



University of
Zurich^{UZH}

Design and Implementation of an Insider Threat Modeling System Using Business Process Models

Jasmin Hochuli
Zürich, Switzerland
Student ID: 20-705-711

Supervisor: Jan von der Assen, Chao Feng
Date of Submission: January 15, 2024

Abstract

Cybersecurity ist heutzutage ein breit diskutiertes Themengebiet in der Literatur, aber auch für jedes Unternehmen von hoher praktischer Relevanz. Trotz bestem Schutz der IT-Systeme vor äusseren Bedrohungen und der Einhaltung aller Sicherheitsmassnahmen, kommt es immer wieder vor, dass Cyberattacken von dem Inneren eines Unternehmens ausgelöst werden. Dieses Phänomen wird in der Fachsprache *"Insider Threat"* genannt. Um diesen Bedrohungen entgegenzuwirken, befasst sich diese Arbeit damit, herauszufinden, welche potenziellen Ziele Insider in einem Unternehmen angreifen könnten. Da die Methoden in der Literatur meist mit hohem Aufwand verbunden sind und sich oft nur auf die technischen Systeme fokussieren, besteht diese Arbeit daraus, eine Lösung zu finden, welche den Aufwand möglichst gering hält, um herauszufinden, wo die jeweiligen Insider Threat-Ziele liegen könnten. Dabei wird versucht, auch die menschliche Einwirkung miteinzubeziehen. Um diese Ziele zu erreichen, wurde entschieden, die Geschäftsprozesse eines Unternehmens genauer zu analysieren. Denn so werden nicht nur die Schwachstellen eines Systems, sondern auch die Zugriffsrechte und Pflichten jedes Akteurs im Prozess beleuchtet.

Diese Arbeit umfasst deshalb eine Methode und einen Prototyp, welche die modellierten Geschäftsprozesse eines Unternehmens untersuchen. Dazu wurde eine Datenbank mit allen in der Literatur gefundenen Insider Threats erstellt, welche dann mit dem eingegebenen Geschäftsprozess abgeglichen werden kann, um so die potenziellen Ziele von Attacken zu lokalisieren. Anhand einer Fallstudie wird der Prototyp an einem Praxisbeispiel eines Geschäftsprozesses angewendet und die Resultate mit den Experten des betroffenen Unternehmens besprochen. Daraus hat sich gezeigt, dass der Prototyp relevante Insider Threats gefunden hat und in einem Unternehmen zur Analyse erfolgreich eingesetzt werden kann.

Nowadays, cybersecurity is a widely discussed topic in academia and is of great practical relevance for every company. Even if IT systems are best possibly protected against external threats and all security measures are adhered to, cyberattacks are often triggered from the inside of an enterprise. In the technical jargon this phenomenon is known as an *"insider threat"*. In order to take action against these threats, this thesis focuses on finding out which potential targets insiders could attack in an organization. As the methodologies from the literature mostly involve a large amount of effort and often solely focus only on technical systems, this thesis is looking for a solution that requires little effort to find out where the respective insider threat targets could be located. Additionally, it should also take human influence into account. To achieve these goals it was decided to analyze a company's business processes in more detail. This not only highlights the weak links of a system but also the access rights and privileges each actor has in the process.

Therefore, this thesis proposes a methodology and a prototype to examine the modeled business processes of a company. For this purpose, a database containing all insider threats found in academia was developed. Its contents can then be compared with the entered business process in order to localize the potential targets of attacks. The prototype was tested in a case study with a real-world business process and the results were discussed with the company's experts. With this, it was shown that the prototype was able to extract relevant insider threats and could be successfully used in a company for analytical purposes.

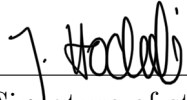
Acknowledgments

I would like to thank my supervisors for guiding me through the process of this thesis. Amongst them, I would like to especially mention Jan von der Assen who was of huge help. Furthermore, I thank IGS GmbH for being willing to share their business process with me such that I was able to evaluate a real-world example. In addition, a big thank you goes to the three experts from IGS GmbH Christian Schinnerl, Thomas Schwarz, and Marcel Nagel who took part in the case study. Without their support, this thesis would not be as insightful and practically oriented.

Declaration of Independence for Written Work

I hereby declare that I have composed this work independently and without the use of any aids other than those declared (including generative AI such as ChatGPT). I am aware that I take full responsibility for the scientific character of the submitted text myself, even if AI aids were used and declared (after written confirmation by the supervising professor). All passages taken verbatim or in sense from published or unpublished writings are identified as such. The work has not yet been submitted in the same or similar form or in excerpts as part of another examination.

Zürich, 15. 1. 2024



Signature of student

Contents

Abstract	i
Acknowledgments	iii
Declaration of Independence	v
1 Introduction	1
1.1 Motivation	2
1.2 Description of Work	2
1.3 Thesis Outline	3
2 Background	5
2.1 Cybersecurity Threats	5
2.1.1 Insider Threats	5
2.1.2 Outsider Threats	6
2.1.3 Attack Vectors	7
2.1.4 Comparison of Insider and Outsider Threats	7
2.2 Risk Management	7
2.2.1 Threat Modeling	8
2.3 Business Process Modeling	8
2.3.1 Flow Objects	9
2.3.2 Connecting Objects	11
2.3.3 Swimlanes	11
2.3.4 Artifacts	11

3	Related Work	13
3.1	Psychological View	13
3.2	Technical View	15
3.3	Limitations	18
4	Architecture	21
4.1	Procedure of Development	21
4.1.1	Insider Threat Database	21
4.1.2	Mapping Process Elements to Insider Threats	26
4.1.3	Visualizing Insider Threat in Business Processes	26
4.2	Proposed Methodology	27
4.2.1	Requirements	28
5	Implementation	31
5.1	Walkthrough of Prototype	31
5.2	Input BPMN Model	35
5.3	Prototype Implementation	37
5.3.1	BPMN.io	37
5.3.2	React	38
5.3.3	Insider Threat Database	38
5.3.4	Computation	40
5.4	Threat Report Output	42
6	Evaluation	43
6.1	Evaluation Method	43
6.2	Case Study	44
6.3	Findings	46
6.3.1	Analysis During Case Study	46
6.3.2	Case Study Results	47

<i>CONTENTS</i>	ix
6.3.3 Feedback Session	49
6.4 Discussion	50
6.4.1 Requirements	50
6.4.2 Case Study	52
6.4.3 Comparison with Related Work	55
6.5 Limitations	57
7 Summary	59
7.1 Future Work	60
Bibliography	61
Abbreviations	65
Glossary	67
List of Figures	67
List of Tables	69
A Installation Guidelines	73
B Insider Threat Database	75
C Supplementary Content	79

Chapter 1

Introduction

Nowadays, many highly advanced detection mechanisms, countermeasures, and other mitigation techniques that help to prevent the occurrence of a cyber incident exist. However, sometimes even the best technology and all its implemented security measures reach their limits because attacks originate from the inside. The human factor in cybersecurity is often the weakest link and can hardly be fully eliminated. Furthermore, the larger the organization, the higher the risk that the attacker is a member of the organization itself [1].

A recent example of this particular problem is the forgery of COVID-19 certificates. During the pandemic in 2020, QR codes were issued to the Swiss population upon the possession of a vaccination certificate or a negative test result to ensure that a person is not infected with the newly discovered virus. Even though the underlying technology was securely built and thoroughly tested, many frauds by authorized personnel were discovered. The accused employees ignored their policies and issued QR codes to people without any certificate or test result. Hence, the design of the business process allowed insiders to adapt their practices for their own benefit. This shows that not only a secure technology but also the human factor must be considered for designing a business process [2], [3].

In the technical jargon, the potential violation of a trusted person within the organization conducting an attack is called *insider threat* [4]. An *insider* can be defined as an "authorized user who has legitimate access to sensitive or confidential material" in an organization [5]. Hence, it is not essentially an employee of an enterprise itself, it can also be another stakeholder who has access to the organization's network such as consultants and other third-party users [6]. Attacks from within are way more difficult to detect and more dangerous because the attackers are trusted by the company and have detailed knowledge about the organization's security measures, counter-measures, the best time to attack, server roles, where the sensitive material is, and how to get out [4].

Insider attacks can be classified into malicious or accidental attacks [1], [5], [7]. Malicious attacks include the theft of data or willfully ignoring security guidelines, whereas examples of accidental attacks are the lack of system knowledge or being the victim of a social engineering attack [7], [8]. Furthermore, the CERT Guide to Insider Threats distinguishes three different types of insider threats: IT sabotage, theft of intellectual property, and fraud [9].

1.1 Motivation

Various sources in academia deal with the question of how to mitigate insider threats. On the one hand, there are sources that look at the problem from a psychological view [10]–[14], where the authors try to understand why a person decides to take malicious actions or what aspects in an organization lead to insider attacks. On the other hand, there are also a lot of technical approaches which analyze the underlying information systems and their processes in order to propose measures on how to prevent insider threats with the support of technology [6], [15]–[20]. The technological sources can be categorized further into risk assessment, threat modeling, and threat monitoring approaches.

As risk assessment approaches cover a whole spectrum of security assessment methods such as the analysis of potential risks, threat assessment, their impact, and sometimes even countermeasures, this thesis focuses mainly on threat assessment to model insider threats. This can be done with various methodologies. After analyzing the sources on insider threat modeling, the most promising approach appears to be the assessment of an enterprise's business processes to identify vulnerabilities that could be exploited by insiders [18]–[20]. One of the reasons for this is, that business process modeling is a widely used method to design and improve business processes. In addition, the process view includes the human factor next to the information systems. This assumption has been validated by taking into account sources that not only concentrate on insider threats but also propose a threat modeling approach for threats in general by analyzing business processes [20]–[23]. Consequently, this thesis aims to leverage this opportunity to model insider threats based on a procedural perspective [20].

1.2 Description of Work

This thesis proposes a methodology for capturing insider threats in an enterprise. More precisely, this thesis develops a threat modeling methodology and a prototype that analyzes the business process of an enterprise in order to automatically reveal potential targets of insider threats in the business process. As such, this thesis is guided by the following research question *"How to model insider threats in an automated way by analyzing an organization's business processes, which include the human factor?"*.

As business process modeling is a widely used approach to model business procedures in enterprises as described in [20], this work focuses on implementing an algorithm that automatically extracts insider threats out of a business process model. Moreover, the methodology matches potential attack targets detected in a business process to a database of possible insider threats.

Therefore, this thesis makes the following contributions. First of all, a database of potential insider threats, that were collected from the literature, sorted and aggregated to small groups, is provided. Additionally, a methodology is proposed, which is a generic procedure that could be implemented in various ways to extract insider threats from a business process. Finally, a prototype, that automatically maps insider threats from the previously

mentioned database to elements in a business process model in BPMN Notation, serves as the product of this thesis.

In order to validate the usefulness and correctness of the implemented prototype, a participatory case study is conducted. Accordingly, based on a real-world business process, the methodology and its prototype are evaluated. Security and process experts then analyze the results obtained from the test and assess the validity. The case study has shown that most of the insider threats extracted by the prototype were relevant to the example business process and expected by the experts.

1.3 Thesis Outline

This thesis report is structured as follows: Chapter 2 gives an overview of the topic and explains the most important vocabulary and concepts needed for the thesis in detail. After this, Chapter 3 reviews the sources that were found on the topic of insider threats and elaborates on where this thesis fits into academia. While Chapter 4 explains the architecture and the methodology for preventing insider threats, Chapter 5 describes precisely how the methodology was implemented and which technology was used to realize it. Next, the prototype is evaluated in Chapter 6, and in the end, a summary of the thesis is presented.

Chapter 2

Background

In cybersecurity, numerous attack types exist. This chapter gives an overview of the types of threat vectors that can occur in enterprises, how they can be distinguished from each other, and how an organization can deal with those threats. Therefore, Section 2.1 gives a high-level overview of various threats in cybersecurity. Then, risk management and threat modeling are elaborated in Section 2.2. Finally, the last section lists and explains the most important elements of the Business Process Modeling Notation (BPMN) that is used for the design and analysis of business processes.

2.1 Cybersecurity Threats

The estimation of the severity of cybersecurity risks requires an organization to be in the picture of what kind of threats it is exposed to. A threat is defined as an effort, that is made to gain access to an organization's information asset to manipulate or impair the integrity, confidentiality, or availability of its system [24]. Threats can be classified into four dimensions depending on whether they are malicious or accidental, internal or external [25]. Additionally, there can be mixed groups where external actors work together with employees within the company to execute an attack [24]. Furthermore, there are other characteristics of how a threat can be classified by distinguishing between the type of threat agent and the threat events. While a threat agent can be defined as "*an entity that initiates an attack*" a threat event is the execution of the attack itself [25]. Threat agents can, as described above, be part of the organization or not. In the following sections, a closer look is taken at how insider and outsider threats differ from each other.

2.1.1 Insider Threats

According to [5], "*insiders are authorized users who have legitimate access to sensitive or confidential material, and they may know the vulnerabilities of the deployed systems and business processes*". Hence, they are part of an organization's structure and trusted by the organization to access and decide about its assets [26]. Being part of an organization's

structure does not necessarily mean that it has to be an employee of an enterprise, it can also be the organization's consultants, contractors, or any third-party personnel, who are not included in the organization itself [6]. The actions of an insider, who misuses this access to the enterprise's assets and violates its security policy, can be called an insider threat [4], [5]. Impacts of this particular threat can for instance be loss or leakage of confidential data [15].

These insider threats don't necessarily have to stem from malicious background. Often there can also be accidental misuse [5], [8]. Reasons for accidental misuse can be stress and other psychological factors. Additionally, the fact that the insider may not have sufficient knowledge either of the system or the organization's security policies can also cause an accidental exploitation of an organization's assets. In contrast, a malicious insider attack can be motivated by monetary gain, data theft, or personal differences [1], [7].

In general, three different types of insider attacks can be distinguished according to [5], [27]: misuse of access, bypassing defenses, and access-control failure. While the first type describes the problem of misusing an organization's system resources, the second one is about insufficient defenses that fail to keep the attacker from mischief. The last case brings up the technical problem of granting access to a system to users that would not be privileged for it [5], [27].

CERT, a division of Carnegie Mellon University, describes a different taxonomy of insider threats which is composed of IT sabotage, theft of intellectual property, and fraud. Sabotage is the term that outlines when a perpetrator wants to harm an organization or an individual. If intellectual property gets stolen, the thief might benefit from a competitive advantage with the unlawfully extracted knowledge. In the case of fraud, data is being modified, deleted, or added [5].

Since an insider already has access to an enterprise's system, they can cover up their tracks with little effort, which makes it extremely hard for a detection system to register an unusual activity [4], [5]. The actor's knowledge of the system, its processes, and vulnerabilities increases the scale of the impact an attack can have [1], [4], [5]. In addition, employees are strongly interested in keeping their intrigues private because the enterprise has detailed knowledge about them, considering a legal punishment that would probably follow an insider attack [17].

2.1.2 Outsider Threats

On the other side, outsider threats describe the event of an actor from the outside of an organization who successfully compromises a network or information system. Hence, the attackers first need to find a way to overcome the security measures implemented by the enterprise to keep adversaries outside a network [26].

The detection of external attacks is under the control of an organization's forensic capabilities and is, therefore, easier to discover by security measures. Moreover, the outsiders are usually not aware of the defensive measures taken by the organization, whereas insiders know everything about the security policies [10].

2.1.3 Attack Vectors

To give the reader an overview of a selected subset of attack vectors, the following paragraph enumerates and explains the ones that appear frequently in academia.

Malware When malicious software or malware is downloaded on a device, the attacker gains access to its network and is able to operate in the system [28].

Phishing Phishing is the application of social engineering on victims to gather sensitive information [28].

Denial of Service A Denial of Service (DoS) attack is when an adversary overloads the network, which decreases the network's capacity in a way that makes it inaccessible to its authorized users [28].

SQL injection attack The amendment of a SQL query to manipulate the entries of a database system is called a SQL injection attack. The impact of this threat can be either data loss or corruption of data [28].

Man-in-the-Middle attack In the Man-in-the-Middle attack, a third party secretly extracts information out of the communication flow between two other parties [28].

2.1.4 Comparison of Insider and Outsider Threats

The following Table 2.1 compares the different threats and shows the differences in whether the attack comes from the inside or outside. Left out are the threats that occur in a similar or identical way no matter where the perpetrator comes from, such as phishing attacks, spam emails, or the intentional crash of a system [29].

2.2 Risk Management

The main goal of cybersecurity is to preserve the confidentiality, integrity, and availability (CIA) of data in an organization, which can be in danger considering the threats that were enumerated in the previous sections [28], [30]. In order to prevent the enterprise from compromises, the risks of its information systems need to be clearly understood so that an informed decision about security measures can be made. A risk in cybersecurity is the impact of a possible exploitation of a vulnerability of a system, such as accessing confidential data, introducing malicious commands, or attacking an organization [30]. During the security risk management process, various risks are identified, assessed, classified, and evaluated [30]. Four factors are considered to assess the risks: *"hazards, assets, threats, and vulnerabilities* [15]. Since the enterprise is constantly evolving and in an unsteady environment, risk assessment is a task that is recurring and includes continuous monitoring and controlling of events to reevaluate potential threats and vulnerabilities [15].

Table 2.1: Comparison of Insider and Outsider Threats [29]

Attribute	Outsiders	Insiders
Authentication	Penetration, attacks on public key or authentication infrastructures, war dialing	Misuse of intended authority by over-authorized users, usurpation of superuser access and root keys
Authorization	Unprivileged exploitation of inadequate controls	Privileged manipulation of access controls
Confidentiality	Unencrypted password capture or compromise of encrypted passwords	National security leaks and other disclosures; access to crypto keys(!)
Integrity	Creating Trojan horses in untrusted components, Word macro viruses, untrustworthy Web code, in-the-middle attacks	Inserting Trojan horses or trapdoors in trusted (and untrusted) components; altering configurations, schedules, and priorities
Denials of Service (DoS)	External net attacks, flooding, physical harm to exposed equipment	Disabling of protected components, exhaustion of protected resources
Accountability	Masquerading, DoS attacks on accounting infrastructures	Hacking beneath the audit trails, altering audit logs, compromising misuse detection
Other misuses	Planting pirated software on the web	Running a covert business, insider trading, resource theft

2.2.1 Threat Modeling

Threat modeling is a part of many risk assessment approaches [6], [15]. According to [31], threat modeling is *"the process of identifying and analyzing the security threat to an information system, application or network"*. In addition, security vulnerabilities and risks that can occur in an organization's network or technical architecture can be modeled. When specific threats are extracted, a targeted mitigation technique and countermeasures against the corresponding threats can be evaluated [25]. There are two threat modeling approaches. One is graphical where diagrams, graphs, or tables are used to model threats. The other one is a formal approach that uses Mathematics and Stochastics [25].

2.3 Business Process Modeling

Whereas *"a business process is the combination of a set of activities within an enterprise with a structure describing their logical order and dependence whose objective is to produce a desired result"*, business process modeling is a technique to design, understand and communicate the enterprise's business processes [32]. There are numerous modeling techniques for keeping track of an enterprise's business processes [32]. However, there is a graphical notation called Business Process Modeling Notation (BPMN) which has become the standard modeling technique used by most organizations in the industry and in academia [20], [33], [34].

The modeling technique in BPMN has four different categories: *"Flow Objects, Connecting Objects, Swimlanes, and Artifacts"* [34], [35]. Each category is briefly explained in the following subsections.

2.3.1 Flow Objects

Flow Objects are significant elements of BPMN for this thesis and therefore a quick description is given for each type. First, it is distinguished between Activities, Events, and Gateways. Then, each of these is narrowed down further.

In the following Figure 2.1, the most important notation elements of Flow Objects in BPMN are portrayed.

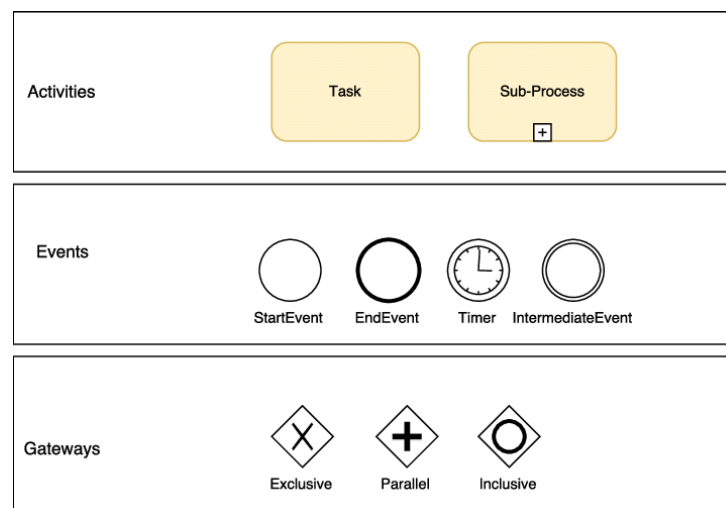


Figure 2.1: Elements of BPMN [33]

Activities

Activity An Activity can either be a process, a subprocess, or a task, which describes the work done by the members of an organization [35].

Process A process is a set of consecutive activities that are triggered by a certain event and stop in an end state [35], [36].

Subprocess A part of a process is called a subprocess, that is not modeled in detail on the main process [35], [36].

Task A task is a discrete workflow that has a start and an end point [35]. Various subtypes of tasks specify how the task is conducted [35], [36].

User Task A user of an application software is assigned a task that is fulfilled by the support of the application [36].

Send Task A send task is responsible for sending a message [35], [36].

Receive Task A message is received in this specific task [35], [36].

Script Task The modeler of the BPMN writes a script that is read by the process engine and then carried out automatically [36].

Service Task The task is fulfilled automatically by software [36].

Manual Task The manual task is not supported by any software and therefore fulfilled by a human being [36].

Business Rule Task Business rules are meant to specify the actions that should be taken under given circumstances. Usually, they are not described in the process model itself [36].

Figure 2.2 shows the graphical representation of each task type that exists in BPMN.

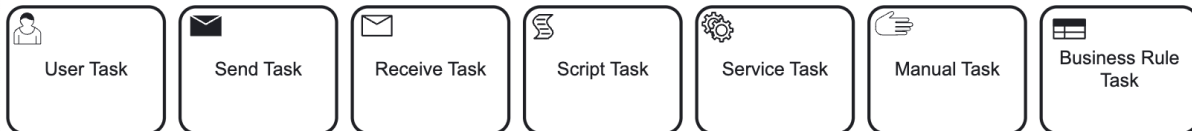


Figure 2.2: Overview of Different Types of Tasks in BPMN

Events

A process flow always begins and ends with an event [36]. The various types are listed in Figure 2.3 below. For this thesis, the Message Events are the most important ones.



Figure 2.3: Events in BPMN [37]

Gateways

The controls in a business process are called Gateways. They are used for *branching, forking, merging, or joining of paths within the process* [35]. The three major ones are mentioned in Figure 2.1 above.

2.3.2 Connecting Objects

Activities have a specific sequence that is portrayed using connecting objects. Sequence Flows are arrows that link the activities in the specified order, whereas Message Flows show the path a message follows to another task or event. Message Flows can only link to an element outside its own pool [36].

2.3.3 Swimlanes

Swimlanes are a graphical tool to distinguish different actors in a process. All activities that are embedded in one lane are conducted by the same actor. Different actors of the same organization can then be combined in a pool [35].

2.3.4 Artifacts

Artifacts are elements in BPMN that do not have a direct influence on the business process itself. They are modeled as *"data objects, data stores, groups, and annotations"* [35]. They are used to visualize the information that is part of the process. Figure 2.4 provides the visual representations of the artifact types [35].

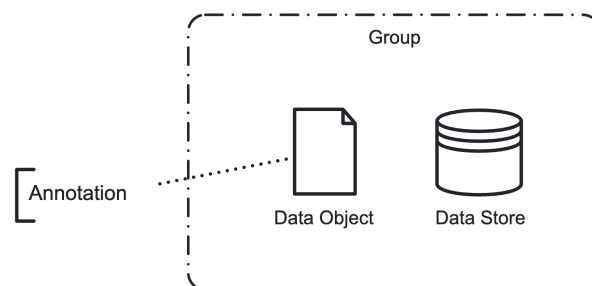


Figure 2.4: Artifacts in BPMN

Chapter 3

Related Work

The qualitative research of this paper was conducted with the help of the search engines Google Scholar and Swisscovery in August 2023. The main research term was "*insider threat modeling*" and was then expanded to similar terms and related topics. At the end of the literature review, 15 papers were selected for closer consideration. Further on, the sources were analyzed according to their methods and results. Consequently, an overview was established containing the information of the publishing year, the view of the authors, what dimension of the organization they looked at, what methodologies they proposed, and if they made use of a prototype or automated tool that supported the methodology. The following paragraphs briefly summarize the most important aspects of the sources. Additionally, a short insight is given concerning what threat modeling tools are used in the industry. At the end of this chapter, there is a table that presents the overview described earlier and shows how this thesis can be put into context regarding the topic of insider attacks.

3.1 Psychological View

Since the concept of *insider threats* is a human-related topic, a lot of papers in academia can be found that shed light on insider threats from a social perspective. Therefore, they also look at models which are based on psychological or sociological theories.

A theory that often appears with insider threats is Game Theory. [10] takes up this topic and describes an adversarial risk analysis approach that takes into account the organization's culture, its already implemented defensive measures, and whether the attack gets detected or not. The authors modeled the insider threat problem in a two-player conflict situation where both have incomplete information. Then, a defend-attack-defend model describes the decisions each player makes and what influence they have on the counterparty. The authors show that applying their model to an organization helps its risk management to refine an optimal defensive strategy to prevent insider attacks. The model is not supported by a prototype, as it is a theoretical model [10].

[11] goes a step further and not only looks at game theoretic models but expands the work to five modeling and simulation approaches that are part of the computational and mathematical organization theory. These approaches include Bayesian belief networks, agent-based modeling, system dynamics, and network analysis. At a meeting that took place in the summer of 2014, experts discussed how those models can explain the actions and interactions of individuals with their context regarding the special issue of insider threats. The finding of the workshop was that all the above-mentioned approaches are useful to apply in organizations and are best when several models are combined to cover the whole spectrum of insider threats. Even though they propose a combination of the different methods, they did not create a tool that would integrate them [11].

While the previously described papers focus on explaining why insider attacks happen, [12] is mainly about the impact of the threat and especially investigates unintentional attacks. To be more specific, they have a closer look at what effects *"user vulnerabilities and user leakage due to user interaction"* have on the information system of an organization [12]. After applying their model called SecureInT (Securing Insider Threats) to understand the system state, the look-ahead analysis outputs potential attacks to determine which cybersecurity risks an organization faces. With this, the authors showed that the probabilistic modeling of user vulnerability and leakage helps understand under which conditions the risks of cyber threats increase. The study only proposes a methodology to identify possible risks. Hence, no prototype is explained that would support the model in an automatic way [12].

[13] conducted a thorough accident analysis to propose effective insider attack prevention. Therefore, they evaluated 43 insider attacks that took place in China, the United States, and Israel between 2009 and 2021 by applying analysis techniques such as a hybrid model of fuzzy set theory, Bayesian networks, and improved Human Factors Analysis and Classification System (IHFACS). To visualize the biggest risks, they simulated the dependencies amongst the factors in the Bayesian network in a program called GeNIe 2.3. Employing this hybrid approach, they aimed to discover which human factors bring about a successful insider attack in enterprises. The outcome of the study was that *"deficiencies in resource management, poor organization climate, technical detection vulnerabilities, and bad personal factors"* were the main reasons identified [13].

The higher order logic (HOL) proof assistant Isabelle/HOL is introduced by [14] to give a social explanation based on the theory of the sociologist Max Weber. They spotlight the insider threat type theft of intellectual property and model the corresponding human behavior accordingly. Furthermore, they provide a *"mechanized logical framework for insider threat analysis"* [14]. Thus, they validated that human behavior can be modeled by applying Max Weber's three steps of social explanation to explain insider threats. In addition, they showed that violations of global policies can be detected by implementing an extension of the Isabelle/HOL proof assistant, which is an automated tool that supports proofs [14].

3.2 Technical View

So far, the described related works predominantly focus on how to explain insider threats by modeling the human interactions in an organization. However, some authors take a more technical view and develop a strategy such that insider attacks can be prevented on the information security side.

One of the technical papers describes how risk assessment helps to identify, assess, and prioritize the potential risks an organization's information system is exposed to so that negative events can be circumvented. NIST SP 800-30 (National Institute of Standards and Technology Special Publication 800-30), FRAP (Facilitated Risk Assessment Process), OCTAVE (Operational Critical, Threat, Asset and Vulnerability Evaluation), and CRAMM (The Central Risk Analysis and Management Method) are the risk assessment methods discussed in further detail to lower the possibility of an insider threat. The authors claim that NIST SP 800-30 is the most suitable risk assessment procedure as it is well-documented, more dynamic, and applicable to either quantitative or qualitative research. As the paper is a review that only compares and analyzes the different risk assessment methods, there is no automation or tool-support provided [15].

[16] elaborates a model-based methodology for insider threat assessment which is compliant with the NIST risk assessment procedure that is also described in [15]. Their adapted procedure contains six steps, which include investigating the system under analysis, identifying potential insiders, determining the corresponding insider threats, finding attack paths that could lead to insider threats, establishing a countermeasures' selection, and lastly conducting iterations and updates to not lose track of any changes. For the step of finding a possible attack path, they take ADVISE (ADversary VIEW Security Evaluation) as a tool example which takes into account time, costs, and success probability of the attack steps. With this methodology, the authors demonstrate how a systematic investigation combined with quantitative analysis can contribute to a successful insider threat prevention strategy [16].

By going beyond a risk assessment approach, [6] provides a complete strategic planning process that should help risk managers reduce insider threats along with the enterprise's business objectives. Therefore, they explain how to assess the threats and risks of an organization by applying various tools such as a tailored risk integration process (TRIP), a threat assessment matrix, an information security scorecard, and multiple others. Unlike the NIST approach described in [15] or [16], the TRIP methodology takes into account the whole business process dimension rather than a single information system. From the critical process starting point, they then define the accompanying critical information systems and applications that should be analyzed [6].

The authors from [17] look at insider threats as an NP-hard (non-deterministic polynomial-time hard) problem that can be explained with graphs that depict the different paths an attacker needs to follow to compromise a system. With the so-called key challenge graph, they aim at understanding the global perspective of insider threats. Various algorithms such as brute force and greedy heuristic are investigated to find the path with the lowest cost to compromise the information system. Cost here means the effort an attacker has to make to overcome a security measure the organization has implemented. They concluded

that the greedy heuristic algorithm was efficient for small graphs whereas the brute force algorithm became very large even for a small number of nodes. The authors suggest developing an automated tool that would apply the described methodology and algorithms for future work [17].

The last four examples showed approaches that model insider threats and assess risks from a technical view to understand them better and define the corresponding countermeasures. Most of the models, however, use a methodology that affords a lot of time and effort which are often not voluntarily spent to a large extent by organizations. Therefore, an automated methodology seems to be more desirable. Additionally, except for [6], the models used in the technological papers described so far focus on the dimension of an information system in an enterprise. However, as the human aspect should also be considered, it is more appropriate to investigate the dimension of business processes. The following two papers were found that combine those two requirements.

[18] analyzes an enterprise's business processes to identify insider threats. The main focus is on sabotage and data exfiltration attacks which are analyzed by applying two different strategies. The first approach is a Fault Tree Analysis which takes a hazard as a starting point and then identifies which combinations of events have to occur to cause the hazard. The analysis is conducted with an automated tool that was developed by the authors in a previous work. The second way is Finite-State Verification which checks the possibility of artifacts having been corrupted by an insider. With these strategies, the researchers managed to find a solution on how to automatically identify vulnerabilities in an enterprise's business processes and how such attacks can be prevented. Nonetheless, as they only focus on the two insider attack types described above, they do not cover all insider attack vectors existent on the business process level [18].

[19] proposes a monitoring technique to secure an organization's business processes in an even more automated way than [18]. They analyze the real-time activities and performance of the business process and can therefore immediately detect any deviations or suspicious behavior. In addition, they add the human factor to their methodology by monitoring their social media behavior. The architecture consists of three modules: on-line monitoring, business process monitoring, and threat management. Despite the ethical and legal issues this approach raises, it shows that taking the business process perspective to address insider threats is a suitable approach because the logs can be assigned to specific users and therefore user behavior can be analyzed. However, the paper deals more with detection rather than with modeling and prevention. It would be beneficial to analyze critical business processes and predict risks and threats in advance to prevent any occurrences of insider threats [19].

The sources mentioned until now restricted their work on insider threats. To find an appropriate method for modeling insider threats by using business processes, the author of this thesis started to also consider sources that included other types of threat agents. Based on previous studies that argued that it is useful to identify insider threats by analyzing business processes, the literature review focused on analyzing business processes to model threats [18], [19].

[20] proposes a framework for eliciting security requirements of a business process considering threats that can originate from insider and outsider attackers. They came up with

a methodology on how to map the potential threats in an enterprise's business processes as [18]. It confirms that locating vulnerabilities in a business process model is a valid methodology to model threats in general. The methodology which is used for eliciting the security requirements is Software Quality Requirements Engineering (SQUARE). Therefore, an iterative process, where they sequentially analyze the business process, is applied. However, no automated prototype that would support the framework is mentioned as they propose to carry out workshops [20].

In contrast to [20], [21] checks the compliance of the previously specified security requirements rather than deriving them from the vulnerabilities in the business process model. This verification framework includes the SecBPMN modeling language (SecBPMN-ml), SecBPMN query language (SecBPMN-Q), and a query engine to compare whether the SecBPMN-Q policies are aligned with the SecBPMN-ml specifications. [21] determines for each BPMN element which security principle it can be linked with and provides a graphical annotation for the security principles. Additionally, they implemented their framework in an automated tool and clearly specified in a methodology which steps and experts are needed to apply their framework. To verify their approach, an empirical study, a scalability analysis, and a large case study were conducted. These evaluation methods have demonstrated the usefulness and applicability for enterprises when using the SecBPMN framework [21].

The authors from [22] model threats in an early stage of the development process in an automated way. Additionally, next to simply modeling the threats, the functionality to interpret the found threats and assess the corresponding risks is integrated. The prototype is designed for business specialists, so that no security expert is needed to find cybersecurity threats on the business process level. The business specialist can annotate the BPMN elements with information that is non-technical and not security-related. These annotations are then automatically processed and reconciled with the threats from the ENISA Threat Landscape. Finally, the OWASP Risk Rating Methodology is applied to interpret the risk for each threat. The approach the paper uses has shown that the automated exfiltration of threats out of a business process is time-efficient and less costly than hiring a security expert. However, the paper has optimized its prototype on eGovernment processes which lacks the variety of business processes in general [22].

[23] follows a similar approach as [22], as they created a prototype that automatically excerpts threats by analyzing business processes that are modeled with BPMN. Furthermore, this methodology is designed for non-security experts as well. In contrast, the two papers can be differentiated by their output. [23] maps the business process elements to vulnerabilities from known databases to produce an attack graph. With this, they aim to avoid the modification of the business process model itself but create a new artifact as output. The procedure that creates this graph representation is conducted by using Meta Attack Language which is based on Domain-Specific Language called coreLang. This language automatically maps the elements of BPMN to the list of known vulnerabilities that was created by NIST. By applying their prototype to a real-life example, they showed that it is possible to automatically analyze a business process in BPMN to provide a non-invasive simulation of cybersecurity threats [23].

Apart from the methodologies and prototypes that are proposed in academia, various tools are used in the industry that automate threat modeling. For this reason, the overlapping threat modeling tools excerpted from the lists of [38] and [39] were inspected on their feasibility for this thesis' goals. Nevertheless, no solution was found that specifically addresses insider threats. The industry mainly relies on graphical threat modeling tools that automatically detect threats in diagrams created by users as part of the application. As this automation is usually done by the tool provider, it is hard to overwrite this logic to align it with insider threats. In addition, each threat modeling tool has its individually defined threat modeling elements and therefore gives a limited space for modeling an organization's business processes. Hence, even though it would be possible, there is a large overhead for understanding the tool's graphical elements such that the business process could be modeled [40]–[44].

3.3 Limitations

The previous paragraphs summarized the main sources that can be found in academia considering the modeling of insider threats and their related topics. This section aims to give the reader an overview of the sources and points out which literature gap this thesis fits in. Table 3.1 gives an overview of the papers discussed above and shows the different categorizations in terms of the publication year, view on the topic, dimension in the enterprise, what methodology is used to analyze the threats, whether there is (or what kind of) a prototype proposed by the authors, and whether the focus only lies on insider threats or threats in general. The last row presents the work that is being developed in this thesis.

To summarize, as shown in Table 3.1, many sources attempt to explain insider threats from a human-centered view and make use of models and methodologies that consider human interactions in an organization. However, some authors look at insider threats from a purely technical view. These technical sources can be categorized further in risk assessment, threat modeling, and threat monitoring. In addition, there are differences in the way, what dimensions of risks and threats are assessed. The psychological sources mainly spotlight the whole organization, while authors from the technological view either look at threats on the information system or process level. To determine the best way of modeling insider threats, also sources that included outsider threats were considered. Additionally, the industry was reviewed on threat modeling tools. However, no solution was found that focuses on insider threats.

Albeit [18] and [19] combine automated insider threat modeling with the dimension of business processes, there is still no approach that automatically produces a threat assessment of all possible insider threats by analyzing an enterprise's business processes. As shown by sources that do not only focus on insider threats, analyzing business processes is a legitimate method [20], [22], [23]. Therefore, this work focuses on implementing an automated methodology to extract insider threats directly out of a business process model and define the potential attack targets in the business process.

Table 3.1: Overview of the Related Work

Source	View	Dimension	Methodology	Prototype	Insider only
[10] 2021	Psychological	Organization	adversarial risk analysis approach	no	yes
[11] 2016	Psychological	Organization	agent-based modeling, game theory, system dynamics, Bayesian belief network, network analysis	no	yes
[12] 2017	Psychological	Information System	SecureInT, Look-ahead analysis	no	yes
[13] 2023	Psychological	Organization	Fuzzy Set Theory, Bayesian networks, IHFACS	GeNIe 2.3 Academic program	yes
[14] 2017	Sociological	Organization	three steps of social explanation	Isabelle/HOL	yes
[15] 2018	Risk Assessment	Information System	NIST, FRAP, OCTAVE, CRAMM	no	yes
[16] 2014	Risk Assessment	Information System	NIST	ADVISE	yes
[6] 2008	Risk Assessment	Process	TRIP	no	yes
[17] 2005	Threat Modeling	Information System	key challenge graph	no	yes
[18] 2014	Threat Modeling	Process	Fault Tree Analysis, Finite State Verification	Automated fault tree analysis tool	yes
[19] 2014	Threat Monitoring	Process	analyze business process logs, social media behavior	social media & process monitoring tools	yes
[20] 2020	Threat Modeling	Process	SQUARE	no	no
[21] 2017	Threat Modeling	Process	SecBPMN-ml, SecBPMN-Q	query engine	no
[22] 2023	Threat Modeling	Process	ENISA, OWASP	BPMN modeler with annotations	no
[23] 2021	Threat Modeling	Process	NIST known vulnerabilities list	coreLang	no
This work 2024	Threat Modeling	Process	insider threat mapping to business process elements	BPMN.io modeler	yes

Chapter 4

Architecture

This chapter presents a high-level overview of what the contribution of this thesis is. To provide the reader with an understanding of the process involved in developing the products of this thesis, the procedure is described first, followed by an explanation of the structure and key aspects of the proposed methodology.

4.1 Procedure of Development

The research question for this qualitative thesis is *"How to model insider threats in an automated way by analyzing an organization's business processes, which include the human factor?"*. To answer this question, the author came up with a qualitative methodology to connect insider threats with the elements of business process models.

Several steps were needed in the process of finding a feasible solution. To begin with, all the potential attack vectors originating from insiders had to be extracted from the literature. Next, these insider threats were sorted and then connected to specific elements of the business process. Lastly, the potential insider threat targets were visually represented in the business process diagram.

4.1.1 Insider Threat Database

First of all, multiple sources were being studied in order to elicit potential insider threats. At the end of the research phase, 99 insider threats were identified from five different sources [6], [16], [29], [45], [46]. Figure 4.1 shows the number of threats that were retrieved per source. As evident in the table, the most insider threats were extracted from [6].

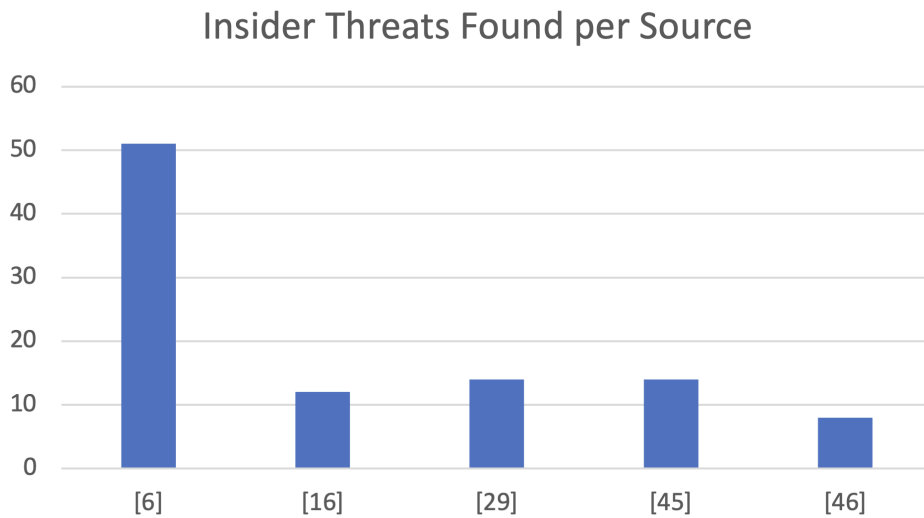


Figure 4.1: Insider Threats Found in the Literature per Source

After having collected the threats, they were subjected to a more detailed analysis. This included ordering the threats and classifying them to find duplicates and similar threats from different sources. As some sources like [29] already distinguished the different threat vectors by their security principle, the best way at this point appeared to order the extracted threats in the same way.

For this, it was decided to choose the security principles *Confidentiality*, *Integrity*, *Availability*, *Accountability*, and *Authenticity* as classification subjects. Whereas the three fundamental principles *Confidentiality*, *Integrity*, and *Availability* are the major principles mentioned in virtually every security-related source, [6] suggested also taking *Accountability* and *Assurance* as a security objective for insider threats. As it has become more relevant over the years to log user activity, *Accountability* has become a fourth element next to the CIA triad according to [47]. Because of this reason, it has been added to the classification subjects. During the analysis of the different insider threats, it was discovered that *Authenticity* plays a key role as well, which is also listed by [47], [48]. *Assurance* was removed from the list of security principles to avoid duplicate threats for different objectives [6], [21].

The list of security principles including a definition for each principle is presented here. After the list, Figure 4.2 shows the number of insider threats that were assigned to each security principle.

Confidentiality of Data or Systems is the protection of *"information from threats or hazards"* [6].

Integrity of Data or Systems *"relates to processes, policies, and controls"* which are subject to not being altered to keep *"data accuracy, completeness and reliability"* [6].

Availability is the *"access to information and systems for legitimate users"* [6].

Accountability is a security principle responsible for *“tracing actions to their source”*. It *“supports nonrepudiation, deterrence, intrusion detection and prevention, after-action recovery, and legal admissibility of records”* [6].

Authenticity is *“the property of being genuine and being able to be verified and trusted; confidence in the validity of a transmission, a message, or message originator. This means verifying that users are who they say they are and that each input arriving at the system came from a trusted source”* [48].

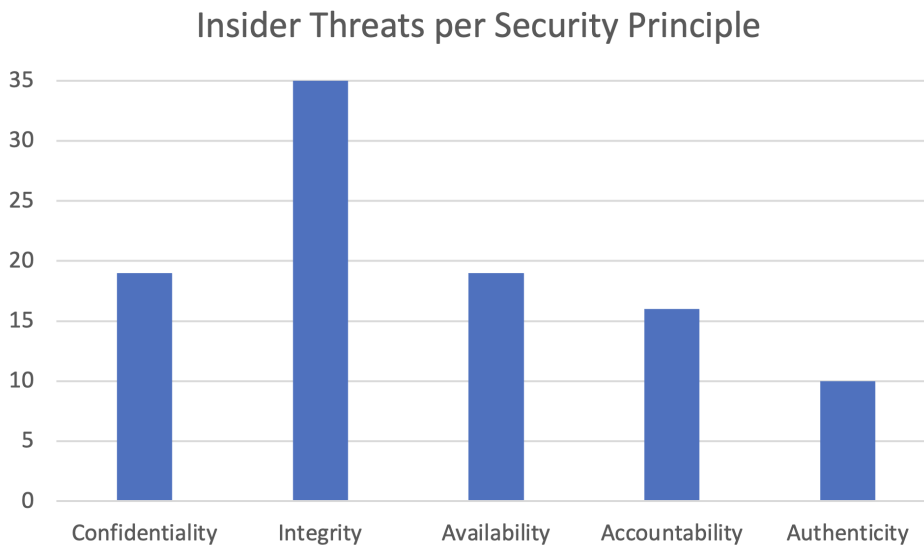


Figure 4.2: Threats Found per Security Principle

As nearly a hundred insider threats were retrieved from the sources and some of the threats were similar or even duplicates within the security principle, the threats were further classified into smaller groups. The clustering of the threats made sense because, at the business process level, the underlying information system and its detailed infrastructure are not modeled. Hence, it is redundant to specify which system vulnerability could be compromised to conduct a specific insider attack. In that sense, the selection takes a pragmatic stance on the selection of the subset of threats.

For instance, cookie tampering is an attack vector that was found in the literature. This attack can be used to manipulate information stored in a web browser. But as the business process most likely does not specify whether the used system runs locally or in a web browser, the attack vector cookie tampering can be added to the cluster of data corruption. Therefore, it is sufficient to know the types of insider threats that can occur, without needing to know the details of what exactly is being compromised.

In the following Table 4.1, each group of insider threats is listed according to the corresponding security principle and briefly explained. The complete table of all listed insider threats categorized into security principles and subgroups can be found in Table B.1 in the appendix.

Table 4.1: Insider Threat Database

Security principle	Insider threat group	Description
Confidentiality	Confidential data acquisition	Data that is being stolen or used inappropriately fits in this category. The attack can target not only a computer system but also a web service when for instance a session is being hijacked [6], [45].
	Confidential data view	If sensitive data is being inspected apart from the normal usage, the attack belongs to confidential data view [6], [16].
	Confidential data transfer	The illegal distribution of confidential files such as password lists, financial information, and other sensitive material is a part of confidential data transfer [6], [16], [29], [45].
	Unauthorized access to credentials	Attacks in this category happen when an insider gets access to crypto keys and other credentials without authorization [6], [16], [29].
Integrity	Data corruption	With data corruption, the fraudulent modification of data can be understood. It happens when information is manipulated within either an application or also a system. Incidents in the past have shown that tampering with cookies is a widely used technique to corrupt data in an unauthorized manner [6], [16], [45].
	Malicious code modification	In software code programming small modifications can have a huge impact. Logic bombs, Trojan horses, and other malicious code injections are examples of this attack group [6], [16], [29].
	Malware installation	The installation of malware can originate from various sources. The use or download of illegal software or offensive material has a higher chance of containing Trojan horses or trapdoors in order to compromise a computer system [6], [16], [45], [46].
	System control manipulation	When default configurations are being modified or the protection of components gets disabled, attackers manipulate system controls [6], [16], [29], [45].

Security principle	Insider threat group	Description
Availability	Hardware attack	All attacks that include hardware are aggregated in this group. Especially when hardware is defective it can get vulnerable to insider attacks. However, an insider is also capable of adding or removing components of hardware to harm a computer system [6], [16], [29], [45].
	Resource exhaustion attack	In resource exhaustion attacks, the availability of the system is being compromised. Examples that belong to this category are DoS, buffer overflow, and replay attacks [6], [29], [45], [46].
	Network exhaustion attack	Unlike resource exhaustion attacks, not the system but the network is not available because of an overload. This can happen when a large amount of data is being downloaded in a small time frame such that the network is not able to process other packets [6], [45].
	Data deletion	The loss of data because of its destruction by an insider is labeled as data deletion [6].
Accountability	System control circumvention	There are various ways in which system controls can be circumvented. In the sources, the altering or disabling of audit logs has been mentioned most frequently [6], [16], [29].
	Unauthorized privilege elevation	In case of the modification of user access rights, privileges in a system can be elevated. This gives the user the capability to get unauthorized access to information or systems [6], [16].
	Misuse of privileges	Even if users are allowed to access certain data or systems, they can still misuse their privileges to attack an organization. They could for instance abuse an adjustment transaction or error-correction procedures to hide their intrigues [6], [29].

Security principle	Insider threat group	Description
Authenticity	Social engineering attack	Attack vectors in social engineering that were found in the context of insider threats are tailgating, ingratiation, phishing, pretexting, and baiting. These techniques are applied to deceive an employee in order to gain unauthorized access [46].
	Impersonation attack	Masquerading as an employee of an enterprise is a typical impersonation attack in insider threats [6], [46].
	Man-in-the-middle attack	When attackers place themselves in between a client and a server and intercept all the messages that are sent between those two parties, they are called a man in the middle [6].

4.1.2 Mapping Process Elements to Insider Threats

Based on the issued list of insider threats, the business process has to be analyzed in detail and each element needs to be checked for potential threats. This creates a mapping, where for every business process element the potential insider threats are listed. There are various ways how this mapping could be conducted.

[21] established a mapping of several security principles to the high-level categories of BPMN elements: activities, data objects, and message flows. In their paper, the goal was to design secure business processes. They also categorized the threats according to the security principles as elaborated in the last section of this thesis and then mapped them on the BPMN elements described before. The only things left out were that *Non-repudiation* and *Privacy* were not selected as major security principles in this thesis. Furthermore, it was decided that message flows can also be exploited to harm *Authenticity* such as in social engineering attacks. The definition of [48] for *Authenticity* in the last section even mentions messages.

Further details of this procedure can be found in Chapter 5, where the technical details of the mapping are mentioned, and each element in the business process is mapped to a specific insider threat group.

4.1.3 Visualizing Insider Threat in Business Processes

As described in Chapter 3, [20] showed that it is a valid approach to derive where potential threats could occur from the business processes. Especially tasks where human actors access data or authenticate themselves are examples where the methodology needs to

have a closer look and validate whether there are potential vulnerabilities for an insider attack.

As business process modeling is a graphical approach to model an enterprise's business processes, this thesis analyzed various graphical threat modeling approaches to select the best solution for visualizing the potential insider threats in the business process.

4.2 Proposed Methodology

After elaborating on how the author of this thesis came up with the proposed methodology, this section describes on a high level how the methodology works and how it could be realized. First, a broad overview of the structure and the elements that are needed to implement the methodology are given. Then, the requirements for the implementation of the prototype are explained.

Step 1: Define security requirements

First of all, the user decides which security requirements are the most important ones in the selected process to achieve a result that is close to the enterprise's business objectives. The options are the five security principles that were discussed before: *Confidentiality, Integrity, Availability, Accountability, and Authenticity*.

Step 2: Input business process model

Next, the user inputs a business process model. This model can have various forms, but it needs information on the flow of consecutive activities, the actors that are part of the process, and a visualization of the data that is needed in the process. This can be a visual representation or in the form of a .xml file. Once the business process model is ready, it can be loaded into the system.

Step 3: Analyze business process

Then, the prototype automatically analyzes the elements in the business process and outputs a list of potential insider threats per element that was found in the business process. These insider threats are aligned with the security requirements that were inserted in the first step.

Step 4: Investigate list of threats

To continue, the user can investigate the set of threats that was returned and put it into the organization's context. This is important, as the automated version is not aware of the underlying systems or any controls that are already configured. In the best case, the prototype presents a visual form showing the location of potential insider threats in the process, so that the user can decide whether the found threats for a certain element in the process are relevant.

Step 5: Select important threats

After the analysis, the user decides which threats are important at what stage of the process. This involves selecting the process elements that are a potential target for a specific threat.

Step 6: Visualize important threats

Finally, the system visualizes the threats selected by the user in the business process model. Furthermore, a list of all potential threats that were extracted from the literature and the corresponding descriptions of each threat are given in a report. Also, depending on how many times a threat was identified as important in Step 5, a sorted list of process elements to show the user where to invest the most to secure the part of the process is given.

Next steps

Until Step 6, the steps were supported by the prototype. For the remaining steps, the business analyst would need to work with a security expert to find out whether the found threats are already mitigated by any controls or countermeasures and where they need to be improved. This is not supported by the proposed methodology as only the business process itself is taken as a source.

Figure 4.3 visualizes the steps above in a BPMN diagram to make the reader more familiar with the use of modeling business processes with BPMN. In addition, the prototype's supported and unsupported features are shown.

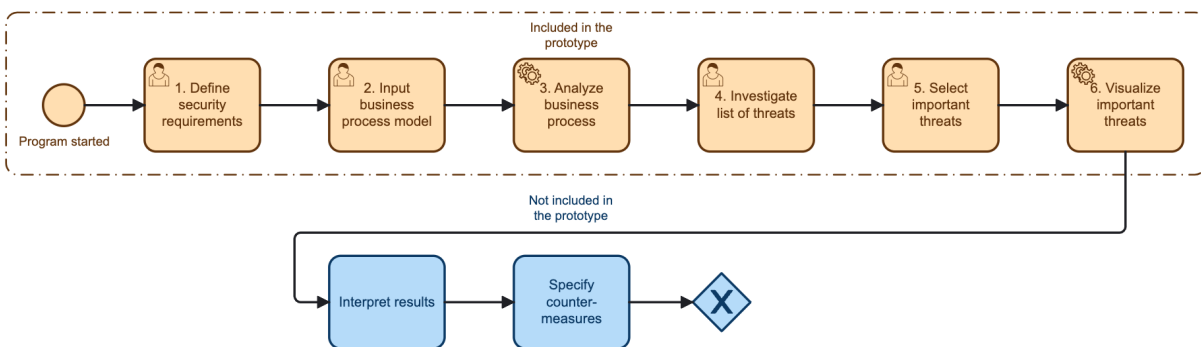


Figure 4.3: Visualization of Methodology

4.2.1 Requirements

Based on the previous descriptions of what the prototype shall fulfill, there are various requirements that the prototype needs to meet. As user stories are a widely used approach to keep track of the requirements in particular for agile development, the following paragraph shows a list of requirements in the form of user stories [49].

As a user, I want to...

1. ...automatically receive a list of insider threats, so that I can evaluate suitable mitigation techniques. (functional)
2. ...input my business process as is, so that I don't have to hire somebody to look at the business process from a security perspective. (business)
3. ...receive feedback about the underlying IT systems supporting my business process, so that I know which IT systems need further analysis. (functional)
4. ...see a visualization of potential targets in my business process, so that I get an overview of the critical elements in my business process. (functional)
5. ...receive feedback on which critical elements in my business process relate to which insider threats so that I can evaluate where to introduce mitigation techniques. (functional)
6. ...receive an ordered list of insider threats, which is ordered by the severity of the risk, so that I can prioritize where to invest in finding mitigation techniques. (functional)
7. ...receive feedback on both intentional and unintentional insider threats, so that I can also understand where mistakes can happen. (functional)
8. ...be able to use the prototype for various business processes in many industries, so that if something changes in my business process, I can still use the prototype. (functional)

Chapter 5

Implementation

This chapter discusses the different elements of the prototype. It sheds light on the way the methodology was implemented and what reasons led to the decisions made during the process. Section 5.1 gives an overview of the prototype and presents a short walkthrough by reflecting the same steps as described in Section 4.2 but this time with more technical details. The following sections are structured in the way the prototype is executed guided by its input, computation, and output. Thus, Section 5.2 elaborates on the modeling language of the business process model that was used as input for the prototype. Then, the third section explains the logic of the prototype itself, especially how the elements of the BPMN model were mapped to the database of insider threats that was introduced in Chapter 4. Finally, Section 5.4 explains how the prototype presents the findings on insider threats in the different parts of the business process.

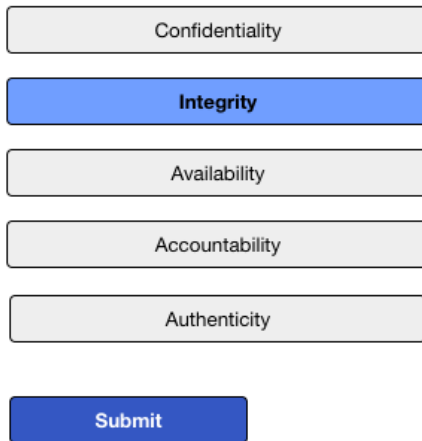
5.1 Walkthrough of Prototype

Step 1: Define security requirements

When the prototype is started, the list of the five security principles: Confidentiality, Integrity, Availability, Accountability, and Authenticity appears. The user can select as many security requirements as considered important for the chosen business process and then submit the selection by clicking on the *Submit*-button (see Figure 5.1). The chosen security requirements are then given to the next page to narrow down the list of insider threats.

Security Principle Selection

Please select the security principles that are the most important ones for the process you would like to



Confidentiality

Integrity

Availability

Accountability

Authenticity

Submit

Figure 5.1: Define Security Requirements (Step 1)

Step 2: Input business process model

Once the security requirements are defined, a new page appears. A drag-and-drop functionality supports the user in uploading the BPMN diagram. After dropping the .bpmn file into the user interface (UI), it is directly shown in the canvas. How the visualization of the BPMN model works is further elaborated in Section 5.3.

Step 3: Analyze business process

When the user clicks on the *Show Threats*-button the prototype automatically iterates through all the threat groups that are present in the insider database and linked with the previously selected security requirement. For each threat, it checks whether the elements that are connected to the threat are also present in the BPMN file that was given as input. If no element can be found in the diagram, the threat is not added to the threat list. Once all threats in the database have been checked, the threat list is generated as output.

Step 4: Investigate list of threats

Figure 5.2 is a screenshot of the page where the extracted insider threats are displayed. The output of the threat list is shown to the left of the visualized diagram and is interactive. Hence, every time the user clicks on a threat, all elements in the BPMN model which are potential attack targets for the corresponding threat are highlighted in orange. In the canvas, the user can navigate and zoom in or out using the mouse.

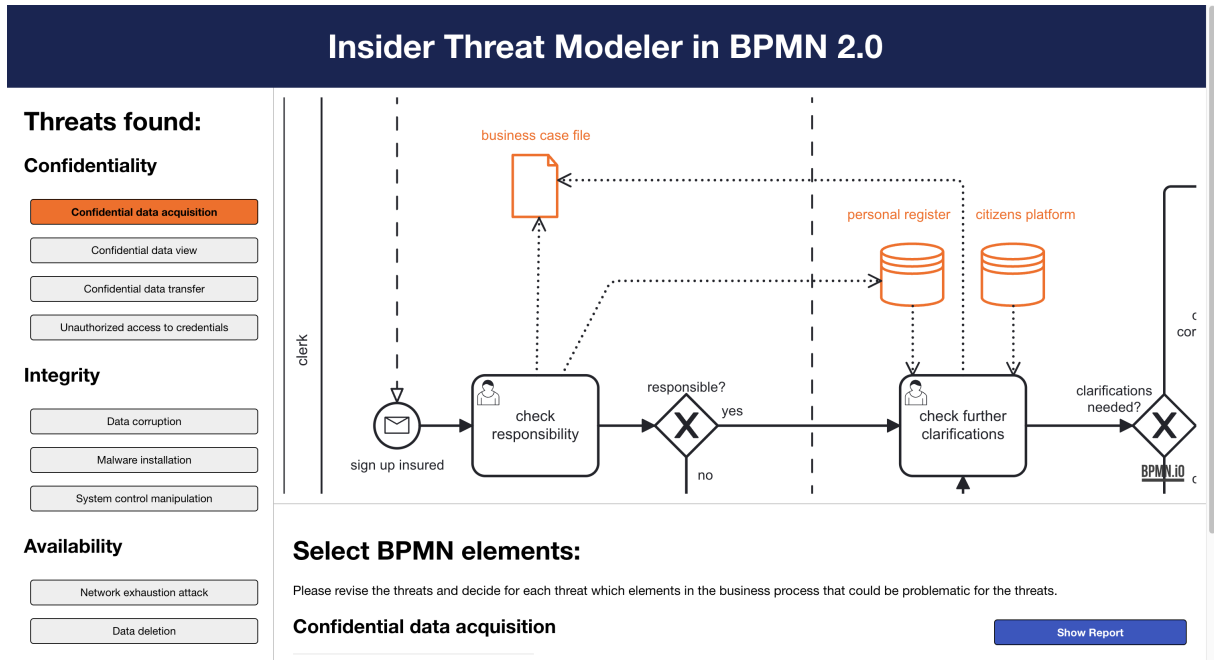


Figure 5.2: Investigate List of Threats (Step 4)

Step 5: Select important threats

Next to the coloring of the BPMN elements in the business process model, each name of the affected elements is listed below the diagram (see Figure 5.2). The user then has to decide for each threat, which BPMN elements could be actual targets for the threat. The user can choose as many elements as relevant to each threat and then submit the selection. Figure 5.3 shows the component where the user can conduct the selection.

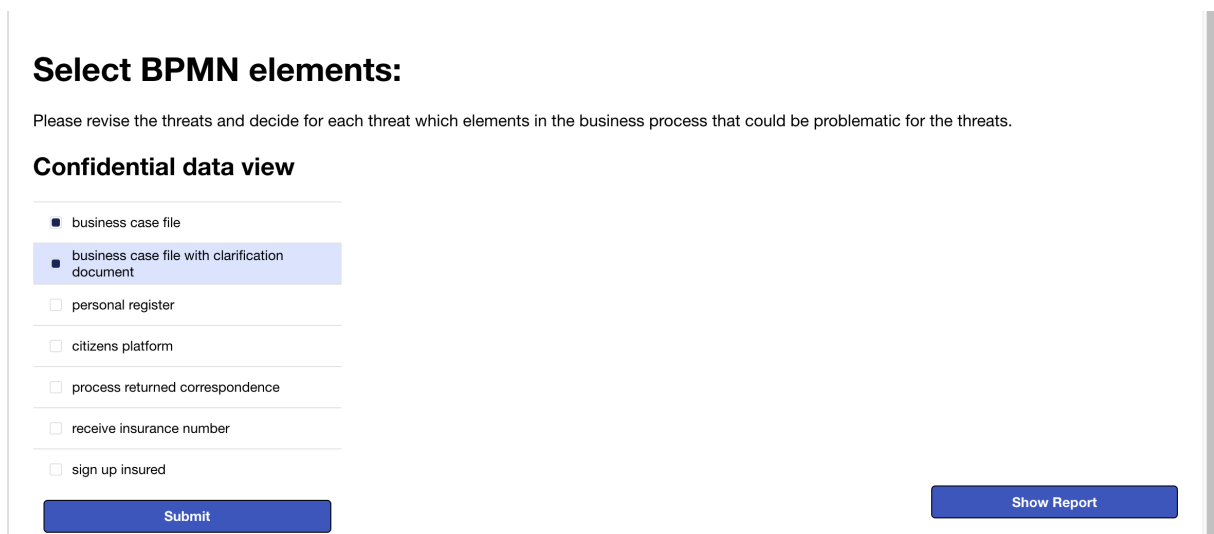


Figure 5.3: Select Important Elements (Step 5)

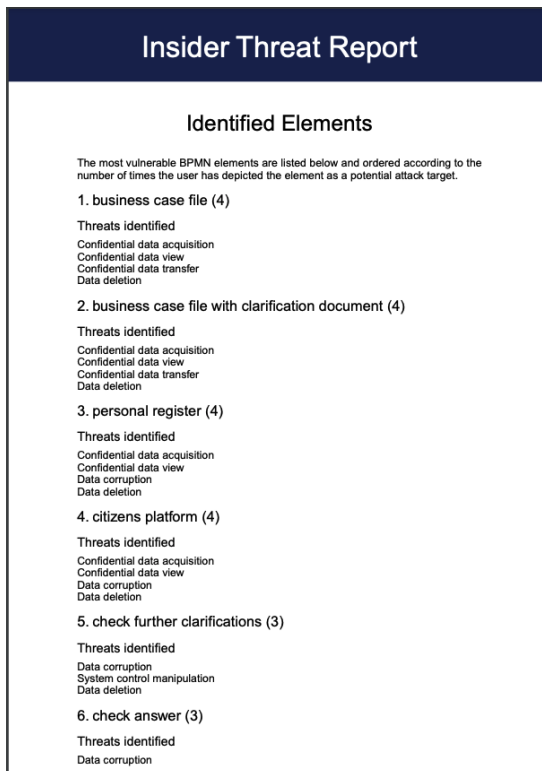


Figure 5.5: Elements in PDF Report

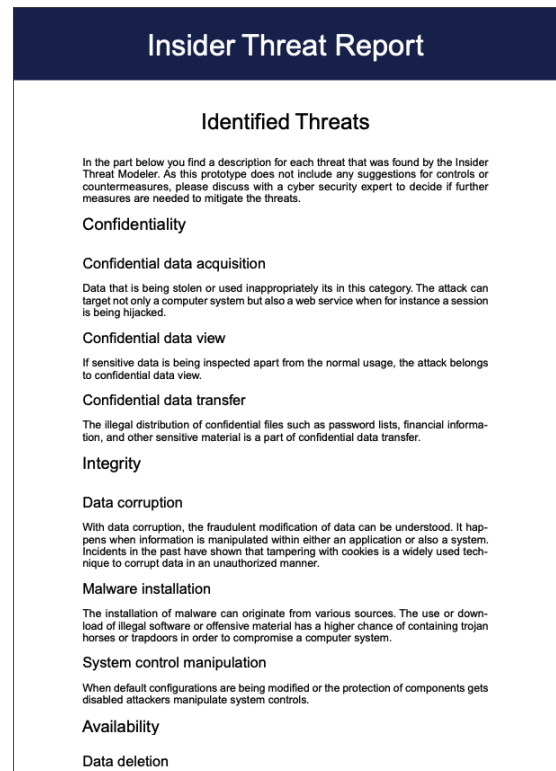


Figure 5.6: Threats in PDF Report

5.2 Input BPMN Model

[20] used BPMN as business process modeling language, because it is a *”widely used graphical representation for business requirements and makes it easier to define and communicate business processes between different stakeholders of the system”*. Another advantage is its *”flexible extension mechanism for any kind of representation and the ability to extend the model for interoperability issues”* [20]. Besides [20], [23] and [22] showed that it is a valid approach to model threats.

In addition, as it is the de-facto standard for modeling business processes, there is a long list of tools that are optimized for BPMN models and make business process modeling way more efficient [21]. Additionally, the human factor is included in the model [18]. Pools and lanes declare who carries out which activities in the business process [35]. Furthermore, there exist various approaches such as [20] where an extension or additional graphical models are used to annotate the BPMN model with security features.

Therefore, the implementation of the insider threat modeling tool relies on modeling business processes with BPMN. Usually, BPMN models are in XML format and are saved with the .bpmn suffix. This enables users to add more XML elements to the BPMN model and customize it as preferred [50].

The rules of modeling a business process in BPMN are clearly defined, but as everyone can draw their own process following their own conceptual model, some assumptions need to be made. Moreover, as this prototype automatically analyzes BPMN files, some

restrictions for the model itself have to be introduced. In the following paragraphs, these assumptions and restrictions are elaborated.

Just as [23], this thesis also assumes that an insider does not have access to the BPMN model itself and therefore is not able to modify the business process by manipulating the model. This assumption is made because the modification of a whole business process could happen at any point in the process. Visualizing this threat would result in redundant and duplicate information. In addition, it is assumed that an insider is able to manipulate the input before passing through a control of the model. Therefore, [23] looks at five different BPMN elements that are vulnerable to attacks in general: conditions, expressions, scripts, service calls, and data flow. Furthermore, they do not consider control flow elements because they are controlled by the process engine and they assume that process engines are not vulnerable to cybersecurity attacks. Hence, nothing that is set by the process modeler is considered for the insider threat mapping such as rules or transitions [23].

The following BPMN elements are not considered as potentially vulnerable elements to insider threats: Gates and Events. The only exceptions are Message Events. *Message Catch* and *Message Throw Events* can happen during the process but a *Catch Event* can also be used to set the process in motion, which would then be called a *Message Start Event*. Furthermore, as *Script Tasks* are implemented in a process engine by a business process modeler, these are also excluded from modification.

In BPMN, messages can be sent either through tasks such as Send and Receive Tasks, but also by using Intermediate or Start Events. It is assumed, that no matter if the message is modeled as a Task or Event in the BPMN model, the same attack vectors can result from them. For this reason, the terminology used as of now is going to be *Message Send* for Send Tasks and Message Throw Events and *Message Receive* for Receive Tasks, Message Catch Events, and Message Start Events.

After explaining all elements that are not included in the prototype, the following list summarizes the BPMN tasks that are taken into consideration for the mapping of the insider threats to establish the insider threat database:

- Manual Task
- User Task
- Service Task
- Message Send (including Send Tasks and Message Throw Events)
- Message Receive (including Receive Tasks, Intermediate Catch Events, and Message Start Events)
- Data Object
- Data Store

Apart from the list of elements that are considered in the mapping, some assumptions about the graphical modeling were made. Sometimes a task needs access to a database system or documents. It is expected of the process modeler to include these elements in the BPMN model, as the prototype would not be able to extract them from the name of the task if the graphical representation is missing.

Moreover, each element in the business process should be labeled. The prototype needs to be able to distinguish the same element tasks from each other. For instance, if there is more than one database system in the process the prototype needs to differentiate between them.

5.3 Prototype Implementation

To process the BPMN diagram, the prototype uses the following components: a tool to parse the BPMN model, the insider threat database, a service that connects the elements in the model to the insider threats, and a tool to visualize the threats and the specific elements where the attacks could originate from.

5.3.1 BPMN.io

BPMN.io is an open-source software tool by Camunda Services GmbH that is made for modeling BPMN, Decision Model and Notation (DMN), and Forms [50]. Its BPMN Viewer and Modeler are projects uploaded on GitHub, that can be downloaded and customized [51]. Also, there are examples for extensions such as coloring the BPMN elements, commenting, or bundling functions [52].

BPMN.io was found to be suitable for this thesis because it can parse and visualize BPMN diagrams. Furthermore, the possibility of customization made it possible to add the logic from this thesis' methodology. Moreover, the example extension functions are clearly documented which enabled a straightforward implementation [52].

In addition, there is an implemented functionality to download the diagram in its XML format as a .bpmn file or as an image in an SVG format. This was particularly valuable to preserve the findings of the business process analysis on insider threats.

Hence, the tool of BPMN.io not only helped with the processing of the BPMN model, it also supported the analysis and visualization phase in the prototype. Therefore, the decision was made to include it in the prototype.

5.3.2 React

Even though the tool of BPMN.io was customizable, the overhead of writing all the logic for the element mapping in a script of HTML would have been too big. In this case, a React project was set up as the web client, which helped to combine the graphical representation with the logical mapping. Additionally, the web application gained higher usability, performance, and speed with the support of React and therefore enhanced the user experience [53].

A web-based solution was chosen for the following reasons. In general, web-based solutions are easily accessible to users and do not need a lot of effort to be created by the programmer. Moreover, they run on nearly all devices and have high usage rates [54].

5.3.3 Insider Threat Database

The core of this thesis is the mapping of insider threats to the business process elements. The following paragraphs go into more detail about how this was carried out for the prototype.

[21] checks the compliance of BPMN models with an organization's security policy. They implemented a framework where rules for each security principle can be specified in order to align them with the security policy. They not only handle Confidentiality, Integrity, Availability, Accountability, and Authenticity as proposed by this thesis, but also include Auditability, Non-repudiation, and Privacy [21].

They specify, which type of BPMN element applies to which security principle by only looking at activities, data objects, and message flows of BPMN. According to [21], all aforementioned types can conflict with the Availability and Integrity of the system. However, Accountability is only applicable to activities. The security principles Authenticity and Confidentiality can both be compromised by data objects, while Authenticity also includes activities and Confidentiality message flows.

Based on this research paper [21], the coarse mapping of BPMN elements to the defined security principles was made. However, this thesis does not agree with excluding message events as elements that could conflict with the security principle of Authenticity as social engineering attacks are common to be part of a message [46]. Therefore, also Message Events and Tasks are included as potential threat targets.

Once the security principles were connected with the BPMN elements, it became possible to go more into detail about the threat groups that were described in Chapter 4. For each threat group, it was decided which BPMN elements could be a potential target for an insider attack in the corresponding group. The gathering of all potential threats in each threat group that were collected from the literature helped in determining the mapping of the BPMN elements. Table 5.1 shows the result of the mapping phase.

Table 5.1: Mapping Insider Threats to BPMN Elements

Insider threat group	Manual Task	User Task	Service Task	Message Send	Message Receive	Data Object	Data Store
Confidential data acquisition	X	X	X	X	X	✓	✓
Confidential data view	X	X	X	X	✓	✓	✓
Confidential data transfer	X	X	X	✓	X	✓	X
Unauth. credential access	X	X	X	X	X	✓	✓
Data corruption	✓	✓	✓	✓	X	✓	✓
Malicious code modification	X	X	✓	X	X	X	X
Malware installation	X	X	X	✓	✓	✓	X
System control manipulation	X	✓	X	X	X	X	X
Hardware attack	✓	X	X	X	X	X	X
Resource exhaustion attack	X	X	✓	X	X	X	X
Network exhaustion attack	X	X	✓	✓	X	✓	X
Data deletion	X	✓	✓	X	X	✓	✓
System control circumvention	X	✓	X	X	X	X	X
Unauth. privilege elevation	✓	✓	X	X	X	X	X
Misuse of privileges	X	✓	X	X	X	X	X
Social engineering	X	X	X	✓	X	X	X
Impersonation attack	X	✓	X	✓	✓	X	X
Man-in-the-middle attack	X	X	X	✓	✓	X	X

As soon as the database schema was established, it was implemented in the prototype. Figure 5.7 visualizes how the database was structured. On the top level, the security principle points to an array of threat objects, that represent the threat groups as shown in the table above. Each threat has a description and a list of BPMN elements that could be a potential target of the threat.

For the entities security principle, threat group, and BPMN element ENUMs were created. This ensures that only the set of a defined list of each entity can be accessed in the code and prevents spelling mistakes in the code.

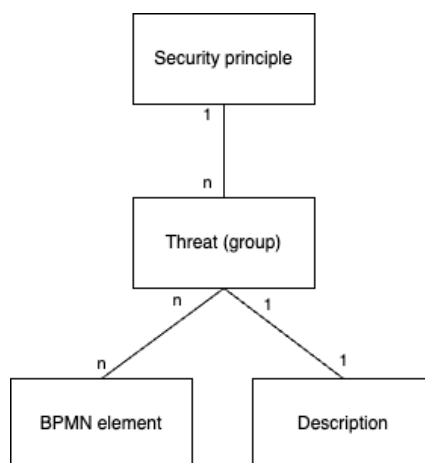


Figure 5.7: Entity Relationship Diagram Insider Threat Database

5.3.4 Computation

After the database was established and the methodology was defined, the back-end implementation phase of the prototype started. Each step of the functionality as mentioned in Section 5.1 is briefly explained in the following paragraphs.

The first step in the implementation is the selection of the security principle. It is essential for further steps that this information is saved. Only then, the prototype can match the input with the database to be able to output the potential threats in the given BPMN model. Therefore, the preservation of the input is ensured by using `useState` of React. This hook updates the state every time the variable changes.

As soon as the model is put into the canvas in Step 2, its XML coding is parsed with a function provided by BPMN.io and then visualized on the web page. All elements from the model are extracted as objects that contain essential information, such as the name it was given and what type of BPMN element it is. This information is stored per default of BPMN.io in the variable `elementRegistry`. Hence, it is the list of objects that are part of the BPMN model which was parsed by the BPMN.io tool.

For each security requirement that was selected by the user in Step 1, a helper function called `visualizeThreats()` is called. This function takes a security requirement and the `elementRegistry` as arguments. First, the function iterates through all threats that are linked to the security principle. Then, it checks for each threat whether its mapped BPMN elements are also part of the BPMN model, which is what the `elementRegistry` is needed for. This ensures that only threats are included in the output that can be mapped to an element in the given business process. In detail, for each insider threat, it is checked, which elements in the `elementRegistry` match the type of BPMN element with the BPMN element types that are linked to the threat in the database. The following code snippet (Listing 5.1) shows this check of elements:

Listing 5.1: Check Which Elements of the Database Are Part of the BPMN Model

```

1  const bpmnElements = threat.elements; //from database
2  bpmnElements.forEach(bpmnElement => {
3    elements = elementRegistry.filter(e =>
4      e.type === 'bpmn:' + bpmnElement);
5  })

```

At the end of the iterations, all threats of a security principle that are part of the BPMN model are pushed in a `useState` called `allThreatsFound`. Hence, this variable should then contain at least one object with a security principle and a list of applicable insider threats. Additionally, each threat is represented as an object with the threat's name and the corresponding elements that were collected from the `elementRegistry`.

Step 4 in the methodology is where the user can analyze the list of threats. A function called `showElementsOfThreat()` filters out all the elements in the BPMN model, which can be linked to the threat. As the `useState` `allThreatsFound` has saved all elements that apply to a threat, it can simply retrieve the selected threat to extract the elements in the BPMN model. Then, the elicited elements are colored orange with a function provided

by BPMN.io. This function takes a list of elements that need to be colored and a color code as arguments such that the canvas directly takes over the instructions.

For each threat, the selected elements from Step 5 are saved as an array. Hence, if the user decides to go back to a certain threat afterwards, the UI gives feedback on which elements were already selected for the threat. The user is still able to edit the list of elements until the *Show Report*-button is clicked.

After Step 5 is finished and the report page is shown, only the elements that were at least once selected by the user get collected. Now the mapping is done differently, as for each BPMN element, an array of threats is saved. The elements additionally get sorted according to the number of threats that were found. This number is saved as `count`. This and the rank information are aggregated in a sorted final list of elements which is then used for the report. For each element, the following information presented in Listing 5.2 is collected and then the elements are sorted according to the `count` number in a decreasing way and then saved in a `useState` called `rankingElements`.

Listing 5.2: Information Stored in an Element in the Final List of Elements

```

1 element: {
2   count: number,
3   threats: [insiderThreats],
4   bpmnElementId: bpmnElement.id
5 }
```

In Step 6, the BPMN model is not only transformed by adding colors but also the names of the elements are changed. The color is defined by the number of threats an element has, hence, by the value of `count`. The name of the element is modified by adding the rank of the sorted final list (called `rankedElements`) in front of it. These two modifications are added in the diagram and also shown in the list of critical elements on the left of the canvas. Therefore, one should have a broad overview of all the threats and also directly spot the elements in the BPMN model.

As the actual source of the list of final threats is rather complicated, Figure 5.8 visualizes how the threats are narrowed down to the final list of threats that appears in the report at the end of executing the steps of the prototype.

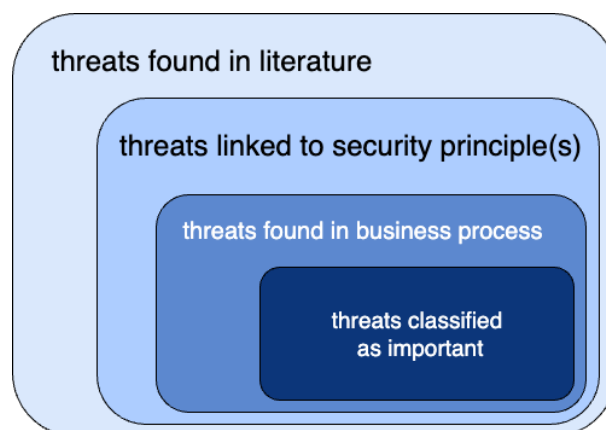


Figure 5.8: Venn Diagram of Insider Threats

5.4 Threat Report Output

As previously explained in the walkthrough of the prototype in Section 5.1 and 5.3, a screen in the prototype visualizes the final set of threats that were found in the business process and classified as important in the form of a list and a modified BPMN diagram.

As an extra feature, the user is able to download the findings of the prototype. It is possible to export three different files: the diagram as an SVG or .bpmn file, and a PDF report of all threats. An SVG is a vector image that keeps the readability if downloaded. The .bpmn file can be downloaded in an XML format, such that the diagram can be further modified in another modeler. The PDF report contains the list of threats and critical elements sorted by the count of how many times it was selected as a potential target by the user. In addition, the identified threats are elaborated briefly and itemized by the security principle they belong to.

There are different file types and outputs of the prototype because the analysis of the insider threats is probably not finished by the time the report is visualized to the user. The user can decide independently how to use the insights gained from executing the prototype with the business process. On the one hand, as a security expert should be informed about the results to work out measures for mitigating insider threats, it can be beneficial to share the list of threats from the PDF report. On the other hand, it could be beneficial to discuss the results also with a process expert, who may then improve the business process to prevent insider threats.

While the functionality of downloading the BPMN model as an SVG or XML file exists in the BPMN.io tool, the PDF report had to be implemented on its own. The data for the report already existed on the report page, it only had to be exported to a PDF format. This functionality was added by implementing the package `@react-pdf/renderer` [55]. Consequently, all information about the threats and the critical BPMN elements in the process can be written in React and then automatically converted to a PDF.

Chapter 6

Evaluation

In order to evaluate whether the proposed methodology and prototype are effective and useful in the real world, a practical evaluation was conducted and its exact structure is described in this chapter. It is divided into five sections. While the first section explains the theory of method, Section 6.2 elaborates how the theory was applied and how the case study was conducted. In Section 6.3, the outcomes of the case study are described. Section 6.4 then discusses these findings and compares the prototype with the related works from Chapter 3 and developed requirements from Chapter 4. The final section provides a summary of all the discoveries made and highlights the limitations of the proposed contributions.

6.1 Evaluation Method

The evaluation method applied in this thesis is a participatory case study. A case study is the application of the proposed products in a real-world environment to evaluate their quality [56]. According to [56], a *participatory case study* involves both academic and non-academic actors. Therefore, a real-world business process is being analyzed by the author of this thesis. As the implemented prototype also needs the input of a domain expert in the business process, this knowledge was collected from the non-academic participants and applied during the evaluation. The findings are then assessed with the enterprise that provided the example business process.

The decision for this method was made because in theory, there are numerous ways of how complex a business process can be modeled. Hence, it was beneficial to analyze a business process that is also used in practice. Moreover, with a real-world process, it can be determined to what extent insider threats can be found that are indeed relevant to enterprises nowadays. If a hypothetical process had been chosen for the evaluation, the interpretation of the results could have been biased by the assumptions that were already made during the construction of the business process itself. For example, the process could have tried to include many different task types to test whether the prototype finds the most relevant threats in all of them. Nevertheless, this would have probably not reflected a real-world process as it may have only included a subset of types of tasks.

In addition, a high-fidelity prototype was employed for the evaluation that included all the described functionality from the last chapter and was fully connected to the insider threat database. The reason for this is, that not only the methodology but also the prototype's usability could be assessed. In a low-fidelity prototype, the user could have run into some problems that were not fully implemented by the time and therefore the functionality would have been limited.

6.2 Case Study

For the case study, potential companies were requested to provide an example business process. First of all, a proposal was written and distributed to various companies to find a real-world business process that could be evaluated. The focus of the research was on companies whose operations are regulated by the government because a higher chance of having the business process written down or even modeled in a business process model was assumed. Four companies were sent the proposal and one accepted to participate in the case study.

The company that agreed to share one of its business processes is called *Informatikgesellschaft für Sozialversicherungen GmbH (IGS GmbH)*. Their business model is to provide IT solutions for social insurance organizations that are managed by the cantons in Switzerland to support and digitize their business processes [57].

Once the company agreed, the next step was to define a suitable business process. The requirement for the example process was the involvement of information systems and human actors in the process of finding insider threats. According to these requirements, IGS GmbH proposed a business process from their side. In a virtual meeting, the chosen process was elaborated by a process architect of IGS GmbH to provide more information on the details of the systems, artifacts, participants, and their actions in the process. Furthermore, the context of the process was explained.

The business process provided by IGS GmbH models the tasks a clerk in an insurance organization needs to accomplish when a member or insured signs up to issue a new insurance number in the form of an insurance card. It contains ten tasks, two data stores, two artifacts, and an intermediate catch event to receive a message. The whole business process is shown in Figure 6.1.

As the business process was not yet in XML format, it was remodeled with the BPMN modeler that is part of the bpmn-js Examples on GitHub [52]. In this process, not everything was copied in the exact same way as in the example process to meet the requirements of the prototype that were described in Section 5.2. For instance, in the example process, the message flows sometimes did not start or end in a separate element. If the receive task or the intermediate catch event was not present in the diagram, the prototype would have not been able to register it. Therefore, the insider threats that would have been connected to the message receive elements would have not been listed as a potential threat. In addition, all the names of specific programs or databases were removed to keep the confidentiality of the enterprise and its business process.

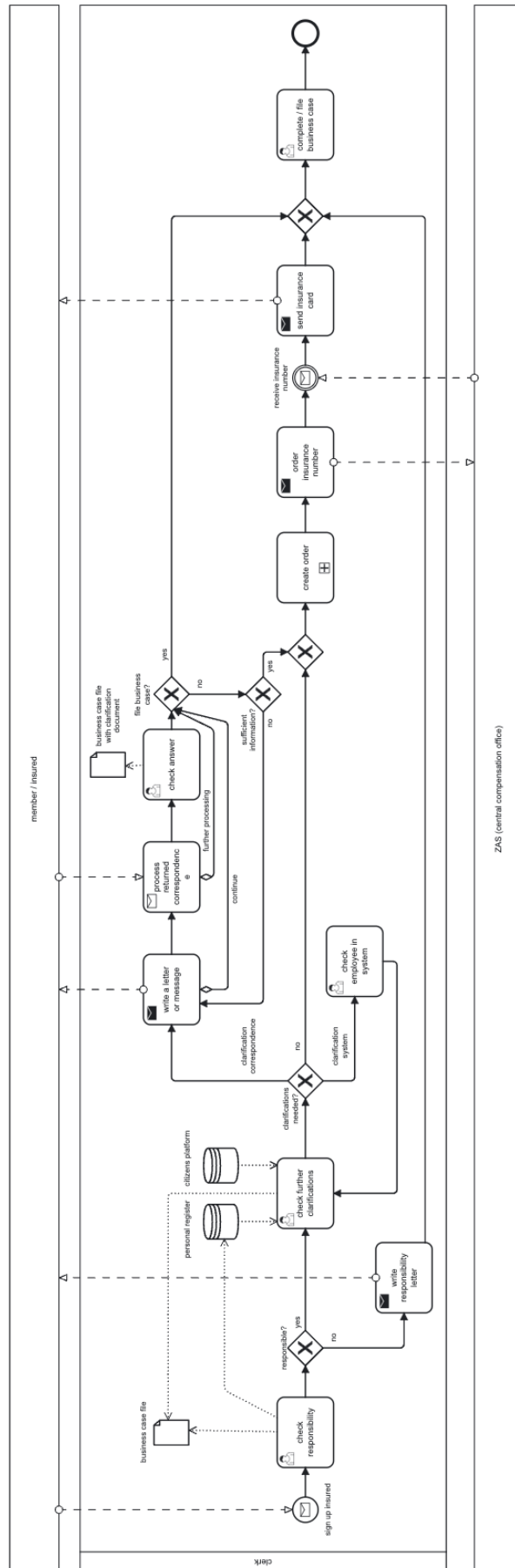


Figure 6.1: Example Business Process Provided by IGS GmbH

Once the business process was prepared, the actual evaluation could start. For this, IGS GmbH informed the researchers about the security requirements they considered as most important. These were then inserted in the prototype together with the business process. During the testing of the prototype, an evaluation report template was filled out by the author to keep track of every decision made. This report template included some basic questions about the business process, fields for writing down the reasons for the decisions, and in the end, a short questionnaire about the threats in the report.

The questionnaire in the report targeted answering the following questions:

- Q1)** Did the prototype filter out all the critical elements in the process?
- Q2)** Did the prototype filter out all relevant threats in the process?
- Q3)** Are there known threats that were not included in the report?
- Q4)** Are the elements that were found relevant regarding already implemented controls?
- Q5)** Do the security requirements and the threats in the output match their security principles?
- Q6)** How useful is it to identify threats in a semi-automated manner by relying on process models?

The questionnaire was filled out in a feedback session with the process owners and security experts of the company that provided the example business process. All results that were collected during the evaluation phase are elaborated and discussed in the next sections.

6.3 Findings

Here, the findings of the evaluation are listed. Subsection 6.3.1 gives an overview of the findings that were made by the author during the conduction of the case study, while Subsection 6.3.2 presents the results from the case study. Finally, Subsection 6.3.3 evaluates the feedback that was given by the domain experts.

6.3.1 Analysis During Case Study

User Interface

One insight was that the higher the number of threats found with the same count, the harder it gets to actually distinguish the BPMN elements from each other. The reason for this is that elements with the same count of insider threats share the same color code. Therefore, it is beneficial that there are also identifiers in the form of numbers implemented that help to find the corresponding BPMN elements faster.

Also, there is no possibility to go back to the selection stage of the threats from the report. On the one hand, this is an advantage, as one could argue that if something needs to be changed, it is better to start over and rethink each step. On the other hand, it could also be unfavorable, as somebody might press the button to show the report before clicking through each threat and selecting the important elements.

In addition, it is quite laborious to click through every threat one by one. It would be favorable to have an automatic iteration through all different threats. However, when clicking on a threat that was dealt with before, the selected elements are pre-selected from before. This ensures that the user can still make adjustments to the selection of elements of previous threats.

Furthermore, the PDF report has some design issues. The pages are not nicely broken between the list of threats. Even though nothing is cut off, the layout could still be improved.

Mapping Insider Threat Database

To begin with, it was discovered that the threat *Data acquisition* is most of the time applicable to the same elements as *Data view*. If information can be viewed, it could also be acquired or at least copied in a way that it can be used beneficially for the insider. However, in the database, it was decided that a Message Receive is not considered a potential target element for *Data acquisition*, even though it is linked to *Data view*. Hence, these attacks could be merged.

In addition, the threat vector *Unauthorized access to credentials* is hard to detect on the business process level. There is no BPMN element, which would give a hint that there are actually credentials stored. This might be more of a lower-level threat that would need to be part of another threat modeling procedure.

Furthermore, to decide whether *Data corruption* or *Data deletion* is applicable, one needs to know the access rights of a certain database or an underlying system. This might be easy for the security expert in the company, but hard for a business process analyst.

6.3.2 Case Study Results

The results of the case study included 13 critical elements and 7 different threats that belonged to 3 different security principles. The following Table 6.1 presents the critical elements and extracted threats that were found when evaluating with the example process. They are presented in the table for better readability. The threat report downloaded from the prototype can be found in the appendix. Figure 6.2 visualizes the number of threats as well as critical elements per security principle.

Table 6.1: Threat Report of Case Study

BPMN element	Insider threat group	BPMN element type
business case file	Confidential data acquisition Confidential data view Confidential data transfer Data deletion	Data Object
business case file with clarification document	Confidential data acquisition Confidential data view Confidential data transfer Data deletion	Data Object
personal register	Confidential data acquisition Confidential data view Data corruption Data deletion	Data Store
citizens platform	Confidential data acquisition Confidential data view Data corruption Data deletion	Data Store
check further clarifications	Data corruption System control manipulation Data deletion	User Task
check answer	Data corruption System control manipulation Data deletion	User Task
check employee in system	Data corruption System control manipulation Data deletion	User Task
check responsibility	Data corruption System control manipulation Data deletion	User Task
process returned correspondence	Confidential data view Malware installation	Message Receive
sign up insured	Confidential data view Malware installation	Message Receive
order insurance number	Confidential data transfer Data corruption	Message Send
send insurance card	Confidential data transfer	Message Send
write a letter or message	Data corruption	Message Send

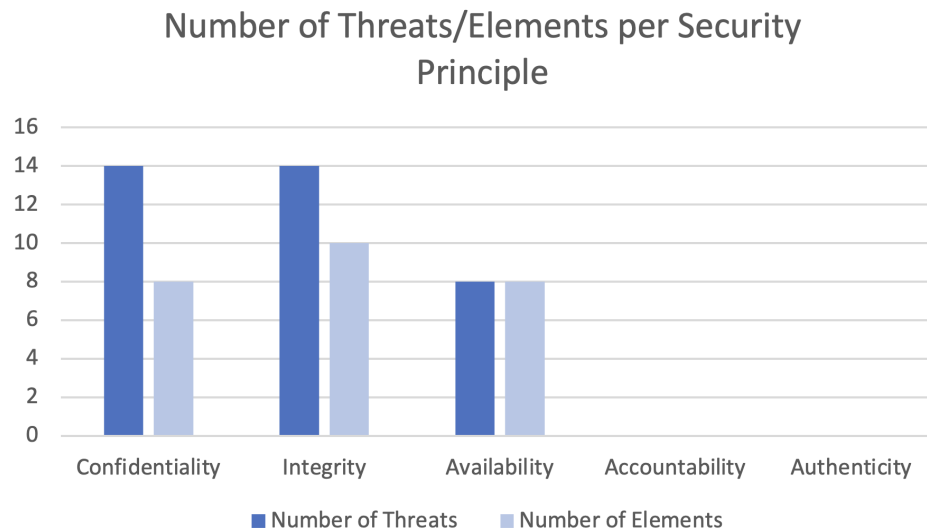


Figure 6.2: Number of Threats/Elements per Security Principle

6.3.3 Feedback Session

As soon as the author was finished with the evaluation, the domain experts were asked to give feedback on the results that were issued by the prototype. The online meeting took place on 21. December 2023. Three experts from IGS GmbH and the author of this thesis formed the group of participants. Amongst the experts were Chrisitan Schinnerl, Chief Information Security Officer (CISO), his deputy, Thomas Schwarz, and Marcel Nagel, the process architect. As a first step of the feedback session, the database, methodology, and prototype of the thesis were briefly explained by the author. Then, the execution of the evaluation was elaborated to give them an insight into where the results came from, which were then presented in a further step. The author gave a broad overview of the found threats, where they were found, and what assumptions were made for some of the elements in the business process.

After the author had explained everything, a semi-structured interview was conducted. The following questions were formulated by the author. To distinguish them from the questions formulated in Section 6.2 the abbreviation *FQ* was chosen for *Feedback Question*.

FQ1) True positives: Are the threats that were found relevant/known?

Overall, the threats were deemed as relevant and also expected to be part of the output. The experts especially agreed on the attacks that compromise the security principle Confidentiality as there is a lot of sensitive data included in the process.

FQ2) True negatives: Are there threats that were not found?

According to the security expert, injection attacks were missing in the threat report. These could be used to extract confidential information from a database. Moreover, privilege elevation and social engineering are other threats that could occur in the business process but were not mentioned.

FQ3) Are there countermeasures or controls already implemented where the critical elements were identified?

On the organization level, the CISO explained that employee education, access control, and confidentiality agreements are part of the security controls that are implemented to mitigate insider attacks. To educate the employees, there are regular training sessions that need to be solved. In addition, the access control policy is restrictive, so that not everyone can access everything with all rights. Moreover, everyone who has access to any files or the network of IGS GmbH needs to sign a confidentiality agreement. Furthermore, logs ensure that all the updates that are made in the files can be traced back to the user. Additionally, for important decisions, the four-eyes principle is applied.

On the system level, there are some specific threats from the report where controls are already implemented to counteract insider attacks. It was argued that *Data deletion* is a threat that is less likely to occur because backups and archives preserve the data in the processes. Furthermore, *Data corruption* is mitigated by logging every update a user makes in a database. Also, the *Malware attacks* that have been detected in the receive tasks are counteracted with a filter. This filter sanitizes the messages that are received from outside the network and converts the attachments into PDFs to remove any malicious threats.

FQ4) Are there elements, where you think other threats than the ones identified could be possible?

According to the security expert databases could be a target for resource exhaustion attacks. They argued that huge queries could bring the database system down.

FQ5) Are there threats where you think an element is missing apart from the ones that were identified?

It was mentioned by the process expert that some threats could target the elements in the sub-process as somebody could corrupt the order of the insurance card.

6.4 Discussion

This section analyzes the results of the evaluation. In Section 6.4.1 the requirements that were defined in Chapter 4 are examined on their fulfillment. Section 6.4.2 discusses the answers given by the experts in the feedback session. Lastly, the thesis contribution is compared with other approaches that were found in the literature in Section 6.4.3.

6.4.1 Requirements

In Section 4.2.1 a list of requirements that the prototype should fulfill was provided. This section revisits these specifications and contrasts them with the experiences obtained from the case study.

1. The first requirement was the automatic generation of an insider threat list. This functionality has been realized in the prototype. There is a list of threats on the report page below the diagram. Additionally, it can also be downloaded as a PDF. Even though it is not fully automated, there is still a threat list of potential threats issued by the prototype which is then adjusted by the user to produce a final list of insider threats.
2. Inserting the business process as is, is the first part of the second requirement. This is met with the drag-and-drop functionality provided by BPMN.io. A user can input a .bpmn file of its BPMN-modeled business process. If the business process is modeled in another format, a remodeling in a BPMN modeler would be needed. If this is given, however, the business process can be inserted into the prototype directly. The second part of the requirement states that no security expert should be needed for the use of the prototype. This is not fulfilled by the prototype. The selection of the critical elements requires a security expert who knows on the one hand, how the tasks are supported by IT systems and on the other hand, whether the listed threats would be realistic for the corresponding elements in the business process.
3. The requirement about receiving feedback on which IT system needs further analysis is partially implemented. The user does get feedback on elements in the process that are potential attack targets. However, if the IT system is not modeled in the process, the prototype cannot register it. An IT expert inside the company would need to support the user of the prototype to assess which tasks are supported by which IT systems.
4. The visualization of potential targets in the business process is a key feature of the prototype. It was realized in order to give the user feedback directly in the business process. Each critical element has been highlighted by a shade of red color. Additionally, a number was added to the name of the element, such that the user can find it faster in the business process.
5. Requirement 5 expects the prototype to show the user which insider threats are connected to which elements in the process. The implementation of this functional feature was realized on the report page. The list of critical elements is shown on the left of the business process. If a user clicks on a critical element, a drop-down component reveals a list of all insider threats that have been found.
6. The next feature requires an ordered list of insider threats according to the severity of the risk. This feature was not realized. All potential threats as well as all critical elements are listed in the report. Nevertheless, there is no ordered list of the threats, as the proposed methodology does not provide any risk assessment approach. Section 4.2 describes the six steps of the methodology and includes a visualization of the functionality which is out of scope in Figure 4.3. The interpretation of the results is not part of this thesis. However, there is an ordered list of critical elements. During the implementation phase, it appeared to be more important to know which element in the process is most critical. The *severity* in this case was measured by the number of insider threats that were found for the corresponding element.

7. The next requirement is about intentional and unintentional insider threats. It expects the prototype to give feedback about both of them. As the insider threat database collected all potential insider threats that were found in the course of the literature review, intentional as well as unintentional threats were included. These threats were then mapped to the BPMN elements. Therefore, both are part of the proposed prototype. Nevertheless, they are not clearly labeled as such. Again, the interpretation of the insider threats and critical elements is not included in the proposed methodology. Hence, an expert would need to decide which threats could happen accidentally or maliciously.
8. The last requirement demands that the prototype can be applied intersectoral and across many industries. As the collection of insider threats entitles many different sources, they should cover a variety of threat vectors not depending on a specific industry. Additionally, BPMN is a widely used modeling language for business processes and is accepted as a standard. This ensures that the use of the prototype is not limited to a subset of enterprises.

6.4.2 Case Study

The goal of this subsection is to answer the questions from Section 6.2, which were formulated to evaluate the methodology and the prototype in the case study. This is accomplished by analyzing the results of the feedback session, which was described in Section 6.3.3.

Q1) Did the prototype filter out all the critical elements in the process?

According to the experts, nearly all the critical elements appeared in the prototype. It was mentioned that in the subprocess *"create order"* potential insider threats could also be detected. However, as the BPMN element type subprocess was not considered in the prototype, it should not come as a surprise that there are no threats listed for this element in the report. Too few details about the subprocess exist that a link to any insider threat could be automated. Nevertheless, one could argue that besides the critical elements that are being proposed by the prototype other elements in the business process should be available for the selection. In contrast, the whole logic of the selection feature would then need to be overthought. The subset of threats should help the user to reduce the workload of checking the elements that are selected.

Q2) Did the prototype filter out all relevant threats in the process?

Most of the threats were viewed as relevant by the experts. They especially agreed on the importance of the threat vectors that are part of the security principle Confidentiality. In numbers, 14 threats were found that would compromise the Confidentiality of the process out of the 36 insider threats in total. They mentioned that some of the threats that were found in the report are not important as they are being mitigated already. On the one hand, this can be a valid argument because the risk of the threat vector being applied in an attack is minimized a lot. On the other

hand, even though there might be countermeasures, the insider might know how to circumvent them in a sophisticated way and under the radar. In addition, the fact that the threats have been mitigated already attests to the relevance of the threat.

These arguments can be applied to the examples that were mentioned in the feedback session. For instance, the threat *Data deletion* was said to be mitigated very well with access control limiting the users who have the privilege to delete and with backups, where data could be retrieved if it was deleted nevertheless. In contrast, as *Data corruption* is being counteracted with logs and the four-eyes principle, it could nonetheless be argued that this threat is relevant, because there might still be a possibility to disable logs or to impersonate an employee which compromises the Accountability of a user. Moreover, the logs would need to be inspected every time someone updates a version of a file and this is probably not done in every single case.

Accordingly, Figure 6.3 shows the division of the threats that were perceived as relevant, relevant but mitigated, and the others. Based on the arguments mentioned before, the author concluded that the true positive rate of insider threats is 88.9% by including also mitigated risks.

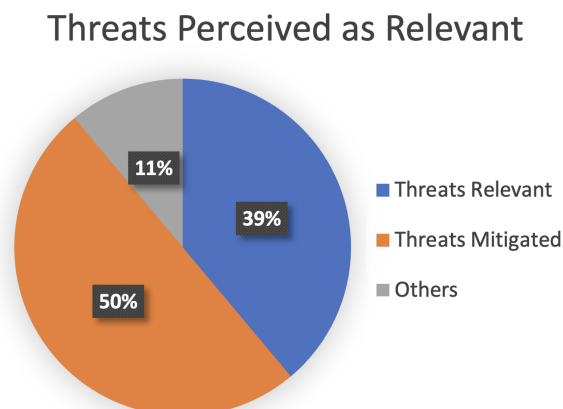


Figure 6.3: Number of Threats Perceived as Relevant

Q3) Are there known threats that were not included in the report?

As the cyber security expert stated, some threats were missing in the report. First of all, he would have expected injection attacks to be in the output. They were included in the output, however not as injection attack itself. The threat vector SQL injection was part of the data corruption attack group. However, he argued that injections can be used to manipulate data but also to view confidential data. Therefore, it could be used for both *Confidential data view* and *Data corruption* or a new attack group could be added to the database only for injection attacks.

Furthermore, the *Resource exhaustion attack*-group was also missing in the threat report according to the CISO. In his view, it could be mapped to data stores, which are the BPMN elements of databases. This remark is legitimate as an insider could come up with huge queries, which would exhaust the limit of the capacity of the

computing power in the database. When inspecting the mapping of the insider threats to the BPMN elements, one can only find a link from resource exhaustion attacks to service tasks. Hence, the data store was not included in the mapping. It would make sense to add it to the resource exhaustion attack so that it would be part of the threat report when availability is selected as an important security principle.

Q4) Are the elements that were found relevant regarding already implemented controls?

The elements that were found were also perceived to be relevant by the experts. Even though some attack vectors were not fully agreed with by the experts, the critical elements themselves still encountered some relevant insider threats. For example, the start event where a message is received was given two threat vectors namely *Confidential data view* and *Malware installation*. As *Malware installation* is mitigated by a filter that gets rid of any malicious data, this insider threat is less important. Regardless, the experts agreed on being able to view confidential data. Therefore, the element is still classified as relevant.

Q5) Do the security requirements and the threats in the output match their security principles?

Yes, the threat report included potential attack vectors from all three principles that were selected in the first step. However, *Privilege elevation* and *Social engineering attacks* were missing in the report in the view of the CISO's deputy. These attack groups would have certainly been a part of the threat report if the security principles Accountability and Authenticity were selected as well. Thus, they were eliminated from the potential attacks in Step 1 of the methodology where the security principle is selected. This is anticipated because the concept was designed in this way. Yet, the approach might need to be reconsidered to not lose any relevant threats along the way. There will always be a trade-off between providing too many attacks versus including all relevant attacks but adding a lot of noise which might also mean an overhead of workload.

Q6) How useful is it to identify threats in a semi-automated manner by relying on process models?

The feedback session has shown, that the prototype has extracted most of the threats that were relevant to the business process. However, it also became evident, that because of human input, some of the relevant attack vectors were lost because not all security principles were selected. Hence, it could be argued that the prototype should output all of the insider threats possible in the business process regardless of any security principle. In contrast, this would increase the list of insider threats vastly, which would in consequence increase the workload for the interpretation of the results by the experts again.

Based on these insights, it can be assumed that a company designing or analyzing a process could successfully use the prototype to extract potential insider threats to find suitable mitigation techniques at a later stage. This would make the business process itself more secure and would lower the risk of being attacked from the inside.

6.4.3 Comparison with Related Work

This subsection deals with the analysis of the approaches in the literature. They are compared with the methodology and prototype this thesis proposes and discussed in the course of the next paragraphs. As this thesis focuses on the technological perspective, the sources found in the psychological domain are not included in this discussion. Moreover, since the proposed methodology comprises only the threat modeling approach and not a profound risk assessment method, the three sources [6], [15], [16] are excluded from the comparison as well. Hence, this part discusses the approaches in the literature where authors analyze business processes to extract insider threats or threats in general.

[18] mainly spotlights sabotage and data-exfiltration attacks in the domain of insider threats. These already cover a majority of the threats that were written in this thesis' insider threat database. However, attack vectors that compromise the security principle Availability are not part of their analysis. Examples of these are *Network* or *Resource exhaustion attacks*. Furthermore, their approach only automatizes the Fault Tree Analysis, which encounters sabotage attacks. The prototype of this thesis in contrast encompasses all insider threats found in the literature. On the contrary, they are able to show a potential path the adversary could take by analyzing each event that would need to happen for an attack to be successful. Hence, they can construct multiple attack targets that are connected. The methodology of this thesis is only capable of identifying single targets. Nevertheless, the prototype of this work visualizes the insider threats directly in the business process. This is beneficial because no other modeling language than BPMN is needed to understand the output. Thus, the interpretation of the output with various stakeholders such as process architects or business consultants is easier. Similar to this thesis is the case study [18] has conducted with the example of an election process in the United States [18].

As a real-time monitoring system is a different approach than threat modeling in general, there is not too much to discuss about [19]. The methodology of this thesis is applied as a preparation or analysis of a model. [19] analyzes the logs of the system to indicate at what point insider threats could occur. This might be effective if the tool is added on top of a system after its implementation. Nonetheless, during the design phase or for a detailed analysis, threat modeling might be beneficial, as it should prevent attacks before they even happen. Additionally, [19] has not applied its methodology to evaluate their contributions. It was only proposed as future work [19].

[20]'s goal is to define the security requirements for a system that is being developed and hasn't existed before, whereas this thesis does not specify whether the business process is already implemented. Instead of taking the security principles as a basis for analyzing the business process, their approach derives the security principles from the threat modeling procedure. Moreover, [20] annotates each BPMN element with icons to visualize the vulnerabilities in the business process. These icons represent security goals, secure data types, threats, access mechanisms, privileges, and transfer ways. This is valuable as it adheres to the business process model, but with so many icons, the model gets overloaded and loses its readability. To prevent this from happening, this thesis has not introduced icons or graphical representations of threats but used coloring and numbers only to specify the potential targets of an insider attack. Furthermore, [20] did not implement their

methodology in an automated tool. The idea is that the extended model is annotated manually during workshops. This needs a lot of time from several experts, which is expensive for an enterprise. An example workshop has been carried out in a case study of a manufacturing company that produces aircrafts [20].

[21] on the other hand, automatized their methodology in the SecBPMN framework. The idea behind their approach has been partially taken over for this thesis as they established a mapping of security principles to the BPMN elements. Nevertheless, this thesis went a step further and mapped each threat group to the BPMN elements rather than just staying on the security principle level. This allows the user of this work's prototype to specify mitigation techniques rapidly. As [20], [21] used graphical annotations to extend the business process model as well. However, they only use icons for each security principle, which makes it less confusing than [20]. Still, the user might not get as much information if only the security principle and not a specific attack group is shown in the business process. Additionally, [21] not only evaluated their methodology in a case study, but they also conducted an empirical study and a scalability analysis, which give the approach more credibility and validity [21].

[22] managed to come up with a solution, where a business specialist is able to fulfill the needs of expertise such that the prototype generates a list of threats out of the business process. Nonetheless, the annotations that are the subject of the business analyst to add to the business process model add another layer of complexity. As BPMN diagrams can get very large and complicated, such that it is hard to keep the overview, these annotations intensify this effect. Therefore, it is better not to add a lot of new elements. This thesis aimed at keeping the business process model as simple as possible and only adding shades of colors and numbering in the naming without adding any more elements or annotations. Another insight from the analysis of this thesis and the article of [22] is, that even though threat modeling with BPMN can be automated, there is still some user input needed such that the prototype is able to output a list of threats. This thesis supports the involvement of a security expert by visualizing the threats in a list view but also inside the business process itself to give a clear overview. In addition, [22] developed their prototype, especially for the e-Government context, while this thesis prototype tried to make it applicable to many industries and fields. Nevertheless, they also conducted a case study with an example process in this context [22].

As already stated in Chapter 3, [23] designed their methodology for non-security experts like [22]. They, however, did not need user input at all, as they mapped the business process to the NIST list of known vulnerabilities. This, on the one hand, economizes the costs and labor of experts, but on the other hand, the vulnerabilities might not be applicable to every single business process. Therefore, it might still need an expert to draw insights from their prototype. Another difference to this thesis is that the authors from [23] set the objective, that the prototype should be non-invasive to the business process. Hence, they created an attack graph, instead of modifying the business process model itself. This is advantageous, as it does not add more complication compared to the approach of [22]. Nonetheless, the attack graph itself looks quite overwhelming and it is hard to derive any conclusions that could improve the business process. To evaluate their methodology, a case study was conducted with an *"invoicing integration process"* [23].

6.5 Limitations

In the last section, the prototype was analyzed and discussed from different points of view. Positive as well as negative aspects of the proposed methodology and prototype were discovered. To round them up, this section elaborates on the limitations that were found in the course of the evaluation.

First of all, it was discovered, that in many cases, a security expert is still needed to assess the outcomes of an automated prototype that elicits threats out of a business process. Therefore, it is a challenge to fully automate a threat modeling approach. This thesis requires a security specialist as well to select the potential target elements from a potential attack. The reason for this is, that a business process' context, unwritten rules, or not modeled systems need human understanding and knowledge of the company to be able to assess threats in a business process.

Another insight is, that the threat modeling approaches are limited to a certain size of its input. Even though this thesis does not add other elements to its business process, the BPMN model still gets confusing the bigger and more complex the process is. Mostly, the visibility or graphical understanding deteriorates. In addition, the same threats appear on different elements, which might lead to other problems.

Then, as there are no technical details revealed in a business process, some threats are redundant or missing. This thesis tried to overcome this problem by letting the user decide for a given artifact, system, or task, whether a threat is applicable. Nevertheless, if the user does not have a deep understanding of the technical setup of these elements, the usefulness of the prototype is very limited.

Moreover, the prototype has only been tested for a few examples. Because of this lack of data, it cannot be proven that the mapping of the insider threats is correct. It would be essential for the system to undergo tests with many different processes to verify the validity of the mapping of insider threats to the BPMN elements. Furthermore, experts from both the business and the security side would need to give feedback to improve the mapping in the database.

Additionally, it was discovered in the case study that some of the threats were lost along the way because of the selection of the security principle in the first step of the methodology. This is problematic, as it would have been found as part of the threat report if there was no restriction because of these principles.

Chapter 7

Summary

To summarize, this thesis has contributed a database, that collects all insider threat vectors that were found in the literature, a methodology that describes how an enterprise can find potential attack targets of insider threats in its business processes, and an automated prototype, which takes a BPMN model as input and visualizes the extracted insider threats in the business process as output. The proposed methodology contains six steps: 1. define security requirements, 2. input business process model, 3. analyze business process, 4. investigate list of threats, 5. select important threats, and 6. visualize important threats. The prototype takes these steps as a basic structure and implements them in a web application.

The web application integrates the BPMN.io tool to automatically parse and visualize a BPMN model. The elements of this model are then compared with the mapping in a database, where each insider threat group is linked to a number of BPMN elements. The user has the possibility to select the threats that are important in the next step to give the report a more concise view of the insider threats found in the business process.

The validity of the methodology and the prototype were evaluated in a participatory case study with an exemplary business process provided by IGS GmbH. Even though the author of this thesis conducted the evaluation when running the prototype, security, and process experts of IGS GmbH were involved by explaining the business process and providing the selection of the security principles in the early stage of the evaluation. At a later stage, they interpreted the results that were given as a threat report and the edited BPMN model to validate whether the prototype fulfilled the expectations of the experts. The evaluation showed that the methodology and the prototype extracted relevant insider threats with a high true positive rate from the business process. Even though some threats were missing and others were mitigated already, the majority of the threats were in line with the expectations the experts had.

Therefore, it can be stated that the contributions of the thesis are not only of academic value but could also be applied in practice. However, more testing with different business processes would be necessary to ensure the correctness of the mapping of the insider threats to the BPMN elements.

7.1 Future Work

As discussed before, one area of future work would be to transform the proposed methodology, so that no security expert is needed anymore. Therefore, a business process expert could use the tool in the design phase to not only build efficient processes but also construct them securely. A possibility could be for the user to answer a set of questions, which would also be processed by the prototype and conclude with results similar to an expert evaluating the threats proposed by the prototype now.

In addition, it would be beneficial for the mapping of the insider threats to the BPMN elements to apply the prototype to other business processes. With more data, the threat report is assumed to become more accurate and therefore, it would decrease the workload of a security expert.

Bibliography

- [1] J. Abulencia, “Insider attacks: Human-factors attacks and mitigation”, *Computer Fraud & Security*, vol. 2021, no. 5, pp. 14–17, 2021.
- [2] F. S. Nils Pfändler, “”wir angebot sputnik v”: Wie das geschäft mit den gefälschten impfpässen funktioniert”, *Neue Zürcher Zeitung*, 2021.
- [3] N. C. S. C. (NCSC), “Covid-19-certificate - public security test - current findings”, 2021, Last accessed: 19. October 2023. [Online]. Available: <https://www.ncsc.admin.ch/ncsc/de/home/aktuell/im-fokus/covid-zertifikat-pst.html>.
- [4] A. Duncan, S. Creese, and M. Goldsmith, “An overview of insider attacks in cloud computing”, *Concurrency and Computation: Practice and Experience*, vol. 27, no. 12, pp. 2964–2981, 2015.
- [5] I. Homoliak, F. Toffalini, J. Guarnizo, Y. Elovici, and M. Ochoa, “Insight into insiders and it: A survey of insider threat taxonomies, analysis, modeling, and countermeasures”, *ACM Comput. Surv.*, vol. 52, no. 2, Apr. 2019.
- [6] K. Brancik, *Insider computer fraud: an in-depth framework for detecting and defending against insider IT attacks*. CRC Press, 2007.
- [7] G. Magklaras and S. Furnell, “Insider threat prediction tool: Evaluating the probability of it misuse”, *Computers & Security*, vol. 21, no. 1, pp. 62–73, 2001.
- [8] L. Daubner, M. Macak, R. Matulevičius, B. Buhnova, S. Maksović, and T. Pitner, “Addressing insider attacks via forensic-ready risk management”, *Journal of Information Security and Applications*, vol. 73, p. 103 433, 2023.
- [9] D. M. Cappelli, A. P. Moore, and R. F. Trzeciak, *The CERT guide to insider threats: how to prevent, detect, and respond to information technology crimes (Theft, Sabotage, Fraud)*. Addison-Wesley, 2012.
- [10] C. Joshi, J. R. Aliaga, and D. R. Insua, “Insider threat modeling: An adversarial risk analysis approach”, *IEEE transactions on information forensics and security*, vol. 16, pp. 1131–1142, 2021.
- [11] A. P. Moore, K. A. Kennedy, and T. J. Dover, “Introduction to the special issue on insider threat modeling and simulation”, *Computational and Mathematical Organization Theory*, vol. 22, pp. 261–272, 2016.
- [12] T. Baluta, L. Ramapantulu, Y. M. Teo, and E.-C. Chang, “Modeling the effects of insider threats on cybersecurity of complex systems”, in *2017 Winter Simulation Conference (WSC)*, IEEE, 2017, pp. 4360–4371.

- [13] M. Zeng, C. Dian, and Y. Wei, “Risk assessment of insider threats based on ihfacbn”, *Sustainability (Basel, Switzerland)*, vol. 15, no. 1, p. 491, 2023.
- [14] F. Kammüller and C. W. Probst, “Modeling and verification of insider threats using logical analysis”, *IEEE systems journal*, vol. 11, no. 2, pp. 534–545, 2015.
- [15] N. A. Hashim, Z. Z. Abidin, A. Puvanasvaran, N. A. Zakaria, and R. Ahmad, “Risk assessment method for insider threats in cyber security: A review”, *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 11, 2018.
- [16] N. Nostro, A. Ceccarelli, A. Bondavalli, and F. Brancati, “Insider threat assessment: A model-based methodology”, *SIGOPS Oper. Syst. Rev.*, vol. 48, no. 2, pp. 3–12, 2014.
- [17] R. Chinchani, A. Iyer, H. Ngo, and S. Upadhyaya, “Towards a theory of insider threat assessment”, in *2005 International Conference on Dependable Systems and Networks (DSN’05)*, IEEE, 2005, pp. 108–117.
- [18] M. Bishop, H. M. Conboy, H. Phan, *et al.*, “Insider threat identification by process analysis”, in *2014 IEEE Security and Privacy Workshops*, IEEE, 2014, pp. 251–264.
- [19] V. Stavrou, M. Kandias, G. Karoulas, and D. Gritzalis, “Business process modeling for insider threat monitoring and handling”, in *Trust, Privacy, and Security in Digital Business*, 2014, pp. 119–131.
- [20] S. Zareen, A. Akram, and S. Ahmad Khan, “Security requirements engineering framework with bpmn 2.0.2 extension model for development of information systems”, *Applied Sciences*, vol. 10, no. 14, 2020.
- [21] M. Salnitri, F. Dalpiaz, and P. Giorgini, “Designing secure business processes with secbpmn”, *Software and systems modeling*, vol. 16, no. 3, pp. 737–757, 2017.
- [22] D. Granata, M. Rak, G. Salzillo, G. Di Guida, and S. Petrillo, “Automated threat modelling and risk analysis in e-government using bpmn”, *Connection Science*, vol. 35, no. 1, p. 2284645, 2023.
- [23] S. Hacks, R. Lagerström, and D. Ritter, “Towards automated attack simulations of bpmn-based processes”, in *2021 IEEE 25th International Enterprise Distributed Object Computing Conference (EDOC)*, IEEE, 2021, pp. 182–191.
- [24] N. Kolokotronis and S. Shiaeles, *Cyber-security threats, actors, and dynamic mitigation*. Boca Raton: CRC Press, 2021.
- [25] M. Tatam, B. Shanmugam, S. Azam, and K. Kannoorpatti, “A review of threat modelling approaches for apt-style attacks”, *Heliyon*, vol. 7, no. 1, 2021.
- [26] C. W. Probst, J. Huncker, M. Bishop, and D. Gollmann, *Insider threats in cyber security*. New York, NY: Springer Science & Business Media, 2010, vol. 49.
- [27] J. Huncker and C. W. Probst, “Insiders and insider threats-an overview of definitions and mitigation techniques.”, *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, vol. 2, no. 1, pp. 4–27, 2011.
- [28] M. Humayun, M. Niazi, N. Jhanjhi, M. Alshayeb, and S. Mahmood, “Cyber security threats and vulnerabilities: A systematic mapping study”, *Arabian journal for science and engineering (2011)*, vol. 45, no. 4, pp. 3171–3189, 2020.

- [29] P. G. Neumann, “Combatting insider threats”, in *Insider Threats in Cyber Security*, C. W. Probst, J. Hunker, D. Gollmann, and M. Bishop, Eds. Boston, MA: Springer US, 2010, pp. 17–44.
- [30] A. S. C. Junior and C. H. Arima, “Threat modeling: A study on its application in digital transformation from the perspective of risk”, *Revista de Gestão e Secretariado (Management and Administrative Professional Review)*, vol. 14, no. 1, pp. 1158–1169, Jan. 2023.
- [31] D. Granata and M. Rak, “Systematic analysis of automated threat modelling techniques: Comparison of open-source tools”, *Software quality journal*, 2023.
- [32] R. S. Aguilar-Savén, “Business process modelling: Review and framework”, *International Journal of Production Economics*, vol. 90, no. 2, pp. 129–149, 2004.
- [33] A. Suchenia, P. Wiśniewski, and A. Ligeza, “Overview of verification tools for business process models”, Sep. 2017, pp. 295–302.
- [34] R. Flowers and C. Edeki, “Business process modeling notation”, *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 3, pp. 35–40, 2013.
- [35] G. Aagesen and J. Krogstie, “Bpmn 2.0 for modeling business processes”, in *Handbook on Business Process Management 1*, ser. International Handbooks on Information Systems, Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 219–250.
- [36] J. Gopfert and H. Lindenbach, *Geschäftsprozessmodellierung mit BPMN 2.0: Business Process Model and Notation*. Germany: De Gruyter, 2014.
- [37] MediaWiki, Last accessed: 14. January 2024. [Online]. Available: <http://mlwiki.org/index.php/BPMN>.
- [38] P. Kirvan, Last accessed: 14. January 2024. [Online]. Available: <https://www.techtarget.com/searchsecurity/tip/Top-threat-modeling-tools-plus-features-to-look-for>.
- [39] C. Allen-Addy, Last accessed: 14. January 2024. [Online]. Available: <https://www.irusrisk.com/resources-blog/11-recommended-threat-modeling-tools11-recommended-threat-modeling-tools>.
- [40] IRIUSRISK, S.L., Last accessed: 14. January 2024. [Online]. Available: <https://www.irusrisk.com/threat-modeling-platform>.
- [41] Microsoft, Last accessed: 14. January 2024. [Online]. Available: <https://learn.microsoft.com/en-us/azure/security/develop/threat-modeling-tool>.
- [42] OWASP Foundation, Inc, Last accessed: 14. January 2024. [Online]. Available: <https://owasp.org/www-project-threat-dragon/>.
- [43] Security Compass, Last accessed: 14. January 2024. [Online]. Available: <https://www.securitycompass.com/sdelements/>.
- [44] C. Schneider, Last accessed: 14. January 2024. [Online]. Available: <https://threagile.io/>.
- [45] G. Magklaras and S. Furnell, “Insider threat specification as a threat mitigation technique”, in *Insider threats in cyber security*, Springer, 2010, pp. 219–244.

- [46] B. A. L, “Information security insider threats in organizations and mitigation techniques”, in *2019 International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC)*, IEEE, 2019, pp. 1–4.
- [47] E. Wheeler, *Security risk management: Building an information security risk management program from the Ground Up*. Elsevier, 2011.
- [48] W. Stallings, *Network security essentials: applications and standards*. Pearson, 2016.
- [49] M. A. Kuhail and S. Lauesen, “User story quality in practice: A case study”, *Software*, vol. 1, no. 3, pp. 223–243, 2022.
- [50] Camunda Services GmbH, Last accessed: 6. December 2023. [Online]. Available: <https://bpmn.io/>.
- [51] Camunda Services GmbH, Last accessed: 22. December 2023. [Online]. Available: <https://github.com/bpmn-io/bpmn-js>.
- [52] Camunda Services GmbH, Last accessed: 22. December 2023. [Online]. Available: <https://github.com/bpmn-io/bpmn-js-examples>.
- [53] S. Surve, Last accessed: 10. January 2024. [Online]. Available: <https://www.freecodecamp.org/news/why-use-react-for-web-development/>.
- [54] RAPID Crews, Last accessed: 10. January 2024. [Online]. Available: <https://rapidcrews.com/10-benefits-of-web-based-applications-systems/>.
- [55] D. Muracciole, Last accessed: 22. December 2023. [Online]. Available: <https://react-pdf.org/>.
- [56] C. Hudon, M.-C. Chouinard, M. Bisson, *et al.*, “Case study with a participatory approach: Rethinking pragmatics of stakeholder engagement for implementation research”, *The Annals of Family Medicine*, vol. 19, no. 6, pp. 540–546, 2021.
- [57] IGS GmbH, Last accessed: 31. December 2023. [Online]. Available: <https://www.igs-gmbh.ch/>.

Abbreviations

ADVISE	ADversary VIEw Security Evaluation
BPMN	Business Process Modeling and Notation
CIA	Confidentiality, Integrity and Availability
CISO	Chief Information Security Officer
CRAMM	The Central Risk Analysis and Management Method
DDoS	Distributed Denial of Service
DMN	Decision Model and Notation
DoS	Denial of Service
FRAP	Facilitated Risk Assessment Process
HOL	Higher Order Logic
IGS GmbH	Informatikgesellschaft für Sozialversicherungen GmbH
IHFACS	Improved Human Factors Analysis and Classification System
NIST	National Institute of Standards and Technology
NP-hard	non-deterministic polynomial-time hard
OCTAVE	Operational Critical, Threat, Asset and Vulnerability Evaluation
SecureInT	Securing Insider Threats
SecBPMN-ml	SecBPMN modeling language
SecBPMN-Q	SecBPMN query language
SQUARE	Software Quality Requirements Engineering
TRIP	Tailored Risk Integration Process
UI	User Interface

Glossary

Authorization Authorization is the decision of whether an entity is allowed to perform a particular action, e.g., whether a user can attach to a network or not.

Framework A framework is the *overview of interlinked items which supports a particular approach to a specific object* [24].

Methodology A methodology is a *”procedure, protocol, and technique for acquiring and analyzing research data”* [25].

Taxonomy A taxonomy is the *”effort of naming, defining and classifying a threat”* [24].

List of Figures

2.1	Elements of BPMN [33]	9
2.2	Overview of Different Types of Tasks in BPMN	10
2.3	Events in BPMN [37]	10
2.4	Artifacts in BPMN	11
4.1	Insider Threats Found in the Literature per Source	22
4.2	Threats Found per Security Principle	23
4.3	Visualization of Methodology	28
5.1	Define Security Requirements (Step 1)	32
5.2	Investigate List of Threats (Step 4)	33
5.3	Select Important Elements (Step 5)	33
5.4	Report Page (Step 6)	34
5.5	Elements in PDF Report	35
5.6	Threats in PDF Report	35
5.7	Entity Relationship Diagram Insider Threat Database	39
5.8	Venn Diagram of Insider Threats	41
6.1	Example Business Process Provided by IGS GmbH	45
6.2	Number of Threats/Elements per Security Principle	49
6.3	Number of Threats Perceived as Relevant	53

List of Tables

2.1	Comparison of Insider and Outsider Threats [29]	8
3.1	Overview of the Related Work	19
4.1	Insider Threat Database	24
5.1	Mapping Insider Threats to BPMN Elements	39
6.1	Threat Report of Case Study	48
B.1	Insider Threat Database With Threats from the Literature	75

Appendix A

Installation Guidelines

To start the prototype locally, please refer to the README.md in the project. It can be found in the folder of the source code.

Appendix B

Insider Threat Database

The following database contains all insider threats found in the literature. They are categorized by the security principle and grouped together as described in Chapter 4.

Table B.1: Insider Threat Database With Threats from the Literature

Security principle	Insider threat group	Specific Threats in Literature
Confidentiality	Confidential data acquisition	inappropriate acquisition of data [6] inappropriate use of confidential data [6] information theft [45] session hijacking [6] exploitation of web services where identities are being hijacked and data is being stolen [6] attacks on session-dependent information [6] tampering with URL query strings [6]
	Confidential data view	inappropriate viewing of data outside normal usage [6] view confidential data [16]
	Confidential data transfer	transfer confidential files [16] national security leaks and other disclosures [29] illegal distribution and transmission of acquired financial information to outsiders [6] suspicious attachments (e.g. password files) [45] illegal data transaction [6] mail to suspicious addresses (e.g. large number of recipients) [45]
	Unauthorized access to credentials	ease of access to a computer system or application entry due to security weaknesses of an older version of an application [6] access to crypto keys [16], [29] access to live master file [6]

Security principle	Insider threat group	Specific Threats in Literature
Integrity	Data corruption	input manipulation (bypass normal data input controls) [6] computer data manipulation within an application or system [6] fudging control totals [6] command execution attacks (SQL database calls to change database) [6] fraudulent modification of vital information [45] corrupt data [16] breakage (siphoning off small sums from numerous sources) [6] cookie tampering [6] attempt to manipulate cookies to "spoof" server-side authentication mechanism [6] cookie content manipulation due to the absence of encryption of the cookie [6]
	Malicious code modification	software code modification: "Logic Bomb" [6] software code modification: "Trojan Horse" [6] putting Trojan horses [16] malcode software injection [6] self-dealing transaction to capitalize on the destruction created by the software code modification [6] bufferoverflow: attacker can cause web application to execute malicious code that is designed to take over a system [6] malicious content: spread viruses / Trojan horse programs within valid XML messages [6] HTML injection (cross-site scripting) [6] inserting trojan horses or trapdoors in trusted (and untrusted) components [29]
	Malware installation	use of unauthorized programs [6] Trojan horse [46] possibility of downloading Trojan horse [45] download illegal software [45] download offensive material [45] install vulnerable supporting software [16] exploitation of operating system vulnerabilities [6]
	System control manipulation	altering audit logs [29] altering audit trails and logs [16] inadequate network journaling for forensic purposes [6] compromising misuse detection [6] deliberate misreporting [6] disabling protection of components [16] disabling of protected components [29] altering configurations, schedules, and priorities [29] modification of default configuration [45]

Security principle	Insider threat group	Specific Threats in Literature
Availability	Hardware attack	hardware destruction [6] use of defective hardware [16] hardware malfunction [6] removal or addition of hardware components [45] resource theft [29]
	Resource exhaustion attack	damage to system availability (i.e. DDoS attack) [6] denial of service: lack of use of any load testing tools to generate web traffic [6] denial of service (DoS) [45] distributed denial of service (DDoS) [46] buffer overflow attack [45] exhaustion of protected resources [29] replay attack [6] coercive parsing [6]
	Network exhaustion attack	downloading large amounts of data in a small time period [45] oversize payloads [6] using over a certain number of network endpoints [45] using a large network burst (throughput) rate [45]
	Data deletion	terminate user session in a seemingly random way, causing loss of data [6] computer data destruction within an application or system [6]
Accountability	System control circumvention	disable system logs [16] circumvention of security controls [6] undocumented transaction codes [6] hacking beneath the audit trail [29]
	Unauthorized privilege elevation	inappropriate account provisioning [6] improper user management [16] computer access level modification [6] elevate user privileges [16]
	Misuse of privileges	misuse of adjustment transaction [6] misuse of error-correction procedures [6] misuse of intended authority by over-authorized user [29] running a covert business [29] insider trading [29] extraneous transactions [6] usurpation of superuser access and root keys [29] privileged manipulation of access controls [29]

Security principle	Insider threat group	Specific Threats in Literature
Authenticity	Social engineering attack	tailgating [46] ingratiation [46] phishing [46] pretexting [46] baiting [46]
	Impersonation attack	masquerading as an employee [6] masquerader [46] employee impersonation and transmission of unauthorized e-mails to corporate clients [6] misuse of system's capabilities (impersonation of another insider to send threatening emails to another insider) [6]
	Man-in-the-middle attack	man-in-the middle attacks [6]

Appendix C

Supplementary Content

Source Code

Proposal Letter

Example Process (BPMN and SVG)

Evaluation Documentation

Evaluation Output (BPMN, SVG, and PDF)

Evaluation Discussion Slides