devices and applications to unique, strong passwords to prevent unauthorized access and enhance security posture.
Denial of Service (DoS) Protection	Denial of Service (DoS) Protection involves implementing measures to detect, mitigate, and prevent DoS attacks, ensuring network availability and performance by filtering malicious traffic and maintaining service continuity.
Deploy Endpoint Detection and Response (EDR)	Deploying Endpoint Detection and Response (EDR) involves implementing solutions that monitor, detect, and respond to threats on endpoints, providing real-time visibility and automated responses to enhance security posture.

Device and Media Controls	Device and Media Controls ensure secure handling, storage, and disposal of hardware and media to prevent unauthorized access, data loss, or breaches, safeguarding sensitive information throughout its lifecycle.
Disable Macros by Default	"Disable Macros by Default" prevents automatic execution of potentially malicious scripts in documents, reducing the risk of malware infections by requiring user intervention to enable macros only when necessary.
Disable Password Authentication	"Disable Password Authentication" involves configuring systems to reject password-based logins, instead requiring stronger authentication methods like public key cryptography, enhancing security by reducing vulnerability to password-based attacks.
Disable RDP if Not Needed	Disabling Remote Desktop Protocol (RDP) when not needed reduces attack surfaces, preventing unauthorized access and mitigating risks of brute force attacks, ransomware, and other remote exploitation threats.
Disable Remote Management	"Disable Remote Management" prevents unauthorized access by turning off remote management features, reducing the attack surface and mitigating risks of unauthorized control or data breaches from remote locations.
Disable USB Ports	"Disable USB Ports" is a security control that prevents unauthorized data transfer and malware introduction by

	disabling USB port functionality, reducing the risk of data breaches and system compromise.
DLL Signature Verification	DLL Signature Verification ensures that Dynamic Link Libraries (DLLs) are digitally signed by trusted sources, preventing unauthorized or malicious code execution by verifying the integrity and authenticity of the DLL files.
DNS Anomaly Detection	DNS Anomaly Detection identifies unusual patterns or deviations in DNS traffic, helping to detect potential threats like data exfiltration, command-and-control communications, or domain generation algorithm (DGA) activities.
DNS Logging and Analysis	DNS Logging and Analysis involves monitoring and analysing DNS queries and responses to detect anomalies, identify malicious activities, and enhance network security by providing insights into potential threats and vulnerabilities.
DNS Threat Intelligence	DNS Threat Intelligence involves monitoring and analysing DNS traffic to identify and block malicious domains, enhancing network security by preventing phishing, malware distribution, and command-and-control server communications.
Educate Users on Secure Practices	Educate users on secure practices by conducting regular training sessions to increase awareness of cybersecurity threats, safe online behaviour, and proper handling of

	sensitive information to minimize human-related security risks.
Email Security Awareness Training	Email Security Awareness Training educates employees on recognizing phishing attempts, avoiding malicious links, and safeguarding sensitive information, thereby reducing the risk of email-based cyber threats and enhancing organizational security posture.
Enable API Server Authentication and Authorization	This control ensures that all API server requests are authenticated and authorized, preventing unauthorized access and actions by verifying user identities and permissions before granting access to resources.
Enable TLS for All Sensitive Data Transfers	This control ensures that all sensitive data transfers are encrypted using Transport Layer Security (TLS), protecting data integrity and confidentiality during transmission over networks, and mitigating risks of interception or tampering.
Encrypt SSH Sessions	Encrypting SSH sessions ensures that data transmitted between a client and server is secure, preventing eavesdropping, man-in-the-middle attacks, and unauthorized access to sensitive information during remote management.
Encryption of Removable Media	"Encryption of Removable Media" ensures data confidentiality by encrypting files on portable storage devices, preventing unauthorized access if lost or stolen, and maintaining data integrity during transfers.

Endpoint Detection and Response	Endpoint Detection and Response (EDR) is a security solution that continuously monitors and collects data from endpoints to detect, investigate, and respond to potential threats in real-time.
Enforce HTTPS for All Traffic	Enforce HTTPS for all traffic ensures data encryption between clients and servers, protecting against eavesdropping, man-in-the-middle attacks, and data integrity issues by mandating secure, encrypted connections for all communications.
Enforce MFA for Remote Access	Enforce Multi-Factor Authentication (MFA) for remote access to enhance security by requiring users to provide two or more verification factors, reducing the risk of unauthorized access to sensitive systems and data.
Enforce Principle of Least Privilege	The control 'Enforce Principle of Least Privilege' ensures users and systems have the minimum access necessary to perform their functions, reducing potential attack vectors and limiting damage from breaches.
Enforce Secure Key Management	Enforce Secure Key Management ensures cryptographic keys are generated, stored, distributed, and retired securely, minimizing unauthorized access and misuse, while maintaining confidentiality, integrity, and availability of sensitive data.
Exploit Protection	Exploit Protection is a security measure designed to prevent the execution of malicious code by identifying

	and blocking exploitation techniques used by attackers to exploit software vulnerabilities.
File Type Verification	File Type Verification ensures that files are not misrepresented by checking their actual content against expected types, preventing malicious files disguised with incorrect extensions from being executed or opened.
File Upload Security	File Upload Security involves implementing measures to validate, sanitize, and scan uploaded files to prevent malicious content, ensuring only safe and authorized files are accepted by the system.
Grsecurity/Pax	Grsecurity/Pax is a set of security enhancements for the Linux kernel, providing advanced access control, memory corruption protection, and exploit mitigation to enhance system security and prevent unauthorized access.
HTTPS	HTTPS encrypts data exchanged between a user's browser and a web server, ensuring confidentiality and integrity, protecting against eavesdropping, man-in-the-middle attacks, and data tampering.
Implement Access Control Checks	Implement Access Control Checks involves verifying user permissions before granting access to resources, ensuring only authorized users can access sensitive data, thereby preventing unauthorized access and potential data breaches.
Implement API Rate Limiting	Implement API Rate Limiting restricts the number of API requests a user or system can make within a specified

	timeframe, preventing abuse, ensuring availability, and protecting against denial-of-service attacks.
Implement HTTPS	Implementing HTTPS ensures encrypted communication between clients and servers, protecting data integrity and confidentiality, preventing eavesdropping, man-in-the-middle attacks, and ensuring authenticity through SSL/TLS certificates.
Implement Real-time Alerting	Implement Real-time Alerting involves continuously monitoring systems and networks to detect suspicious activities or anomalies, promptly notifying security teams to enable immediate response and mitigate potential threats.
Implement Segmentation and Network Isolation	Implement Segmentation and Network Isolation to limit access and contain threats by dividing networks into smaller, isolated segments, reducing attack surfaces and preventing lateral movement within the network.
Incident Analysis	Incident Analysis involves examining security incidents to determine their cause, impact, and scope, enabling organizations to improve defences, prevent recurrence, and enhance response strategies through detailed investigation and reporting.
Key Logging	Key logging is a monitoring technique that records keystrokes on a keyboard, often used maliciously to capture sensitive information like passwords, but can also be used for legitimate security auditing.

Keylogger Detection Tools	Keylogger Detection Tools are software solutions designed to identify and block keylogging malware, which captures keystrokes to steal sensitive information, ensuring user data confidentiality and system integrity.
Limit Access to Kubernetes Dashboard	Restrict access to the Kubernetes Dashboard by implementing authentication, role-based access control (RBAC), network policies, and secure connections to prevent unauthorized access and protect sensitive cluster information.
Limit Browser Plugins	"Limit Browser Plugins" involves restricting the installation and use of browser plugins to reduce attack surfaces, prevent vulnerabilities, and enhance security by allowing only essential, vetted plugins.
Limit Exposure of Sensitive Data	Limit Exposure of Sensitive Data involves minimizing access, encrypting data, implementing data masking, and using secure channels to reduce the risk of unauthorized access and data breaches.
Log and Monitor SSH Access	"Log and Monitor SSH Access" involves recording all SSH login attempts and activities, analysing logs for anomalies, and alerting administrators to unauthorized access attempts to enhance security and incident response.
Log Integrity	Log Integrity ensures that log files are protected from unauthorized access, modification, or deletion,

	maintaining their accuracy and reliability for forensic analysis and compliance purposes.
Log Tamper Detection	Log Tamper Detection involves monitoring and alerting for unauthorized changes to log files, ensuring integrity by using cryptographic hashes or checksums to detect alterations, thereby maintaining reliable audit trails.
Logging and Auditing of DNS Queries	This control involves monitoring and recording DNS query activities to detect anomalies, ensure compliance, and facilitate incident response by providing visibility into potential threats and unauthorized access attempts.
Malware Analysis	Malware Analysis involves examining malicious software to understand its behaviour, origin, and impact, enabling effective detection, prevention, and response strategies to protect systems and data from cyber threats.
Malware and Ransomware Training	This control involves educating employees on identifying, avoiding, and responding to malware and ransomware threats, enhancing awareness, and promoting safe online practices to mitigate potential security breaches and data loss.
Memory Access Monitoring	Memory Access Monitoring involves tracking and analysing access to system memory to detect unauthorized or suspicious activities, helping prevent data breaches, malware execution, and ensuring compliance with security policies.

Monitor Network Traffic	"Monitor Network Traffic" involves continuously observing data flow across a network to detect anomalies, unauthorized access, or malicious activities, ensuring timely identification and response to potential security threats.
Node Resource Management	Node Resource Management involves monitoring and regulating the allocation and usage of computational resources in a network to prevent resource exhaustion, ensure availability, and maintain optimal performance and security.
Phishing Simulation	Phishing Simulation involves conducting mock phishing attacks to assess and enhance employees' ability to recognize and respond to phishing attempts, thereby strengthening organizational resilience against real-world phishing threats.
Privilege Access Management (PAM)	Privilege Access Management (PAM) involves securing, managing, and monitoring access to critical systems and data by controlling privileged accounts, reducing risks of unauthorized access and potential data breaches.
Privilege Escalation Monitoring	Privilege Escalation Monitoring involves continuously tracking and analysing user activities to detect unauthorized privilege increases, ensuring timely alerts and responses to prevent potential security breaches and maintain system integrity.

Privileged Account Discovery	Privileged Account Discovery identifies, and inventories privileged accounts across systems and applications, enabling organizations to manage access, reduce risks, and ensure compliance by monitoring and securing these high-risk accounts.
Quarantine Suspicious Emails	Quarantine Suspicious Emails involves isolating potentially harmful emails in a secure environment, preventing them from reaching users' inboxes, allowing for safe analysis and reducing the risk of phishing or malware attacks.
Rate Limiting APIs	Rate Limiting APIs control restricts the number of API requests a user or system can make within a specified timeframe, preventing abuse, ensuring fair usage, and protecting against denial-of-service attacks.
Regular Plugin Updates	Regular Plugin Updates involve routinely updating software plugins to their latest versions to patch vulnerabilities, enhance security features, and ensure compatibility, thereby reducing the risk of exploitation by cyber threats.
Regularly Update SSH Software	Regularly updating SSH software ensures the latest security patches are applied, mitigating vulnerabilities and protecting against exploits, unauthorized access, and potential breaches in secure shell communications.
Remote Desktop Protocol (RDP) Hardening	RDP Hardening involves securing Remote Desktop Protocol by enforcing strong authentication, using

	network-level authentication, applying encryption, limiting access via firewalls, and regularly updating software to prevent unauthorized access.
Resource Monitoring	Resource Monitoring involves continuously tracking and analysing system resources like CPU, memory, and network usage to detect anomalies, optimize performance, and identify potential security threats in real-time.
Resource Utilization Metrics	"Resource Utilization Metrics" involves monitoring and analysing system resource usage to detect anomalies, optimize performance, and identify potential security threats through unusual patterns or spikes in resource consumption.
Rootkit Detection	Rootkit Detection involves identifying and removing malicious software that hides its presence on a system, often using specialized tools to scan for hidden files, processes, or system modifications.
Runtime Protection	Runtime Protection involves monitoring and securing applications during execution to detect and prevent malicious activities, unauthorized changes, or vulnerabilities, ensuring the integrity and security of the running software.
Runtime Security Monitoring	Runtime Security Monitoring involves continuously observing applications during execution to detect and respond to threats in real-time, ensuring immediate

	identification and mitigation of vulnerabilities or malicious activities.
Secrets Management	Secrets Management involves securely storing, accessing, and managing sensitive information like passwords, API keys, and tokens to prevent unauthorized access and reduce the risk of data breaches.
Secure Application Development Practices	Secure Application Development Practices involve integrating security measures throughout the software development lifecycle, including threat modelling, secure coding standards, code reviews, and security testing, to mitigate vulnerabilities and enhance application security.
Secure Email Gateway	A Secure Email Gateway filters and monitors inbound and outbound emails to protect against threats like phishing, malware, and spam, ensuring secure communication and data protection for organizations.
Secure IPC Mechanisms	"Secure IPC Mechanisms" control ensures safe inter-process communication by implementing authentication, encryption, and access controls to prevent unauthorized data access and mitigate risks of data interception or tampering.
Secure WebSocket Configuration	Ensure WebSocket connections use TLS for encryption, validate server certificates, implement strict origin checks, and configure secure headers to protect against unauthorized access and data interception.

Service and Daemon Management	Service and Daemon Management involves monitoring, configuring, and securing system services and background processes to minimize vulnerabilities, reduce attack surfaces, and ensure only necessary services are active and properly configured.
Social Engineering Awareness	Social Engineering Awareness involves educating employees to recognize, resist, and report deceptive tactics used by attackers to manipulate individuals into divulging confidential information or performing unauthorized actions.
SSH Hardening	SSH Hardening involves securing SSH configurations by enforcing strong authentication, disabling root login, using key-based authentication, restricting access by IP, and regularly updating SSH software to mitigate vulnerabilities.
Threat Detection	Threat Detection involves monitoring systems and networks to identify suspicious activities or anomalies, enabling timely alerts and responses to potential security incidents, thereby minimizing damage and maintaining system integrity.
Token Management	Token Management involves securely generating, storing, and validating tokens used for authentication and authorization, ensuring they are protected against misuse, theft, and replay attacks to maintain system integrity.

Use Azure Bastion for Secure RDP/SSH	Azure Bastion provides secure RDP/SSH access to Azure VMs without exposing them to the internet, reducing attack surface and enhancing security by using a fully managed platform service.
Use Credential Management Tools	"Use Credential Management Tools" involves deploying software to securely store, manage, and retrieve user credentials, ensuring strong encryption, reducing password reuse, and facilitating secure authentication processes across systems.
User Access Management	User Access Management involves processes to ensure that only authorized individuals have access to specific systems and data, including user provisioning, authentication, role-based access, and regular access reviews.
Vulnerability Management	Vulnerability Management involves identifying, evaluating, prioritizing, and mitigating software and hardware vulnerabilities to reduce risk and protect systems from exploitation by continuously monitoring and applying patches or updates.
Zero Trust Architecture	Zero Trust Architecture is a security model that requires strict identity verification for every user and device attempting to access resources, regardless of their location, assuming no implicit trust.

Table 24 -Case Study Control Selection

The following images shows the graph with the control nodes added to the graph and relationships between the control nodes and attack nodes being established.

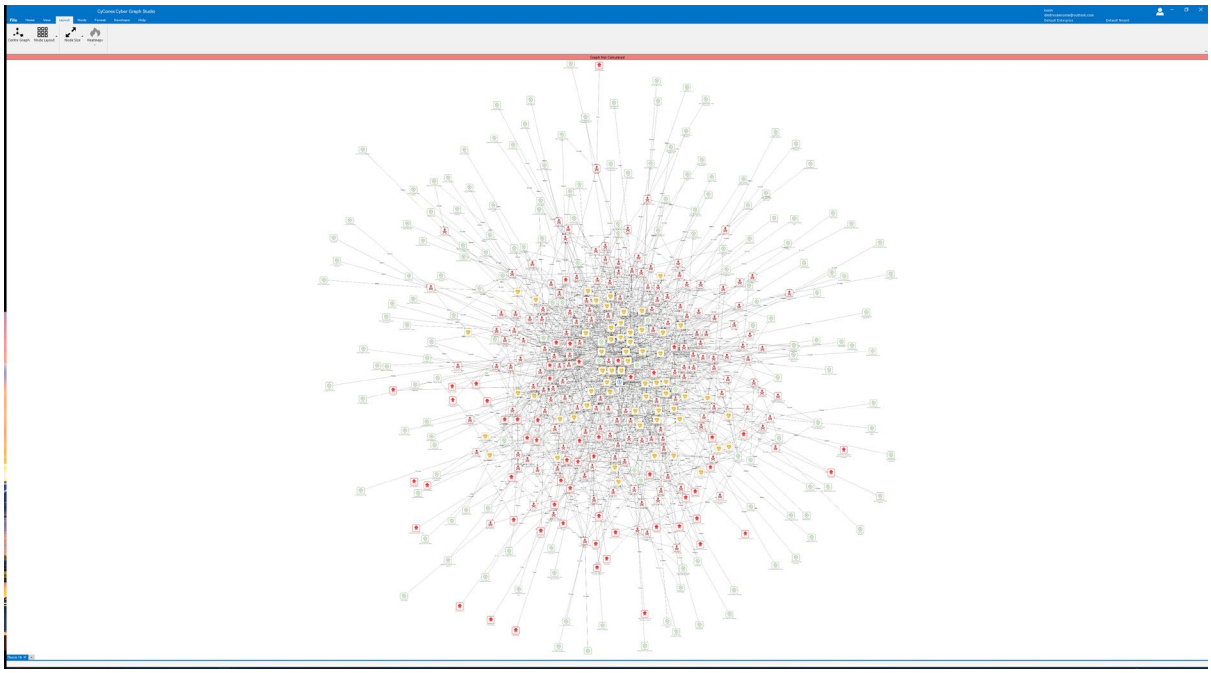

Figure 29 - Case Study Control Nodes Added to Graph

8.9 Post Mitigation Risk Assessment

The next stage of the assessment was to undertake a Post Mitigation Risk Assessment, which is used to determine the final risk faced by the target of evaluation asset.

A post mitigation risk assessment focuses on evaluating after implementing controls. This stage establishes the final understanding of the organisation's inherent risk exposure and identifies critical vulnerabilities, attacks, and potential impacts.

The process is similar to the Pre-Mitigation risk Assessment which involves the following key steps:

Identify Inherent Risks:

Analyse the vulnerabilities and attacks associated with assets in their current state without considering any controls. This step helps determine the raw risk level posed by existing weaknesses.

Estimate Likelihood and Impact:

Assess the likelihood of potential attacks exploiting identified vulnerabilities and the associated impacts on confidentiality, integrity, and availability. This estimation provides insight into the severity of risks.

Map Attacks to Vulnerabilities:

Establish a clear connection between identified vulnerabilities and threat actors' methods to exploit them. This mapping helps visualise the organisation's exposure and identify high-priority risks.

Map Controls to Attacks:

Establish a clear connection between identified controls and attacks reducing the impact of he attacks.

Calculate Risk Levels:

Assign risk levels based on likelihood and impact. These levels represent the organisation's risk profile in the absence of mitigations.

Again, as part of the Post Mitigation Risk Assessment calculations a Monte Carlo analysis is used. A Monte Carlo analysis is a computational technique to understand the impact of uncertainty, variability, or randomness in a system, model, or process.

In this Post Mitigation Risk Assessment, a Monte Carlo simulation consisting of 100 iterations was performed to analyse the behaviour and outcomes of the graph-based model. During each iteration, the attributes of nodes and edges within the graph were randomly selected from a predefined statistical distribution. These distributions could represent characteristics such as probabilities, weights, or other metrics relevant to the nodes and edges, ensuring that the simulation captured a range of scenarios.

Again, randomly assigned attributes were then used to initialize or "seed" the graph for computation, allowing the simulation to reflect the stochastic nature of real-world conditions. For each seeded graph, computations were carried out to evaluate various metrics or values of interest derived from the graph structure.

On completion of each iteration, the results, whether node values, edge weights, or other graph properties, were aggregated into a cumulative distribution. This aggregate distribution provided a comprehensive view of the range and likelihood of different outcomes across the simulation. From this final distribution, the mode (the most frequently occurring value) of any required metric was selected, offering a representative value for decision-making or analysis.

Confidentiality Risk

The following table lists the top 10 post mitigation risks to the Confidentiality of the Asset, (a full list of Confidentiality Risks can be found in the appendices A4):

Vulnerability	Asset	Likelihood	Impact	Risk Value
Insufficient Authorization	Microsoft Azure	32	47	15
Cross-Account Access Misconfigurations	Microsoft Azure	22	47	10

Improper Identity and Access Management	Microsoft Azure	21	47	10
Insecure Remote Management Access	Microsoft Azure	20	47	9
Default Credentials	Microsoft Azure	17	47	8
Insecure Default Settings	Microsoft Azure	16	47	8
Broken Authentication and Session Management	Microsoft Azure	16	47	8
Inadequate Secrets Management	Microsoft Azure	14	47	7
Insecure Access Control Policies	Microsoft Azure	14	47	7
Inadequate Data Backup and Recovery	Microsoft Azure	14	47	7

Table 25 - Pre-Mitigated Risk to Confidentiality

Integrity Risk

The following table lists the top 10 post mitigation risks to the Integrity of the Asset, (a full table of Integrity Risks can be found in the appendices A5).

Vulnerability	Asset	Likelihood	Impact	Risk Value
Insufficient Authorization	Microsoft Azure	21	48	10
Improper Identity and Access Management	Microsoft Azure	15	48	7

Insecure Remote Management Access	Microsoft Azure	14	48	7
Inadequate Protection Against Insider Threats	Microsoft Azure	13	48	6
Broken Authentication and Session Management	Microsoft Azure	12	48	6
Cross-Account Access Misconfigurations	Microsoft Azure	11	48	5
Inadequate IAM Policies	Microsoft Azure	10	48	5
Insecure Default Settings	Microsoft Azure	9	48	4
Default Credentials	Microsoft Azure	9	48	4
Insecure DevOps Practices	Microsoft Azure	9	48	4

Table 26 - Pre-Mitigated Risk to Integrity

Availability Risk

The following table lists the top 10 post mitigation risks to the Availability of the Asset, (a full table of Availability Risks can be found in the appetencies A6).

Vulnerability	Asset	Likelihood	Impact	Risk Value
Inadequate Data Backup and Recovery	Microsoft Azure	25	39	10
Insufficient Incident Response Procedures	Microsoft Azure	14	39	5
Improper Identity and Access Management	Microsoft Azure	12	39	5

Broken Authentication and Session Management	Microsoft Azure	12	39	5
Insecure Remote Management Access	Microsoft Azure	11	39	4
Backup Failures	Microsoft Azure	11	39	4
Insufficient Authorization	Microsoft Azure	10	39	4
Insecure Access Control Policies	Microsoft Azure	9	39	4
Insecure Key Management Practices	Microsoft Azure	9	39	4
Inadequate IAM Policies	Microsoft Azure	9	39	4

Table 27 - Pre-Mitigated Risk to Availability

Accountability Risk

The following table lists the top 10 post mitigation risks to the Accountability of the Asset., (full table of Accountability risks can be found in the appendices A7).

Vulnerability	Asset	Likelihood	Impact	Risk Value
Insufficient Authorization	Microsoft Azure	32	36	12
Cross-Account Access Misconfigurations	Microsoft Azure	22	36	8
Improper Identity and Access Management	Microsoft Azure	21	36	8
Insecure Remote Management Access	Microsoft Azure	20	36	7

Default Credentials	Microsoft Azure	17	36	6
Insecure Default Settings	Microsoft Azure	16	36	6
Broken Authentication and Session Management	Microsoft Azure	16	36	6
Inadequate Secrets Management	Microsoft Azure	14	36	5
Insecure Access Control Policies	Microsoft Azure	14	36	5
Inadequate Data Backup and Recovery	Microsoft Azure	14	36	5

Table 28 - Pre-Mitigated Risk to Accountability

The following image illustrates the distribution of Impact and Likelihood of risk present with the graph:

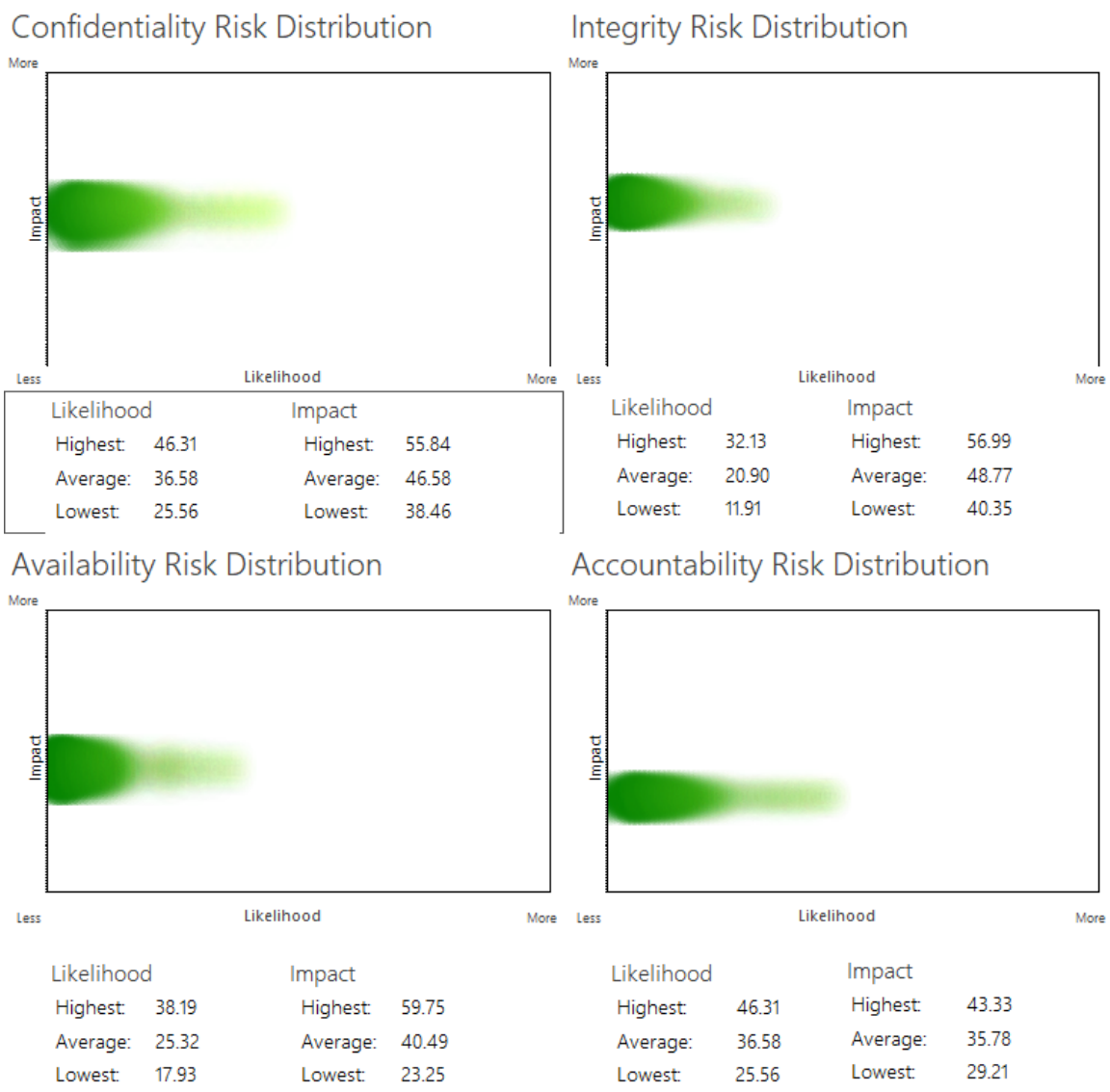

Figure 30 - case Study - Pre-Mitigation Risk Distribution

8.10 Case Study Feedback

Following completion of the case study, a review meeting was held to understand and capture the positive and negative aspects of the Case Study. The review meeting consisting of a small panel of cyber security subject matter experts including cyber risk assessors and security architects. The

members of the panel had all participated in the case study and were engaged in all aspects of the development of the graph used in the study.

Cyber Risk Assessor Feeback

Positive Feedback

Transparent Methodology:

> The structured approach to using directed graphs for risk assessment provides a transparent and innovative approach. This is highly beneficial for organisations with complex cybersecurity ecosystems.

Dynamic Analysis:

> Incorporating Monte Carlo simulations demonstrates a strong emphasis on capturing variability in real-world conditions, which enhances the model's reliability.

Real-World Application:

> The case study's focus on the government sector aligns with critical infrastructure protection, showcasing practical applicability.

Visualisation Strength:

> The graphical representation of risks, controls, and vulnerabilities is a strong point, enabling stakeholders to comprehend complex dependencies effectively.

Comprehensive Scope:

> Including pre-mitigation and post-mitigation assessments reflects a thorough analysis of the risk lifecycle.

Constructive Feedback

Data Validation:

Greater emphasis on validating the input data for the directed graphs would ensure consistency and accuracy, especially for large-scale implementations.

Stakeholder Involvement:

While the technical aspects are robust, the case study could further elaborate on stakeholder engagement during the risk assessment.

Real-Time Monitoring:

Consideration of how the directed graph model could incorporate real-time threat intelligence would strengthen its applicability in dynamic threat environments.

Scalability Challenges:

Addressing potential computational limitations for large graphs in complex organizations could enhance the model's practicality.

Cultural Factors:

Including insights on how organizational culture and human factors influence the framework's adoption could provide a more holistic view.

Security Architect Feedback

Positive Feedback

Innovative Framework:

The directed graph approach is a standout feature, as it allows for visualizing and analysing relationships between assets, threats, and controls, which is critical for designing secure architectures.

Adaptability:

The model's ability to dynamically update with changing cybersecurity landscapes makes it highly relevant for real-world implementations.

Granularity:

The detailed representation of nodes (assets, threats, vulnerabilities) and edges (relationships, dependencies) provides architects with the necessary depth for designing layered security measures.

Risk-Based Insights:

Including pre- and post-mitigation risk assessments is invaluable for designing proactive and reactive security controls.

Scalable Methodology:

The case study demonstrates scalability in modelling large organizational networks, essential for designing enterprise-level security architectures.

Constructive Feedback

Integration with Existing Frameworks:

While the methodology is innovative, more detail on integrating with established frameworks like NIST CSF, ISO 27001, or TOGAF would improve its alignment with industry standards.

Automation Capabilities:

Elaborating on the potential for automating the generation and analysis of directed graphs would make the model more appealing for continuous monitoring.

Cloud and Hybrid Environments:

The case study could expand on how the framework handles complexities introduced by modern architectures, such as cloud and hybrid environments.

Threat Intelligence:

Incorporating mechanisms for ingesting real-time threat intelligence feeds would significantly enhance its relevance for adaptive security architectures.

Performance Optimisation:

Addressing computational overheads when applying the model to large-scale systems or networks could improve adoption feasibility in high-demand environments.

Chapter 9 - Reflection and Appraisal

This section of the thesis undertakes a reflection of the original research aims, objectives and questions originally described in Chapter 1.

9.1 Analysis of the Research Aims

This section discusses the extent to which the research aims have been addressed.

Research Aim 1: Develop a Directed Graph Schema capable of accurately modelling information security risk and compliance. Our primary goal is to create an advanced information security-directed graph model that can be used to accurately represent all elements and relationships related to information security within an enterprise environment.

We believe this research aim has been strongly achieved. The research has developed a comprehensive graph schema that is capable of accurately modelling information security risk and compliance in an enterprise environment. The research proposes a standardised set of node types, including Threat Actor, Attack, Vulnerability, Asset, Controls and Objective which provide specific and meaningful context to the graph allowing users with information security experience to implicitly understand the construct and relationships within the graph. Further, the graph schema enforces and

controls relationships between node types, ensuring that only semantically correct graphs can be created.

Research Aim 2: Elucidate complex interactions and dependencies: This research seeks to formalise and elucidate the many interdependent interactions and dependencies among people, policies and technology that comprise an enterprise's information security foundation. By critically examining each element's effect and influence on surrounding elements, a complete picture of its information security ecosystem will emerge.

We believe this research aim has been fully achieved but has significantly developed as the research progressed. By using the graph schema, it is entirely possible to model people, policy, and technology elements as abstract elements but in several different ways, allowing the graph to be flexibly implemented. For example, the graph schema supports the policy aspect in different ways depending on how the user considered it could be best used to model their needs, i.e. a single Control node can be used to represent policy, or alternatively, an Objective node could be used with parent Control nodes representing different policies or policy elements, or alternatively again an Asset node could be used to represent policy with associated Objective and Control nodes.

Research Aim 3: Establish a standardised assessment model: This research seeks to create a rigorous and repeatable model for evaluating information security maturity within an enterprise, making the assessment consistent across scales and contexts.

Again, the research has achieved these aims by creating the graph schema along with the associated formulas for determining consistent assessments; further, the schema and formulas have been implemented as algorithms within the CyConex application.

Research Aim 4: Integrating granular taxonomy and metrics: Central to this research effort is developing a granular taxonomy that accurately depicts varying maturity levels within specific

controls and elements and quantifiable metrics that can measure the effectiveness of information security controls and processes.

Once again, the research aims have been achieved in this regard. As part of the graph schema, nodes have a defined taxonomy of parameters used to support the production of accurate and quantifiable metrics. For example, Threat Actor nodes have a defined Taxonomy of parameters such as Access (TAa), Capability (TAc), Resources (TAr), and Motivation (TAm), resulting in a Threat Actor Value (TAv). The schema further provides for each parameter guidance on assessing each parameter.

Research Aim 5: Facilitate risk Reduction and informed decision-making: Through this model, this research seeks to give organisations tools that will allow for more informed decision-making regarding information security risks while reducing these risks.

This research also sought to give organisations tools that will allow for more informed decision-making regarding information security risks while reducing these risks. This aim has been fully achieved through the design of the graph schema and the development of the CyConex application which implements the graph schema and associated calculations. This coupled with additional capabilities within the application to visualise risk and compliance aspects of graphs through dashboards and heat maps for example, users of the application can have significantly improved understanding and better-informed decision-making.

Research Aim 6: Explore applications in the cybersecurity industry and beyond. An additional goal of developing this model is to explore its applications in other areas, such as cyber insurance underwriting, information security consulting, regulatory compliance assessments and organisational risk management. In doing this, one can assess how it could provide value for these domains while strengthening the overall cybersecurity posture.

The CyConex application will be available free of charge for non-commercial use or for evaluation in commercial organisations from www.cyconex.com. This will allow organisations to develop their own directed graphs for specific circumstances.

Research Aim 7: Contribute to academic and practical knowledge: Finally, this research seeks to make a substantial impactful contribution to both academic literature and practical knowledge in information security. By filling gaps in existing frameworks and setting a precedent for further investigation of maturity assessment techniques in information security maturity assessment processes.

Finally, this research sought to make a substantial contribution to both academic literature and practical knowledge in information security. By filling gaps in existing models and setting a precedent for further investigation of maturity assessment techniques in information security maturity assessment processes. We believe this aim has been achieved through the publication of this research and the development of the CyConex application.

9.2 Analysis of the Research Objectives

This section discusses the extent to which the research objectives have been achieved.

Research Objective 1: How can a directed graph-based framework be designed to offer a more comprehensive and accurate representation of the interactions and dependencies among human factors, policy elements and technological components within an enterprise? What advantages does this approach have over traditional frameworks?

We believe this objective has been achieved as the graph schema presented in this research and the CyConex application allow extremely complex and detailed directed graph models to be created which can accurately represent the interactions and dependencies among people, policy, and technology. This approach has some significant advantages over traditional frameworks such as the ability to visualise relationships which provides for greater insight and understanding coupled with the

graph schema and formulas which ensure a rigorous and consistent approach to compliance and risk assessments.

Research Objective 2: How does a directed graph-based maturity assessment framework enhance the understanding and evaluation of enterprise information and cybersecurity maturity? How can this approach be leveraged to develop more granular taxonomies and metrics and facilitate better-informed decision-making for risk reduction?

Again, we believe this objective has been fully achieved. For example, the case study in Chapter 7 specifically demonstrated how directed graphs can be used to undertake a maturity assessment and provide greater understanding and insight into enterprise information and cybersecurity. The graph schema provides a granular taxonomy and structure in assessing metrics for the schema parameters. When the resulting graph metric is calculated we believe the approach provides better-informed decision-making and consequently risk reduction.

Research Objective 3: What are the challenges and considerations in implementing a directed graph-based maturity assessment framework within an enterprise and how can they be addressed to ensure the effectiveness and scalability of the model?

The research has identified a small number of challenges and considerations when implementing directed graphs. For example, correctly formalising relationships between nodes to accurately represent the organisational context requires expert knowledge of cybersecurity coupled with a good knowledge of the target of evaluation the graph is modelling. Further, when modelling graphs, it is important to understand what questions the graph is attempting to answer as this can significantly impact how relationships need to be modelled.

Research Objective 4 (Additional): How can the directed graph-based maturity assessment framework be applied in domains such as cyber insurance, regulatory compliance assessments and organisational risk management and what value does it bring to these areas?

The directed graph-based maturity assessment framework can be applied in domains such as cyber insurance, regulatory compliance assessments, and organizational risk management in several ways.

Cyber insurance providers can use the framework to assess the maturity of a potential customer's cybersecurity posture, which can help them to:

Price premiums more accurately: By assessing the customer's maturity in areas such as asset management, risk management, security controls, and incident response, cyber insurance providers can get a better understanding of the customer's overall cybersecurity risk. This information can then be used to price premiums more accurately.

Identify areas where the customer may need to improve their security posture to be eligible for coverage: By identifying the customer's strengths and weaknesses in terms of cybersecurity maturity, cyber insurance providers can provide the customer with recommendations for improvement. This can help the customer to reduce their risk of cyberattacks and to become more eligible for cyber insurance coverage.

Regulatory compliance assessments

Organizations can use the framework to assess compliance with various regulations, such as the General Data Protection Regulation (GDPR) and the Cybersecurity Maturity Model Certification (CMMC). This can help them to:

Avoid fines and penalties: Organizations can identify and address any gaps in their security and compliance programs by assessing their compliance with regulations. This can help them to avoid fines and penalties from regulatory bodies.

Protect their reputation: A data breach or other cybersecurity incident can damage an organization's reputation. By assessing their compliance with regulations, organizations can reduce the risk of such incidents and protect their reputation.

Gain a competitive advantage: In some industries, compliance with regulations is a requirement for doing business. By assessing their compliance with regulations, organizations can gain a competitive advantage over their competitors.

Organizations can use the framework to assess their overall risk posture, including cybersecurity risks. This can help them to:

Identify and prioritize risks: The framework can help organizations identify and prioritize all risks, including cybersecurity. This information can then be used to develop and implement risk mitigation strategies.

Develop and implement risk mitigation strategies: Once the risks have been identified and prioritized, the framework can be used to develop and implement risk mitigation strategies. This can reduce the likelihood and impact of risks occurring.

Monitor and report on risks: The framework can also be used to monitor and report risks. This can help organizations track their progress in mitigating risks and to identify any new or emerging risks.

The directed graph-based maturity assessment framework brings some benefits to these domains, including:

Holistic view: The framework provides a holistic view of an organization's cybersecurity posture, regulatory compliance, or organizational risk posture. This can help organizations identify and address gaps in their security and compliance programs.

Risk-based approach: The framework is risk-based, which focuses on the areas that pose the most significant risk to the organization. This helps organizations prioritize their security and compliance efforts.

Measurable improvement: The framework provides a quantitative measure of maturity, which allows organizations to track their progress over time and to identify areas where they need to improve.

The directed graph-based maturity assessment framework can be used to assess an organization's cybersecurity posture by considering the following factors:

Asset management: How well does the organization identify and classify its assets, including information systems, data, and hardware?

Risk management: How well does the organization identify, assess, and manage cybersecurity risks?

Security controls: How well does the organization implement and maintain security controls to protect its assets?

Incident response: How well is the organization prepared to respond to and recover from cybersecurity incidents?

The organization can also use the framework to track their progress over time. By reassessing its cybersecurity posture regularly, the organization can see how its maturity score is changing and identify areas where it needs to focus its improvement efforts.

9.3 Analysis of the Research Questions

This section discusses the extent to which the research questions set out in Chapter 1 have been addressed.

Research Question 1 (RQ1):

> How well do prevalent information security frameworks encompass and illustrate the complex interactions and dependencies among People, Policy, and Technology within an enterprise setting? What are the specific areas where these frameworks might fall short in addressing the synergistic relationships between these elements?

In Chapter 4, we undertook a review of the common cybersecurity maturity and risk management frameworks. This review examined the frameworks and their specific strengths and weaknesses in

how they managed the interactions and dependencies among People, Policy, and Technology. Some examples included the NIST CSF which is a structured approach for organisations to manage and improve their cybersecurity posture. However, it has some shortcomings, such as a lack of specific control implementation guidance, challenges in customising and adapting the framework and a lack of specific maturity assessment criteria. The Cybersecurity Capability Maturity Model (C2M2) has several strengths, such as taking a holistic approach to cybersecurity by addressing people, policy and technology and acknowledging the interdependencies between controls within and across domains. However, the C2M2 also has some shortcomings, such as it needs to provide detailed, step-by-step implementation guidance for individual controls; it does not provide specific criteria or metrics for measuring the maturity of controls in detail. ISO/IEC 27001 also has shortcomings, such as a lack of specific control implementation guidance, challenges in customising and adapting the framework and a lack of maturity assessment criteria. Similarly, the Center for Internet Security (CIS) Controls have specific shortcomings such as like the NIST CSF; the CIS Controls provide high-level descriptions of controls but do not offer detailed, step-by-step implementation guidance; the CIS Controls provide a prioritised list of controls, organisations must still customise them to their specific needs, the CIS Controls offer a framework for assessing an organisation's cybersecurity maturity. However, they do not provide specific metrics or criteria for measuring the maturity of controls related to people, policy, and technology.

Research Question 2 (RQ2):

> How does employing a directed graph-based framework improve the representation of an enterprise's intricate interactions and dependencies among People, Policy, and Technology? In what ways does this graph-based approach provide more insights or depth compared to traditional frameworks?

In Chapter 5, we looked at basic graph theory, a branch of mathematics that studies structures called graphs. These are abstract representations with nodes and edges connecting them. Graphs are categorised as directed (digraphs) or undirected based on the directionality of their edges. Directed graphs (digraphs) illustrate asymmetric relationships through their edges, which have intrinsic directionality. Each edge in a directed graph starts from a source node and ends at a destination node. Edges in directed graphs show unidirectional interactions or influences between nodes. In a graph, Nodes represent entities or objects, and Edges depict relationships among the nodes/entities. In Chapter 5 we looked at using directed graphs for modelling information security maturity, and similarly, in Chapter 6, we examined using directed graphs for modelling information security risk. We found that they are a powerful tool for assessing an organisation's compliance with a cybersecurity framework. They can be used to visualise the organisation's cybersecurity landscape, identify data sources, understand interdependencies, assess maturity levels, and identify gaps. In the planning stage of a cybersecurity maturity review, directed graphs can be used to define the scope and objectives of the review, identify the data sources that need to be collected and understand the interdependencies between different components of the organisation's cybersecurity practices. In the data-gathering stage, they can organise and categorise the data sources, identify the relationships between data sources, assess the completeness of data gathering and visualise the flow of data within the organisation's cybersecurity practices. In the assessment stage, directed graphs can be used to visualise the maturity levels defined in the cybersecurity framework, conduct a comparative analysis of the organisation's current practices against the desired practices, identify areas of strength and weakness in the organisation's cybersecurity practices, understand the interdependencies and relationships between different components of the organisation's cybersecurity practices and assign scores or ratings to the maturity levels of different nodes.

Research Question 3 (RQ3):

How does employing a directed graph in a maturity assessment framework contribute to a more nuanced and actionable understanding of enterprise information and cybersecurity maturity?

Employing a directed graph in a maturity assessment framework can contribute to a more nuanced and actionable understanding of enterprise information and cybersecurity maturity by visualising relationships. As stated in Chapter 5, a directed graph can be used to visualise the relationships between different entities and processes, such as assets, systems, and policies. This can help organisations understand how these entities and processes interact and how they contribute to the overall cybersecurity posture. For example, the graph could visualise the relationships between assets like computers, servers, and networks. This could help organisations identify which assets are most critical to their operations and most vulnerable to attack. As discussed in Chapter 5, a directed graph can be used to identify gaps between the organisation's current maturity level and the desired maturity level. This can help organisations prioritise their efforts and focus on the areas where they need to improve most. For example, the graph could identify gaps between the organisation's current security controls and the controls required to achieve the desired maturity level. We also discussed in Chapter 6 that a directed graph can measure the organisation's progress over time as it improves its cybersecurity posture. This can help organisations track their progress and ensure they are on track to achieve their goals. For example, the graph could track the number of security incidents that have occurred over time, or the number of security controls implemented. In Chapter 7, we discussed that a directed graph can be used to communicate the findings of the maturity assessment to stakeholders. This can help stakeholders to understand the organisation's cybersecurity posture and the areas where improvements are needed. For example, the graph could visually represent the organisation's cybersecurity maturity or create a report summarising the assessment findings. Overall, employing a directed graph in a maturity assessment framework can be valuable for organisations serious about

improving their cybersecurity posture. It can help organisations visualise the relationships between different entities and processes, identify gaps, measure progress, and communicate findings.

Research Question 4 (RQ4):

> How does employing a directed graph in a cybersecurity risk assessment contribute to a more nuanced and actionable understanding of enterprise information and cybersecurity risk?

We discussed visualising the relationships between assets, such as computers, servers, and networks. This could help organisations identify which assets are most critical to their operations and most vulnerable to attack. The graph could also visualise the relationships between malware, phishing, and social engineering threats. This could help organisations identify which threats are most likely to occur and which have the most significant impact. We also discussed that directed graphs can identify gaps between the organisation's current and desired risk profiles. This can help organisations prioritise their efforts and focus on the areas where they need to improve most. Overall, employing a directed graph in a cybersecurity risk assessment can contribute to a more nuanced and actionable understanding of enterprise information and cybersecurity risk by visualising relationships as stated in Chapter 5, identifying gaps as stated in Chapter 6, Measuring progress as stated in Chapter 7 and communicating findings: As stated in Chapter 7, the graph can be used to communicate the findings of the risk assessment to stakeholders. This can help stakeholders understand the organisation's cybersecurity risk profile and the areas where improvements are needed.

Research Question 5 (RQ5):

> How does employing a directed graph to undertake combined cybersecurity maturity and risk assessment contribute to a more nuanced and actionable understanding of enterprise information and cybersecurity maturity and risk?

Like questions RQ3 and RQ4, we discussed an improved understanding of cybersecurity posture in Chapter 5 and how combining maturity and risk assessment can help organisations get a more holistic

view of their cybersecurity posture. Better prioritisation of security investments: as discussed in Chapter 6, organisations can better prioritise their security investments by understanding the relationship between maturity and risk. This can help them to get the most bang for their buck and to reduce their overall risk exposure. Increased efficiency and effectiveness of security controls can be achieved, as discussed in Chapter 7; by understanding the maturity level of their security controls, organisations can ensure that they are using the proper controls in the right places. This can help them improve their security posture's efficiency and effectiveness. Additionally, enhanced compliance with security standards is possible, as discussed in Chapter7. By understanding their maturity level and risk exposure, organisations can better assess their compliance with security standards. Finally, improved decision-making, as discussed in Chapter 7, by combining maturity and risk assessment, organisations can make better decisions about their cybersecurity posture. This can help them to reduce their risk exposure and to improve their overall security posture.

9.4 Reflections on the Research

The research on combined cybersecurity maturity and risk assessment is still in its initial stages. This means that there is still a lot that we still need to learn about this approach. For example, we need to find out the full benefits of this approach or the challenges organisations may face when implementing it.

However, the research did find that combined maturity and risk assessment can help organisations to:

Develop a more holistic view of their cybersecurity posture.

Better prioritise their security investments.

Increase the efficiency and effectiveness of their security controls.

Enhance their compliance with security standards.

Make better decisions about their cybersecurity posture.

The research also found several challenges to implementing combined maturity and risk assessment. These challenges include:

One of the challenges of combined maturity and risk assessment is the need for a mutual understanding of maturity and risk. This is because maturity and risk are two different concepts that different people can interpret differently. For example, one organisation may define maturity as the extent to which an organisation has implemented security controls. In contrast, another organisation may define maturity as the extent to which an organisation has implemented security controls that effectively reduce risk.

Another challenge of combined maturity and risk assessment is the need for a methodology to combine the two approaches effectively. This is because maturity and risk assessment are two different methodologies with different strengths and weaknesses. For example, maturity assessment can identify areas where an organisation needs to improve its security posture, while risk assessment can be used to prioritise security investments.

Finally, the combined maturity and risk assessment research is focused on large organisations. We must determine how this approach can be applied to small and medium-sized organisations. Small and medium-sized organisations may need more resources to implement a combined maturity and risk assessment, or they may need more expertise.

Despite these challenges, the research suggests that combined maturity and risk assessment is a promising approach to improving cybersecurity posture. Organisations that are serious about improving their cybersecurity posture should consider this approach.

The research is still in its early stages, and more research is needed to fully understand the benefits and challenges of combined maturity and risk assessment.

The research is focused on large organisations, and more research is needed to understand how this approach can be applied to small and medium-sized organisations.

- The research is focused on traditional cybersecurity risks, and more research is needed to understand how this approach can be applied to emerging risks, such as artificial intelligence and quantum computing.

Overall, the research on combined cybersecurity maturity and risk assessment is promising. This approach has the potential to help organisations improve their cybersecurity posture and reduce their risk exposure. However, more research is needed to understand the benefits and challenges of this approach entirely.

Chapter 10 - Concluding Remarks and Future Work

This chapter summarises our contributions relating to the research questions and discusses the potential future directions derived from the findings in this thesis.

10.1 Contributions

10.1.1 Graph Schema

The research has developed a comprehensive graph schema that is capable of accurately modelling information security risk and compliance in an enterprise environment. The research proposes a standardised set of node types, including Threat Actor, Attack, Vulnerability, Asset, Controls and Objective which provide specific and meaningful context to the graph allowing users with information security experience to implicitly understand the construct and relationships within the graph.

10.1.2 Standardised Assessment Model

The research provides a rigorous and repeatable model for evaluating information security maturity within an enterprise, making the assessment consistent across scales and contexts along with the associated formulas for determining consistent assessments; further the schema and formulas have been implemented as algorithms within the CyConex application. As part of the graph schema, nodes have a defined taxonomy of parameters used to support the production of accurate and quantifiable metrics.

10.1.3 Maturity and Risk Reduction Tools

The research provides organisations with tools that will allow for more informed decision-making regarding information security risks while reducing these risks. This aim has been fully achieved through the design of the graph schema and the development of the CyConex application which implements the graph schema and associated calculations. This coupled with additional capabilities within the application to visualise risk and compliance aspects of graphs through dashboards and heat

maps for example, users of the application can have significantly improved understanding and better-informed decision-making.

The CyConex application will be available free of charge for non-commercial use or for evaluation in commercial organisations from www.cyconex.com. This will allow organisations to develop their own directed graphs for specific circumstances.

10.2 Future Work

This section concludes the thesis by summarising some ideas for future research in the use of directed graphs for assessing information security maturity and risk assessment.

10.2.1 Enhancements to the Graph Schema

The current graph schema only includes a limited number of entities. We can add more entities and relationships to make the graph schema more comprehensive. For example, the creation of a Threat node would allow risks to be calculated without necessarily identifying specific vulnerabilities or attacks.

Use more sophisticated data types; for example, the current graph schema uses simple data types, such as strings and integers. Using more sophisticated data types, such as dates or even unstructured data, could allow for a more accurate representation of cybersecurity maturity and risk.

10.2.2 Machine Learning and Artificial Intelligence

Machine learning could be examined in future work on the graph schema; for example, machine learning could be used to identify patterns in the data. This could be used to identify relationships between entities and relationships that are not easily identifiable by humans. Machine learning could also be used to predict future events. This could be used to predict the likelihood of a risk occurring or the impact of a risk if it does occur. Machine learning could also be used to automate tasks. This could be used to automate the process of collecting and analysing data, as well as the process of generating reports and recommendations.

Artificial intelligence (AI) could be examined in future work; for example, AI could be used to identify and prioritise risks. This could be done by using AI to analyse the data in the graph and identify risks that are most likely to occur or have the most significant impact. AI could be used to develop mitigation strategies. This could be done by using AI to generate recommendations for how to mitigate the risks that have been identified. AI could also be used to automate tasks. This could be done by using AI to automate the process of collecting and analysing data, as well as the process of generating reports and recommendations.

References

Aksu, G., Ozturk, F., & Karakose, M. (2017). A new asset and vulnerability-centric quantitative model for IT system risk assessment. Computers & Security, 70, 330-343.

Alam, M. M., Islam, M. S., Hossain, M. A., & Hossain, M. A. (2023). A graph-based model for cybersecurity risk management. IEEE Access, 11, 103122-103135.

Alam, M. M., Islam, M. S., Hossain, M. A., & Hossain, M. A. (2023). A graph-based model for cybersecurity risk assessment in cloud computing. Computers & Security, 104, 102496.

Alhajri, R. M., Alsunaidi, S. J., Zagrouba, R., Almuhaideb, A. M., & Alqahtani, M. A. (2019). Dynamic Interpretation Approaches for Information Security Risk Assessment. In 2019 International Conference on Computer and Information Sciences (ICCIS) (pp. 1-8). IEEE.

Baras, J., Van de Velde, S., & Van den Brande, D. (2014). A critical review of information security management systems models. Information & Management, 51(5), 491-508.

Bhattacharjee, J., Sengupta, A. & Mazumdar, C. (2013). "A Formal Methodology for Enterprise Information Security Risk Assessment," *International Conference on Risks and Security of Internet and Systems*.

Cheng, X. & Zhang, Z. (2010). Risk Assessment of Information System using the Shell Theory and Attack Graph. 2010 3rd International Conference on Computer Science and Information Technology, 5, 537–541. https://doi.org/10.1109/iccsit.2010.5563917

Dehghantanha, A., Conti, A., & Wang, H. (2023). A survey on graph-based methods for cybersecurity risk management. IEEE Communications Surveys & Tutorials, 25(1), 101-134.

Fernandez, A. and Garcia, D. F. (2016). "Complex vs Simple Asset Modeling Approaches for Information Security Risk Assessment Evaluation with MAGERIT methodology," *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, pp. 542–549. doi 10.1109/intech.2016.7845064.

Genchev, P. (2020). “An approach to support information security risk assessment,” *2020 International Conference on Biomedical Innovations and Applications (BIA)*, 00, pp. 125–128. doi: 10.1109/bia50171.2020.9244516.

GraphSPD. (2023). Graph-based Security Patch Detection using Graph Neural Networks. IEEE. https://ieeexplore.ieee.org/document/10179479

Juma A, Arman A, Hidayat F. (2023). Cybersecurity Assessment Framework: A Systematic Review .IEEE. https://ieeexplore.ieee.org/document/10291832

Keramati, M. (2016). An Attack Graph-Based Procedure for Risk Estimation of Zero-Day Attacks. 2016 8th International Symposium on Telecommunications (IST), pp. 723–728. https://doi.org/10.1109/istel.2016.7881918

Koch, M., Mancini, L. V. & Parisi-Presicce, F. (2000). Computer Security - ESORICS 2000, 6th European Symposium on Research in Computer Security, Toulouse, France, October 4-6, 2000. Proceedings. Lecture Notes in Computer Science, 122–139. https://doi.org/10.1007/10722599_8

Kumar, R., et al. (2023). *Graph Theory Matrix Approach in Cryptography and Network Security*. IEEE. https://ieeexplore.ieee.org/document/10202460

Lagraa, S., et al. (2023). Graph-based Data Representation for Network Security Monitoring. ACM Digital Library. https://dl.acm.org/doi/10.1007/s10207-023-00742-7

Leszczyna. R (2024). Cybersecurity Assessment MethodsWhy Aren’t They Used?. IEEE. https://ieeexplore.ieee.org/document/10693545

Opplige, R. (2018). “Quantitative Risk Analysis in Information Security Management: A Modern Fairy Tale,” *EEE Computer and Reliability Societies*.

Nyanchama, M. & Osborn, S. (1999). The role graph model and conflict of interest. ACM Transactions on Information and System Security (TISSEC), 2(1), 3–33. https://doi.org/10.1145/300830.300832

Oppliger, R. (2015, May). Quantitative risk analysis in information security management: A modern fairy tale. In 2015 IEEE International Workshop on Managing the Software Process (MSP) (pp. 136-145). IEEE.

Sengupta, A., Manna, A. & Mazumdar, C. (2013). A Graph-Based Approach for Managing Enterprise Information System Security. 2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies, pp. 137–143. https://doi.org/10.1109/cube.2013.33

Shamala, P., Ahmad, R., Zolait, A. H. & Sahib, S. bin (2015). "Collective information structure model for Information Security Risk Assessment (ISRA)," *Journal of Systems and Information Technology*, Volume 17(Issue 2), pp. 193–219. doi: 10.1108/jsit-02-2015-0013.

Sommestad, T., Ekstedt, M. & Johnson, P. (2009). Cyber Security Risks Assessment with Bayesian Defense Graphs and Architectural Models. 2009 42nd Hawaii International Conference on System Sciences, 1, 1–10. https://doi.org/10.1109/hicss.2009.141

Springer Cybersecurity. (2018). Graph-based Visual Analytics for Cyber Threat Intelligence. SpringerOpen. https://cybersecurity.springeropen.com/articles/10.1186/s42400-018-0017-4

Phillips, C., & Swiler, L. (1998). A Graph-Based System for Network-Vulnerability Analysis. ACM Digital Library. https://dl.acm.org/doi/pdf/10.1145/310889.310919

Välja, T., Rõõm, T., & Veltman, K. (2015). Enhancing Attack Graph Analysis with Interoperability and Availability Considerations. Security & Privacy, IEEE, 13(5), 46-54.

Wangen, G. (2017). "Information Security Risk Assessment: A Method Comparison," *Norwegian University of Science and Technology*.

Wangen, H., Hallstensen, S., & Snekkenes, E. (2018). Identifying critical assets using a network-based approach. Computers & Security, 77, 259-271.

Appendices

A1 - Case Study Vulnerabilities Added to the Graph

API Key Exposure	API Key Exposure occurs when API keys are unintentionally exposed in public repositories' code or logs, allowing unauthorised access to services and potentially leading to data breaches or service misuse.
API Misconfiguration	API Misconfiguration occurs when APIs are improperly set up, leading to unauthorised access data exposure or security flaws due to inadequate authentication, excessive permissions, or lack of encryption.
Backup Failures	"Backup Failures" refer to the inability to create, maintain, or restore data backups, leading to potential data loss, increased downtime, and vulnerability to ransomware or other data-compromising incidents.
Broken Authentication and Session Management	Weak or broken authentication mechanisms allow attackers to bypass login protections, hijack sessions, or exploit user credentials.
Broken Function Level Authorization	Broken Function-Level Authorization occurs when applications fail to properly enforce user permissions. This allows unauthorized users to access restricted functions or data, potentially leading to data breaches or privilege escalation.
Bypassed URL Filtering	Bypassed URL Filtering is a vulnerability in which security mechanisms fail to restrict access to specific URLs. This allows

	unauthorised users to access blocked or sensitive content, potentially leading to data breaches.
Certificate Validation Flaws	Certificate Validation Flaws occur when systems improperly verify digital certificates, allowing attackers to intercept, alter, or forge communications, potentially leading to man-in-the-middle attacks and unauthorised data access.
Cloud Security Misconfigurations	Cloud Security Misconfigurations occur when cloud resources are improperly set up. These misconfigurations can lead to unauthorised access, data exposure, or breaches due to incorrect permissions, a lack of encryption, or inadequate security controls.
Container Image Vulnerabilities	Container Image Vulnerabilities refer to security weaknesses in container images, often due to outdated software misconfigurations or embedded secrets, which can be exploited to compromise containerised applications and environments.
Container Security Flaws	Container security flaws involve misconfigurations, outdated images, or inadequate isolation, allowing unauthorised access privilege escalation or data breaches.
Cross-Account Access Misconfigurations	Cross-account access Misconfigurations occur when permissions are improperly set, allowing unauthorised access between different accounts or cloud environments, potentially leading to data breaches or unauthorised resource manipulation.
Cross-Region Data Replication Risks	Cross-Region Data Replication Risks involve unauthorised access data breaches or compliance violations due to improper configuration or insufficient security controls when replicating data across different geographic regions.

Cryptographic Flaws	Cryptographic flaws refer to encryption algorithms or implementations that allow attackers to decrypt, alter, or forge data, compromising confidentiality integrity.
Data Leakage through Misconfigured Storage	Data Leakage through Misconfigured Storage occurs when sensitive data is exposed due to improperly secured storage systems, such as cloud buckets or databases, which allow unauthorised access and potential data breaches.
Data Loss from Accidental Deletion	Data Loss from Accidental Deletion occurs when users unintentionally delete critical data, often due to inadequate access controls, lack of backups, or insufficient user training, leading to potential operational disruptions.
Default Credentials	Default Credentials vulnerability occurs when systems use factory-set usernames and passwords, making them susceptible to unauthorised access if not changed, as these credentials are often publicly known or easily guessable.
Environment Variable Manipulation	Environment Variable Manipulation involves altering environment variables to influence program behaviour. This can potentially lead to unauthorised access privilege escalation or the execution of arbitrary code within the affected application.
Excessive Privileges	Excessive Privileges occur when users or applications have more access rights than necessary, increasing the risk of unauthorised actions, data breaches, or system compromise due to misuse or exploitation.
Exposed Secrets and Keys	Exposed Secrets and Keys vulnerability occurs when sensitive information, such as API keys, passwords, or cryptographic keys,

	is inadvertently included in public repositories or logs, risking unauthorised access and data breaches.
Failure to Implement Secure Coding Practices	Developers who do not follow secure coding standards introduce vulnerabilities that attackers can exploit, such as SQL injection or buffer overflow.
Improper Authentication	Improper Authentication occurs when a system fails to verify user identities correctly, allowing unauthorised access. This can result from weak password policies, flawed authentication mechanisms, or inadequate session management.
Improper Identity and Access Management	Improper Identity and Access Management occurs when user identities and permissions are poorly managed, leading to unauthorised access to data breaches and compromised systems due to inadequate authentication and authorisation controls.
Inadequate Data Backup and Recovery	Inadequate Data Backup and Recovery refers to insufficient or poorly managed backup systems that risk data loss and prolonged downtime during incidents due to incomplete, outdated, or inaccessible backups and recovery processes.
Inadequate Encryption of Data at Rest	This vulnerability occurs when sensitive data stored on a device or server is not encrypted or is inadequately encrypted, making it susceptible to unauthorised access and potential data breaches.
Inadequate Encryption Strength	Inadequate Encryption Strength refers to the use of weak cryptographic algorithms or insufficient key lengths, which makes encrypted data susceptible to decryption by attackers and compromises confidentiality and data integrity.

Inadequate IAM Policies	Inadequate IAM Policies refer to insufficient identity and access management controls, which can lead to unauthorised access privilege escalation and potential data breaches due to poorly defined roles, permissions, and authentication mechanisms.
Inadequate Input Validation	Inadequate Input Validation occurs when a system fails to properly check user inputs. This allows attackers to inject malicious data, potentially leading to unauthorised access data breaches or system compromise.
Inadequate Monitoring and Alerting	Inadequate Monitoring and Alerting refers to insufficient systems or processes for detecting logs and alerting on suspicious activities or anomalies, increasing the risk of undetected breaches and delayed incident response.
Inadequate Protection Against Insider Threats	Insufficient insider threat protection allows employees or contractors to misuse their access to steal data, sabotage systems, or engage in other malicious activities.
Inadequate Resource Isolation	Inadequate Resource Isolation occurs when systems or applications fail to properly segregate resources, allowing unauthorised access or interference, potentially leading to data breaches, privilege escalation, or service disruptions.
Inadequate Scalability	Inadequate Scalability is a vulnerability in which a system fails to handle increased load or growth, leading to performance degradation, potential downtime, and increased susceptibility to attacks during high-demand periods.
Inadequate Secrets Management	Inadequate Secret Management refers to the improper handling, storage, or transmitting sensitive information like passwords, API

	keys, or tokens, which can lead to unauthorised access and potential data breaches.
Inadequate Security Group Rules	"Inadequate Security Group Rules" refers to overly permissive or improperly configured network access controls that allow unauthorised access or exposure to potential threats, increasing the risk of data breaches or attacks.
Inadequate VLAN Segmentation	Inadequate VLAN Segmentation occurs when network traffic is improperly isolated, allowing unauthorised access between VLANs, increasing the risk of data breaches, lateral movement, and exposure to attacks within the network.
Incomplete Visibility into Cloud Usage	"Incomplete Visibility into Cloud Usage" refers to the lack of comprehensive monitoring and understanding of cloud resources and activities, which can lead to potential security risks due to untracked data applications or user actions.
Insecure Access Control Policies	Insecure Access Control Policies occur when systems fail to enforce proper restrictions on user permissions, allowing unauthorised access to sensitive data or functions, potentially leading to data breaches or system compromise.
Insecure API	Insecure API vulnerabilities arise when APIs lack proper authentication authorisation or data validation, allowing attackers to exploit endpoints, access sensitive data, or perform unauthorised actions.
Insecure API Access	Insecure API Access occurs when APIs lack proper authentication authorisation or encryption, allowing unauthorised users to

	exploit them, potentially leading to data breaches, unauthorised data access
Insecure API Endpoints	Insecure API endpoints expose sensitive data or functionality due to inadequate authentication authorisation or input validation, allowing attackers to exploit these weaknesses for unauthorised access or data breaches.
Insecure API Exposure	Insecure API Exposure occurs when APIs lack proper authentication authorisation or encryption, allowing unauthorized access to sensitive data, leading to data breaches and potential exploitation by attackers.
Insecure API Gateways	Insecure API Gateways expose systems to threats by lacking proper authentication authorisation and data validation, potentially allowing unauthorised access to data breaches and exploitation of backend services.
Insecure API Management	Insecure API Management refers to inadequate security controls in API design implementation or configuration, which can lead to unauthorised access to data breaches or exploitation by attackers due to exposed endpoints or insufficient authentication.
Insecure Default Settings	Default configurations often prioritise convenience over security, leaving systems vulnerable to attack. Failure to change insecure defaults allows attackers easy access.
Insecure DevOps Practices	Insecure DevOps Practices involve inadequate security measures in development and operations, leading to risks like exposed credentials, insufficient access controls, and unpatched software,

	which can potentially compromise application integrity and data confidentiality.
Insecure Handling of User Data	Insecure Handling of User Data occurs when applications improperly store, transmit or process user data, leading to unauthorised access data breaches or exposure to malicious actors.
Insecure Key Management	Insecure Key Management refers to the improper handling, storage or transmission of cryptographic keys leading to unauthorised access data breaches or compromised encryption.
Insecure Key Management Practices	Insecure Key Management Practices involve improper handling, storage, or distribution of cryptographic keys, leading to unauthorised access data breaches or compromised encryption.
Insecure OAuth Implementations	Insecure OAuth implementations can lead to unauthorised access by improperly handling tokens that lack proper validation or misconfiguring scopes, allowing attackers to exploit authentication and authorisation processes.
Insecure Permissions	Insecure Permissions occur when files, directories or resources have overly permissive access controls allowing unauthorised users to read, modify or execute them.
Insecure Remote Management Access	Insecure Remote Management Access refers to inadequate security measures in remote management interfaces, which allow unauthorised access to data breaches or system control due to weak authentication, unencrypted connections, or default credentials.

Insecure REST API Configurations	Insecure REST API configurations involve improper authentication authorisation or data validation, exposing sensitive data or system functionality to unauthorised access and potentially leading to data breaches or system compromise.
Insecure Storage Configurations	Insecure Storage Configurations refer to improperly secured data storage systems that expose sensitive information to unauthorised access due to misconfigurations, weak encryption, or inadequate access controls, increasing the risk of data breaches.
Insecure Third-Party Components	Insecure Third-Party Components refer to vulnerabilities arising from using outdated or untrusted external libraries, plugins, or modules, which can potentially introduce security risks and exploits into an otherwise secure system.
Insecure Third-Party Libraries	Insecure Third-Party Libraries are external code dependencies with known vulnerabilities that attackers can exploit, potentially compromising the security of applications that rely on them.
Insecure Transport Layer Security	Insecure Transport Layer Security refers to using outdated or misconfigured TLS protocols, which expose data to interception or tampering during transmission and compromise the confidentiality and integrity of communications.
Insecure VM Migration	Insecure VM Migration refers to the risk of data interception or unauthorised access during the transfer of virtual machines between hosts, often due to a lack of encryption or inadequate authentication measures.
Insufficient Audit Trail	"Insufficient Audit Trail" refers to inadequate logging and monitoring of system activities. This hinders the ability to detect,

	investigate, and respond to unauthorised access or anomalies that compromise security and compliance.
Insufficient Authorization	Insufficient Authorization occurs when a system fails to properly enforce permissions. This allows unauthorised users to access restricted resources or perform actions beyond their intended privileges, leading to potential data breaches.
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Weak security in IaaS environments allows attackers to compromise cloud infrastructure, leading to unauthorised access, data breaches, or service disruptions.
Insufficient DDoS Protection	Insufficient DDoS Protection refers to inadequate measures to detect, mitigate, or withstand Distributed Denial of Service attacks, potentially leading to service disruption, degraded performance, or complete unavailability of online resources.
Insufficient Incident Response Procedures	"Insufficient Incident Response Procedures" refers to inadequate or poorly defined processes for detecting, responding to, and recovering from security incidents. These can lead to prolonged exposure, increased damage, and ineffective mitigation.

A2 - Case Study Attacks Added to the Graph

Abuse Elevation Control Mechanism	Abuse Elevation Control Mechanism involves exploiting legitimate elevation control mechanisms to gain higher-level privileges on a system. Attackers might use methods like bypassing User Account Control (UAC) on Windows to execute payloads with elevated privileges, allowing them to manipulate

	protected system resources or execute code with administrative rights.
Account Access Removal	Account Access Removal involves removing or disabling user accounts to disrupt access to systems or services. Attackers may use this technique to lock out legitimate users, maintain control over compromised accounts, or disrupt operations.
Account Discovery	Account Discovery involves identifying accounts on a system or within an Active Directory environment. Attackers use this technique to find privileged accounts, service accounts, or other user accounts that they can exploit for lateral movement or privilege escalation.
Account Manipulation	Account Manipulation refers to the alteration or creation of accounts on a system to maintain access, escalate privileges, or impersonate other users. Attackers may create new accounts with elevated privileges or modify existing ones to suit their needs.
Activation Of Payloads	Activation of Payloads refers to executing malicious code on a compromised system. Attackers may use various triggers to activate payloads, such as specific dates, user actions, or system events, to achieve their objectives, such as data theft or system disruption.
Appcert Dlls	AppCert DLLs is a technique where attackers exploit the AppCertDLLs registry key in Windows to load a malicious DLL when a process starts. This method is used to achieve persistence,

	execute code in the context of a trusted process, or evade security controls.
Applescript	AppleScript is a scripting language used on macOS systems. Attackers use AppleScript to automate tasks, execute commands, or manipulate applications on a compromised macOS system, often as part of a broader attack strategy.
Application Access Token	Application Access Token refers to tokens used by applications to authenticate with other services. Attackers may steal these tokens to gain unauthorized access to services, allowing them to bypass authentication mechanisms and impersonate legitimate users.
Application Layer Protocol	Application Layer Protocol involves using standard network protocols, such as HTTP, HTTPS, or DNS, to communicate with compromised systems or command-and-control servers. Attackers use these protocols to blend in with legitimate traffic, making it difficult for security tools to detect malicious activity.
Application Layer Protocol-Dns	Application Layer Protocol: DNS refers to using the DNS protocol for command and control (C2) communication. Attackers may encode data within DNS queries and responses to communicate with compromised systems, often evading detection by blending in with legitimate DNS traffic.
Application Layer Protocol-HttpS	Application Layer Protocol: HTTP/S involves using the HTTP or HTTPS protocols for command and control (C2) communication. Attackers use these protocols to blend in with normal web traffic, making it difficult for security tools to detect and block malicious activity.

Application Layer Protocol-Websocket	Application Layer Protocol: WebSocket is a protocol that provides full-duplex communication channels over a single TCP connection. Attackers may use WebSockets for command and control (C2) communication to maintain persistent, real-time connections with compromised systems.
Application Window Discovery	Application Window Discovery is the process of identifying open application windows on a system. Attackers use this technique to understand the context of a user's session or to identify specific applications that are running, which can help in planning further attacks or understanding user behaviour.
Automated Collection	Automated Collection refers to the use of scripts or automated tools to collect data from compromised systems without manual intervention. Attackers use this technique to efficiently gather large volumes of information, such as files, credentials, or system configurations, often in preparation for exfiltration.
Automated Collection- Input Capture	Automated Collection: Input Capture refers to the automated capture of user input, such as keystrokes or mouse movements, by malicious tools. Attackers use this technique to systematically collect input data over time, allowing them to gather credentials, search terms, or other sensitive information.
Automated Exfiltration	Automated Exfiltration involves using scripts or automated tools to exfiltrate data from a compromised system to an attacker-controlled location. This technique reduces the need for manual intervention, allowing attackers to stealthily transfer large volumes of data over time.

Bash History	Bash History involves accessing or manipulating the command history file (.bash_history) on Linux or Unix systems to recover sensitive information such as previously executed commands or credentials. Attackers may also clear or alter the bash history to hide their activities.
Boot Or Logon Autostart Execution	Boot or Logon Autostart Execution is a technique where attackers configure malicious code to execute automatically during the system boot or user logon process. This is often achieved by adding entries to startup folders, modifying registry keys, or abusing legitimate autostart mechanisms.
Boot Or Logon Initialization Scripts	Boot or Logon Initialization Scripts involve placing or modifying scripts that run during system boot or user logon to execute malicious code. These scripts can be used to establish persistence or to execute payloads under specific user contexts.
Browser Extensions	Browser Extensions refers to the use or manipulation of browser extensions to execute malicious code or steal sensitive information. Attackers may develop or modify browser extensions to capture credentials, monitor user activity, or manipulate web traffic.
Brute Force	Brute Force is a method of gaining access to accounts by systematically trying all combinations of passwords until the correct one is found. Attackers often automate this process to attempt multiple passwords in quick succession, targeting accounts with weak or default credentials.

Clipboard Data	Clipboard Data involves capturing the contents of the clipboard on a compromised system. Attackers use this technique to capture sensitive information such as passwords, account numbers, or other data that users copy and paste between applications.
Cloud Infrastructure Discovery	Cloud Infrastructure Discovery involves identifying the components and services used in a cloud environment. Attackers use this technique to understand the architecture, find misconfigurations, or locate targets for further attacks, such as virtual machines, storage, or network settings.
Cloud Service Dashboard	Cloud Service Dashboard refers to accessing or discovering management interfaces or dashboards for cloud services. Attackers use this technique to gain visibility into the cloud environment, manage resources, or execute actions that can lead to further compromise.
Cloud Service Discovery	Cloud Service Discovery involves identifying cloud services in use within an organization. Attackers use this technique to understand the cloud environment, locate valuable assets, or find potential misconfigurations that could be exploited for further access or data exfiltration.
Cloud Storage Object Discovery	Cloud Storage Object Discovery is the process of identifying storage objects, such as files or databases, within cloud environments. Attackers use this technique to locate sensitive data, backup files, or configuration information that can be leveraged for further exploitation or data theft.

Command And Scripting Interpreter	Command and Scripting Interpreter refers to the use of command line interfaces and scripting languages (e.g., PowerShell, Bash) to execute malicious code on a system. Attackers use these interpreters to perform a wide range of actions, including executing commands, downloading payloads, and manipulating files.
Create Account	Create Account is a technique where attackers create new user accounts on a compromised system to maintain access. The new account may have elevated privileges and can be used to carry out further attacks while evading detection.
Create Or Modify System Process	Create or Modify System Process is a technique where attackers create or alter system processes to execute malicious code. This could involve creating new processes with elevated privileges, injecting malicious code into existing processes, or modifying the execution parameters of critical system processes to achieve their objectives.
Credentials In Files	Credentials in Files involves searching for stored credentials in files on the system. Attackers look for configuration files, scripts, or documents that contain plaintext or easily decryptable passwords, tokens, or keys, which can then be used to gain unauthorized access to systems or services.
Credentials In Registry	Credentials in Registry involves searching the Windows Registry for stored credentials. Attackers may look for keys or values where applications or services store plaintext or weakly encrypted

	passwords, allowing them to gain unauthorized access to systems or services.
Data Compressed	Data Compressed refers to the use of compression algorithms to reduce the size of data before exfiltration. Attackers use compression to minimize the amount of data transmitted over the network, reducing the chances of detection by security tools.
Data Destruction	Data Destruction is a technique where attackers delete or overwrite data to disrupt operations, prevent recovery, or destroy evidence. Attackers may use this technique to cover their tracks, inflict financial damage, or achieve other malicious goals.
Data Encoding	Data Encoding refers to transforming data into a different format, such as Base64, to evade detection or facilitate exfiltration. Attackers use encoding to bypass content filters, avoid detection by security tools, or ensure data integrity during transmission to external locations.
Data Encrypted	Data Encrypted involves encrypting data before exfiltration to protect its contents and avoid detection by security tools. Attackers use encryption to ensure that intercepted data cannot be easily read or analysed by defenders.
Data Encrypted For Impact	Data Encrypted for Impact involves encrypting data on a target system to make it unusable, often as part of a ransomware attack. Attackers demand payment in exchange for the decryption key, using encryption to hold data hostage and extort victims.
Data From Cloud Storage Object	Data from Cloud Storage Object refers to the collection of data stored in cloud storage services, such as files, databases, or

	backups. Attackers target cloud storage objects to exfiltrate sensitive information, access configuration data, or identify further targets for exploitation.
Data From Cloud Storage Object- Automated Collection	Data from Cloud Storage Object: Automated Collection involves the use of automated tools to continuously monitor and collect data from cloud storage services. Attackers use this technique to ensure they capture all relevant information, such as newly added files or updated data, without manual intervention.
Data From Information Repositories	Data from Information Repositories involves collecting data from structured or unstructured information repositories, such as databases, document management systems, or code repositories. Attackers target these repositories to access sensitive information, intellectual property, or operational data.
Data From Information Repositories- Automated Collection	Data from Information Repositories: Automated Collection refers to the use of automated tools to gather data from information repositories, such as databases or document management systems. Attackers use this technique to streamline the collection process and ensure they capture a comprehensive set of data.
Data From Local System	Data from Local System involves collecting data directly from the storage or memory of a compromised system. Attackers may search for valuable files, configuration settings, or sensitive information stored on the local system, which can then be used for further attacks or exfiltration.
Data From Network Shared Drive	Data from Network Shared Drive refers to the collection of data from shared network drives accessible within a network.

	Attackers target these shared resources to gather information, such as shared documents, backups, or user data, that can be used for further exploitation.
Data From Network Shared Drive- Automated Collection	Data from Network Shared Drive: Automated Collection involves using automated tools to continuously monitor and collect data from shared network drives. Attackers use this technique to gather data over time, ensuring they capture any new or modified files that may contain valuable information.
Data Manipulation	Data Manipulation refers to the unauthorized modification of data to achieve malicious objectives. Attackers may alter data to disrupt operations, manipulate outcomes, or cover their tracks, often targeting financial, operational, or reputational aspects of a business.
Data Manipulation- Stored Data Manipulation	Data Manipulation: Stored Data Manipulation involves altering stored data to achieve malicious objectives. Attackers may modify databases, files, or configuration data to disrupt operations, manipulate outcomes, or cover their tracks.
Data Obfuscation	Data Obfuscation involves altering the appearance of data to make it harder to detect or analyse. Attackers use techniques like encryption, compression, or encoding to hide malicious payloads, configuration settings, or other indicators of compromise from security tools.
Data Staged	Data Staged refers to the process of preparing collected data for exfiltration. Attackers may compress, encrypt, or segment the data and place it in specific locations within the network or on

	compromised systems, ready for extraction to an external location.
Data Transfer Size Limits	Data Transfer Size Limits refers to limiting the size of data transfers to avoid triggering network security thresholds or alarms. Attackers use this technique to stay under the radar of data loss prevention (DLP) systems or other monitoring tools.
Data Transfer Size Limits-Email Collection	Data Transfer Size Limits: Email Collection refers to limiting the size of data exfiltrated via email to avoid detection by email security filters or DLP systems. Attackers send small, incremental chunks of data to evade detection thresholds set by monitoring tools.
Deactivate Security Software	Deactivate Security Software involves disabling or tampering with security tools like antivirus, firewalls, or intrusion detection systems to avoid detection. Attackers may stop services, modify configurations, or exploit vulnerabilities in security software to reduce its effectiveness or bypass its protections.
Deobfuscate/Decode Files Or Information	Deobfuscate/Decode Files or Information is a technique used by attackers to reveal hidden or obfuscated data. Attackers may use this technique to access encoded or encrypted payloads, configuration files, or other malicious content that was hidden to avoid detection. Once decoded, the malicious content can be executed or used in further attacks.
Digital Certificate Validation	Digital Certificate Validation involves examining digital certificates used for authentication and secure communication. Attackers may use this technique to identify weaknesses in

	certificate management, exploit expired or misconfigured certificates, or impersonate legitimate services.
Direct Volume Access	Direct Volume Access is a technique where attackers directly interact with a system's storage devices to read or write data, bypassing normal file system access controls. This can be used to hide data, create malicious partitions, or exfiltrate information without triggering file-based security monitoring.
Distributed Component Object Model	Distributed Component Object Model (DCOM) is a Microsoft technology for communication between software components. Attackers exploit DCOM to execute commands or move laterally within a network, often using legitimate administrative privileges or exploiting weak configurations.
Domain Trust Discovery	Domain Trust Discovery is the process of identifying trust relationships between domains in an Active Directory environment. Attackers use this technique to understand the network's trust topology, identify targets for lateral movement, or find potential paths for privilege escalation.
Drive-By Compromise	Drive-by Compromise involves compromising a website to deliver malicious content to visitors without requiring user interaction. Attackers embed malicious scripts or exploit kits into legitimate websites, or create fake sites, which automatically exploit vulnerabilities in the visitor's browser or plugins, leading to malware installation on their systems.

Dynamic Resolution	Dynamic Resolution is a technique where attackers use domain generation algorithms (DGAs) or other dynamic methods to create domains for command and control (C2) communication. This technique allows attackers to evade detection by frequently changing domains, making it harder for defenders to block C2 channels.
Dynamic Resolution- Domain Generation Algorithms	Dynamic Resolution: Domain Generation Algorithms involve using algorithms to generate domain names for command and control (C2) communication. Attackers use DGAs to create large numbers of domains, making it difficult for defenders to predict and block all potential C2 channels.
Dynamic Resolution- Fast Flux Dns	Dynamic Resolution: Fast Flux DNS is a technique where attackers frequently change the IP addresses associated with their domains to evade detection. Fast flux makes it difficult for defenders to block malicious domains, as the underlying IP addresses are constantly changing.
Email Collection	Email Collection involves collecting emails and attachments from a compromised system. Attackers may search for emails containing sensitive information, credentials, or communication patterns that can be exploited for further attacks or intelligence gathering.
Encrypted Channel	Encrypted Channel refers to the use of encryption to protect the confidentiality and integrity of command and control (C2) communications. Attackers use encryption to prevent detection

	and analysis of their traffic by security tools, making it more difficult for defenders to identify malicious activity.
Endpoint Denial Of Service	Endpoint Denial of Service (DoS) is a technique where attackers exhaust the resources of a target endpoint, rendering it inoperable. Attackers may use this technique to disrupt services, prevent access, or create a diversion for other malicious activities.
Endpoint Denial of Service-Resource Hijacking	Endpoint Denial of Service: Resource Hijacking involves consuming the resources of an endpoint to the point of rendering it inoperable. Attackers use this technique to disrupt services, prevent access, or as part of a broader attack strategy.
Endpoint Denial of Service-Service Exhaustion Flood	Endpoint Denial of Service: Service Exhaustion Flood is a type of DoS attack that targets specific services on an endpoint, overwhelming them with requests to exhaust their resources. This technique is used to disrupt services, prevent legitimate access, or create a diversion for other attacks.
Event Triggered Execution	Event Triggered Execution refers to executing malicious code in response to specific system events, such as a user login, system startup, or other predefined triggers. Attackers use this technique to ensure their payload executes at critical times or under specific conditions.
Event Triggered Execution-Application Shimming	Event Triggered Execution: Application Shimming is a technique that involves manipulating the application compatibility features of Windows to execute malicious code. Attackers create shims that intercept and modify the normal execution flow of

	applications, allowing them to inject malicious code or exploit vulnerabilities for privilege escalation.
Event Triggered Execution-Component Object Model Hijacking	Event Triggered Execution: Component Object Model Hijacking involves manipulating the Windows Component Object Model (COM) to trigger the execution of malicious code. Attackers register a malicious DLL or executable as a COM object, causing it to be loaded and executed by applications that use COM for inter-process communication.
Event Triggered Execution-Screensaver	Event Triggered Execution: Screensaver is a technique where attackers configure a malicious executable to run as a screensaver on Windows systems. When the screensaver activates (typically after a period of user inactivity), the malicious code is executed, allowing attackers to achieve persistence or execute arbitrary payloads.
Execution Guardrails	Execution Guardrails is a technique where attackers use environmental conditions to determine whether to execute their payload. This might include checking for specific network conditions, domain memberships, or user privileges. The goal is to prevent detection and analysis by ensuring the malware only runs in the intended environment.
Execution Through Api	Execution through API involves attackers using application programming interfaces (APIs) to execute malicious code on a system. APIs provide programmatic access to system functions,

	and attackers may exploit vulnerabilities in APIs or abuse them to perform unauthorized actions on a compromised system.
Exfiltration Over Alternative Protocol	Exfiltration Over Alternative Protocol refers to using unconventional or lesser-known network protocols for data exfiltration. Attackers may use protocols like FTP, SCP, or custom protocols to avoid detection by security tools that monitor common exfiltration channels.
Exfiltration Over Alternative Protocol- Domain Name System	Exfiltration Over Alternative Protocol: Domain Name System involves using DNS queries and responses to exfiltrate data. Attackers encode data within DNS requests or replies, exploiting the DNS protocol's ubiquity and lack of thorough inspection by security tools.
Exfiltration Over Alternative Protocol- Secure Shell	Exfiltration Over Alternative Protocol: Secure Shell involves using SSH for data exfiltration. Attackers use SSH tunnels or direct SCP transfers to move data, often leveraging the protocol's encryption and legitimacy to avoid detection.
Exfiltration Over C2 Channel	Exfiltration Over C2 Channel involves using established command and control (C2) channels for data exfiltration. Attackers use the same channel they use for C2 communications to send stolen data out of the network, blending in with normal C2 traffic.
Exfiltration Over Physical Medium	Exfiltration Over Physical Medium involves using removable media, such as USB drives or external hard drives, to exfiltrate data physically. Attackers may copy data to a device and then remove it from the premises to bypass network security controls.

Exfiltration Over Usb	Exfiltration Over USB refers to using USB drives for data exfiltration. Attackers copy sensitive data to a USB drive and physically remove it from the target environment, avoiding network monitoring and detection.
Exfiltration Over Web Service	Exfiltration Over Web Service involves using web-based services, such as Google Drive or Dropbox, to exfiltrate data. Attackers leverage these trusted platforms to transfer stolen data, taking advantage of their wide usage and legitimacy to avoid detection.
Exploit Public-Facing Application	Exploit Public-Facing Application is a technique where attackers target vulnerabilities in publicly accessible web applications or services. Exploitation may allow the attacker to gain unauthorized access, execute arbitrary code, or cause a denial of service, often serving as an initial entry point into a network.
Exploitation For Client Execution	Exploitation for Client Execution involves exploiting vulnerabilities on client software (e.g., browsers, document readers) to execute arbitrary code. Attackers target users who interact with compromised websites, open malicious documents, or run infected software to gain execution control on the victim's system.
Exploitation For Credential Access	Exploitation for Credential Access involves exploiting software vulnerabilities to access credential material. Attackers may exploit vulnerabilities in applications, operating systems, or network devices to extract passwords, keys, or tokens from memory or storage.

Exploitation For Privilege Escalation	Exploitation for Privilege Escalation refers to the use of software vulnerabilities to gain higher privileges on a compromised system. Attackers exploit vulnerabilities in operating systems, applications, or drivers to execute arbitrary code with elevated privileges, allowing them to perform unauthorized actions or gain deeper access to the system.
Exploitation Of Remote Services	Exploitation of Remote Services involves targeting vulnerabilities in remote services like RDP, SSH, or SMB to gain unauthorized access to systems. Attackers may exploit weak configurations, vulnerabilities, or credentials to move laterally within a network or maintain persistence.
Exploitation Of Remote Services- Ssh Hijacking	Exploitation of Remote Services: SSH Hijacking involves taking over an existing SSH session to gain unauthorized access to systems. Attackers may use SSH hijacking to move laterally within a network without initiating new SSH connections, maintaining stealth and avoiding detection.
External Remote Services	External Remote Services involves attackers leveraging external-facing services such as VPNs, SSH, or web-based management interfaces to gain unauthorized access. Attackers may use stolen credentials, default passwords, or exploit vulnerabilities to connect and access internal networks, often as a precursor to further attacks.
Extra Window Memory Injection	Extra Window Memory Injection involves injecting malicious code into the memory space of another process by modifying window memory objects. Attackers use this method to execute

	code within the context of another process, making their activities harder to detect and analyse.
Fallback Channels	Fallback Channels are secondary communication paths used by attackers when their primary command and control (C2) channel is disrupted or blocked. Attackers use fallback channels to maintain communication with compromised systems, often switching to different protocols or domains to avoid detection.
File And Directory Discovery	File and Directory Discovery involves locating files and directories on a system to find valuable data or determine the file structure. Attackers use this technique to search for sensitive information, configuration files, or locations where they can place malicious payloads.
File And Directory Permissions Modification	File and Directory Permissions Modification involves altering the access controls of files and directories to grant unauthorized access or prevent legitimate users from accessing them. Attackers may change permissions to hide malicious files, gain access to sensitive information, or disrupt operations by restricting access to critical resources.
File Deletion	File Deletion is a technique where attackers remove files from a system to hide their presence, prevent recovery, or disrupt operations. This can include deleting logs, malware binaries, or data files. Attackers use this technique to cover their tracks or to cause damage by removing important files.
Firmware Corruption	Firmware Corruption involves modifying or tampering with a device's firmware to disrupt its normal operation or render it

	unusable. Attackers use firmware corruption to cause hardware failures, create persistent access, or prevent recovery efforts.
Forced Authentication	Forced Authentication is a technique where attackers force a system to authenticate to an attacker-controlled server. This can be used to capture credentials or hashes, which can then be cracked or replayed to gain unauthorized access to other systems or services.
Hardware Additions	Hardware Additions refer to physical devices added to a system or network to facilitate unauthorized access or data exfiltration. This can include rogue devices such as USB drives, network implants, or hardware keyloggers introduced by attackers with physical access to the target environment.
Hide Artifacts	Hide Artifacts involves concealing malicious files, processes, or network connections to avoid detection by security tools and administrators. Attackers use techniques such as file attribute manipulation, rootkits, or encryption to hide their activities and maintain a covert presence on a compromised system.
Hijack Execution Flow	Hijack Execution Flow is a technique where attackers manipulate the normal execution flow of a program to execute malicious code. This can involve DLL hijacking, process injection, or other methods to divert the program's execution to the attacker's payload.
Hijack Execution Flow- Dll Search Order Hijacking	Hijack Execution Flow: DLL Search Order Hijacking involves placing a malicious DLL in a location that is searched before the legitimate DLL, causing the operating system to load the

	malicious DLL instead. This technique is used to execute arbitrary code under the guise of a legitimate application.
Hijack Execution Flow- Dll Side-Loading	Hijack Execution Flow: DLL Side-Loading involves placing a malicious DLL alongside a legitimate application, tricking the application into loading the malicious DLL instead of a legitimate one. This technique is often used to evade detection by executing code in the context of a trusted process.
Implant Container Image	Implant Container Image involves the use of malicious or compromised container images that, when deployed, execute malicious code. Attackers may inject malware into container images or use them to establish persistence within containerized environments.
Indicator Blocking	Indicator Blocking is a technique where attackers block or manipulate indicators of compromise to prevent detection and response. This may include blocking IP addresses, altering logs, or disrupting security tools' ability to communicate alerts. The goal is to hinder detection efforts and prolong the attacker's presence on the network.
Indicator Removal on Host	Indicator Removal on Host involves removing evidence of an attack from a compromised system to avoid detection. This can include deleting log files, clearing event logs, removing malware binaries, or modifying timestamps. Attackers use this technique to erase signs of their activities and make it difficult for defenders to investigate and respond to the attack.

Inhibit System Recovery	Inhibit System Recovery is a technique where attackers disable or tamper with system recovery options to prevent the victim from restoring operations. This can include deleting backups, disabling recovery partitions, or corrupting recovery files to make recovery more difficult or impossible.
Input Capture	Input Capture involves capturing user input, such as keystrokes or mouse movements, to obtain credentials or other sensitive information. Attackers use keyloggers or screen capture tools to record input, often running in the background to avoid detection.
Input Prompt	Input Prompt involves prompting the user to enter credentials or sensitive information. Attackers may create fake input prompts that mimic legitimate prompts, tricking users into entering their information. This data can then be captured and used to gain unauthorized access or escalate privileges.
Internal Spearphishing	Internal Spearphishing is a targeted phishing attack that occurs within an organization's network. Attackers use compromised accounts to send convincing phishing emails to other users, often aiming to steal credentials, distribute malware, or escalate privileges.
Inter-Process Communication	Inter-Process Communication (IPC) involves mechanisms that allow processes to communicate with each other within a system. Attackers may exploit IPC mechanisms to escalate privileges, inject malicious code, or manipulate processes to achieve their objectives.

Kerberoasting	Kerberoasting is a technique that involves abusing the Kerberos authentication protocol to extract service account credentials from Active Directory. Attackers request service tickets and then extract the encrypted portion containing the service account password hash, which can be cracked offline.
Keychain	Keychain is a credential storage feature on macOS that stores passwords, keys, and certificates. Attackers target the keychain to extract sensitive information such as stored credentials or encryption keys, which can be used for further attacks.
Lateral Tool Transfer	Lateral Tool Transfer involves moving tools or payloads from one system to another within a network. Attackers use this technique to propagate malware, establish persistence, or prepare for further attacks by transferring files through legitimate or compromised channels.
Lsass Driver	LSASS Driver refers to attacks targeting the Local Security Authority Subsystem Service (LSASS), a process in Windows responsible for enforcing security policies and managing user credentials. Attackers exploit LSASS to dump credentials, escalate privileges, or move laterally within a network.
Man In the Browser	Man in the Browser (MitB) is an attack technique where malware intercepts and manipulates communication between a user and their web browser. Attackers use this technique to steal credentials, modify web transactions, or perform unauthorized actions on behalf of the user.

Man-In-The-Browser	Man-in-the-Browser (MitB) is an advanced attack where malware manipulates the data flow between a web browser and its security mechanisms. This technique allows attackers to intercept and modify transactions, steal credentials, or inject malicious content into a legitimate browsing session.
Manipulate Network Traffic	Manipulate Network Traffic involves intercepting, modifying, or redirecting network traffic to achieve malicious objectives. Attackers may use this technique to perform man-in-the-middle attacks, disrupt communications, or reroute traffic to malicious sites.
Manipulation Of Insecure Content	Manipulation of Insecure Content involves modifying or injecting malicious content into insecure applications or services. Attackers use this technique to exploit vulnerabilities, deliver payloads, or manipulate data to achieve their goals.
Masquerading	Masquerading involves disguising or altering a process, service, or file to appear as something legitimate or benign. This can involve renaming malicious files to resemble legitimate system files, changing file paths, or modifying attributes like timestamps. The goal is to deceive users or security tools into thinking the malicious entity is a normal part of the system, helping attackers evade detection.
Multi-Hop Proxy	Multi-hop Proxy is a technique where attackers route their command and control (C2) traffic through multiple intermediate systems before reaching the destination. This technique is used to

	obscure the true origin of the traffic and make it more difficult for defenders to trace back to the attacker.
Multi-Stage Channels	Multi-Stage Channels refer to the use of multiple stages in the command and control (C2) communication process. Attackers may use different protocols or channels at each stage to evade detection, complicate analysis, and ensure reliable communication with compromised systems.
Native Api	Native API refers to the use of core operating system APIs, which are provided by the OS kernel and user-mode subsystems. Attackers use native APIs to directly interact with the operating system in ways that may bypass higher-level security monitoring. This method can be used for tasks such as process injection, memory manipulation, or interacting with the file system and registry, often avoiding detection by security tools.
Network Denial of Service	Network Denial of Service (DoS) involves overwhelming a network or service with traffic to render it unavailable. Attackers may use techniques like flooding or amplification attacks to exhaust network bandwidth or resources, disrupting normal operations.
Network Service Discovery	Network Service Discovery involves identifying services running on remote systems by scanning open ports and querying service banners. Attackers use this technique to map the network, identify potential entry points, or discover services that could be vulnerable to exploitation.

Network Service Scanning	Network Service Scanning is the process of identifying open ports and network services running on remote systems. Attackers use this technique to map the network, identify potential targets, and discover vulnerable services that can be exploited for further attacks.
Network Share Discovery	Network Share Discovery involves identifying shared folders or network drives accessible on a network. Attackers use this technique to locate shared resources that might contain valuable data, provide additional access, or serve as points of further compromise.
Network Sniffing	Network Sniffing involves capturing network traffic to monitor and analyse data flowing through a network. Attackers use this technique to capture credentials, session tokens, or other sensitive information transmitted over the network, often targeting unencrypted or weakly protected communications.
Non-Application Layer Protocol	Non-Application Layer Protocol involves using network protocols that operate below the application layer, such as ICMP or UDP, for command and control (C2) communication. Attackers use these protocols to bypass application-layer security controls and avoid detection.
Non-Standard Port	Non-Standard Port refers to the use of uncommon or non-default ports for network communication to evade detection. Attackers may configure their command and control (C2) servers to listen on ports that are not commonly monitored by security tools, making their traffic less likely to be detected.

Non-Standard Port- Tcp/Udp	Non-Standard Port: TCP/UDP refers to using uncommon or non-default TCP or UDP ports for network communication. Attackers may configure their command and control (C2) servers to use these ports to avoid detection by security tools that monitor standard ports.
Os Credential Dumping	OS Credential Dumping is a technique used to extract credentials from the operating system. Attackers target locations where credentials are stored, such as the Windows Security Accounts Manager (SAM), LSASS memory, or Active Directory, to extract password hashes or plaintext credentials.
Password Spraying	Password Spraying is a technique where attackers attempt to gain unauthorized access to multiple accounts by trying common passwords against a large number of usernames. This technique avoids triggering account lockouts by using a low volume of attempts per account, making it harder to detect.
Permission Groups Discovery	Permission Groups Discovery is the process of identifying user or group permissions within a system or network. Attackers use this technique to understand the access levels of different accounts and groups, which can inform privilege escalation or lateral movement strategies.
Phishing	Phishing is a social engineering technique where attackers send fraudulent messages, often emails, that appear to come from a trustworthy source to trick individuals into revealing sensitive information or installing malware. These messages may contain

	malicious links or attachments designed to steal credentials, deliver malware, or direct the victim to a spoofed website.
Port Knocking	Port Knocking is a stealthy technique used by attackers to communicate with a system without exposing open ports. Attackers send a series of network packets in a specific sequence to a closed port, which signals the system to temporarily open a port for incoming connections. This technique can be used to bypass firewalls and intrusion detection systems.
Portable Executable Injection	Portable Executable Injection is a technique where attackers inject malicious code into the memory space of a running process. The code is usually injected into a legitimate process, allowing the attacker to execute their payload while avoiding detection. This technique is often used to run malware stealthily or to achieve persistence.
PowerShell	PowerShell is a scripting language, and command-line shell used primarily in Windows environments. Attackers use PowerShell to execute commands, download and run scripts, and perform reconnaissance, often exploiting its integration with the Windows operating system to evade detection.
Private Keys	Private Keys refers to the theft or misuse of cryptographic private keys, which are used to authenticate and encrypt communications. Attackers may locate and extract private keys from files, memory, or network traffic, allowing them to impersonate legitimate users or decrypt sensitive data.

Protocol Tunnelling	Protocol Tunnelling involves encapsulating one protocol within another to bypass security controls. Attackers use tunnelling to hide command and control (C2) traffic within legitimate protocols, such as HTTP or DNS, making it more difficult for defenders to detect and block malicious activity.
Remote Access Software	Remote Access Software refers to legitimate software tools used for remote management and access, such as TeamViewer or VNC. Attackers use these tools to maintain access to compromised systems, blending in with legitimate traffic and evading detection by using trusted software.
Remote Desktop Protocol	Remote Desktop Protocol (RDP) is a Microsoft protocol that allows remote access to a computer's desktop interface. Attackers exploit RDP by using stolen credentials, exploiting vulnerabilities, or brute-forcing weak passwords to gain unauthorized access and move laterally within a network.
Remote Service Session Hijacking	Remote Service Session Hijacking is a technique where attackers take over an existing remote service session, such as RDP or SSH. Attackers may hijack these sessions to gain access to a system without triggering new authentication events, allowing them to move laterally within a network.
Remote Services	Remote Services refers to network-based services that provide remote access to systems, such as RDP, SSH, or VNC. Attackers leverage these services to access systems over a network, using valid credentials, exploiting vulnerabilities, or abusing legitimate tools to move laterally.

Remote Services- Ssh	Remote Services: SSH involves using the SSH protocol to access remote systems over a network. Attackers may use stolen credentials or exploit vulnerabilities in SSH configurations to gain unauthorized access, move laterally within a network, or maintain persistence on a compromised system.
Remote System Discovery	Remote System Discovery involves identifying other systems and devices on a network that can be targeted for further exploitation. Attackers may use network scanning tools or built-in commands to list remote systems, their configurations, and their operating status.
Resource Hijacking	Resource Hijacking refers to the unauthorized use of a victim's resources, such as CPU, memory, or network bandwidth, to perform malicious activities. Attackers may use compromised systems to mine cryptocurrency, send spam, or conduct distributed denial-of-service (DDoS) attacks.
Screen Capture	Screen Capture involves capturing images or video of the contents displayed on a compromised system's screen. Attackers use this technique to gather sensitive information, monitor user activity, or capture visual data that can be used for further exploitation.
Security Software Discovery	Security Software Discovery is the process of identifying security software installed on a system, such as antivirus, firewalls, or intrusion detection systems. Attackers use this information to understand the defensive posture of a network and to identify tools that need to be disabled or bypassed.

Server Software Component	Server Software Component refers to the use or abuse of server software components to execute malicious code or maintain access. Attackers may exploit vulnerabilities in server software or inject malicious components to manipulate server behaviour for their own purposes.
Shared Modules	Shared Modules involves attackers using shared code libraries or modules (e.g., DLLs in Windows) to execute malicious code. Attackers may inject code into these modules or replace legitimate modules with malicious ones, allowing their code to run within the context of a trusted application or process.
Signed Binary Proxy Execution	Signed Binary Proxy Execution is a technique where attackers misuse signed binaries that are trusted by the operating system to execute malicious code. By proxying their execution through these trusted binaries, attackers can bypass security controls that flag unsigned or suspicious executables.
Spearphishing Link	Spearphishing Link involves sending a targeted phishing email that includes a link to a malicious website. The website may be designed to steal credentials, deliver malware, or trick the victim into downloading malicious files.
Steal Application Access Token	Steal Application Access Token involves stealing tokens used by applications to authenticate to services. Attackers extract tokens from memory, storage, or intercepted network traffic, which can then be used to impersonate legitimate users or access sensitive resources.

System Binary Proxy Execution	System Binary Proxy Execution involves using legitimate system binaries to execute malicious code. Attackers may abuse trusted binaries to proxy execution of their code, thereby bypassing security controls that monitor for malicious activity. This technique leverages the trust and legitimacy of system binaries to evade detection.
System Information Discovery	System Information Discovery involves gathering information about a system, such as its operating system, hardware, software, and configuration settings. Attackers use this information to identify potential vulnerabilities, tailor their attacks, or plan further exploitation.
System Owner User Discovery	System Owner/User Discovery is the process of identifying the owner or user of a system. Attackers use this technique to understand the context of a compromised system, identify high-value targets, or find accounts that can be used for further exploitation.
System Services	System Services refers to the manipulation or abuse of system services, such as creating, modifying, or starting a service, to execute malicious code. Attackers may use this technique to achieve persistence, escalate privileges, or execute arbitrary code with elevated permissions.
System Shutdown Reboot	System Shutdown/Reboot involves forcing a system to restart or shut down, potentially disrupting services and erasing evidence of the attacker's activities. Attackers use this technique to cause a

	denial of service, prevent recovery, or implement malicious changes during the reboot process.
Taint Shared Content	Taint Shared Content involves manipulating or poisoning shared content, such as files or directories, to execute malicious code when accessed by users. Attackers use this technique to spread malware or escalate privileges by infecting content commonly accessed by multiple users.
Transfer Data to Cloud Account	Transfer Data to Cloud Account involves exfiltrating data to a cloud storage account controlled by the attacker. Attackers use cloud storage services, such as AWS S3 or Azure Blob Storage, to store stolen data, leveraging the cloud's ubiquity and accessibility to avoid detection.
Trusted Relationship	Trusted Relationship involves exploiting trust relationships between organizations to gain unauthorized access. Attackers target partners, vendors, or service providers that have legitimate access to the victim's network, using compromised accounts or systems to move laterally within the network.
Unencrypted Communication	Unencrypted Communication refers to the use of plaintext communication channels for command and control (C2). Attackers may use unencrypted protocols to avoid detection by security tools that do not inspect plaintext traffic or when encryption might raise suspicion.
User Execution	User Execution is a technique that relies on tricking a user into executing malicious content. This can include actions such as opening a malicious document, clicking on a harmful link, or

	running an untrusted application. User interaction is required for the attack to succeed, and social engineering techniques are often employed.
User Execution- Malicious File	User Execution: Malicious File involves tricking a user into opening or executing a file that contains malicious code. The file might appear to be a legitimate document, image, or application but is designed to deliver malware once opened.
User Execution- Malicious Link	User Execution: Malicious Link is a technique where attackers entice a user to click on a link that redirects to a malicious website or file download. This can lead to credential theft, drive-by downloads, or the installation of malware.
Valid Accounts	Valid Accounts refers to the use of legitimate credentials to access systems and resources. Attackers may obtain these credentials through phishing, credential dumping, or brute-force attacks, allowing them to blend in with normal user activity and making it harder to detect unauthorized access.
Virtualization Sandbox Evasion	Virtualization/Sandbox Evasion involves detecting and avoiding virtual environments or sandboxes used by defenders to analyse malware. Attackers use this technique to determine whether their code is being analysed in a controlled environment and can then alter or stop execution to avoid detection.

A3 - Case Study Pre-Mitigated Vulnerability Likelihood Table

Vulnerability	Asset	Likelihood

Default Credentials	Microsoft Azure	60
Broken Authentication and Session Management	Microsoft Azure	53
Cross-Account Access Misconfigurations	Microsoft Azure	50
Insecure Remote Management Access	Microsoft Azure	50
Exposed Secrets and Keys	Microsoft Azure	49
Improper Identity and Access Management	Microsoft Azure	49
Insecure Access Control Policies	Microsoft Azure	48
Insecure API Access	Microsoft Azure	48
Inadequate IAM Policies	Microsoft Azure	47
API Key Exposure	Microsoft Azure	46
Inadequate Secrets Management	Microsoft Azure	45
Cloud Security Misconfigurations	Microsoft Azure	44
Certificate Validation Flaws	Microsoft Azure	40
Insecure API Exposure	Microsoft Azure	40
Inadequate Security Group Rules	Microsoft Azure	39
Insecure API Endpoints	Microsoft Azure	39
Insecure Permissions	Microsoft Azure	38
Insecure API Gateways	Microsoft Azure	37
Insecure API	Microsoft Azure	35
Insecure REST API Configurations	Microsoft Azure	35
Cryptographic Flaws	Microsoft Azure	34
Data Leakage through Misconfigured Storage	Microsoft Azure	34
Incomplete Visibility into Cloud Usage	Microsoft Azure	34
Insecure Default Settings	Microsoft Azure	34

Container Image Vulnerabilities	Microsoft Azure	33
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	33
Inadequate Input Validation	Microsoft Azure	32
Insecure OAuth Implementations	Microsoft Azure	32
Failure to Implement Secure Coding Practices	Microsoft Azure	31
Inadequate Encryption of Data at Rest	Microsoft Azure	31
Inadequate Resource Isolation	Microsoft Azure	31
Insecure Third-Party Libraries	Microsoft Azure	31
Insecure DevOps Practices	Microsoft Azure	30
Insufficient Incident Response Procedures	Microsoft Azure	29
Container Security Flaws	Microsoft Azure	28
Insecure Handling of User Data	Microsoft Azure	27
Environment Variable Manipulation	Microsoft Azure	26
Inadequate Monitoring and Alerting	Microsoft Azure	26
Inadequate VLAN Segmentation	Microsoft Azure	26
Broken Function Level Authorization	Microsoft Azure	25
Bypassed URL Filtering	Microsoft Azure	25
Insecure Storage Configurations	Microsoft Azure	25
Inadequate Data Backup and Recovery	Microsoft Azure	22
Inadequate Protection Against Insider Threats	Microsoft Azure	22
Backup Failures	Microsoft Azure	21
Improper Authentication	Microsoft Azure	21
Insufficient Authorization	Microsoft Azure	21

API Misconfiguration	Microsoft Azure	20
Excessive Privileges	Microsoft Azure	20
Insecure Key Management	Microsoft Azure	20
Insecure Key Management Practices	Microsoft Azure	19
Inadequate Encryption Strength	Microsoft Azure	17
Insecure API Management	Microsoft Azure	16
Insufficient Audit Trail	Microsoft Azure	13
Insecure Third-Party Components	Microsoft Azure	12
Insecure VM Migration	Microsoft Azure	12
Cross-Region Data Replication Risks	Microsoft Azure	10
Inadequate Scalability	Microsoft Azure	10
Insufficient DDoS Protection	Microsoft Azure	4
Data Loss from Accidental Deletion	Microsoft Azure	1

A4 - Case Study Pre-Mitigated Confidentiality Impact Table

Vulnerability	Asset	Likelihood	Impact	Risk Value
Default Credentials	Microsoft Azure	41	47	19
Insufficient Authorization	Microsoft Azure	34	47	16
Insecure Remote Management Access	Microsoft Azure	33	47	16
Inadequate Secrets Management	Microsoft Azure	32	47	15
Insecure Access Control Policies	Microsoft Azure	31	47	15

Inadequate IAM Policies	Microsoft Azure	31	47	15
Broken Authentication and Session Management	Microsoft Azure	31	47	15
Improper Identity and Access Management	Microsoft Azure	30	47	14
Cross-Account Access Misconfigurations	Microsoft Azure	30	47	14
API Key Exposure	Microsoft Azure	26	47	12
Inadequate Encryption of Data at Rest	Microsoft Azure	26	47	12
Data Leakage through Misconfigured Storage	Microsoft Azure	26	47	12
Incomplete Visibility into Cloud Usage	Microsoft Azure	25	47	12
Insecure Key Management Practices	Microsoft Azure	24	47	11
Insecure Handling of User Data	Microsoft Azure	24	47	11
Insecure API Access	Microsoft Azure	24	47	11
Insecure Default Settings	Microsoft Azure	23	47	11
Inadequate Resource Isolation	Microsoft Azure	21	47	10
Insecure Permissions	Microsoft Azure	20	47	9
Exposed Secrets and Keys	Microsoft Azure	19	47	9

Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	19	47	9
Improper Authentication	Microsoft Azure	19	47	9
Insecure REST API Configurations	Microsoft Azure	19	47	9
Insecure API Gateways	Microsoft Azure	19	47	9
Container Security Flaws	Microsoft Azure	18	47	8
Insecure API	Microsoft Azure	18	47	8
Container Image Vulnerabilities	Microsoft Azure	18	47	8
Insecure API Exposure	Microsoft Azure	17	47	8
Insufficient Incident Response Procedures	Microsoft Azure	17	47	8
Cloud Security Misconfigurations	Microsoft Azure	17	47	8
Exposed Secrets and Keys	Microsoft Azure	17	47	8
Inadequate Encryption Strength	Microsoft Azure	16	47	8
Insecure DevOps Practices	Microsoft Azure	16	47	8
Insecure API Endpoints	Microsoft Azure	16	47	8
Insecure Key Management	Microsoft Azure	16	47	8
Inadequate VLAN Segmentation	Microsoft Azure	16	47	8
Cryptographic Flaws	Microsoft Azure	16	47	8

Insecure OAuth Implementations	Microsoft Azure	14	47	7
Inadequate Security Group Rules	Microsoft Azure	14	47	7
Insecure Storage Configurations	Microsoft Azure	14	47	7
Inadequate Data Backup and Recovery	Microsoft Azure	14	47	7
Backup Failures	Microsoft Azure	14	47	7
Inadequate Input Validation	Microsoft Azure	14	47	7
Bypassed URL Filtering	Microsoft Azure	14	47	7
Certificate Validation Flaws	Microsoft Azure	14	47	7
Insecure API Management	Microsoft Azure	14	47	7
Excessive Privileges	Microsoft Azure	12	47	6
Insecure VM Migration	Microsoft Azure	12	47	6
Environment Variable Manipulation	Microsoft Azure	12	47	6
Insecure Third-Party Libraries	Microsoft Azure	12	47	6
Inadequate Protection Against Insider Threats	Microsoft Azure	11	47	5
Broken Function Level Authorization	Microsoft Azure	10	47	5
API Misconfiguration	Microsoft Azure	9	47	4

Insecure Third-Party Components	Microsoft Azure	8	47	4
Inadequate Monitoring and Alerting	Microsoft Azure	8	47	4
Cross-Region Data Replication Risks	Microsoft Azure	6	47	3
Inadequate Scalability	Microsoft Azure	4	47	2
Insufficient Audit Trail	Microsoft Azure	3	47	1
Insufficient DDoS Protection	Microsoft Azure	0	47	0
Data Loss from Accidental Deletion	Microsoft Azure	0	47	0

A5 - Case Study Pre-Mitigated Integrity Impact Table

Vulnerability	Asset	Likelihood	Impact	Risk Value
Default Credentials	Microsoft Azure	37	48	18
Cross-Account Access Misconfigurations	Microsoft Azure	27	48	13
Insecure Remote Management Access	Microsoft Azure	26	48	12
Improper Identity and Access Management	Microsoft Azure	25	48	12
Inadequate IAM Policies	Microsoft Azure	25	48	12

Inadequate Secrets Management	Microsoft Azure	24	48	12
Insecure Access Control Policies	Microsoft Azure	23	48	11
Insufficient Authorization	Microsoft Azure	22	48	11
Broken Authentication and Session Management	Microsoft Azure	22	48	11
Insecure DevOps Practices	Microsoft Azure	21	48	10
API Key Exposure	Microsoft Azure	20	48	10
Insecure Key Management Practices	Microsoft Azure	19	48	9
Insecure Third-Party Libraries	Microsoft Azure	19	48	9
Insecure Permissions	Microsoft Azure	19	48	9
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	18	48	9
Cryptographic Flaws	Microsoft Azure	18	48	9
Insecure Default Settings	Microsoft Azure	17	48	8
Inadequate Protection Against Insider Threats	Microsoft Azure	17	48	8
Inadequate Encryption of Data at Rest	Microsoft Azure	17	48	8
Cloud Security Misconfigurations	Microsoft Azure	17	48	8

Insecure API Access	Microsoft Azure	17	48	8
Inadequate VLAN Segmentation	Microsoft Azure	17	48	8
Exposed Secrets and Keys	Microsoft Azure	16	48	8
Incomplete Visibility into Cloud Usage	Microsoft Azure	16	48	8
Insecure REST API Configurations	Microsoft Azure	16	48	8
Exposed Secrets and Keys	Microsoft Azure	15	48	7
Certificate Validation Flaws	Microsoft Azure	15	48	7
Inadequate Resource Isolation	Microsoft Azure	15	48	7
Broken Function Level Authorization	Microsoft Azure	15	48	7
Container Security Flaws	Microsoft Azure	14	48	7
Insecure API	Microsoft Azure	13	48	6
Insecure API Exposure	Microsoft Azure	12	48	6
Excessive Privileges	Microsoft Azure	12	48	6
Inadequate Security Group Rules	Microsoft Azure	12	48	6
Inadequate Input Validation	Microsoft Azure	12	48	6
Insecure API Endpoints	Microsoft Azure	12	48	6
Container Image Vulnerabilities	Microsoft Azure	12	48	6
Insecure Key Management	Microsoft Azure	12	48	6

Environment Variable Manipulation	Microsoft Azure	11	48	5
Insecure API Management	Microsoft Azure	11	48	5
Insecure API Gateways	Microsoft Azure	11	48	5
Insecure Third-Party Components	Microsoft Azure	10	48	5
Insecure OAuth Implementations	Microsoft Azure	10	48	5
Insufficient Audit Trail	Microsoft Azure	10	48	5
Insufficient Incident Response Procedures	Microsoft Azure	10	48	5
Insecure Handling of User Data	Microsoft Azure	9	48	4
Insecure VM Migration	Microsoft Azure	8	48	4
Backup Failures	Microsoft Azure	8	48	4
Inadequate Encryption Strength	Microsoft Azure	7	48	3
Inadequate Data Backup and Recovery	Microsoft Azure	7	48	3
Bypassed URL Filtering	Microsoft Azure	7	48	3
Cross-Region Data Replication Risks	Microsoft Azure	6	48	3
Data Leakage through Misconfigured Storage	Microsoft Azure	6	48	3
API Misconfiguration	Microsoft Azure	5	48	2

Insecure Storage Configurations	Microsoft Azure	5	48	2
Inadequate Monitoring and Alerting	Microsoft Azure	4	48	2
Inadequate Scalability	Microsoft Azure	2	48	1
Insufficient DDoS Protection	Microsoft Azure	0	48	0
Improper Authentication	Microsoft Azure	0	48	0
Data Loss from Accidental Deletion	Microsoft Azure	0	48	0

A6 - Case Study Pre-Mitigated Availability Impact Table

Vulnerability	Asset	Likelihood	Impact	Risk Value
Inadequate Data Backup and Recovery	Microsoft Azure	26	40	10
Insufficient DDoS Protection	Microsoft Azure	25	40	10
Insufficient Incident Response Procedures	Microsoft Azure	22	40	9
Insecure Remote Management Access	Microsoft Azure	19	40	8
Improper Identity and Access Management	Microsoft Azure	18	40	7

Insecure Access Control Policies	Microsoft Azure	17	40	7
API Key Exposure	Microsoft Azure	16	40	6
Inadequate IAM Policies	Microsoft Azure	16	40	6
Backup Failures	Microsoft Azure	16	40	6
Broken Authentication and Session Management	Microsoft Azure	16	40	6
Cross-Account Access Misconfigurations	Microsoft Azure	14	40	6
Container Security Flaws	Microsoft Azure	14	40	6
Insecure Key Management Practices	Microsoft Azure	13	40	5
Inadequate Resource Isolation	Microsoft Azure	13	40	5
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	12	40	5
Insecure API Access	Microsoft Azure	12	40	5
Insufficient Authorization	Microsoft Azure	11	40	4
Cryptographic Flaws	Microsoft Azure	11	40	4
Insecure Default Settings	Microsoft Azure	10	40	4
Inadequate Security Group Rules	Microsoft Azure	10	40	4
Insecure Storage Configurations	Microsoft Azure	10	40	4

Incomplete Visibility into Cloud Usage	Microsoft Azure	10	40	4
Bypassed URL Filtering	Microsoft Azure	10	40	4
Inadequate Secrets Management	Microsoft Azure	9	40	4
Inadequate Scalability	Microsoft Azure	9	40	4
Data Loss from Accidental Deletion	Microsoft Azure	9	40	4
Inadequate Protection Against Insider Threats	Microsoft Azure	8	40	3
Cloud Security Misconfigurations	Microsoft Azure	8	40	3
Insecure API	Microsoft Azure	8	40	3
Insecure DevOps Practices	Microsoft Azure	8	40	3
Insecure API Management	Microsoft Azure	8	40	3
Insecure OAuth Implementations	Microsoft Azure	7	40	3
Inadequate Input Validation	Microsoft Azure	7	40	3
Exposed Secrets and Keys	Microsoft Azure	7	40	3
Environment Variable Manipulation	Microsoft Azure	7	40	3
Insecure Third-Party Libraries	Microsoft Azure	7	40	3
Container Image Vulnerabilities	Microsoft Azure	7	40	3

Insecure Key Management	Microsoft Azure	7	40	3
Inadequate VLAN Segmentation	Microsoft Azure	7	40	3
Insecure API Exposure	Microsoft Azure	6	40	2
Insecure REST API Configurations	Microsoft Azure	6	40	2
Certificate Validation Flaws	Microsoft Azure	6	40	2
Broken Function Level Authorization	Microsoft Azure	6	40	2
Insecure API Endpoints	Microsoft Azure	6	40	2
Insecure Permissions	Microsoft Azure	6	40	2
Insecure API Gateways	Microsoft Azure	6	40	2
Insufficient Audit Trail	Microsoft Azure	5	40	2
Inadequate Monitoring and Alerting	Microsoft Azure	4	40	2
API Misconfiguration	Microsoft Azure	4	40	2
Insecure Third-Party Components	Microsoft Azure	3	40	1
Excessive Privileges	Microsoft Azure	3	40	1
Insecure VM Migration	Microsoft Azure	3	40	1
Cross-Region Data Replication Risks	Microsoft Azure	3	40	1
Inadequate Encryption of Data at Rest	Microsoft Azure	3	40	1

Inadequate Encryption Strength	Microsoft Azure	2	40	1
Insecure Handling of User Data	Microsoft Azure	2	40	1
Exposed Secrets and Keys	Microsoft Azure	2	40	1
Data Leakage through Misconfigured Storage	Microsoft Azure	2	40	1
Default Credentials	Microsoft Azure	0	40	0
Improper Authentication	Microsoft Azure	0	40	0

A7 - Case Study Pre-Mitigated Accountability Impact Table

Vulnerability	Asset	Likelihood	Impact	Risk Value
Default Credentials	Microsoft Azure	41	36	15
Insufficient Authorization	Microsoft Azure	34	36	12
Insecure Remote Management Access	Microsoft Azure	33	36	12
Inadequate Secrets Management	Microsoft Azure	32	36	12
Insecure Access Control Policies	Microsoft Azure	31	36	11
Inadequate IAM Policies	Microsoft Azure	31	36	11
Broken Authentication and Session Management	Microsoft Azure	31	36	11

Improper Identity and Access Management	Microsoft Azure	30	36	11
Cross-Account Access Misconfigurations	Microsoft Azure	30	36	11
API Key Exposure	Microsoft Azure	26	36	9
Inadequate Encryption of Data at Rest	Microsoft Azure	26	36	9
Data Leakage through Misconfigured Storage	Microsoft Azure	26	36	9
Incomplete Visibility into Cloud Usage	Microsoft Azure	25	36	9
Insecure Key Management Practices	Microsoft Azure	24	36	9
Insecure Handling of User Data	Microsoft Azure	24	36	9
Insecure API Access	Microsoft Azure	24	36	9
Insecure Default Settings	Microsoft Azure	23	36	8
Inadequate Resource Isolation	Microsoft Azure	21	36	8
Insecure Permissions	Microsoft Azure	20	36	7
Exposed Secrets and Keys	Microsoft Azure	19	36	7
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	19	36	7
Improper Authentication	Microsoft Azure	19	36	7

Insecure REST API Configurations	Microsoft Azure	19	36	7
Insecure API Gateways	Microsoft Azure	19	36	7
Container Security Flaws	Microsoft Azure	18	36	6
Insecure API	Microsoft Azure	18	36	6
Container Image Vulnerabilities	Microsoft Azure	18	36	6
Insecure API Exposure	Microsoft Azure	17	36	6
Insufficient Incident Response Procedures	Microsoft Azure	17	36	6
Cloud Security Misconfigurations	Microsoft Azure	17	36	6
Exposed Secrets and Keys	Microsoft Azure	17	36	6
Inadequate Encryption Strength	Microsoft Azure	16	36	6
Insecure DevOps Practices	Microsoft Azure	16	36	6
Insecure API Endpoints	Microsoft Azure	16	36	6
Insecure Key Management	Microsoft Azure	16	36	6
Inadequate VLAN Segmentation	Microsoft Azure	16	36	6
Cryptographic Flaws	Microsoft Azure	16	36	6
Insecure OAuth Implementations	Microsoft Azure	14	36	5
Inadequate Security Group Rules	Microsoft Azure	14	36	5

Insecure Storage Configurations	Microsoft Azure	14	36	5
Inadequate Data Backup and Recovery	Microsoft Azure	14	36	5
Backup Failures	Microsoft Azure	14	36	5
Inadequate Input Validation	Microsoft Azure	14	36	5
Bypassed URL Filtering	Microsoft Azure	14	36	5
Certificate Validation Flaws	Microsoft Azure	14	36	5
Insecure API Management	Microsoft Azure	14	36	5
Excessive Privileges	Microsoft Azure	12	36	4
Insecure VM Migration	Microsoft Azure	12	36	4
Environment Variable Manipulation	Microsoft Azure	12	36	4
Insecure Third-Party Libraries	Microsoft Azure	12	36	4
Inadequate Protection Against Insider Threats	Microsoft Azure	11	36	4
Broken Function Level Authorization	Microsoft Azure	10	36	4
API Misconfiguration	Microsoft Azure	9	36	3
Insecure Third-Party Components	Microsoft Azure	8	36	3
Inadequate Monitoring and Alerting	Microsoft Azure	8	36	3

Cross-Region Data Replication Risks	Microsoft Azure	6	36	2
Inadequate Scalability	Microsoft Azure	4	36	1
Insufficient Audit Trail	Microsoft Azure	3	36	1
Insufficient DDoS Protection	Microsoft Azure	0	36	0
Data Loss from Accidental Deletion	Microsoft Azure	0	36	0

A4 - Case Study Post Mitigated Confidentiality Impact Table

Vulnerability	Asset	Likelihood	Impact	Risk Value
Insufficient Authorization	Microsoft Azure	32	47	15
Cross-Account Access Misconfigurations	Microsoft Azure	22	47	10
Improper Identity and Access Management	Microsoft Azure	21	47	10
Insecure Remote Management Access	Microsoft Azure	20	47	9
Default Credentials	Microsoft Azure	17	47	8

Insecure Default Settings	Microsoft Azure	16	47	8
Broken Authentication and Session Management	Microsoft Azure	16	47	8
Inadequate Secrets Management	Microsoft Azure	14	47	7
Insecure Access Control Policies	Microsoft Azure	14	47	7
Inadequate Data Backup and Recovery	Microsoft Azure	14	47	7
Inadequate Encryption of Data at Rest	Microsoft Azure	14	47	7
Inadequate IAM Policies	Microsoft Azure	13	47	6
Incomplete Visibility into Cloud Usage	Microsoft Azure	13	47	6
Data Leakage through	Microsoft Azure	13	47	6

Misconfigured Storage				
Insecure Handling of User Data	Microsoft Azure	12	47	6
Inadequate Protection Against Insider Threats	Microsoft Azure	12	47	6
Insecure API	Microsoft Azure	12	47	6
Insecure Permissions	Microsoft Azure	12	47	6
Container Security Flaws	Microsoft Azure	11	47	5
Insecure API Access	Microsoft Azure	11	47	5
Container Image Vulnerabilities	Microsoft Azure	11	47	5
Insufficient Incident Response Procedures	Microsoft Azure	10	47	5
Inadequate VLAN Segmentation	Microsoft Azure	10	47	5

Insecure API Exposure	Microsoft Azure	9	47	4
Insecure Key Management Practices	Microsoft Azure	9	47	4
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	9	47	4
Backup Failures	Microsoft Azure	9	47	4
Cloud Security Misconfigurations	Microsoft Azure	9	47	4
Cryptographic Flaws	Microsoft Azure	9	47	4
Excessive Privileges	Microsoft Azure	8	47	4
Improper Authentication	Microsoft Azure	8	47	4
Inadequate Resource Isolation	Microsoft Azure	8	47	4
Insecure DevOps Practices	Microsoft Azure	8	47	4

Insecure Key Management	Microsoft Azure	8	47	4
Certificate Validation Flaws	Microsoft Azure	7	47	3
Insecure API Endpoints	Microsoft Azure	7	47	3
Insecure Third-Party Libraries	Microsoft Azure	7	47	3
Insecure API Gateways	Microsoft Azure	7	47	3
Inadequate Security Group Rules	Microsoft Azure	6	47	3
Insecure VM Migration	Microsoft Azure	6	47	3
Insecure REST API Configurations	Microsoft Azure	6	47	3
Inadequate Input Validation	Microsoft Azure	6	47	3
Exposed Secrets and Keys	Microsoft Azure	6	47	3
Bypassed URL Filtering	Microsoft Azure	6	47	3

Inadequate Encryption Strength	Microsoft Azure	5	47	2
Insecure Storage Configurations	Microsoft Azure	5	47	2
Broken Function Level Authorization	Microsoft Azure	5	47	2
Environment Variable Manipulation	Microsoft Azure	5	47	2
Insecure API Management	Microsoft Azure	5	47	2
Insecure Third-Party Components	Microsoft Azure	4	47	2
Insecure OAuth Implementations	Microsoft Azure	4	47	2
API Misconfiguration	Microsoft Azure	4	47	2
Inadequate Scalability	Microsoft Azure	4	47	2
Exposed Secrets and Keys	Microsoft Azure	4	47	2

API Key Exposure	Microsoft Azure	4	47	2
Cross-Region Data Replication Risks	Microsoft Azure	4	47	2
Inadequate Monitoring and Alerting	Microsoft Azure	2	47	1
Insufficient Audit Trail	Microsoft Azure	2	47	1
Insufficient DDoS Protection	Microsoft Azure	0	47	0
Data Loss from Accidental Deletion	Microsoft Azure	0	47	0

A5 - Case Study Post Mitigated Integrity Impact Table

Insufficient Authorization	Microsoft Azure	21	48	10
Improper Identity and Access Management	Microsoft Azure	15	48	7

Insecure Remote Management Access	Microsoft Azure	14	48	7
Inadequate Protection Against Insider Threats	Microsoft Azure	13	48	6
Broken Authentication and Session Management	Microsoft Azure	12	48	6
Cross-Account Access Misconfigurations	Microsoft Azure	11	48	5
Inadequate IAM Policies	Microsoft Azure	10	48	5
Insecure Default Settings	Microsoft Azure	9	48	4
Default Credentials	Microsoft Azure	9	48	4
Insecure DevOps Practices	Microsoft Azure	9	48	4
Inadequate Secrets Management	Microsoft Azure	8	48	4

Insecure Key Management Practices	Microsoft Azure	8	48	4
Incomplete Visibility into Cloud Usage	Microsoft Azure	8	48	4
Cloud Security Misconfigurations	Microsoft Azure	8	48	4
Insecure API Access	Microsoft Azure	8	48	4
Cryptographic Flaws	Microsoft Azure	8	48	4
Inadequate Encryption of Data at Rest	Microsoft Azure	7	48	3
Container Security Flaws	Microsoft Azure	7	48	3
Inadequate VLAN Segmentation	Microsoft Azure	7	48	3
Insecure Access Control Policies	Microsoft Azure	6	48	3
Excessive Privileges	Microsoft Azure	6	48	3

Inadequate Security Group Rules	Microsoft Azure	6	48	3
Inadequate Data Backup and Recovery	Microsoft Azure	6	48	3
Insecure Permissions	Microsoft Azure	6	48	3
Insecure API Exposure	Microsoft Azure	5	48	2
Insecure VM Migration	Microsoft Azure	5	48	2
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	5	48	2
Insecure REST API Configurations	Microsoft Azure	5	48	2
Certificate Validation Flaws	Microsoft Azure	5	48	2
Broken Function Level Authorization	Microsoft Azure	5	48	2

Container Image Vulnerabilities	Microsoft Azure	5	48	2
Insecure Third-Party Components	Microsoft Azure	4	48	2
Insufficient Audit Trail	Microsoft Azure	4	48	2
Insufficient Incident Response Procedures	Microsoft Azure	4	48	2
Exposed Secrets and Keys	Microsoft Azure	4	48	2
API Key Exposure	Microsoft Azure	4	48	2
Exposed Secrets and Keys	Microsoft Azure	4	48	2
Environment Variable Manipulation	Microsoft Azure	4	48	2
Insecure Third-Party Libraries	Microsoft Azure	4	48	2
Insecure Key Management	Microsoft Azure	4	48	2

Insecure OAuth Implementations	Microsoft Azure	3	48	1
Inadequate Encryption Strength	Microsoft Azure	3	48	1
API Misconfiguration	Microsoft Azure	3	48	1
Insecure Handling of User Data	Microsoft Azure	3	48	1
Insecure Storage Configurations	Microsoft Azure	3	48	1
Cross-Region Data Replication Risks	Microsoft Azure	3	48	1
Backup Failures	Microsoft Azure	3	48	1
Inadequate Input Validation	Microsoft Azure	3	48	1
Insecure API	Microsoft Azure	3	48	1
Inadequate Resource Isolation	Microsoft Azure	3	48	1
Insecure API Endpoints	Microsoft Azure	3	48	1

Insecure API Management	Microsoft Azure	3	48	1
Insecure API Gateways	Microsoft Azure	3	48	1
Data Leakage through Misconfigured Storage	Microsoft Azure	3	48	1
Bypassed URL Filtering	Microsoft Azure	2	48	1
Inadequate Monitoring and Alerting	Microsoft Azure	1	48	0
Inadequate Scalability	Microsoft Azure	1	48	0
Insufficient DDoS Protection	Microsoft Azure	0	48	0
Improper Authentication	Microsoft Azure	0	48	0
Data Loss from Accidental Deletion	Microsoft Azure	0	48	0

A6 - Case Study Post Mitigated Availability Impact Table

Vulnerability	Asset	Likelihood	Impact	Risk Value
Inadequate Data Backup and Recovery	Microsoft Azure	25	39	10
Insufficient Incident Response Procedures	Microsoft Azure	14	39	5
Improper Identity and Access Management	Microsoft Azure	12	39	5
Broken Authentication and Session Management	Microsoft Azure	12	39	5
Insecure Remote Management Access	Microsoft Azure	11	39	4
Backup Failures	Microsoft Azure	11	39	4
Insufficient Authorization	Microsoft Azure	10	39	4
Insecure Access Control Policies	Microsoft Azure	9	39	4

Insecure Key Management Practices	Microsoft Azure	9	39	4
Inadequate IAM Policies	Microsoft Azure	9	39	4
Inadequate Resource Isolation	Microsoft Azure	9	39	4
Cross-Account Access Misconfigurations	Microsoft Azure	8	39	3
Cryptographic Flaws	Microsoft Azure	8	39	3
Inadequate Scalability	Microsoft Azure	7	39	3
Container Security Flaws	Microsoft Azure	7	39	3
Bypassed URL Filtering	Microsoft Azure	7	39	3
Insecure Default Settings	Microsoft Azure	6	39	2
Inadequate Protection Against Insider Threats	Microsoft Azure	6	39	2

API Key Exposure	Microsoft Azure	6	39	2
Insecure API Access	Microsoft Azure	6	39	2
Inadequate Security Group Rules	Microsoft Azure	5	39	2
Insecure DevOps Practices	Microsoft Azure	5	39	2
Insecure Key Management	Microsoft Azure	5	39	2
Insecure API Exposure	Microsoft Azure	4	39	2
Inadequate Secrets Management	Microsoft Azure	4	39	2
Insecure Storage Configurations	Microsoft Azure	4	39	2
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	4	39	2

Incomplete Visibility into Cloud Usage	Microsoft Azure	4	39	2
Cloud Security Misconfigurations	Microsoft Azure	4	39	2
Insecure API	Microsoft Azure	4	39	2
Insecure API Endpoints	Microsoft Azure	4	39	2
Insecure Third-Party Libraries	Microsoft Azure	4	39	2
Container Image Vulnerabilities	Microsoft Azure	4	39	2
Insecure Handling of User Data	Microsoft Azure	3	39	1
Insecure REST API Configurations	Microsoft Azure	3	39	1
Data Loss from Accidental Deletion	Microsoft Azure	3	39	1
Inadequate Input Validation	Microsoft Azure	3	39	1

Broken Function Level Authorization	Microsoft Azure	3	39	1
Environment Variable Manipulation	Microsoft Azure	3	39	1
Insecure Permissions	Microsoft Azure	3	39	1
Inadequate VLAN Segmentation	Microsoft Azure	3	39	1
Insecure Third-Party Components	Microsoft Azure	2	39	1
Insecure OAuth Implementations	Microsoft Azure	2	39	1
Inadequate Monitoring and Alerting	Microsoft Azure	2	39	1
Insufficient Audit Trail	Microsoft Azure	2	39	1
API Misconfiguration	Microsoft Azure	2	39	1
Excessive Privileges	Microsoft Azure	2	39	1

Insecure VM Migration	Microsoft Azure	2	39	1
Cross-Region Data Replication Risks	Microsoft Azure	2	39	1
Exposed Secrets and Keys	Microsoft Azure	2	39	1
Certificate Validation Flaws	Microsoft Azure	2	39	1
Insecure API Management	Microsoft Azure	2	39	1
Insecure API Gateways	Microsoft Azure	2	39	1
Data Leakage through Misconfigured Storage	Microsoft Azure	2	39	1
Inadequate Encryption Strength	Microsoft Azure	1	39	0
Exposed Secrets and Keys	Microsoft Azure	1	39	0
Inadequate Encryption of Data at Rest	Microsoft Azure	1	39	0

Default Credentials	Microsoft Azure	0	39	0
Insufficient DDoS Protection	Microsoft Azure	0	39	0
Improper Authentication	Microsoft Azure	0	39	0

A7 - Case Study Post Mitigated Accountability Impact Table

Vulnerability	Asset	Likelihood	Impact	Risk Value
Insufficient Authorization	Microsoft Azure	32	36	12
Cross-Account Access Misconfigurations	Microsoft Azure	22	36	8
Improper Identity and Access Management	Microsoft Azure	21	36	8
Insecure Remote Management Access	Microsoft Azure	20	36	7
Default Credentials	Microsoft Azure	17	36	6

Insecure Default Settings	Microsoft Azure	16	36	6
Broken Authentication and Session Management	Microsoft Azure	16	36	6
Inadequate Secrets Management	Microsoft Azure	14	36	5
Insecure Access Control Policies	Microsoft Azure	14	36	5
Inadequate Data Backup and Recovery	Microsoft Azure	14	36	5
Inadequate Encryption of Data at Rest	Microsoft Azure	14	36	5
Inadequate IAM Policies	Microsoft Azure	13	36	5
Incomplete Visibility into Cloud Usage	Microsoft Azure	13	36	5
Data Leakage through	Microsoft Azure	13	36	5

Misconfigured Storage				
Insecure Handling of User Data	Microsoft Azure	12	36	4
Inadequate Protection Against Insider Threats	Microsoft Azure	12	36	4
Insecure API	Microsoft Azure	12	36	4
Insecure Permissions	Microsoft Azure	12	36	4
Container Security Flaws	Microsoft Azure	11	36	4
Insecure API Access	Microsoft Azure	11	36	4
Container Image Vulnerabilities	Microsoft Azure	11	36	4
Insufficient Incident Response Procedures	Microsoft Azure	10	36	4
Inadequate VLAN Segmentation	Microsoft Azure	10	36	4

Insecure API Exposure	Microsoft Azure	9	36	3
Insecure Key Management Practices	Microsoft Azure	9	36	3
Insufficient Controls for Infrastructure as a Service (IaaS) Security	Microsoft Azure	9	36	3
Backup Failures	Microsoft Azure	9	36	3
Cloud Security Misconfigurations	Microsoft Azure	9	36	3
Cryptographic Flaws	Microsoft Azure	9	36	3
Excessive Privileges	Microsoft Azure	8	36	3
Improper Authentication	Microsoft Azure	8	36	3
Inadequate Resource Isolation	Microsoft Azure	8	36	3
Insecure DevOps Practices	Microsoft Azure	8	36	3

Insecure Key Management	Microsoft Azure	8	36	3
Certificate Validation Flaws	Microsoft Azure	7	36	3
Insecure API Endpoints	Microsoft Azure	7	36	3
Insecure Third-Party Libraries	Microsoft Azure	7	36	3
Insecure API Gateways	Microsoft Azure	7	36	3
Inadequate Security Group Rules	Microsoft Azure	6	36	2
Insecure VM Migration	Microsoft Azure	6	36	2
Insecure REST API Configurations	Microsoft Azure	6	36	2
Inadequate Input Validation	Microsoft Azure	6	36	2
Exposed Secrets and Keys	Microsoft Azure	6	36	2
Bypassed URL Filtering	Microsoft Azure	6	36	2

Inadequate Encryption Strength	Microsoft Azure	5	36	2
Insecure Storage Configurations	Microsoft Azure	5	36	2
Broken Function Level Authorization	Microsoft Azure	5	36	2
Environment Variable Manipulation	Microsoft Azure	5	36	2
Insecure API Management	Microsoft Azure	5	36	2
Insecure Third-Party Components	Microsoft Azure	4	36	1
Insecure OAuth Implementations	Microsoft Azure	4	36	1
API Misconfiguration	Microsoft Azure	4	36	1
Inadequate Scalability	Microsoft Azure	4	36	1
Exposed Secrets and Keys	Microsoft Azure	4	36	1

API Key Exposure	Microsoft Azure	4	36	1
Cross-Region Data Replication Risks	Microsoft Azure	4	36	1
Inadequate Monitoring and Alerting	Microsoft Azure	2	36	1
Insufficient Audit Trail	Microsoft Azure	2	36	1
Insufficient DDoS Protection	Microsoft Azure	0	36	0
Data Loss from Accidental Deletion	Microsoft Azure	0	36	0