systems

MDPI

Article

Cyber Evaluation and Management Toolkit (CEMT): Face Validity of Model-Based Cybersecurity Decision Making

Stuart Fowler [1,*], Keith Joiner [2] and Siqi Ma [1]

[1] School of Systems and Computing, University of New South Wales, Canberra 2600, Australia; siqi.ma@unsw.edu.au
[2] Capability Systems Centre, University of New South Wales, Canberra 2600, Australia; k.joiner@unsw.edu.au
* Correspondence: stuart.fowler@unsw.edu.au

Abstract: The Cyber Evaluation and Management Toolkit (CEMT) is an open-source university research-based plugin for commercial digital model-based systems engineering tools that streamlines conducting cybersecurity risk evaluations for complex cyber-physical systems. The authors developed this research tool to assist the Australian Defence Force (ADF) with the cybersecurity evaluation of complicated systems operating in an increasingly contested and complex cyber environment. This paper briefly outlines the functionality of the CEMT including the inputs, methodology, and outputs required to apply the toolkit using a sample model of the process applied to a generic insider threat attack. A face validity trial was conducted on the CEMT, surveying subject-matter experts in the field of complex cybersecurity analysis and risk assessment to present the generic case study and gather data on the expected benefits of a real-world implementation of the process. The results of the face validity broadly supports the effectiveness and usability of the CEMT, providing justification for industry research trials of the CEMT.

Keywords: cybersecurity; model-based systems engineering; CEMT; cyber; digital engineering; mission engineer; cyber-worthiness; model-based risk assessments; threat modelling

Citation: Fowler, S.; Joiner, K.; Ma, S. Cyber Evaluation and Management Toolkit (CEMT): Face Validity of Model-Based Cybersecurity Decision Making. *Systems* **2024**, *12*, 238. https://doi.org/10.3390/systems12070238

Academic Editors: Gregory S. Parnell and Eric Specking

Received: 27 May 2024
Revised: 18 June 2024
Accepted: 29 June 2024
Published: 1 July 2024

1. Introduction

1.1. Background

Making informed and accurate decisions on the acceptability of cybersecurity risk to complicated engineered systems is a challenging problem. There is an understandable urge to try and simplify the problem to a purely technical challenge by mandating a set of technical-, procedural-, and personnel-focused security controls and auditing the compliance to that baseline set of security controls. However, this reductionist approach often neglects the complexity of the system's operating context, mission, and safety criticality, especially where organisational boundaries separate aspects like estate infrastructure from ICT infrastructure and user platforms (i.e., ships, aircraft, vehicles, etc.). Not all controls are relevant to all systems, and not all systems require the same level of assurance when it comes to mitigating cyber threats. Contemporary security assessment methodologies [1,2] have introduced risk-based decision-making processes to allow compliance frameworks to be applied. The specific steps vary between the various methodologies, but the generic approach is that the results of a compliance audit and control effectiveness determination are used as an input to a qualitative risk decision made by an appropriately authorised decision maker.

Figure 1 depicts a generic process for how these risk-based decisions are made. At step 1 in the top right corner, a system is designed with security being one of the quality factors driving that design. A security assessment is then made, typically by auditing a set of best practice mitigation controls (step 2) such as the ISM [3], NIST 800-53 [4], or similar, and assessing the residual risk based on the level of compliance found during that audit

(step 3). This residual risk determination includes an implicit and qualitative analysis by the security assessor of the effectiveness of any compliant controls and the importance of any non-compliant controls. The residual risk is then provided to a decision maker, who holds a level of authority within the organisation that allows them to accept the risk, and they compare this against the organisation's risk tolerance. Based on this comparison, they either accept the risk, authorising the system for operation, or reject the risk, requiring further mitigation of the residual risk by the design team before the system can be accepted into operation (step 4).

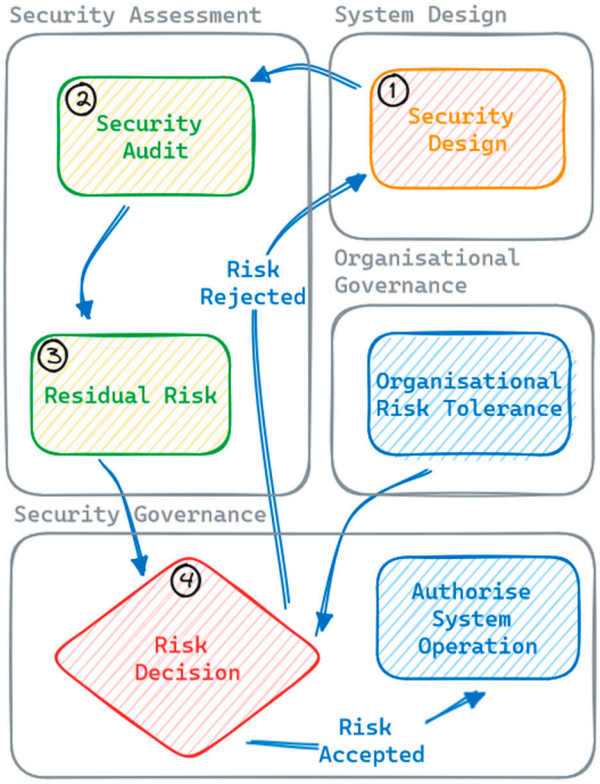

Figure 1. Generic process for making cybersecurity risk decisions under a compliance-driven risk assessment paradigm.

One of the challenges introduced by this approach is that it is reliant on the control baselines used for the audit being directly relevant to the system being audited. The security assessment usually includes a step to exclude controls that are not applicable, but any non-compliances against applicable controls are often difficult to assess objectively because the control baselines are often not prioritised within the context of the system being assessed. The individual security assessor then needs to implicitly assess the importance of a particular control non-compliance and incorporate that into the residual risk. This process is often subjective and opaque, which can make it difficult to explain and justify to other participants in the risk assessment that do not have cyber expertise. Consequently, the decision maker usually either needs greater cyber expertise to understand the assessment, or they need to blindly trust the assessor.

Compliance-driven approaches can leave decisions makers in a position where it is difficult to understand the level of risk being accepted, and consequently hamper effective risk decisions. The need for a better way to describe and assess risk has been identified in contemporary military doctrine, as evidenced by the following statements from the U.S. Department of Navy (DON):

> *"The DON needs a new approach to cybersecurity that goes beyond compliance because our over-reliance on compliance has resulted in insecure systems, which jeopardise the missions these systems support. Instead of a compliance mindset, the DON will shift to*

> *Cyber Ready, where the right to operate is earned and managed every day. The DON will make this transition by adhering to an operationally relevant, threat-informed process that affordably reduces risk and produces capabilities that remain secure after they have been delivered at speed".* [5] (p. 7)

The reliance on the opaque expert judgement of the assessor is exacerbated because these assessments are typically documented in spreadsheets [6,7] and databases [8], which do not effectively describe the system's security risk to a layman. Moreover, such spreadsheets can be difficult to maintain and update as new threats emerge throughout the capability sustainment life cycle, as the underlying control baseline is updated over time and as the capability design evolves through development and in-service. Emerging techniques [8] use 'compliance-as-code' techniques to improve reusability and to reduce the difficulties of amalgamating assessments at a system-of-systems level. However, these techniques primarily address the amalgamation of a compliance level, rather than the amalgamation of residual risk levels, meaning that the challenges of opaque expert judgement remain.

A 'best-practice control' set is more easily achieved in a predominantly information communications technology (ICT) system, as there is a uniformity in ICT architectures compared to cyber-physical systems, or what ICT professionals commonly refer to as the 'edge'. However, the 'edge' forms an integral part of the cyber-attack surface and the kill-chain effects of complex engineered systems, which means that the direct application of the best-practice control sets derived from that comparatively uniform ICT environment can lead to a focus on the wrong type of controls. Additionally, the expert judgement of importance and priority performed by assessors can be influenced by an implicit bias towards those uniform ICT architectures. The best-practice control sets help immensely in terms of assessment efficiency within that uniform environment; however, the cyber-physical environment often requires a return to first-principles security and engineering to obtain an accurate cybersecurity assessment. First-principle cybersecurity assessments of engineered systems are especially useful in developing risk mitigations to recommend.

1.2. Literature Review

The existing alternatives in the literature to this compliance-driven risk assessment are threat-based risk assessments. Threat-based risk assessment methods provide better flexibility in terms of applying the process to different system contexts at the expense of requiring a higher level of expertise and understanding from the assessors.

The widely known 'MITRE ATT&CK' framework [9] facilitates threat-based assessment, but it is more of a taxonomy and repository of known behaviours to understand threats and attack patterns, rather than an evaluation and assessment methodology. While the framework derives mitigations and detections for implementation in systems, a direct application of these would essentially be another compliance-driven assessment, which can suffer from the same challenges introduced earlier.

Similarly, Mission-based Cyber Risk Assessments (MBCRA) such as Cyber Table Top [10] and Mission-based Risk Assessment Process for Cyber (MRAP-C) [11], required under the U.S. Department of Defence acquisition processes, provide an example of threat-based approaches to deliver risk assessments. However, an analysis of the MBCRA implementations concluded in 2022 that the explainability of the product of the MBCRA can still be limited due to the data output format [12]. The MBCRA process is inherently collaborative, which improves the situation in terms of opaque expert judgement, whereby decision makers can rely on the tempering and recording effects of group consensus. However, the limited explainability still constrains the goal of truly informed cyber risk decision making.

The literature also outlines approaches that may help with the explainability of cybersecurity assessments. According to Kordy et al. [13], "*the great advantage of graph-based approaches lies in combining user friendly, intuitive, visual features with formal semantics and algorithms that allow for qualitative and quantitative analysis*". The intuitive nature of these approaches shows promise for facilitating collaboration with stakeholders who may not have a deep cybersecurity background. Kordy et al. [13] categorised graph-based security

analyses into two broad categories: statically modelled graphs and sequentially modelled graphs. Both categories can deal with either just attacks or both attack and defence.

Attack trees, first outlined in 1999 by Weiss [14] and popularised by Schneier [15], are the basis for many of the statically modelled graphs used for cybersecurity analysis. These graphs depict a hierarchical tree structure to describe the combination of events that need to occur for a cybersecurity risk to be realised and are broadly similar to the fault trees common to system safety analysis. Attacks trees have been successfully incorporated into the likelihood calculations of traditional risk assessment methodologies from 2014 [16], and have been extended from 2016 to include the depiction of defensive actions [17] and termed attack-defence trees. In 2015, the risk likelihood assessments that flowed from attack trees were estimated by converting the attack tree structure into a Bayesian network in order to combine probabilities of the various nodes on the attack tree [18,19].

The sequentially modelled graphs outlined in the literature are comprised of a sequence of events that an attacker must perform in order to compromise a target system. This approach provides a focus on the intent of an attacker; one of the attributes missing in the standard systems security approaches and a fundamental advantage when the threat is evolving quicker than the system. Inherently, such approaches can focus the assessment on innovative mitigations to limit the likelihood of successful attack. The concept of insecurity flow graphs was introduced early by Moskowitz and Kang [20] to model this attack sequence, though this concept attempted to reduce the graph into a mathematical model, losing some of the intuitiveness benefits of a graph-based approach.

The construction of misuse case graphs, also known as abuse case graphs, is another approach in the literature that focuses on the actions of the attacker and their interaction with the target system. Appearing in the literature at the turn of the century in McDermott [21] and Sindre et al. [22], misuse cases borrow from the use case concept common in software development and systems engineering practices but focus on how an adversary, rather than an authorised user or operator, might interact with a system. More recent extensions to the misuse case concept include overlaying the misuse cases with the system architecture [23] and developing misuse case coverage metrics as a proxy for security risk [24]. On their own, misuse cases require extensive textual encoding to adequately convey the full risk picture [25].

While misuse cases are often used to elicit security requirements [26], mirroring the purpose of use cases in software development, Matuleviaius [27], in 2017, linked misuse cases to the mal-activity graphs proposed by Sindre [28]. These mal-activity graphs provide a sequential model of the activities taken by the attacker—much like the insecurity flow graphs introduced by Moskowitz and Kang [20]—and can be used to provide additional detail to support misuse case graphs.

The literature does include some comparison studies between the attack tree methodologies and the misuse case methodologies. Both Opdahl et al. [29] and Karpati et al. [30] found that while attack tree methodologies tended to identify more threats, there were significant differences between the types of threats found by each technique. This would suggest that a combined approach such as that outlined by Tondel et al. [31], which looks at both misuse cases and attack trees, might be the most effective approach.

Efficiency in risk assessment is also an important concern, particularly for organisations with many legacy cyber-physical systems to assess that have only recently exploited ICT networking (aka Internet of Things (IoT)). While there has been some literature looking to increase the reusability of these graph-based threat models [32] these techniques still require knowledgeable experts to apply them to systems individually. However, in the development of bespoke, complex cyber-physical systems, the technical knowledge of those involved with the assessment is not necessarily the key limitation of a successful risk assessment—the ability to explain that technical risk assessment to a decision maker who understands that the broader operational and strategic context can be the issue.

1.3. Cyberworthiness

'Cyberworthiness' is a term that was created recently within the Australian Department of Defence and invokes existing worthiness concepts under which Defence is responsible for internally regulating the safe use of their systems. The existing worthiness systems began in the 1990s with airworthiness regulations and governance [33,34], which led to seaworthiness [35] and landworthiness [36] being defined and enforced. 'Worthiness' requires a level of regulation and monitored assurance over design, deployment, and sustainment. As 'worthiness' is often synonymous with continued safety and functionality, these regulations are applied more rigorously based on safety criticality.

Cybersecurity has challenged Australian Defence, as it has the U.S. Defence, though Australia is arguably lagging in this realm [37–40]. As cybersecurity threats became more prevalent and necessary across more safety-critical systems that do not fit into the traditional definition of an ICT system, it became necessary to apply broader and more enforceable assurances than those applied to generic ICT systems. Consequently, cyber-worthiness was first termed [41,42] and is defined as:

> *"The desired outcome of a range of policy and assurance activities that allow the operation of Defence platforms, systems and networks in a contested cyber environment. It is a pragmatic, outcome-focused approach designed to ensure all Defence capabilities are fit-for-purpose against cyber threats".* [43]

The inclusion of the worthiness construct is a deliberate reference to the established technical assurance frameworks that underpin concepts such as airworthiness and seaworthiness. This requires a level of rigour in the engineering analysis that provides confidence and assurance that the claim of worthiness is accurate, complete, and supported by evidence. However, a worthiness claim must also be understandable by those making decisions about the operation of a system, as outlined by the Principles for Governance of Seaworthiness from the Defence Seaworthiness Management System Manual [44]:

> *"2.10 The seaworthiness governance principles require that seaworthiness decisions are made:*
>
> *a. mindfully—decisions are more effective and less likely to have unintended consequences when they are made with a thorough understanding of the context, the required outcome, the options available, and their implications now and in the future*
>
> *b. collaboratively—obtaining input from all stakeholders and engaging in joint problem-solving results in better decisions (bearing in mind that collaboration does not necessarily require consensus)*
>
> *c. accountably—decisions only become effective when people take accountability for making them happen*
>
> *d. transparently—decisions are more effective when everyone understands what has been decided and why".* [44] (p.33)

Cyberworthiness then requires not only the rigorous application of cybersecurity analysis, but also demands the explicit inclusion of a broad range of stakeholders and a focus on genuinely informed decision makers to ensure that they can reasonably take accountability for the decisions they are asked to make.

1.4. Addressing the Problem

The solution to the identified challenges is not as simple as just implementing academic approaches. These approaches are not new, with some almost 25 years old, but they do not have the industry traction in engineered systems development. The primary concerns that limit the direct application of these techniques to complicated cyber-physical systems with evolving complex threats are:

- Usability—there is limited ability to easily create and review these graph-based threat assessments, especially in large, complex systems;

- Efficiency—reusability of these assessments is limited in comparison to compliance-based approaches that re-apply a common control set;
- Maintainability—it is difficult to update complex graph-based assessments without specialised toolsets as the system or threat environment evolves.

In particular, sequential graph methods can be difficult to understand at a macro level, while static graph methods can be unwieldy and difficult to maintain as the system changes. However, structured languages such as those in model-based systems engineering (MBSE) may be able to be used to unify sequential and static graph methods so that the security designers and evaluators can produce one set of graphs and automatically generate views of other graphs based on the relationships defined in the modelling process. The inherent strengths of model-based approaches can also help maintainability through the object-oriented nature of the dataset, allowing for rapid update and consistency checks of complex views.

It is our thesis that a rigorous and streamlined threat-based approach to security analysis that describes threat vectors and risks using graphs and leverages model-based systems engineering techniques to support maintainability can deliver a methodology that overcomes many of the cybersecurity assessment challenges. Our thesis can be evaluated by delivering such a methodology to project teams who face these challenges when using the traditional compliance-based risk assessments currently implemented in government and industry. This threat-based approach would align with the overall concept of a mission-based cyber risk assessment but drive the implementation of that concept towards one that delivers an intuitive and maintainable output of threat graphs for through-life cyber resilience. Such an approach likely helps cybersecurity assessors by leveraging the systems and safety engineering culture and competencies in government and industry.

This threat-based approach would also address the decision-making principles underpinning cyber-worthiness. The approach must be mindful of the context in which the system is operating, be intuitive enough to facilitate collaboration between all stakeholders, be understandable by key decisions makers so that they can reasonably take accountability for their decisions and be transparent in terms of how the conclusions trace to the analysis.

Our research aimed to determine whether this more extensive threat-based methodology and associated toolset would help improve cybersecurity risk assessments across more diverse systems, especially cyber-physical systems at the 'edge', in comparison to traditional compliance-driven approaches. Such improvements would be categorised across dimensions of effectiveness, efficiency, and usability to understand both the benefits and costs associated with an alternative approach. Seeking feedback on the methodology and associated toolkit through industry workshops is key to ensuring that practitioners can provide a preliminary vetting of any proposals before allocating resources to costly implementation trials. Structured workshops with representative domain experts to examine the early and likely validity of the method can be referred to as 'face validity' and has been debated academically by Allen et al. [45].

Our primary research question asked:

- Are integrated threat models, developed using model-based systems engineering (MBSE) techniques, an effective and efficient basis for the assessment and evaluation of cyberworthiness?

To help answer this primary question, we derived the following three sub-questions that characterised effective cyberworthiness based on how informed the decision makers were, efficiency in terms of the reusability of the assessment effort, and maintainability of the assessment through-life as well as a qualitative question seeking preference data from practitioners:

- Do the developed threat models provide decision makers with the necessary understanding to make informed security risk decisions?
- Does the process provide sufficient reusability and maintainability that the methodology is more efficient than prevailing compliance-based approaches?

- Do cybersecurity risk practitioners prefer the integrated threat model approach to traditional security risk assessment processes?

2. Materials and Methods

Our methodology for investigating this problem set requires a level of development of a proposed security assessment approach that brings together threat-based analysis, graph-based presentation techniques, and model-based implementations.

This methodology began by determining what we needed our threat-based analysis to take as inputs and produce as outputs, which we defined as threat-based cybersecurity engineering. We then began prototyping implementations of this approach using model-based systems engineering techniques to iteratively identify the features that worked and any challenges that needed to be overcome [46]. We refined this prototype to develop a candidate toolset for implementation and evaluation [47]. This resulted in the creation of the Cyber Evaluation and Management Toolkit (CEMT), which included a generic example implementation of a common threat to complex systems [48]. This example implementation was then presented at conference tutorials and industry briefings with feedback from participants received via survey [49]. This provided a result set that described the face validity of the approach, outlining preliminary impressions from industry and academic experts as to the feasibility and expected benefits of a wider implementation of the approach.

This section provides a summary of the work carried out to define the threat-based cybersecurity engineering concept and develop the Cyber Evaluation and Management Toolkit (CEMT) alongside a high-level summary of the generic example implementation used for the face validity surveys. The results of these face validity surveys are then presented in the results section of this paper, with a discussion of the results at the end of this paper, which will also outline future work to conduct implementation trials on complicated engineered systems in the real world to finalise the evaluation of the developed approach.

2.1. Threat-Based Cybersecurity Engineering

In order to guide the development of the CEMT, it was necessary to first articulate an evaluation framework in which the toolkit would be operated. For this reason, our threat-based cybersecurity engineering concept was created, and the associated inputs and outputs were articulated.

Threat-based cybersecurity engineering is defined as the process by which a system's capability is continuously assessed to ensure that relevant cybersecurity mechanisms are incorporated into the capability design and the resultant residual cybersecurity risks are accurately determined and explained. This definition was developed within the context of cyberworthiness and the need for an approach to cyber evaluation that provides rigorous assurance to underpin and justify any cybersecurity recommendations.

Threat-based cybersecurity engineering is primarily aimed at securing complicated cyber-physical systems that require a first-principles approach to identifying how to appropriately secure a system from complex threats. It is threat-based and context-focused and derives a set of relevant security controls that should be incorporated within a system. It builds on the systems security engineering processes defined in [50] and incorporates the framing and categorisation of the NIST Cybersecurity Framework [51]. It is a collaborative process that brings together experts from different parts of an organisation and facilitates meaningful discussion through intuitive analysis and quantitative reasoning. The cybersecurity engineering process is one of facilitation, which seeks to incorporate the expertise of various stakeholders and present assessments in a transparent, traceable, and explainable manner, rather than being reliant on cybersecurity polymaths to conduct opaque and ultimately subjective assessments.

The generalised inputs and outputs to and from threat-based cybersecurity engineering are depicted in Figure 2.

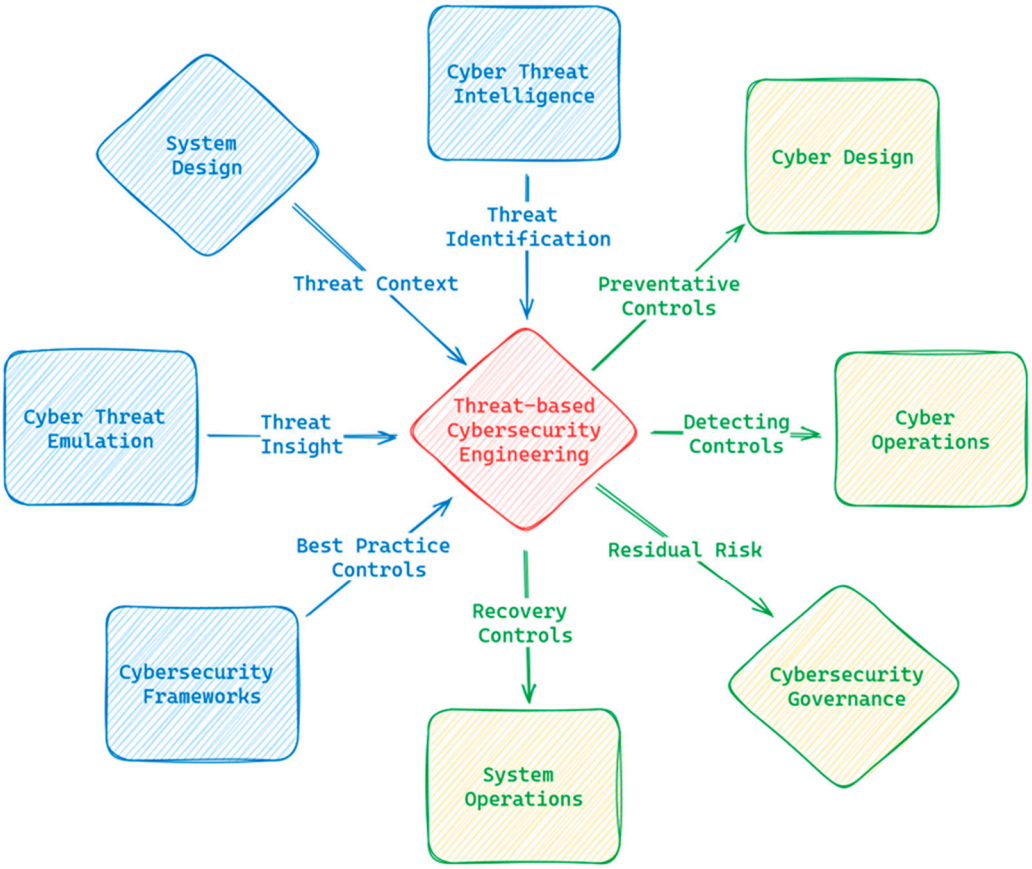

Figure 2. Threat-based cybersecurity engineering context.

The primary inputs to threat-based cybersecurity engineering are:

- Threat Context, derived from the system or capability design/architecture;
- Threat Identification, provided by the Cyber Threat Intelligence function within an organisation;
- Threat Insight, contributed by the Cyber Threat Emulation function within an organisation;
- Best Practice Controls, distilled from the various cybersecurity frameworks available within the cybersecurity body of knowledge.

The Threat Context input represents the functions, interfaces, and components within the system or capability that is being evaluated. This defines the threat surface for the cyber evaluation and informs both the scope and structure of the threat model. The design of the system and the missions which the system performs are the primary source of the threat context.

The Threat Identification input is the attack surface of the system, essentially the set of attack vectors and the threat actors that perform them, which are relevant to the system being evaluated. This is typically contributed by the Cyber Threat Intelligence function, which monitors and identifies potential threat actors, and can consist of both generic threat sets that cover common threats to information systems such as network penetration threats and supply chain attacks, and more specific advanced threat sets such as electronic attack or signal processing exploitation.

The Threat Insight input describes the tactics, techniques, and procedures that would be used by the identified threat actors to achieve their malicious intent. This encompasses the threat vectors by which the system might be attacked and articulates the steps that must be taken by a threat actor in order to compromise the system being evaluated. The Cyber Threat Emulation function, which maintains a detailed understanding of the methods used by threat actors to access systems, is the key contributor of the Threat Insight input.

The Best Practice Controls input encompasses the corpus of information that captures the cybersecurity mitigation mechanisms that can be used to prevent, detect, respond to and recover from a cyber-attack. Importantly, the intent here is to not limit the evaluation to the application of a single control framework audited to determine compliance, rather, the superset of control frameworks is distilled by the cybersecurity engineer when developing and reviewing the threat model, and controls are selected based on their applicability to that threat model, rather than their providence within a particular control framework. A rigorous methodology must simultaneously consider control sets from NIST [2,38], ASD [3], MITRE [9], and others, and select them based on relevance, rather than aligning to any one specific framework.

The primary outputs from threat-based cybersecurity engineering are:

- Preventative Controls, a baseline of preventative cybersecurity controls within the system, for inclusion in the system design;
- Detecting Controls, a baseline of detection and response controls relevant to the system, for implementation by the Cyber Operations function within an organisation;
- Recovery Controls, a baseline of recovery and resilience controls relevant to the system, for implementation by the System Operations function within an organisation;
- Residual Risk, the overall risk presented by the threats to the capability given the mitigation mechanisms that are in place.

The Preventative Controls output encapsulates the mitigation mechanisms that can be applied to the system components to prevent a threat vector from being successfully completed. These are implemented into the cyber design of the system or capability and the level of implementation of these controls drives the overall likelihood of a cybersecurity risk being realised.

The Detecting Controls output includes the mitigation mechanisms that can be incorporated into the system to identify an attack on the system and respond to that attack. These controls are allocated to the Cyber Operations function of an organisation, where analysts are responsible for security monitoring and incident response for the capability being evaluated.

The Recovery Controls output describes the mitigation mechanisms that can contribute to the ability to recover from a successful cyber-attack. These controls are incorporated by the System Operations function of an organisation, where personnel are responsible for ensuring that the system or capability is operational and performant. The implementation of these controls can mitigate the consequence of a cybersecurity risk being realised.

The Residual Risk output represents the overall risk that exists within a system or capability, based on the threats to the system and the control mechanisms in place to mitigate that threat. This residual risk is passed to the Cybersecurity Governance function within an organisation, where it is compared against the organisations' risk appetite to determine whether the level of residual risk is acceptable.

The threat-based cybersecurity engineering artefacts should be expressed using threat graphs that provide explicit traceability between the inputs and outputs. This facilitates the justification of the outputs, based on the inputs, to provide assurance of the relevance of the control set that is ultimately derived. It must also be documented in a way that is agile and responsive to change so that the assessment artefacts can be iteratively updated throughout the system life cycle, ensuring that the outputs of the process remain evergreen and the overall risk management outcomes are enduring.

2.2. Cyber Evaluation and Management Toolkit (CEMT)

Our research then involved the development of a methodology that satisfied this threat-based cybersecurity engineering approach using model-based systems engineering as a potential candidate for solving the challenges identified earlier in this paper. Our early prototype work [46] set the scene for the adaptation of SysML diagrams such as use case diagrams and activity diagrams to model misuse cases and mal-activity flows in the target system. The threat models developed using these views were then used as a lattice for

tracing security mitigations and a map for assessing the priority risks in the system. This high level approach is depicted in Figure 3.

Figure 3. High level approach for our model-based implementation of threat-based cybersecurity engineering.

The initial, informal feedback we received on the prototype was that it required too much focus on how to build the models, rather than the outputs of the assessment. These usability concerns were the primary driver of a refinement activity where a custom plugin was developed to abstract away the complexities of the underlying model-based systems engineering tools and of SysML as a language. The result was the Cyber Evaluation and Management Toolkit (CEMT), which is a plugin to the CATIA Magic software package that implements a taxonomy of custom stereotypes and a suite of custom diagrams to streamline the process initially developed in our prototype work [47].

The CEMT is freely available, and the plugin, source code, and detailed documentation can be publicly accessed on GitHub [48]. The CEMT development included the creation of a generic worked example of two misuse cases, Physical Incursion and Insider Threat, which is also distributed with the toolkit as a sample implementation. This worked example was presented at academic conference tutorials and professional industry workshops, and survey responses were sought from the participants to assess the face validity.

This worked example is summarised in the following section to provide context to the results presented later in this paper.

2.3. CEMT Sample Model

This section outlines the worked example that is provided as a sample implementation with the CEMT. This sample model provides a generic assessment of the Insider Threat and Physical Incursion misuse cases of a generic system. It is not meant to be an exemplar implementation, but is rather meant to illustrate the overall methodology, provide an instructive example of how to develop models using the CEMT, and demonstrate the outputs that are created by the plugin.

The summaries in this section are not meant to be comprehensive descriptions of an entire threat vector and are deliberately kept high-level for brevity, and example diagrams were chosen for readability in this format. All diagram types, along with a comprehensive exploration of the low-level process for creating these diagrams, are available for viewing by the reader in the online documentation available in GitHub [48], where complex diagrams can be more easily displayed and explored.

2.3.1. Threat Modelling

The Threat Modelling phase is the starting point of the CEMT modelling approach and comprises the following steps:

1. Misuse case diagrams;
2. Intermediate mal-activity diagrams;
3. Detailed mal-activity diagrams.

The first step in this phase involves the creation of a CEMT misuse case diagram, which is a customised version of the SysML use case diagram. This diagram is used to define the top-level threats to the system under evaluation by using misuse cases to represent those top-level threats. Threat actors that perform the misuse cases are also placed on these diagrams and linked to the misuse cases they perform. An example of a misuse case diagram is shown in Figure 4.

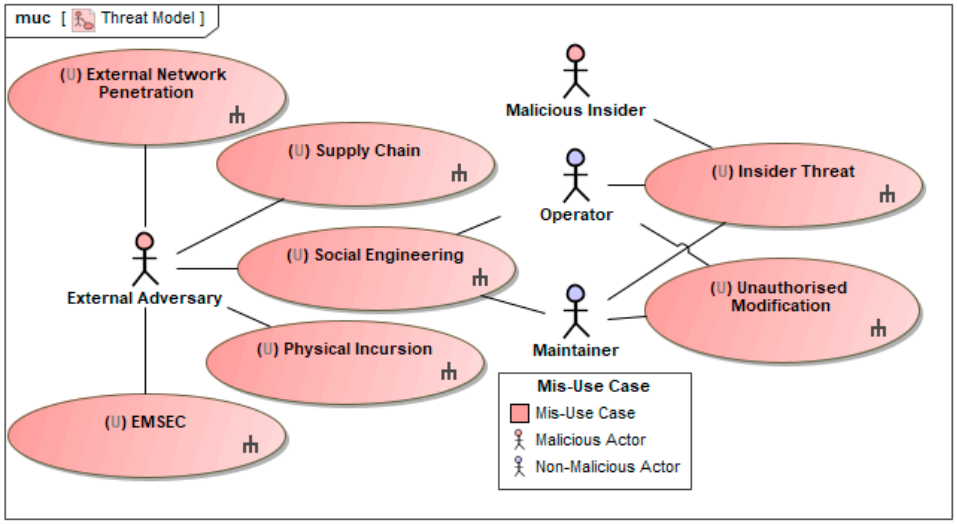

Figure 4. CEMT misuse case diagram from the generic sample model.

For each misuse case, a CEMT mal-activity diagram is then created, which articulates the various threat vectors that could be used to achieve the top-level threat. Figure 5 depicts an example of this for the Insider Threat misuse case, where the steps required to perform this attack are defined. These steps can each have another CEMT mal-activity diagram below them to describe the more specific details. This results in a nested series of flowcharts that show the actions the attacker needs to take as well as the ways in which a defender would be able to detect the attackers' activity. This nested behaviour can be seen in Figure 6, which depicts the internal detail of the Exploit System Access block from Figure 5 with the input and output pins on each figure representing the same object, allowing for the threat flow to pass between diagrams at different levels of the nested hierarchy.

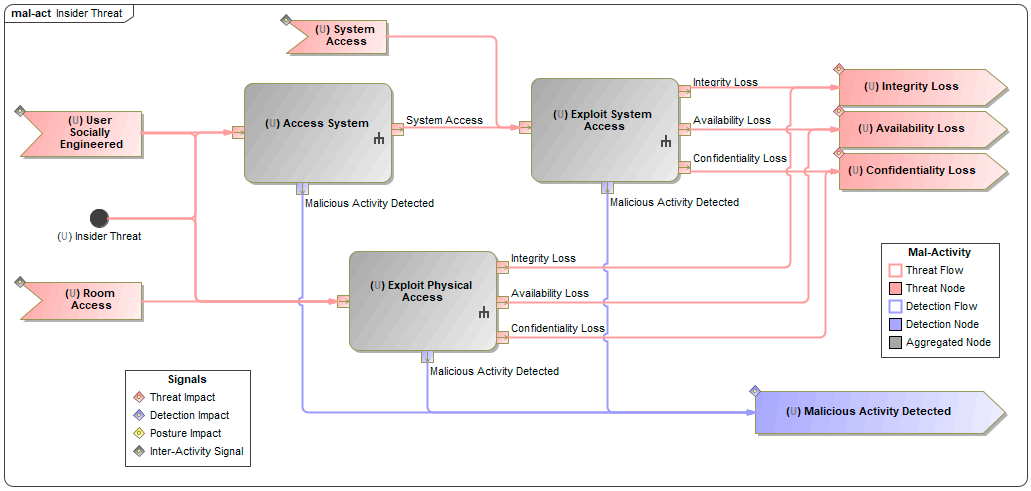

Figure 5. Intermediate CEMT mal-activity diagram for the Insider Threat misuse case.

These nested mal-activities diagrams continue until they reach the bottom level of the hierarchy necessary to articulate the elemental steps that the attacker needs to take to complete the attacks. An example of the lowest level detailed mal-activity diagram is shown in Figure 7, which depicts the internal detail of the Access Sensitive Data block from Figure 6. These detailed diagrams also outline the ability of these threat actions to be detected.

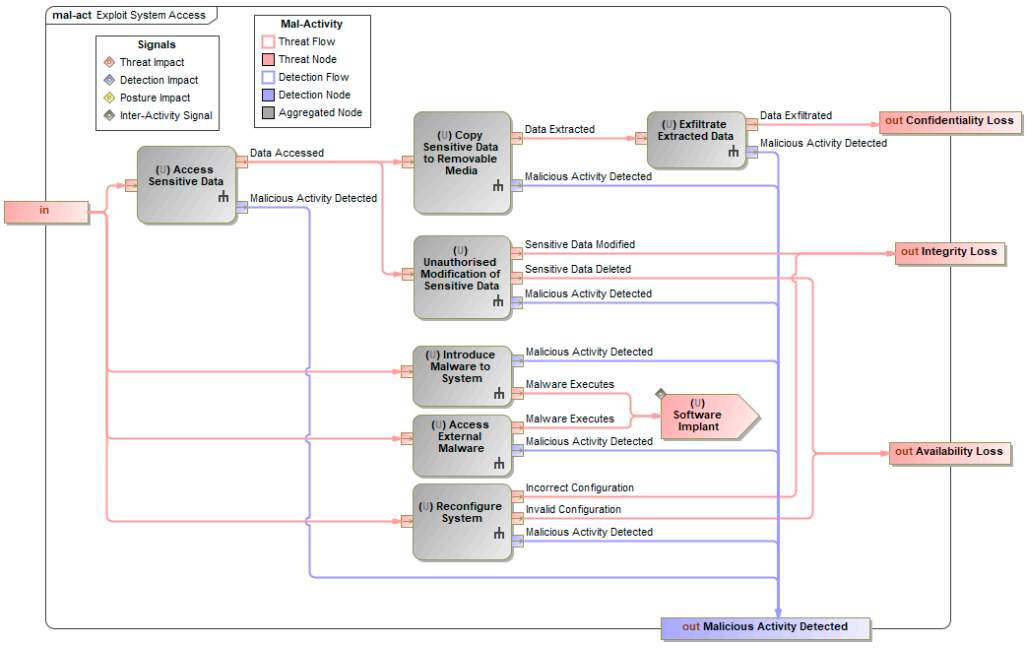

Figure 6. Intermediate CEMT mal-activity diagram for the Exploit System Access mal-activity.

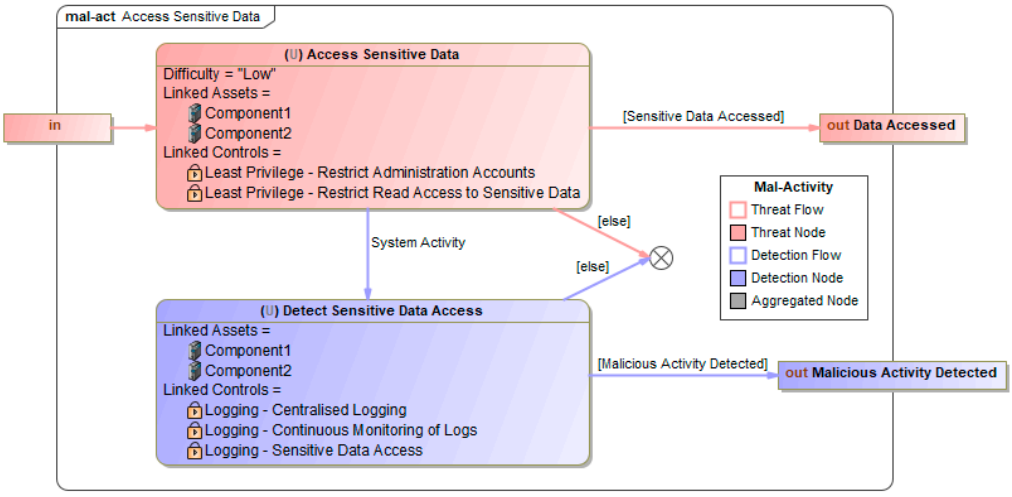

Figure 7. Detailed CEMT mal-activity diagram for the Access Sensitive Data mal-activity.

Following the completion of this Threat Modelling phase, a comprehensive threat model that identifies the threats to the system and the threat vectors through which those threats could compromise the system was developed.

2.3.2. Threat Mitigation

The next phase in the CEMT modelling approach is the Threat Mitigation phase, which comprises the following steps:

1. Allocating assets to the threat model;
2. Tracing controls to the threat model.

The first step involves reviewing the threat nodes and detection nodes in the detailed CEMT mal-activity diagrams in the threat model and identifying the components in the system that are affected by those actions. These components are then created as Asset objects and linked to the specific steps of the threat model. This is shown in Figure 7, where the Component1 and Component2 assets were linked to the relevant nodes of the threat model as they are the components of the system that contain the sensitive data that could be accessed in these actions.

The next step involves reviewing the same actions in the threat model and proposing controls that could potentially be implemented to prevent the threat action from being completed. These are created as objects and linked into the threat model. These controls should be sourced from best-practice frameworks and the knowledge and experience of the modeller. Importantly, the intention is not only to identify controls that are implemented in the system, or even just controls that should be implemented in the system. Ideally, any control that could feasibly be implemented in the system and would directly mitigate the likelihood of the action being completed by the threat actor should be created and linked to the specific step. Figure 7 shows both the preventative controls linked to the relevant threat node and detecting controls linked to the relevant detection node.

The CEMT then instantiates each of the linked security controls onto the linked assets. This creates a list of asset-allocated mitigation mechanisms that are derived from the threat model, ensuring that these controls to be audited are both relevant to the system and are directly contributing to the mitigation of a threat vector. The implementation state of these instantiated objects can be audited and then input into the model to complete the Threat Mitigation phase.

2.3.3. Risk Assessment

The final phase in the CEMT modelling approach is the Risk Assessment phase, which comprises the following steps:

1. Attack tree assessment;
2. Parametric risk analysis;
3. Risk evaluation.

This phase involves the modeller working with various system stakeholders to review the threat vectors, identify those that present the highest risk, and then construct simulation analyses of those high-risk threat vectors to produce quantitative data to support an overall residual risk assessment. This phase is primarily a review phase, where the CEMT takes the data and relationships defined in the earlier phases and builds new views of those data to support the assessment and analysis by stakeholders.

The first such view is an attack tree, an example of which is shown in Figure 8. These attack trees trawl the threat flowcharts defined in the Threat Modelling phase and present this information as a directed acyclic graph where each node is a step taken by the threat actor. This results in an attack tree where every branch of the tree represents a single threat vector, start to finish. The purpose of this diagram is to provide a summary view of the entire attack surface, and allows stakeholders to review the threat model to ensure that there is sufficient coverage of important threat vectors.

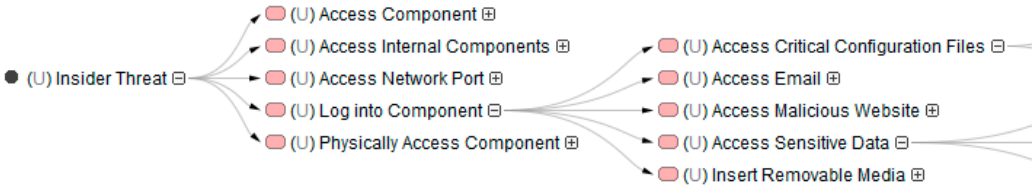

Figure 8. Partial CEMT attack tree—forward diagram from the CEMT sample model.

In addition to the attack trees, the CEMT also builds bowtie diagrams that allow the modeler to select a node in the attack tree, and the bowtie diagrams display the security controls that are relevant to that node, grouped into mitigating controls and detection controls, colour-coded by the implementation state of that control based on the audit results input into the CEMT. In conjunction with the attack trees, this allows stakeholders to query each node to determine what the system is doing to address that step in the threat vector.

As an example, the assessment stakeholders may focus on nodes that are common to a number of branches on the attack tree, as implementing additional controls at those nodes are likely to yield bigger benefits as they will simultaneously mitigate multiple threat

vectors. Equally, a focus may be placed on particularly short branches of the attack tree, as these represent situations where there is not much natural defence-in-depth, which means that the mitigations on the nodes in that branch are often critically important. Ultimately, the purpose of this assessment is to provide a qualitative review of the entire attack surface, ensure that critical nodes are adequately mitigated, and identify potentially high-risk threat vectors for further analysis via quantitative simulation.

Once the high-risk threat vectors have been identified, a detailed analysis can be performed on those threat vectors using the CEMT parametric risk diagram. These diagrams are created by the CEMT for the high-risk threat vectors identified during the analysis of the attack trees. Each node in the threat vector is added to the parametric diagram along with the associated detection action and chains them together to set up a parametric simulation that calculates a residual probability and detection probability for the threat vector. A partial extract of one of these diagrams is shown in Figure 9, which shows two nodes of the threat vector and also depicts the related control effectiveness values and the linked controls, colour-coded by their implementation state.

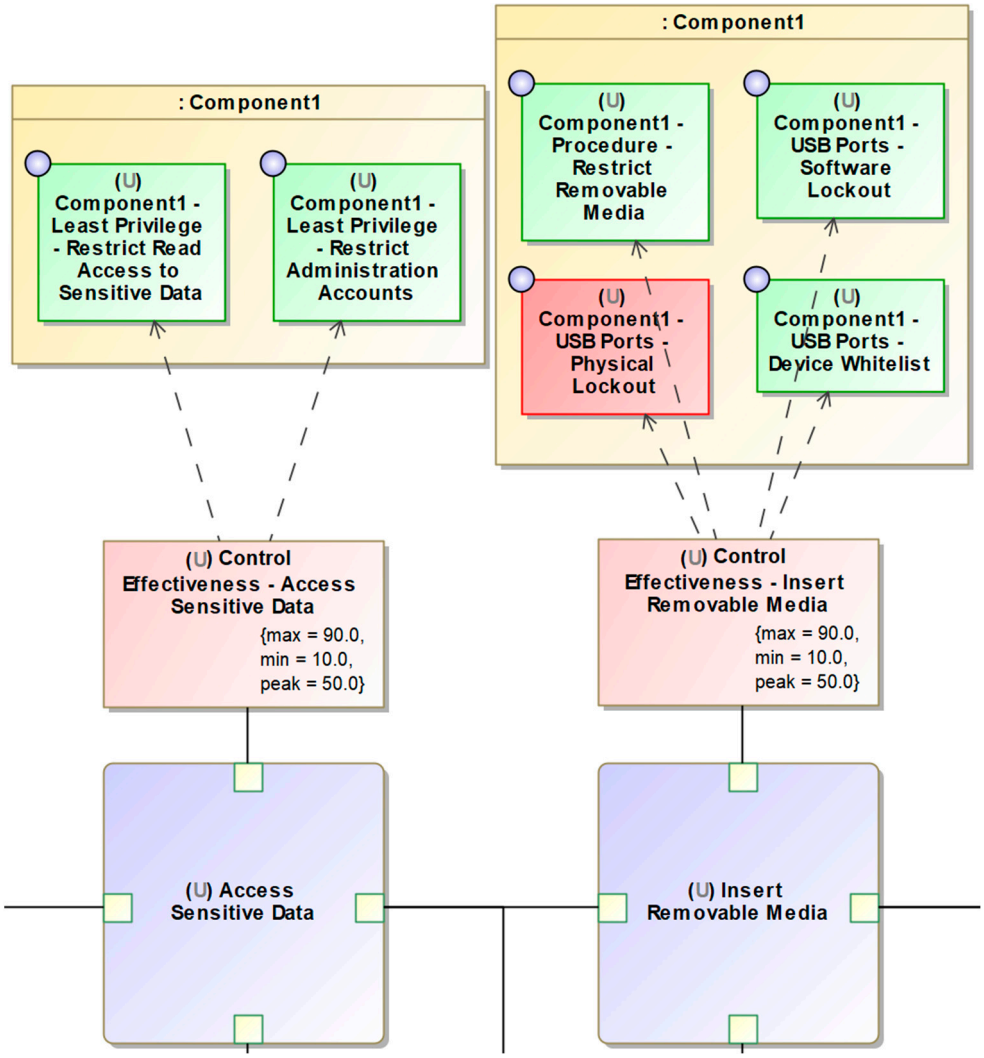

Figure 9. Partial CEMT parametric risk diagram from the CEMT sample model.

Importantly, Figure 9 highlights the control effectiveness value, which is a critical input that needs to be provided to the CEMT during this step. The control effectiveness value is the percentage of attempts of the action that would be prevented by the controls that are in place. This value is likely to be subjective, so the modeler can input a value range to account for the uncertainty in the evaluation of this input value. This function defaults

to a triangular distribution with a minimum, maximum, and peak value. The process is repeated for the associated detection actions, and once these values have all been set, the CEMT can run a Monte-Carlo simulation and produce a histogram of the resulting residual risk probability and detection probability. An example of these histograms is shown in Figure 10.

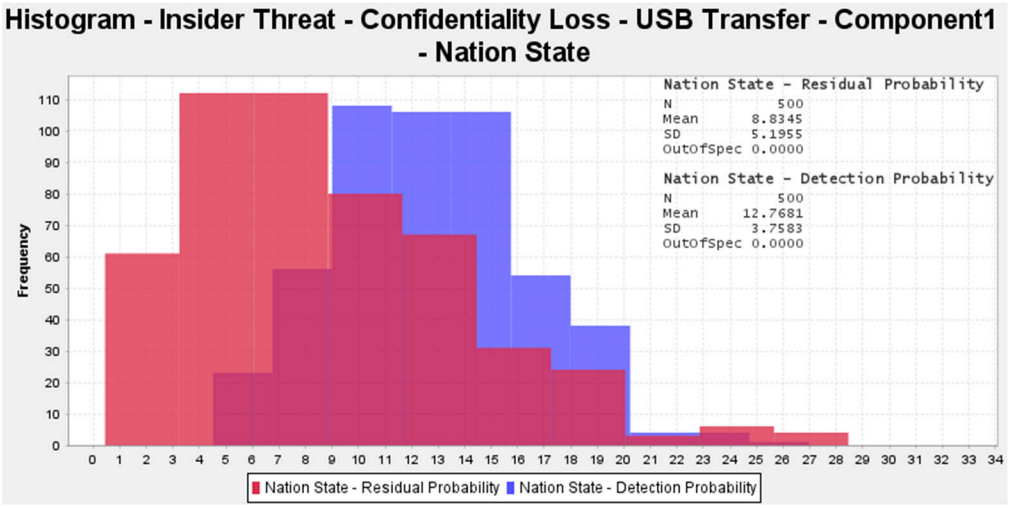

Figure 10. Simulation histogram for the CEMT parametric risk diagram.

The residual risk is only realised when all of the steps are successfully completed by the adversary, so the calculation is a simple multiplication of the probabilities of each step being completed successfully, that is:

$$P_{Residual} = P_{Initial} \times P_{1_T} \times P_{2_T} \times \ldots \times P_{x_T} \quad (1)$$

where $P_{Initial}$ is the probability of the attack being attempted by an adversary, as set by the modeler, and P_{x_T} is the probability of the individual threat node being successfully completed by the attacker. P_{x_T} is calculated from the multiplication of two values, E_{x_T}, which is derived from the control effectiveness value input by the modeler for that threat node, and the difficulty input, D_{x_T}, which is a user-configurable mapping between the difficulty values assigned to each threat node in the Threat Modelling phase and a quantitative probability that they would be able to complete the attack step:

$$P_{x_T} = (E_{x_T} \times D_{x_T}) \quad (2)$$

E_{x_T} is defined as the percentage of attacks that would not be prevented by the controls that have been implemented on that threat node and can be based on either historical data or the expert judgement of the modelers.

D_{x_T} is defined as the percentage of attempted attacks that would be successful in isolation of any controls. D_{x_T} represents the difficulty of the step irrespective of any mitigating controls and is driven by the capability of the threat actor. As an example, a cyber payload delivered via an electronic attack against a signal processor would be inherently more difficult than a simpler node such as inserting a USB device, and this difference is captured in the difficult input term.

This probability calculation treats each threat node as an independent event, and where this does not hold true, for example, if a control set is reused later in a particular risk chain, then the modeler would need to account for this in the effectiveness value for the second appearance of the control in that chain. If the attack progresses because the control was ineffective the first time it appeared, it is unlikely to be effective the second time the attacker encounters it later in the chain, and the modeler must reduce the effectiveness in the later node to ensure accuracy.

Detection, on the other hand, can occur at any step, and a single successful detection node would result in the attack being detected. Thus, the overall detection probability is calculated as a probability summation:

$$P_{Detection} = [P_{Initial} \times P_{1_D}] + [P_2 \times P_{2_D}] + \ldots + [P_x \times P_{x_D}] \quad (3)$$

where P_{1_D} is the probability of the individual detection node being successful and is calculated as the multiplication of two values, E_{x_D}, which is derived from the control effectiveness value input by the modeler for that detection node, and the avoidance input, A_{x_D}, which is a user-configurable mapping between the capability of the threat actor and a quantitative probability that they would avoid the detecting controls:

$$P_{x_D} = (E_{x_D} \times A_{x_D}) \quad (4)$$

E_{x_D} is defined as the percentage of attempts that would be detected by the controls that have been implemented on that detection node and can be based on either historical data or the expert judgement of the modelers. A_{x_D} is defined as the percentage of potential detections that would be avoided by the attacker, based on the capability of the attacker to avoid detection, and can be based on either historical data or the expert judgement of the modelers. P_x is the probability that the detection node will be triggered, which occurs if the previous threat node was successfully completed, therefore:

$$P_x = P_{Initial} \times P_{1_T} \times P_{2_T} \times \ldots \times P_{(x-1)_T} \quad (5)$$

Finally, the CEMT builds a summary risk table using these simulation values, along with the steps required to complete threat vectors, a list of the participating assets, and a list of the relevant mitigation mechanisms. The assessment stakeholders will then map that information to the qualitative likelihood values used within their particular organisation to assess the consequence of the threat vector succeeding and determine the final risk rating for that particular threat vector. Once all critical threat vectors have been assessed, this summary risk table will be populated with a risk assessment for each of the high priority threats to the system, completing the CEMT process.

2.4. Achieving Threat-Based Cybersecurity Engineering

The CEMT approach effectively implements the threat-based cybersecurity engineering concepts outlined earlier in this paper. The CEMT threat model is derived directly from the system context and can be informed by the Cyber Threat Intelligence function within an organisation, who can work collaboratively with the modelers to determine the most appropriate misuse cases based on known threat actors. Threat emulation teams can be engaged to help analyse the system design through the lens of those misuse cases, provide valuable insight into the feasibility of certain attacks, and validate the specific steps that would need to be taken to complete those attacks, as articulated by the mal-activity diagrams.

Best-practice control frameworks are leveraged to determine the set of controls that are tied to the specific steps of the threat model, but the controls from those best-practice frameworks are only incorporated where they are addressing a specific threat-emulation-validated threat action derived from the threat-intelligence-informed misuse cases.

The overall output of the CEMT approach is a set of mitigating and detecting controls that can be implemented to manage risk within the system, and a calculated residual risk based on the implementation state of those mitigating and detecting controls. This achieves all of the outputs depicted in Figure 2, with the exception of the recovery controls. The incorporation of recovery controls into the CEMT approach is an intended extension to the process, and this is discussed further in Section 4.2 of this paper.

This is all achieved within an assessment methodology based on graphical representations of the cybersecurity risks in the system, allowing for transparent collaboration between disparate stakeholders. The underlying structured datasets can also be lever-

aged, much like model-based system design, to increase the efficiency of the modelling by improving the reusability, maintainability, and speed of the assessment using automation.

2.5. Efficiency through Automation

The ability to increase the efficiency of complex system development is seen as one of the potential benefits of the model-based systems engineering approach [52], and this ability to automate in order to minimise the engineering impact of iterative change to the design is a key component of the application of agile methods into system engineering [53]. It is important that the cybersecurity analysis of these complex systems takes advantage of the potential of automation during model-based risk assessments to ensure that it supports, rather than becomes a roadblock to, the need for efficient iteration of the system design. There are three primary methods of automation supported by the CEMT:

- Automated update of complex drawings and simulations to ensure that changes to the design or threat environment can be incorporated efficiently into the threat model;
- Automated model validation to ensure that basic review tasks are automated, allowing expert reviewers to focus on the actual threat assessment component;
- Automated documentation to ensure that the process of creating enduring design artefacts is efficient and accurate.

The automation of complex drawings and simulations in the CEMT is achieved through a combination of bespoke relationship expressions that leverage the capabilities inherent in the underlying digital engineering tool and purpose-built JavaScript API calls that extend these inherent capabilities to achieve the desired views in the CEMT.

The first example of this is the development of the attack trees and bow-tie diagrams in the Risk Assessment phase. While attack trees are not particularly novel and have been used in security analysis since at least 1994 [54], one of their limitations is that they are primarily static [13] and are difficult to update once a significant level of complexity or interconnectedness is reached. The CEMT deliberately uses mal-activity diagrams to allow the modeler to express the threat vectors in a manner that maximises the efficiency of inputting the data, but then automatically generates an attack tree using queries of the structured information within those mal-activity diagrams, which provides the same information in a format that is significantly easier to review. This allows for changes to be made to the mal-activity diagram as the system design matures or the threat environment changes, and the CEMT will automatically keep the attack trees up to date.

This drawing automation is extended through the use of JavaScript API calls to generate new diagrams from existing data in the model where the in-built query language does not sufficiently support the necessary level of automation. The generation of parametric diagrams and the associated simulations in the Risk Assessment phase is automated in this way, allowing a modeler to select a threat vector in the attack tree and generate the simulation in an automated fashion. Similarly, the simulations can be run and the results saved into the model through the automation macros provided in the CEMT.

One of the limitations of using JavaScript API in this way is that the generated diagrams are not kept up to date when the source data in the mal-activity diagrams are changed. To address this limitation, the second type of automation—automated model validation—is also used in the CEMT. This automation leverages the active validation functionality in the underlying digital engineering software to provide real-time validation of a model against a defined set of constraints. The CEMT provides several active validation constraints that will be highlighted to the modeler when one of the diagrams built through the JavaScript API is out-of-date due to a change in the source data, allowing the modeler to rerun the automation macro to resolve the inconsistency. In addition to this, a number of active validation constraints are included that provide guidance to the modeler as they are building out the model to ensure the completeness of the model. This real-time assistance to the modeler helps to resolve issues as early in the modelling process as possible while also removing the burden of identifying these errors from a manual peer review team.

The final type of automation used in the CEMT is automated document generation. While one of the primary purposes of digital engineering is to move away from document-based engineering analysis [55–57], there remains a need to provide a snapshot of the model at particular points in time for the purposes of record-keeping, archiving, and sharing information with stakeholders who may not have access to digital engineering tools. Using a system safety engineering analogy, the threat model can be thought of as the cybersecurity equivalent of the safety case [58], but there is still a practical need for a snapshot of the threat model much like there is a practical need for a safety case report [59]. To support this, the CEMT provides an automated export of the output of the threat model into a Microsoft Word document that summarises the scope of the misuse cases that have been evaluated and presents the implementation status of the control baseline as well as the results of the risk assessment on the critical threat vectors. This provides an artefact that can be quickly and easily generated and regenerated, which can then be distributed into an organisation's existing risk governance processes for endorsement and acceptance.

3. Results

3.1. Face Validity Trial Setup

Initial research was conducted on the CEMT to assess the face validity of the proposed approach. This investigation assessed whether the approach seemed promising to relevant experts working in cybersecurity evaluations of complicated systems to inform continued future work on our approach and associated tools. Presentations, tutorials, and workshops on the CEMT process and the worked example described in Section 2.3 were provided to many cybersecurity engineers, systems engineers, and project managers across the communities of a major Australian government department and the Australian power industry during 2022. Participants were offered the opportunity to complete a short survey to capture their impressions of the approach with respect to the effectiveness, efficiency, and expected benefits.

3.2. Face Validity Trial Data Collection and Setup

A total of 20 responses were received (n = 20). Fourteen respondents identified themselves as having a background as a cybersecurity subject matter expert, with an average experience of 7.8 years. Sixteen respondents identified themselves as having worked with cybersecurity risk assessments in some capacity, with an average experience of 5.5 years.

The breakdown of the organisation type of the respondents consisted of 40% coming from public sector Defence and 60% coming from private industry. There was also an even spread of disciplines represented in the survey. A total of 30% of the respondents described themselves as fulfilling a governance role in the development of cyber-physical systems, 30% identified themselves as fulfilling a management role, 30% identified themselves as fulfilling a technical specialist role, and 10% responding with other.

However, the survey respondents did see themselves as having a technical dimension to the main area of expertise, as outlined in Figure 11. A total of 70% of those surveyed identified their main area of expertise as either IT/Computing or engineering, with only 10 per cent selecting management as their main area of expertise. Figure 12 highlights the spread of expected roles in the CEMT process amongst the survey responses. These results showed a reasonable spread between the different roles, with no one role dominating the responses.

Overall, the demographics of the survey respondents were balanced between different specialisations within the process of cybersecurity risk assessments. The survey participants had a strong level of experience in cybersecurity, both generally and in terms of risk assessment. The bias towards a technical engineering/IT background is to be expected as the respondents were selected from attendees at professional presentations and conference tutorials in the development of complex cyber-physical systems, which would naturally select participants with that type of background experience. Ultimately, this cohort presents

a reasonable foundation from which to draw conclusions about the face validity of the CEMT process.

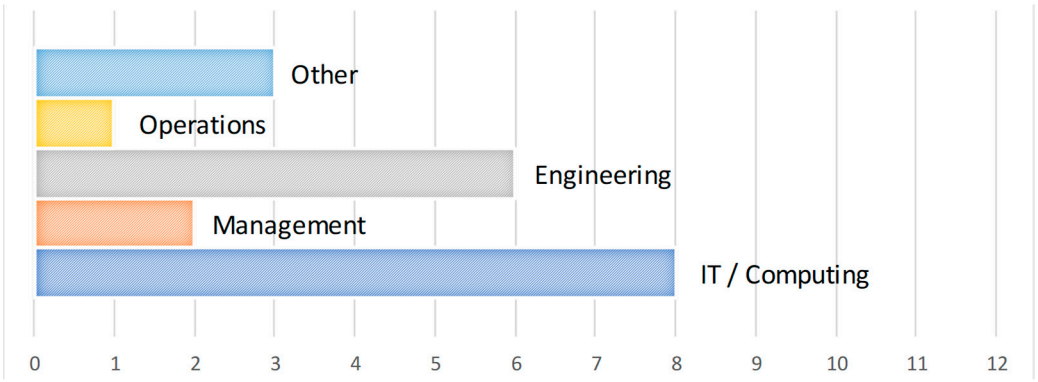

Figure 11. Responses to the question: "What is your main area of expertise?" Bars indicate the number of respondents, out of the 20 respondents, that provided that particular response.

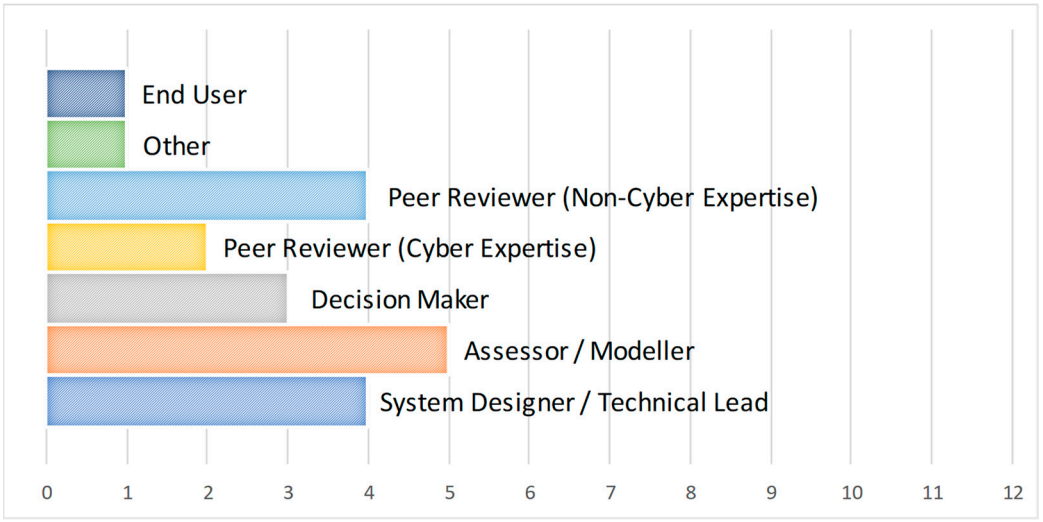

Figure 12. Responses to the question: "What option best describes your expected role in a cyberworthiness assessment process?" Bars indicate the number of respondents, out of the 20 respondents, that provided that particular response.

The survey responses are included in Table 1, with the summary results graphed in Figure 13. The responses were broadly positive, with most responses supporting the expected benefits of a cyber assessment using the CEMT.

These results generally support the claims that the approach can produce contextually relevant, collaborative assessments that provide an improved understanding of a system's cyberworthiness. However, they also show concerns about the difficulty of producing the assessment, its overall accuracy, and the model's ability to address cyberworthiness. Nevertheless, the overall response to the approach in the survey was positive, with 75% of respondents giving the approach an overall rating of either excellent or very good, as depicted in Figure 14.

These face validity findings are not being used as scientific evidence to confirm the technical validity of the approach, but rather as an indicator of the expected benefits of the approach, informing the appropriateness of future research to fully trial its technical validity.

Table 1. Responses to the survey questions. Table cells indicate the percentage (%) of the 20 respondents that selected the particular response to each question.

	Survey Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Q1	The CEMT produces risk assessments that are tailored to the context in which the system operates	0	0	15	50	35
Q2	Cyberworthiness assessments are simple to produce using the CEMT	5	0	40	35	20
Q3	The CEMT is an effective use of time	0	0	30	25	45
Q4	The CEMT process is intuitive	0	5	25	45	25
Q5	The CEMT encourages stakeholders to work collaboratively to determine the residual risk level	0	0	10	35	55
Q6	The CEMT clearly identifies which security controls are important to the system	0	0	5	55	40
Q7	The CEMT produces transparent cyberworthiness assessments	0	5	10	40	45
Q8	The CEMT facilitates informed decision making with respect to the identified cybersecurity risks	0	0	5	50	45
Q9	The CEMT produces cyberworthiness assessments that have ongoing value through the future phases of the capability life cycle	0	0	10	40	50
Q10	The CEMT would improve my understanding of the cyberworthiness of a system	0	0	10	20	70
Q11	The CEMT produces accurate assessments of a system's cyberworthiness	0	10	20	35	35
Q12	The CEMT facilitates the engagement of stakeholders and the provision of meaningful input from those stakeholders into a cyberworthiness assessment	0	0	20	40	40
Q13	The cyberworthiness assessments produced by the CEMT are sufficiently detailed	0	5	20	30	45
Q14	The CEMT identifies the relative impact of security controls with respect to the cyberworthiness of the system	0	5	15	40	40
Q15	The CEMT is not overly dependent on the subjective opinion of subject matter experts	0	0	30	50	20
Q16	The CEMT provides sufficient information to allow decision makers to be accountable for their decisions	0	10	15	35	40
Q17	The CEMT clearly highlights the areas of greatest cyber risk to the system	0	0	15	35	50
Q18	The CEMT adds value to a system and/or project	0	0	5	35	60
Q19	The CEMT provides a complete and comprehensive approach to determining cyberworthiness	5	10	10	50	25
Q20	The CEMT is an improvement over existing cyberworthiness assessment processes	0	5	10	20	65

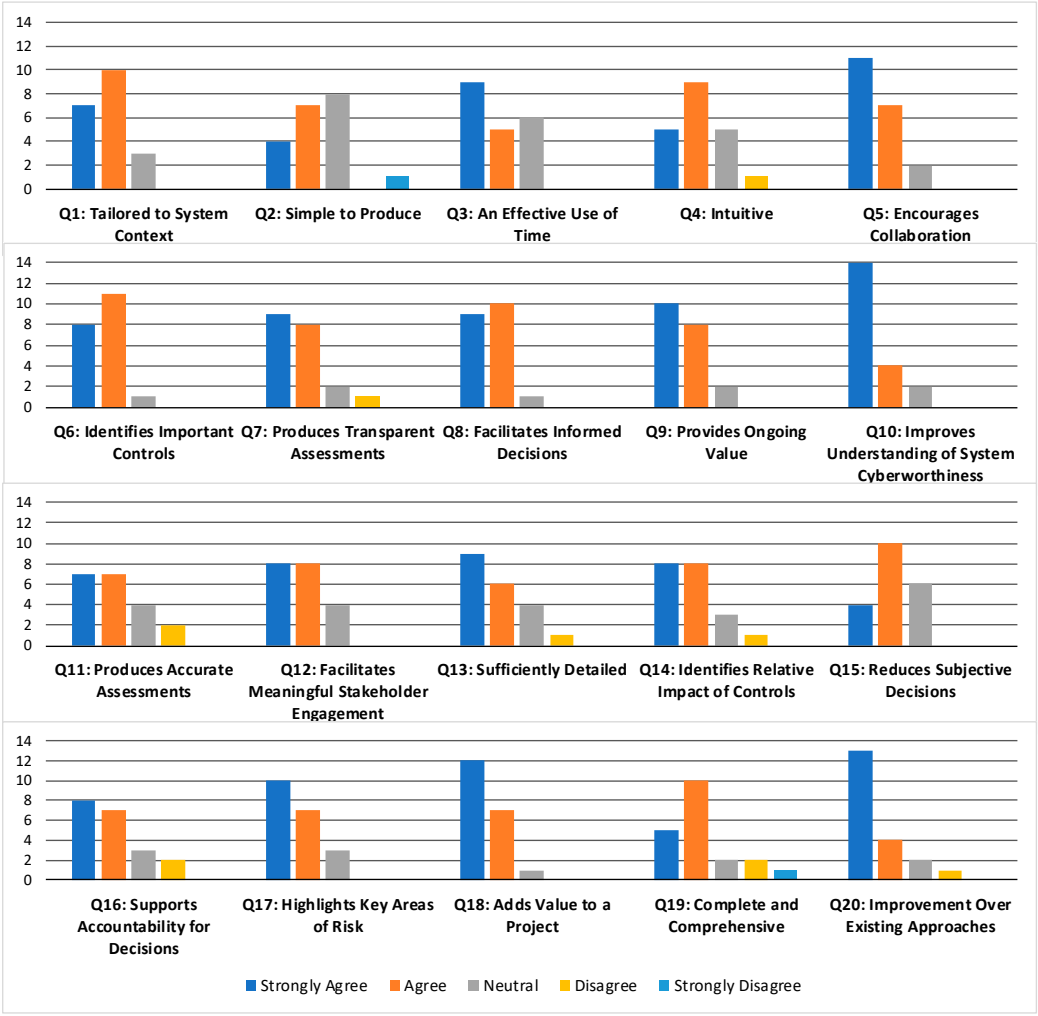

Figure 13. Summary results of the CEMT face validity survey. Bars indicate the number of respondents, out of the 20 respondents, that provided that particular response.

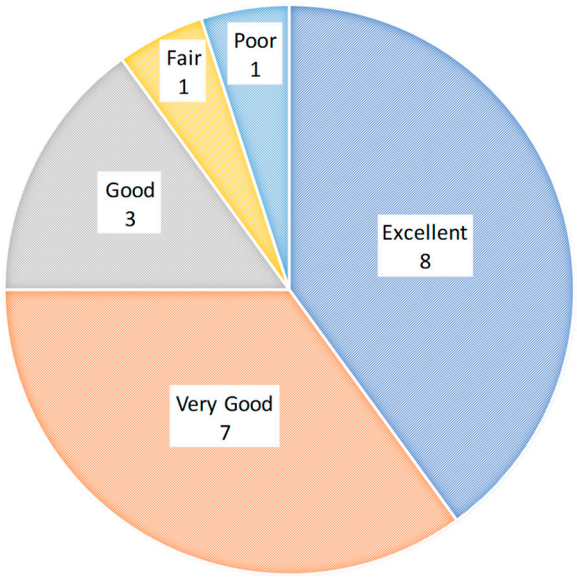

Figure 14. Overall rating of the threat-based cybersecurity engineering approach using the CEMT, based on responses to the face validity survey. Number in each pie segment indicates the number of respondents, out of the 20 respondents, that provided that particular response.

4. Discussion

The CEMT was able to meet most of the expectations of the surveyed users concerning outcomes, where they broadly welcome threat-based cybersecurity engineering using graph-based approaches for cybersecurity analysis created using model-based systems engineering techniques. The tool promises a repeatable and consistent method for producing (1) abuse case graphs, referred to as misuse case diagrams in the CEMT, and (2) insecurity flow graphs, referred to as mal-activity diagrams in the CEMT, which contain the necessary model-based attributes that allow for the dynamic generation of data views. Surveyed users welcomed the feature for static graph techniques such as attack trees to be generated in real-time to help ensure alignment between the attack trees and insecurity flow graphs and to simplify the maintenance of these complementary approaches.

The CEMT also provides a promising framework for prioritising the importance of different mitigation techniques by explicitly tracing mitigation techniques to threat vectors and then tying those threat vectors to a quantitative analysis of the risk likelihood. This feature aims to allow cyber analysts to rapidly reassess different configurations of the design that have different mitigation implementation states and recalculate the likelihood of critical threat vectors. A secondary advantage of using common model-based systems engineering tools as the basis of the CEMT is that it can also be integrated into design models of a system that has been developed using these model-based approaches.

There are still some limitations of the CEMT that were highlighted in the validity workshops and when compared to the generic threat-based cybersecurity engineering model presented at the start of this paper. While the mitigating and detecting controls were a clear output of the CEMT alongside the residual risk ratings, the tool does not currently derive recovery controls. Deriving these controls within the toolkit is an avenue for future development of the tool and could likely be achieved by creating a set of activity diagrams that address recovery from the identified high-risk threat vectors. These diagrams would articulate a set of actions an operations team would need to take to recover the system in the event of that risk being realised, and the recovery controls that would make the team's efforts possible or more effective would be traced to each step in that recovery process. The level of implementation of these controls could then mitigate the consequence of the risk, as the ability and responsiveness of a recovery action would directly mitigate the consequence of a risk being realised.

The results of our face validity trial into the CEMT show promise for the ability of the toolkit to meet the needs of assessing the cyberworthiness of a system using a threat-based cybersecurity engineering approach. The overall rating of the approach amongst the respondents was overwhelmingly positive, with 40% of responses rating it as excellent and a further 35% rating it as very good.

The surveys also confirmed that the respondents believed that the CEMT could address the three primary challenges that the current compliance-based risk assessments face, that is:

- Appropriateness of the assessed controls to the system being assessed, as demonstrated by the responses to Question 1;
- Prioritisation of controls, as demonstrated by the responses to Questions 6 and 14;
- Ability for non-expert decision makers to understand the assessment, as demonstrated by Questions 7, 8, and 17.

The existing compliance-based assessments such as the Australian Government PSPF [1] and NIST RMF [2] provide a consistent list of best-practice controls that are applied against a system regardless of its operating environment, while the threat-based approach of the CEMT starts from the system context and its operational environment to identify threat vectors and then link best-practice security controls that are relevant to those threat vectors. The developed methodology inherently constrains the scope of control assessments to the context of the system being assessed.

As these controls are then explicitly traced to the threat vectors, the importance of controls can be analysed based on the number of threat vectors that each control mitigates

as well as the criticality of those threat vectors in terms of the likelihood of the threat vector being successful and the consequence of the impact of the threat vector. This architectural view provides a basis for having informed discussions with decision makers as to the importance of implementing a particular control, or not.

These discussions with decision makers are further supported by the CEMT due to its graphical nature and explicit modelling. Analysts produce step-by-step flowcharts of each possible threat vector, and control effectiveness is described in terms of reducing the likelihood of a single step being performed by an attacker. This provides a lattice for robust conversations around assumptions and exclusions in the modelling, removing much of the opaque expert judgement inherent in high-level cyber risk assessments.

The results of the face validity survey also confirmed that some concerns around the CEMT approach remain, particularly in terms of the simplicity to produce and the overall efficiency of the project. While significant steps were made between the early prototypes and the CEMT in terms of usability and simplicity, the survey responses indicate that at least some concern remains about these facets of the CEMT approach.

4.1. Significance

In 2015, Nguyen et al. [60] provided a comprehensive review of the literature associated with using model-based techniques for the security assessment of cyber-physical systems. This state of research was revisited by Geissman and Bodden [61] in 2020. They found several approaches and methodologies in the literature that leveraged model-based techniques including SysML-based approaches to address the security assessment of complex cyber-physical systems.

Many of these existing approaches such as the cyber security requirements methodology (CSRM) [62] and Larsen et al. [63] are focused on the definition of security requirements and provide a methodology for managing the implementation of those security requirements through the systems design life cycle. Our CEMT approach distinguishes itself from these approaches by being explicitly threat focused, and uses this threat analysis to derive the relevant security properties within the system.

Several other methodologies including SAVE [64], Navas et al. [65], and Geissman et al. [66] are also threat focused and utilise similar threat modelling techniques such as misuse cases to the CEMT. However, these approaches provide a high-level threat model that interacts with a detailed white-box model of the system. The CEMT extends the misuse case approach with detailed mal-activity diagrams that create a white-box threat model that interacts with either a black-box or white-box system model. This feature enables modellers to create detailed threat models to explain the cybersecurity threat to non-cybersecurity experts including non-technical decision makers.

Some emerging methodologies such as MBSESec [67] also provide more detailed modelling of adversary behaviour in the form of activity diagrams. However, the novelty provided by the CEMT in comparison to these lies in the explicit control traceability to the detailed threat model and the use of specific SysML profiles that support the rapid creation of summary visualisations such as attack trees and bow-tie diagrams. While this traceability and visualisation does not necessarily produce a better or more accurate threat model, it does enhance the ability of non-specialists to interact with and critically review the threat model to make more informed risk decisions.

The threat focus of the CEMT is also the primary point of difference between it and the other existing extension profiles that address security assessments in model-based languages. For example, UAF [68], UMLSec [69], and SysML-Sec [70] all provide taxonomies and extension profiles that focus on the allocation of security requirements and the linkages of assets, vulnerabilities, and mitigations. The primary point of difference in the CEMT is that security requirements are derived from the threat modelling process itself, and that the allocation of mitigations to assets is explicitly performed via common relationships to the threat model.

To illustrate the significance of the CEMT approach, five key attributes that contribute to a comprehensive cyberworthiness evaluation were defined:

- Extended Model-Based Taxonomy—an extension of an open model-based systems engineering language such as UML or SysML; this is provided to facilitate a model-based approach;
- Threat Focused—the threats to the system, rather than a best-practice control baseline or asset hierarchy, is used as the focal point of the assessment;
- Detailed Adversary Modelling—the actions of the adversary are modelled in detail, facilitating a precise discussion and review of any threat analysis;
- Visualisation and Simulation of Threat—detailed adversary modelling is expressed in simplified graphs such as attack trees, and branches of those graphs can be simulated quantitatively;
- Explicit Traceability to Threats—derived security controls are directly traceable to adversary actions, facilitating discussion and review of the importance of each control in terms of the malicious action it mitigates.

Table 2 provides a summary comparison of these published model-based security assessment frameworks to the CEMT across these five key attributes. Figure 15 provides a visual depiction of the level to which these five key attributes are served by these existing frameworks, with green representing those attributes that are well-served by existing techniques, yellow indicating attributes that are served by some of the existing techniques, and red indicating those attributes that are not currently found in existing techniques.

Table 2. Comparison of model-based security assessments along five key attributes.

	Model-Based Security Assessment Approach	Extended Model-Based Taxonomy	Threat Focused	Detailed Adversary Modelling	Visualisation and Simulation of Threats	Explicit Traceability to Threats
1	CSRM [62]	Y	N	N	N	N
2	Larsen et al. [63]	Y	N	N	N	N
3	SAVE [64]	Y	Y	N	N	N
4	Navas et al. [65]	Y	Y	N	N	N
5	Geissman et al. [66]	Y	Y	N	N	N
6	MBSESec [67]	Y	Y	Y	N	N
7	UAF [68]	Y	N	N	N	N
8	UMLSec [69]	Y	N	N	N	N
9	SysML-Sec [70]	Y	N	N	N	N
10	CEMT	Y	Y	Y	Y	Y

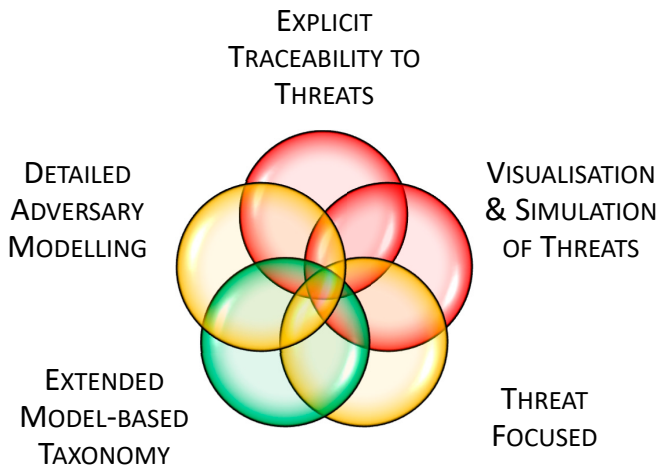

Figure 15. Significance of the CEMT approach across five key attributes of model-based security assessments.

4.2. Future Work

While the results to date have been promising, there are still a number of avenues for future work, which can be broadly grouped into further refinement of the toolkit, further expansion of the toolkit, and further case study trial implementation of the toolkit.

Refinement of the toolkit involves continuing to address the latest concerns around usability, simplicity, and efficiency. This refinement could be achieved by developing further automation within the toolkit to increase the user experience of creating models, producing additional training materials to help with the initial learning process or through the development of reusable libraries of common threat vectors, similar to the generic case study examples.

Expansion of the toolkit could involve the development of new modelling steps or diagram views to increase the effectiveness of the toolkit. This effort would likely include the expansion of the toolkit to address recovery from a successful cyber-attack through the development of response activity diagrams from which are derived a set of relevant recovery controls. It could also include the development of additional summary diagrams to aid in the explanation of the cyber risk to different audiences.

The primary future work, however, is through further implementation of the CEMT approach. The face validity trials provide a view of the expected benefits of the toolkit, but to assess the overall technical validity of the approach, further implementation on real-world systems must be completed and evaluated. These implementation trials are currently underway, with scope for wider trials and research into other industries and countries. We look forward to reporting on the results once the initial trials are complete.

5. Conclusions

The CEMT provides an example of a model-based technique that can support decision making around the acceptability of cybersecurity risk in complicated engineered systems. The cybersecurity assessment methodology is unique in utilising model-based systems engineering techniques to develop a detailed threat model of adversary behaviour that defines explicit traceability between controls and threats and provides a visualisation and simulation of any resultant risks. We have provided preliminary results based on a face validity trial that confirms that there is significant potential for the methodology and toolkit to positively impact the effectiveness of this decision-making process. This face validity has enabled detailed implementation trials to proceed. The widespread availability of the CEMT and its face validity should enable organisations beyond the Australian microcosm to trial this framework and toolset: we welcome wider use and feedback to keep ahead of the advanced persistent threat.

Author Contributions: Conceptualisation, S.F. and K.J.; Methodology, S.F.; Software, S.F.; Investigation, S.F.; Writing—original draft preparation, S.F.; Writing—review and editing, K.J. and S.M.; Visualisation, S.F.; Supervision, K.J. and S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The models and tools presented in the study are openly available in GitHub at https://github.com/stuartfowler/CEMT/ (accessed on 25 April 2024). The raw survey data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: This research is supported by an Australian Government Research Training Program (RTP) Scholarship.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Australian Government—Department of Home Affairs, Protective Security Policy Framework. Available online: https://www.protectivesecurity.gov.au (accessed on 25 April 2024).
2. National Institute of Standards and Technology (NIST) Computer Security Resource Center (CSRC), NIST Risk Management Framework (RMF). Available online: https://csrc.nist.gov/projects/risk-management/about-rmf (accessed on 25 April 2024).
3. Australian Government—Australian Signals Directorate, Information Security Manual (ISM). Available online: https://www.cyber.gov.au/resources-business-and-government/essential-cyber-security/ism (accessed on 25 April 2024).
4. National Institute of Standards and Technology (NIST) Computer Security Resource Center (CSRC), NIST Special Publication 800-53 Rev. 5: Security and Privacy Controls for Information Systems and Organizations. Available online: https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final (accessed on 25 April 2024).
5. U.S. Department of Navy; Cyber Strategy, November 2023. Available online: https://dvidshub.net/r/irstzr (accessed on 25 April 2024).
6. Australian Government—Australian Signals Directorate, System Security Plan Annex Template (March 2024). Available online: https://www.cyber.gov.au/sites/default/files/2024-03/System%20Security%20Plan%20Annex%20Template%20(March%202024).xlsx (accessed on 25 April 2024).
7. National Institute of Standards and Technology (NIST) Computer Security Resource Center (CSRC), Control Catalog (spreadsheet). Available online: https://csrc.nist.gov/files/pubs/sp/800/53/r5/upd1/final/docs/sp800-53r5-control-catalog.xlsx (accessed on 25 April 2024).
8. National Institute of Standards and Technology (NIST), OSCAL: The Open Security Controls Assessment Language. Available online: https://pages.nist.gov/OSCAL/ (accessed on 25 April 2024).
9. MITRE ATT&CK Framework. Available online: https://attack.mitre.org/ (accessed on 25 April 2024).
10. The Department of Defense Cyber Table Top Guide, Version 2, 16 September 2021. Available online: https://www.cto.mil/wp-content/uploads/2023/06/DoD-Cyber-Table-Top-Guide-v2-2021.pdf (accessed on 25 April 2024).
11. Monroe, M.; Olinger, J. Mission-Based Risk Assessment Process for Cyber (MRAP-C). *ITEA J. Test Eval.* **2020**, *41*, 229–232.
12. Kuzio de Naray, R.; Buytendyk, A.M. *Analysis of Mission Based Cyber Risk Assessments (MBCRAs) Usage in DoD's Cyber Test and Evaluation*; Institute for Defense Analyses: Alexandria, VA, USA, 2022; IDA Publication P-33109.
13. Kordy, B.; Pièvre-Cambacédès, L.; Schweitzer, P.P. DAG-based attack and defense modeling: Don't miss the forest for the attack trees. *Comput. Sci. Rev.* **2014**, *13–14*, 1–38. [CrossRef]
14. Weiss, J.D. A system security engineering process. In Proceedings of the 14th Annual NCSC/NIST National Computer Security Conference, Washington, DC, USA, 1–4 October 1991.
15. Schneier, B. Attack trees: Modeling security threats. *Dr Dobb's J. Softw. Tools* **1999**, *12–24*, 21–29. Available online: https://www.schneier.com/academic/archives/1999/12/attack_trees.html (accessed on 25 April 2024).
16. Paul, S.; Vignon-Davillier, R. Unifying traditional risk assessment approaches with attack trees. *J. Inf. Secur. Appl.* **2014**, *19*, 165–181. [CrossRef]
17. Kordy, B.; Pouly, M.; Schweitzer, P. Probabilistic reasoning with graphical security models. *Inf. Sci.* **2016**, *342*, 111–131. [CrossRef]
18. Gribaudo, M.; Iacono, M.; Marrone, S. Exploiting Bayesian Networks for the analysis of combined Attack Trees. *Electron. Notes Theor. Comput. Sci.* **2015**, *310*, 91–111. [CrossRef]
19. Holm, H.; Korman, M.; Ekstedt, M. A Bayesian network model for likelihood estimations of acquirement of critical software vulnerabilities and exploits. *Inf. Softw. Technol.* **2015**, *58*, 304–318. [CrossRef]
20. Moskowitz, I.; Kang, M. An insecurity flow model. In Proceedings of the 1997 Workshop on New Security Paradigms, Cumbria, UK, 23–26 September 1997; pp. 61–74.
21. McDermott, J.; Fox, C. Using abuse case models for security requirements analysis. In Proceedings of the 15th Annual Computer Security Applications Conference, Phoenix, AZ, USA, 6–10 December 1999; pp. 55–64.
22. Sindre, G.; Opdahl, A.L. Eliciting security requirements with misuse cases. *Requir. Eng.* **2004**, *10*, 34–44. [CrossRef]
23. Karpati, P.; Sindre, G.; Opdahl, A.L. Visualizing cyber attacks with misuse case maps. In *Requirements Engineering: Foundation for Software Quality*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 262–275.
24. Abdulrazeg, A.; Norwawi, N.; Basir, N. Security metrics to improve misuse case model. In Proceedings of the 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensics, Kuala Lumpur, Malaysia, 26–28 June 2012.
25. Saleh, F.; El-Attar, M. A scientific evaluation of the misuse case diagrams visual syntax. *Inf. Softw. Technol.* **2015**, *66*, 73–96. [CrossRef]
26. Mai, P.; Goknil, A.; Shar, L.; Pastore, F.; Briand, L.C.; Shaame, S. Modeling Security and Privacy Requirements: A Use Case-Driven Approach. *Inf. Softw. Technol.* **2018**, *100*, 165–182. [CrossRef]
27. Matuleviaius, R. *Fundamentals of Secure System Modelling*; Springer International Publishing: Cham, Switzerland, 2017; pp. 93–115.
28. Sindre, G. Mal-activity diagrams for capturing attacks on business processes. In *Requirements Engineering: Foundation for Software Quality*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 355–366.
29. Opdahl, A.; Sindre, G. Experimental comparison of attack trees and misuse cases for security threat identification. *Inf. Softw. Technol.* **2009**, *51*, 916. [CrossRef]
30. Karpati, P.; Redda, Y.; Opdahl, A.; Sindre, G. Comparing attack trees and misuse cases in an industrial setting. *Inf. Softw. Technol.* **2014**, *56*, 294. [CrossRef]

31. Tondel, I.A.; Jensen, J.; Rostad, L. Combining Misuse Cases with Attack Trees and Security Activity Models. In Proceedings of the 2010 International Conference on Availability, Reliability and Security, Krakow, Poland, 15–18 February 2010; pp. 438–445.
32. Meland, P.H.; Tondel, I.A.; Jensen, J. Idea: Reusability of threat models—Two approaches with an experimental evaluation. In *Engineering Secure Software and Systems*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 114–122.
33. Purton, L.; Kourousis, K. Military Airworthiness Management Frameworks: A Critical Review. *Procedia Eng.* **2014**, *80*, 545–564. [CrossRef]
34. Mo, J.P.T.; Downey, K. System Design for Transitional Aircraft Support. *Int. J. Eng. Bus. Manag.* **2014**, *6*, 45–56. [CrossRef]
35. Hodge, R.J.; Craig, S.; Bradley, J.M.; Keating, C.B. Systems Engineering and Complex Systems Governance—Lessons for Better Integration. *INCOSE Int. Symp.* **2019**, *29*, 421–433. [CrossRef]
36. Simmonds, S.; Cook, S.C. Use of the Goal Structuring Notation to Argue Technical Integrity. *INCOSE Int. Symp.* **2017**, *27*, 826–841. [CrossRef]
37. United States Government Accountability Office. Weapon Systems Cybersecurity: DOD just Beginning to Grapple with Scale of Vulnerabilities. *GAO-19-129*. 2018. Available online: https://www.gao.gov/products/gao-19-128 (accessed on 15 June 2024).
38. Joiner, K.F.; Tutty, M.G. A tale of two allied defence departments: New assurance initiatives for managing increasing system complexity, interconnectedness and vulnerability. *Aust. J. Multi-Discip. Eng.* **2018**, *14*, 4–25. [CrossRef]
39. Joiner, K.F. How Australia can catch up to U.S. cyber resilience by understanding that cyber survivability test and evaluation drives defense investment. *Inf. Secur. J. A Glob. Perspect.* **2017**, *26*, 74–84. [CrossRef]
40. Thompson, M. Towards Mature ADF Information Warfare—Four Years of Growth. *Defence Connect Multi-Domain*. 2020. Available online: https://www.defenceconnect.com.au/supplements/multi-domain-2 (accessed on 15 June 2024).
41. Fowler, S.; Sweetman, C.; Ravindran, S.; Joiner, K.F.; Sitnikova, E. Developing cyber-security policies that penetrate Australian defence acquisitions. *Aust. Def. Force J.* **2017**, *102*, 17–26.
42. Australian Senate. Budget Hearings on Foreign Affairs Defence and Trade, Testimony by Vice Admiral Griggs, Major General Thompson and Minister of Defence (29 May, 2033–2035 hours). 2018. Available online: https://parlview.aph.gov.au/mediaPlayer.php?videoID=399539timestamp3:19:43 (accessed on 15 June 2024).
43. Australian Government. *ADF Cyberworthiness Governance Framework*; Australian Government: Canberra, Australia, 2020.
44. Australian Government. Defence Seaworthiness Management System Manual. 2018. Available online: https://www.defence.gov.au/sites/default/files/2021-01/SeaworthinessMgmtSystemManual.pdf (accessed on 15 June 2024).
45. Allen, M.S.; Robson, D.A.; Iliescu, D. Face Validity: A Critical but Ignored Component of Scale Construction in Psychological Assessment. *Eur. J. Psychol. Assess. Off. Organ Eur. Assoc. Psychol. Assess.* **2023**, *39*, 153–156. [CrossRef]
46. Fowler, S.; Sitnikova, E. Toward a framework for assessing the cyber-worthiness of complex mission critical systems. In Proceedings of the 2019 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia, 12–14 November 2019.
47. Fowler, S.; Joiner, K.; Sitnikova, E. Assessing cyber-worthiness of complex system capabilities using MBSE: A new rigorous engineering methodology. *IEEE Syst. J.* 2022. *submitted*. Available online: https://www.techrxiv.org/users/680765/articles/677291-assessing-cyber-worthiness-of-complex-system-capabilities-using-mbse-a-new-rigorous-engineering-methodology (accessed on 25 April 2024).
48. Cyber Evaluation and Management Toolkit (CEMT). Available online: https://github.com/stuartfowler/CEMT (accessed on 25 April 2024).
49. Fowler, S. Cyberworthiness Evaluation and Management Toolkit (CEMT): A model-based approach to cyberworthiness assessments. In Proceedings of the Systems Engineering Test & Evaluation (SETE) Conference 2022, Canberra, Australia, 12–14 September 2022.
50. National Institute of Standards and Technology (NIST) Computer Security Resource Center (CSRC), NIST Special Publication 800-160 Rev. 2: Developing Cyber-Resilient Systems: A Systems Security Engineering Approach. Available online: https://csrc.nist.gov/pubs/sp/800/160/v2/r1/final (accessed on 25 April 2024).
51. National Institute of Standards and Technology (NIST), CSF 2.0: Cybersecurity Framework. Available online: https://www.nist.gov/cyberframework (accessed on 25 April 2024).
52. Madni, A.; Purohit, S. Economic analysis of model-based systems engineering. *Systems* **2019**, *7*, 12. [CrossRef]
53. Bussemaker, J.; Boggero, L.; Nagel, B. The agile 4.0 project: MBSE to support cyber-physical collaborative aircraft development. *INCOSE Int. Symp.* **2023**, *33*, 163–182. [CrossRef]
54. Amoroso, E.G. *Fundamentals of Computer Security Technology*; Pearson College Div: Englewood Cliffs, NJ, USA, 1994.
55. INCOSE. *Systems Engineering Vision 2020*; International Council on Systems Engineering: Seattle, WA, USA, 2007.
56. Madni, A.M.; Sievers, M. Model-based systems engineering: Motivation, current status, and research opportunities. *Syst. Eng.* **2018**, *21*, 172–190. [CrossRef]
57. Huang, J.; Gheorghe, A.; Handley, H.; Pazos, P.; Pinto, A.; Kovacic, S.; Collins, A.; Keating, C.; Sousa-Poza, A.; Rabadi, G.; et al. Towards digital engineering—The advent of digital systems engineering. *Int. J. Syst. Syst. Eng.* **2020**, *10*, 234–261. [CrossRef]
58. Chelouati, M.; Boussif, A.; Beugin, J.; El Koursi, E.-M. Graphical safety assurance case using goal structuring notation (gsn)–challenges, opportunities and a framework for autonomous trains. *Reliab. Eng. Syst. Saf.* **2023**, *230*, 108–933. [CrossRef]
59. Sujan, M.; Spurgeon, P.; Cooke, M.; Weale, A.; Debenham, P.; Cross, S. The development of safety cases for healthcare services: Practical experiences, opportunities and challenges. *Reliab. Eng. Syst. Saf.* **2015**, *140*, 200–207. [CrossRef]

60. Nguyen, P.H.; Ali, S.; Yue, T. Model-based security engineering for cyber-physical systems: A systematic mapping study. *Inf. Softw. Technol.* **2017**, *83*, 116–135. [CrossRef]
61. Geismann, J.; Bodden, E. A systematic literature review of model-driven security engineering for cyber–physical systems. *J. Syst. Softw.* **2020**, *169*, 110697. [CrossRef]
62. Carter, B.; Adams, S.; Bakirtzis, G.; Sherburne, T.; Beling, P.; Horowitz, B. A preliminary design-phase security methodology for cyber–physical systems. *Systems* **2019**, *7*, 21. [CrossRef]
63. Larsen, M.H.; Muller, G.; Kokkula, S. A Conceptual Model-Based Systems Engineering Method for Creating Secure Cyber-Physical Systems. *INCOSE Int. Symp.* **2022**, *32*, 202–213. [CrossRef]
64. Japs, S.; Anacker, H.; Dumitrescu, R. SAVE: Security & safety by model-based systems engineering on the example of automotive industry. In Proceedings of the 31st CIRP Design Conference, Online, 19–21 May 2021.
65. Navas, J.; Voirin, J.; Paul, S.; Bonnet, S. Towards a model-based approach to systems and cybersecurity: Co-engineering in a product line context. *Insight (Int. Counc. Syst. Eng.)* **2020**, *23*, 39–43. [CrossRef]
66. Geismann, J.; Gerking, C.; Bodden, E. Towards ensuring security by design in cyber-physical systems engineering processes. In Proceedings of the International Conference on the Software and Systems Process, Gothenburg, Sweden, 26–27 May 2018.
67. Mažeika, D.; Butleris, R. MBSEsec: Model-based systems engineering method for creating secure systems. *Appl. Sci.* **2020**, *10*, 2574. [CrossRef]
68. Object Management Group. UAF: Unified Architecture Framework. 2022. Available online: https://www.omg.org/spec/UAF. (accessed on 15 June 2024).
69. Jurjens, J. *Secure Systems Development with UML*; Springer: Berlin/Heidelberg, Germany, 2005.
70. Apvrille, L.; Roudier, Y. Towards the model-driven engineering of secure yet safe embedded systems. *Int. Workshop Graph. Models Secur.* **2014**, *148*, 15–30. [CrossRef]