UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Conversion Master in IT Curriculum

Madis Valk

Security Risk Management in Auditing Processes

Master's Thesis (15 ECTS)

Supervisor: Raimundas Matulevičius, PhD

Tartu 2023

Security Risk Management in Auditing Processes

Abstract:
Financial auditing processes manage a wealth of confidential data from various stakeholders, making it imperative to ensure the security of this information to prevent unauthorised access, leaks, or misuse that may result in severe consequences for both the auditing organisation and its clients. Centralised systems, traditionally employed in these processes, are susceptible to various security risks, including unauthorised access, data misuse, and privacy breaches. This thesis examines traditional, centralised tools and blockchain technology in the context of security risk management for audit processes. The analysis of the traditional, centralised approach focuses on identifying valuable business assets and applying security risk-oriented patterns to identify security risks and derive security requirements. Possible solutions to mitigate the security risks identified in the centralised design are also discussed. Blockchain technology, a decentralised and transparent system, offers potential benefits in enhancing the security of financial auditing processes. However, its limitations, such as confidentiality and scalability, necessitate exploring permissioned blockchains as a viable solution for securing sensitive audit information. Therefore, this study investigates the R3 Corda platform, a permissioned blockchain, as a potential solution for managing security risks in audit processes. This research shows that implementing the R3 Corda platform in the financial auditing process, specifically for receiving information and documents from clients, can offer valuable insights into the impact of blockchain technology on security risks. The analysis reveals that the Corda platform provides enhanced data integrity, traceability, and availability compared to traditional centralised systems, while also addressing the confidentiality requirements of sensitive audit information. This thesis demonstrates that the implementation of the Corda platform in the auditing process results in improved security measures and risk mitigation. Furthermore, comparing centralised and blockchain-based countermeasures provides a deeper understanding of suitable approaches for securing audit information. The findings contribute to the ongoing discourse around the practical implementation of blockchain technology in financial auditing processes and security risk management. This knowledge can help stakeholders make informed decisions when considering implementing blockchain technology in the context of financial auditing and security risk management, offering a secure and reliable alternative to traditional centralised systems.

Keywords:
security risk management, auditing process, security risk-oriented pattern, security requirement, Corda, blockchain

CERCS: T120 - Systems engineering, computer technology

Auditiprotsesside turvariskide juhtimine

Lühikokkuvõte:

Finantsauditi protsessid haldavad hulgaliselt konfidentsiaalseid andmeid erinevatelt sidusrühmadelt, mistõttu on hädavajalik tagada selle teabe turvalisus, et vältida volitamata juurdepääsu, lekkeid või väärkasutust, mis võib põhjustada tõsiseid tagajärgi nii auditeerivale organisatsioonile kui ka selle klientidele. Tavapäraselt kasutatakse nendes protsessides tsentraliseeritud süsteeme, mis on vastuvõtlikud erinevatele turvariskidele, sealhulgas volitamata juurdepääsule, andmete väärkasutamisele ja privaatsuse rikkumisele. Käesolev magistritöö käsitleb traditsioonilisi tsentraliseeritud tööriistu ja plokiahela tehnoloogiat finantsauditi protsesside turvariskide juhtimise kontekstis. Traditsioonilise, tsentraliseeritud lähenemisviisi analüüs keskendub väärtuslike ärivarade tuvastamisele ja turvariskidele orienteeritud mustrite rakendamisele turvariskide tuvastamiseks ja turbenõuete tuletamiseks. Arutletakse ka võimalike lahenduste üle tsentraliseeritud disainis tuvastatud turvariskide maandamiseks. Plokiahela tehnoloogia, mis on detsentraliseeritud ja läbipaistev süsteem, pakub potentsiaalseid eeliseid finantsauditi protsesside turvalisuse suurendamisel. Kuid selle piirangud, nagu konfidentsiaalsus ja skaleeritavus, tingivad vajaduse uurida loalisi plokiahelaid kui tundliku auditiinfo kaitsmiseks sobivat lahendust. Seetõttu uurib see magistritöö R3 Corda platvormi, mis on loaline plokiahel, kui potentsiaalset lahendust auditi protsesside turvariskide haldamiseks. See uurimustöö näitab, et R3 Corda platvormi rakendamine finantsauditi protsessis, klientidelt teabe ja dokumentide vastuvõtmiseks, pakub teadmisi plokiahela tehnoloogia mõjust turvariskidele. Analüüsist selgub, et Corda platvorm tagab andmete tervikluse, jälgitavuse ja käideldavuse, järgides samal ajal ka tundliku audititeabe konfidentsiaalsusnõudeid. See magistritöö näitab, et Corda platvormi rakendamine auditi protsessis tagab turvameetmed ja riskide maandamise. Lisaks annab tsentraliseeritud ja plokiahelapõhiste vastumeetmete võrdlemine arusaama audititeabe turvalisuse seisukohast sobivast lähenemisviisist. Tulemused aitavad kaasa käimasolevale diskussioonile plokiahela tehnoloogia praktilisest rakendamisest finantsauditi protsessides ja turvariskide juhtimises. Need teadmised aitavad sidusrühmadel teha informeeritud otsuseid, kui nad kaaluvad plokiahela tehnoloogia rakendamist finantsauditi ja turvariskide juhtimise kontekstis, pakkudes turvalist ja usaldusväärset alternatiivi traditsioonilistele tsentraliseeritud süsteemidele.

Võtmesõnad:
turvariski juhtimine, auditi protsess, turvariskile orienteeritud muster, turvanõue, Corda, plokiahel

CERCS: T120 - Süsteemitehnoloogia, arvutitehnoloogia

Contents

1 Introduction **5**

2 Background **8**
2.1 Auditing Process and Systems 8
2.2 Security Risk Management 10
2.3 Blockchain-Based Applications 13
2.4 Related Work 17
2.5 Summary 18

3 E-dok-based Auditing System **21**
3.1 Case Description 21
3.2 Security Requirements Elicitation 30
3.2.1 Pattern One: Securing Data From Unauthorised Access 30
3.2.2 Pattern Two: Securing Data That Flows Between Business Entities 33
3.2.3 Pattern Three: Securing Business Activity After Data Is Submitted 36
3.2.4 Pattern Four: Securing Business Service Against DoS Attacks . 38
3.2.5 Pattern Five: Securing Data Stored In/Retrieved From The Data Store 40
3.3 Summary 42

4 Blockchain-based Auditing System **46**
4.1 Blockchain Platforms 46
4.2 Selecting Blockchain Platform for Auditing System 51
4.3 Corda for E-dok 54
4.4 Security Risk Management of Corda-Based E-dok System 59
4.4.1 Corda Platform for Security Threat Mitigation 59
4.4.2 Security Risks in Corda-based Input Folder and their Mitigation 62
4.5 Summary 64

5 Concluding Remarks **66**
5.1 Limitations 66
5.2 Answers to Research Questions 66
5.3 Conclusion 67
5.4 Future Work 68

References **69**

Appendices **82**
I. Security Requirements 82
II. Licence 87

1 Introduction

Security risk management in financial auditing processes encompasses identifying, assessing, and mitigating potential threats that could compromise the confidentiality, integrity, and availability [34, 103] of sensitive information exchanged during the auditing process. As auditing organisations manage confidential data from various stakeholders, such as client companies, their subsidiaries, suppliers, customers, banks, advisors, and other auditors, it is imperative to ensure the security of this information to prevent unauthorised access, leaks, or misuse that may result in severe consequences for both the auditing organisation and its clients.

The repercussions of inadequate security in auditing processes cannot be understated. Unauthorised access to or misuse of sensitive audit information can lead to various adverse outcomes. For instance, client data could be sold to competitors, giving them an unfair advantage in the market. Moreover, the reputation of auditing firms and their clients may suffer significant, long-lasting damage, potentially affecting their ability to attract new business and retain existing clients. Finally, in extreme cases, the financial burden of claims for damages stemming from security breaches could result in the bankruptcy of the auditing firm. In light of these potential consequences, this thesis explores the application of blockchain technology as a viable solution to enhance the security of auditing systems, aiming to mitigate the risks associated with traditional centralised systems and protect the interests of all stakeholders involved in the auditing process.

The significance of this research lies in need to secure the auditing process in the face of emerging and evolving security risks. Financial auditing processes are exposed to a variety of threats, including unauthorised system access, man-in-the-middle attacks, data misuse or modification, malicious script submissions, denial-of-service (DoS) attacks, and data privacy breaches [64]. These risks necessitate a comprehensive understanding and the development of robust countermeasures to protect sensitive audit information, maintain stakeholder trust, and uphold the reputation of auditing firms and their clients.

This thesis delves into security risk management in financial auditing processes, emphasising recognising and mitigating the risks associated with traditional centralised systems. The research includes an audit scenario analysis based on E-dok, an audit software developed by the Estonian Auditors' Association, an examination of five security risk-oriented patterns [4, 64], and an analysis of traditional risk engineerings methods, such as centralised technology solutions and countermeasures. In addition, the thesis explores the possibility of embracing a decentralised approach by implementing blockchain technology, specifically the R3 Corda platform [28], weighing its advantages and limitations compared to traditional systems. By transferring part of the audit process—receiving information and documents from the client—to the Corda platform, the research offers insights into the impact of blockchain technology on security risks. Furthermore, it provides a practical perspective on its potential for enhancing the security

of financial auditing processes.

In recent years, blockchain technology has emerged as a promising solution for enhancing security in various domains [70], including financial auditing processes [95]. As a decentralised, tamperproof, and transparent system, blockchain offers numerous advantages [121], such as improved data integrity, traceability, and availability. However, blockchain technology has inherent limitations [121], including potential challenges related to confidentiality, scalability, and resource consumption. To address this issue, the thesis investigates the potential use of permissioned blockchains, such as the R3 Corda platform [28], that could provide confidentiality for sensitive audit information [115]. In addition, this thesis explores the potential of using blockchain to manage security risks in audit processes, assessing its potential as a robust solution for securing audit information.

This research adopts a comparative approach to examine the traditional centralised technology solutions and blockchain-based approaches in mitigating the identified security risks in financial auditing processes. By analysing the case and assessing the security requirements, this thesis aims to offer insights into the impact of implementing blockchain technology on the security risks of the audit process. The comparison between centralised and decentralised countermeasures facilitates an understanding of suitable approaches for securing audit information.

Furthermore, the thesis contributes to the ongoing discourse around the practical implementation of blockchain technology in financial auditing processes [95, 117]. This knowledge can help stakeholders make informed decisions when considering implementing blockchain technology in the context of financial auditing and security risk management.

The thesis's main research question is: **How to manage security risks in the auditing processes?** The main research question breaks down into the following research questions (RQ):

RQ 1: What is the current state of securing auditing processes?

It examines audit processes and systems, security risk management approaches, blockchain technology, and the research carried out so far.

RQ 2: What are the security requirements for the audit process?

A scenario analysis will be presented to examine assets, centralised process risks, security requirements and mitigation tools.

RQ 3: How does blockchain help to avoid security risks in auditing processes?

A new design of the process presented in the previous question is given and examined in light of the risks identified in the previous question.

This thesis is structured into five chapters to provide a comprehensive understanding of the topic. Chapter 2 presents an overview of the financial audit process and systems, security risk management, blockchain-based applications, and related work. Chapter 3 delves into an audit case based on E-dok, discusses its conceptual model and business process, applies the security requirements elicitation from the business process (SREBP)

method [4, 64] to derive security requirements, and identifies security risks. The chapter also suggests possible solutions to the security risks identified in traditional, centralised design. Chapter 4 focuses on analysing blockchain platforms, the feasibility of implementing blockchain technology, specifically the R3 Corda platform, and selecting an appropriate platform for the audit system. This chapter also demonstrates the migration of a part of the audit process to the Corda platform, further analyses the impact of this migration on the identified security risks of the audit system, and compares centralised and blockchain-based countermeasures. Additionally, it examines the security risks arising from migrating the audit system to the Corda platform and discusses possible mitigation options. Finally, Chapter 5 offers concluding remarks.

2 Background

This chapter concentrates on the research question **RQ 1, What is the current state of securing auditing processes?** which is further divided into four sub-questions to answer it better:

RQ 1.1: What are the auditing processes and systems?

RQ 1.2: What are the security risk management approaches?

RQ 1.3: What is blockchain technology that could secure auditing processes?

RQ 1.4: What are the current solutions to mitigate security risks by applying blockchain technology?

The first section explains audit processes and systems, the second section security risk management, the third section blockchain technology and its applications, and the fourth section gives insight into related research.

2.1 Auditing Process and Systems

The objective of a financial audit is to obtain reasonable assurance to allow the auditor express opinion, whether the audited financial statements present fairly, in all material respects, the financial position of the company, and its financial performance and its cash flows for the reporting period in accordance with the applicable reporting standards [87]. The audit process, which is based on the Standards for the Professional Practice of the Sworn Auditor, consists of several stages, including pre-engagement activities, risk assessment, risk response, summarising and reporting. Pre-engagement activities include client and engagement acceptance procedures, agreeing on the terms and conditions of the engagement and entering into a client agreement. In the risk assessment phase, an understanding of the entity and its environment, the client's internal control system, the accounting process and the accounting framework is gained; an analytical review is performed, risks are assessed, materiality levels are determined, and responses to risks are planned, i.e., the audit approach and the procedures to be performed during the audit, i.e., the audit plan is prepared. The planned procedures are carried out during the risk response phase. The stage includes checking internal controls and their effectiveness, performing substantive procedures, evaluating the client's annual report compliance with presentation requirements of the financial reporting framework, and evaluating identified errors. Finally, during the audit summary and reporting phase, audit procedures of subsequent events are performed, management representations are obtained, an independent auditor's report is issued, and, if necessary, a memorandum to management is issued.

In order to obtain reasonable assurance, sufficient appropriate evidence [88] is gathered through all stages and various procedures of the audit. In doing so, a significant amount of documentary evidence is collected from different persons and parties. The information obtained may be of different quality, including consistent or not, as well as

substantially consistent or inconsistent with previously known or acquired documentation. The resulting documentation may contain misstatements detected by the submitter and corrected by sending the corrected document. It is also possible that the auditor will identify the misstatement during the audit, and the person providing the information will decide whether or not to correct it. In the latter case, if not trivial, it is written to the audit summary as an uncorrected error.

The parties from whom information is obtained can vary, from the client's management and accountants to sales and warehousing staff, as well as the client's banks, legal advisers, suppliers and customers, also from the auditors of the client's subsidiaries and others. In Estonian law, an auditor is not entitled to request information about a company from third parties on behalf of the company. Instead, the auditor makes such requests for information to the client, who in turn forwards them to the addressees of the requests. However, the auditor needs to know who responded to the inquiry and whether it is the party from whom the auditor sought the information. This is not complicated to identify in Estonia, as it is possible to issue digitally stamped or encrypted documents with the auditor's identification code, the senders of which are known or can be verified. The situation may be complicated with documents received from other countries. In some cases, International Standards on Auditing require that inquiries be made under the auditor's control [89]. This is to ensure that the request is made and the response is received from the party from whom the auditor needs information or confirmation for the audit evidence.

Audit systems are computer programs that assist auditors in carrying out their work. From small systems designed for small and medium-sized businesses to large enterprise systems used for multinational organisations, many options of them are available. Audit software is the environment to document an auditor's engagements; record, systematise and process the evidence collected; archive and maintain a completed engagement file for the period required by regulations. These systems often automate many of the tasks involved in the audit process, such as data collection, analysis, and reporting. In addition to assisting in the documentation of engagements following an applicable methodology that complies with standards for the professional practice of sworn auditor, they also assist in the systematisation of the work performed and the evidence obtained; audit systems typically include several key functionalities, such as engagement project management, data extraction and analysis, sampling for testing, and reporting. Data extraction and analysis functionality allows auditors to collect and analyse large amounts of financial data quickly and accurately. However, such a system is time-consuming to set up, so it is not yet practical to use it for rapidly evolving small and medium-sized customers whose ERP software is adapted to the entity's needs from year to year. Therefore, in many cases, the same set-up cannot be used next year without significant changes. Sampling functionality helps auditors to compile samples of the data to be checked. Finally, reporting functionality enables auditors to communicate the audit results to management

and stakeholders.

Until now, in small and medium-sized practices, obtaining documentary information has been solved either by giving the auditor access to the client's ERP software or relevant databases or receiving information via e-mail, the cloud server folder accessed by the auditor and client's representative or a particular audit software folder if the audit software has such functionality, or on a separate data carrier (e.g., flash drive, external hard drive). The larger the number of information providers, the more difficult or labour-intensive it is to use such solutions. In practice, obtaining information from third parties, especially where it is required to be under the auditor's control, is done by e-mail. However, the audit team must verify the consistency of all information received using appropriate procedures.

The collection, systematisation and processing of information, including documents, is one of the main functions of the audit. For example, in the audit software E-dok created by the Estonian Auditors' Association, for each client, there is one folder (input folder) in which the auditor can grant access to different users who can then submit information and documents in different formats into that folder. Anyone who has access to this folder will be able to see all the files stored there and download them. In practice, due to privacy and functionality reasons, auditing offices often use for collecting information alternative solutions for E-dok input folders, such as cloud solutions. Regardless of how the information is collected, the information used in the engagement is stored in the engagement file as the procedures are performed.

The problem with the functionality of collecting documentary evidence provided by external parties includes: (i) determining which queries have been answered, (ii) the chronological order in which each specific query has been answered, assuming the most recent response should be followed by the auditor. Historical information may be required to document and analyse changes made during the audit; (iii) to determine that responses to inquiries that need to be made under the auditor's control have been obtained from relevant parties/entities from whom information is required. The information provider is whom we expect to be, not someone else, such as the audited entity itself. These questions need to be answered during the audit.

2.2 Security Risk Management

The security risk management process is critical to ensuring financial audit systems' integrity, confidentiality and availability. This section covers several important concepts that organisations need to understand, including information security, security, security risk, risk management, and security risk management, and their interrelationships, in order to develop effective security risk management processes for their systems.

Information security is concerned with safeguarding information and information systems from unauthorised access, use, modification, disruption, destruction, or disclosure to provide availability, integrity, and confidentiality [34, 103]. In information systems, secu-

rity encompasses protecting system components, including hardware, software, and data, against unauthorised access, use, disclosure, disruption, modification, or destruction [59]. Security measures [59] can be classified into three categories: physical, technical, and administrative controls. Security risk is defined as the potential for loss, harm, or damage to an information system due to a threat event that exploits a vulnerability [34, 66], with various sources and potential consequences for organisations.

Risk management is a systematic process of identifying, assessing, and prioritising risks, followed by coordinating and applying resources to minimise, monitor, and control the probability and impact of adverse events [102]. Security risk management is the application of risk management principles and practices to address the security risks associated with information systems [105, 66], involving identifying, analysing, evaluating, treating, and monitoring security risks to minimise the likelihood and impact of security incidents [105].

The Information Systems Security Risk Management (ISSRM) domain model [4, 34, 64, 66] is a comprehensive conceptual framework that integrates key concepts, relationships, and definitions related to security risk management. Developed by surveying security-related standards, risk management standards, and risk management methods [4, 34, 64, 66], the ISSRM domain model aims to provide a common understanding of domain terminology for stakeholders involved in managing security risks.

The ISSRM domain model is built on three main sets of concepts: asset-related, risk-related, and risk treatment-related. Asset-related concepts [64, 66] focus on identifying valuable assets that need protection and establishing security criteria for these assets. Assets can be categorised as business or information system (IS) assets. Business assets describe the capabilities, skills, processes, and information essential to the business to achieve its core objectives, while IS assets are components or parts of an information system that support business assets, such as software, hardware, networks, or even personnel and facilities that play a role in the system's security. Security criteria describe the security needs applicable to business assets, typically confidentiality, integrity, and availability [34, 103]. Confidentiality refers to preventing unauthorised access to sensitive information, ensuring that only authorised parties can access the data. Integrity involves maintaining the accuracy and consistency of information and systems and preventing unauthorised modification or tampering. Finally, availability ensures that information and systems are accessible and usable by authorised parties when needed.

Risk-related concepts [64, 66] introduce definitions of risk, its primary components, and their relationships. A security risk is a potential for loss, harm, or damage to an information system due to a threat event that exploits a vulnerability. Threats are potential attacks or incidents initiated by threat agents who use attack methods to target one or more information system (IS) assets by exploiting their vulnerabilities [4, 34, 64, 66]. On the other hand, vulnerabilities refer to the characteristics of IS assets that expose weaknesses or flaws, making them susceptible to being exploited by threats [34, 52, 64,

66]. The combination of a threat and vulnerabilities constitutes a risk event, while the impact represents the consequence of the risk. Security risks can arise from various sources, including natural disasters, human error, malicious insiders, cyber criminals, and nation-state actors. These risks can have significant consequences for organisations, including financial losses, damage to reputation, legal liabilities, and regulatory penalties.

Risk treatment-related concepts [64, 66] describe the decisions, requirements, and controls to be defined and implemented to mitigate risks. There are four categories of risk treatment decisions: risk avoidance, risk reduction, risk transfer, and risk retention. Security requirements are environmental conditions that should be met to mitigate risks, contributing to covering one or more risk treatments for the target system. Finally, controls (safeguards or countermeasures) are designed means to improve security by implementing these security requirements.

The ISSRM domain model is centred around a process [64, 66] that describes activities to perform for effective security risk management. This process consists of several steps: studying the organisation's context and identifying its assets, determining security objectives, analysing risks, making risk treatment decisions, eliciting security requirements, and implementing security controls. The ISSRM process is iterative, with several iterations until an acceptable risk level is reached. Therefore, regular reviews and continuous monitoring of risks are necessary to maintain the required security level.

By understanding and applying the ISSRM domain model and its central concepts, organisations can develop effective security risk management processes for their information systems. Emphasis on assets, risks, threats, vulnerabilities, and security requirements is important for identifying and addressing potential security risks, selecting security countermeasures, and ensuring compliance with regulatory requirements and industry standards.

In the realm of security risk management, several standards have been established to guide organisations in developing, implementing, and maintaining robust and effective risk management systems. One such standard is ISO 31000, which provides a universally applicable framework for managing risks across various industries and sectors [102]. The standard is focused on the principles, framework, and process of risk management, emphasising the need for a structured and systematic approach to ensure that organisations can effectively identify, assess, and treat risks [102].

Another set of standards related to security risk management is the ISO/IEC 27000 family of standards, which focus specifically on information security management systems (ISMS) [103]. This family of standards comprises numerous guidelines and requirements that address various aspects of information security, such as risk assessment, risk treatment, and the implementation of security controls [104]. The ISO/IEC 27001 standard, in particular, establishes the requirements for an ISMS, providing a systematic approach to managing sensitive information and ensuring its confidentiality, integrity, and availability [104].

In addition to these international standards, regional standards can shape security risk management practices. The Estonian Information Security Standard (E-ITS) is an example of a regional standard that addresses information security, taking into account Estonian legal framework while being based on the German BSI IT-Grundschutz baseline security method [7, 98]. E-ITS is designed to be compatible with the ISO/IEC 27001 standard requirements, ensuring comprehensive protection of business processes and information systems used for public tasks while achieving a consistent level of information security across all components [7]. Furthermore, the standard encompasses various aspects of information security management, including requirements and guidelines for implementing and maintaining an ISMS, risk management processes for both typical and atypical target objects, and the provision of external assurance through auditing and certification processes [7].

In conclusion, the ISO 31000 standard, the ISO/IEC 27000 family of standards, and the E-ITS standard all contribute to the field of security risk management by providing organisations with a comprehensive set of principles, frameworks, and processes to effectively manage risks and ensure the security of their information assets.

2.3 Blockchain-Based Applications

Blockchain technology has attracted considerable attention in recent years due to its potential to transform various industries. Initially developed as the underlying technology for cryptocurrencies such as Bitcoin [74], blockchain's decentralised and immutable nature has expanded its applications beyond finance to various domains. This section provides an overview of blockchain technology, different platforms, and blockchain-based applications, illustrating the possibilities for transforming industries and addressing real-world challenges.

Blockchain is a decentralised, distributed ledger technology that enables secure and transparent recording of transactions across a peer-to-peer network of nodes [74]. The technology functions through a consensus mechanism that validates transactions, which are then grouped into blocks and added to the chain in a linear and chronological order [74]. In addition, cryptographic techniques ensure the immutability and security of the data, making it resistant to tampering and fraud [121].

The blockchain network comprises multiple nodes, each maintaining a copy of the ledger [41]. This distributed architecture ensures data redundancy and prevents any single point of failure, contributing to the overall robustness and resilience of the system [41]. Furthermore, transactions are validated through a consensus mechanism, such as proof-of-work [74] or proof-of-stake, which involve nodes competing to solve complex mathematical problems or committing a certain amount of a native cryptocurrency as collateral, respectively [41].

As Xu et al. [121] identified, the five fundamental properties of blockchain technology set it apart from conventional centralised databases and computational platforms. These

properties – immutability, non-repudiation, integrity, transparency, and equal rights – contribute to establishing trust within the blockchain network by enabling secure, transparent, and auditable interactions between participating nodes.

Immutability ensures that once a transaction is added to the blockchain, it becomes virtually impossible to alter, enhancing the record-keeping system's security and reliability. Non-repudiation provides assurance that committed transactions cannot be denied by any party involved, further strengthening trust and accountability within the network. Data integrity is maintained through cryptographic tools, such as hashing and digital signatures, which protect the data from unauthorised modifications or tampering. Transparency, an essential aspect of public blockchain technology, allows every node in the network to access the entire transaction history, fostering trust between parties and enabling real-time tracking and auditing of records. Lastly, equal rights ensure that every participant can access and manipulate the blockchain based on their computational power or stake, promoting cooperation and preventing an undue concentration of power among nodes.

Blockchain technology can be categorised based on its accessibility and the governance model employed. Typically, blockchains are divided into two primary classes: public and private [21, 36]. However, these classifications can be further refined into permissionless and permissioned blockchains [118, 121].

Public blockchains, synonymous with permissionless blockchains [41], are open and decentralised networks that allow any participant to join the consensus process and validate transactions without restrictions [82]. Examples of public blockchains include Bitcoin and Ethereum, which facilitate peer-to-peer transactions of digital assets and enable the development of decentralised applications (dApps), respectively [20, 74].

Private blockchains are controlled by one organisation [41]. These limit access to a specific group of trusted participants [61], offering greater control, privacy, and scalability compared to public blockchains [21]. Private blockchains are permissioned; however, the rights of different users may vary [41]. Permissioned private blockchains require participants to authenticate their identity and grant transaction visibility only to the involved entities. Notable examples of permissioned private blockchains [70, 121] are Hyperledger[1], MultiChain[2], R3 Corda[3], and Ripple[4].

Public permissioned blockchain networks, as outlined by Ruiz [93], have surfaced as an innovative method for reconciling the differences between public-permissionless networks and private-permissioned (or consortium) networks. These networks integrate the permissioning characteristics of private consortiums and a decentralised governance structure, striving to obtain the benefits of each system [93]. Permissioning entails the identification of participating nodes with real-world identities, offering compliance

[1] https://www.hyperledger.org/
[2] https://www.multichain.com/
[3] https://www.r3.com/
[4] https://ripple.com/

benefits while also allowing for better consensus algorithms, improved performance, and energy efficiency. The decentralised governance model enhances trust and confidence in the network, particularly for external users who access services indirectly, thus creating a more inclusive and transparent blockchain environment [93].

Blockchain technology offers advantages contributing to its adoption across industries. One benefit is decentralisation, which eliminates the need for a central authority, thereby increasing trust and reducing the risk of single points of failure [41]. This decentralised nature also enables transparency and auditability [107], as every participant in the network can access and verify the transactions stored on the blockchain. Another advantage is the enhanced security provided by cryptographic techniques used in blockchains, such as hashing and digital signatures, which make the transaction data resistant to tampering and fraud [41]. Furthermore, blockchain technology can streamline processes and reduce operational costs by automating tasks and minimising the need for transaction intermediaries [41].

While blockchain technology offers various advantages, several limitations need to be addressed. One concern is scalability, which encompasses transaction processing rates, data size, and data transmission latency [121]. For example, public blockchains, such as Bitcoin and Ethereum, face challenges in handling increased transaction volumes, leading to higher transaction fees and longer confirmation times [121]. This issue is exacerbated by the global replication of data across all full nodes, resulting in limitations on data size and increased write latency due to the need for updates to propagate across a global network [121].

Moreover, the energy consumption and environmental impact associated with some blockchain consensus mechanisms, like Bitcoin's proof-of-work, raise sustainability concerns [111]. In addition, the uncertain regulatory environment surrounding blockchain technology and digital assets, which varies across jurisdictions, poses challenges to compliance and adoption [111]. Integrating blockchain solutions with existing systems and processes may also require substantial resources and effort, further complicating matters for organisations adopting the technology [41]. Furthermore, public blockchains face a trade-off between data privacy and transparency, as they inherently provide limited privacy due to the absence of privileged users, necessitating a balance between preserving data privacy and maintaining the technology's inherent transparency [121].

Blockchain-based applications, which means applications that use blockchain to a significant degree [121], leverage the underlying technology of blockchain to address various use cases and industry challenges, utilising its decentralised, secure, and transparent nature [121]. These applications extend beyond cryptocurrencies and financial services, encompassing various sectors, such as supply chain management, healthcare, voting systems, digital identity, intellectual property, and the Internet of Things (IoT) [121].

Corda, developed by R3, is a distributed ledger technology (DLT) platform designed specifically for the financial services industry, focusing on ensuring interoperability,

scalability, and privacy [70]. While Corda's primary applications are geared towards financial services, its versatile architecture has also made it suitable for various other industries. For example, Mohanty [70] describes Corda use cases in the following areas: insurance, travel, manufacturing and supply chain, healthcare, telecommunication services, tokenisation, agriculture, and government/land registry. Below is a summary of selected applications of Corda's blockchain technology, listed on Corda's website [32], where security (including confidentiality, integrity and availability) and risk management (including regulatory compliance and auditability) are fundamental.

One noted application of Corda is automating the bond lifecycle in the Agora use case [3]. By creating digital "smart bonds," Agora streamlines processes, reduces costs, and mitigates security vulnerabilities. Nasdaq has also leveraged Corda to develop the Nasdaq Digital Assets Suite (NDAS) [75, 76]. This robust and scalable platform facilitates the seamless trading and management of digital assets while ensuring compliance with regulatory standards. Corda's technology has also been applied to enhance the efficiency and security of trade finance and supply chain management. Contour [27, 84] and the Marco Polo Network [63], for example, use Corda's blockchain to streamline global trade finance activities, while DLT Ledgers [33] and aXedras [14, 83] utilise the platform to improve the transparency and traceability of supply chains. Moreover, Corda has been employed in developing various platforms to address challenges in capital markets, cash management, and interbank reconciliation. HQLAX [47], Instimatch Global [49, 86], and Spunta [85, 101], for instance, leverage Corda's privacy and security features to enable more efficient and secure transactions across various financial sectors.

There is little literature on the application of blockchain technology in auditing. Due to historical competition, the four major audit networks (the Big Four) develop their own software, which is not available to other auditors. Consequently, information on the development of this software is limited, as is the literature on the subject. Bonsón and Bednárová [17], however, shed light on the existence of blockchain-based Big Four developments, which, as a rule, are not audit software.

The development of blockchain-based applications by the Big Four accountancy firms has generated considerable interest and investment as these companies increasingly recognise the potential of this technology for accounting and auditing [17]. Deloitte, for example, has launched Rubix, a software platform enabling users to develop customised blockchain and smart contracts for various purposes, including automating financial reconciliations and providing real-time assurance for financial statements [68]. KPMG, in collaboration with Microsoft, has focused on creating prototype models to address blockchain implementation challenges in industries such as financial services, the public sector, and healthcare [58]. Ernst & Young has developed EY Ops Chain, a project centred on payments, invoicing, inventory information, pricing, digital contract integration, and supply chain management, while also engaging in the Libra project, a start-up focused on distributed ledgers [6, 69]. Finally, PWC has launched a report on blockchain's

implications for energy concerns and developed the De Novo platform, which emphasises the implementation of blockchain in supply chain management [43]. The active participation of the Big Four in blockchain initiatives underscores the transformative potential of this technology in the accounting and auditing landscape [17].

In summary, blockchain technology has the potential to revolutionise various industries and applications, offering a decentralised, secure, and transparent solution for managing data and transactions. Corda's blockchain platform, in particular, demonstrates its adaptability and versatility in addressing security, risk management, and regulatory compliance challenges across diverse industries and applications. The technology's key strengths lie in its privacy features, transaction-level security, and ability to provide seamless, efficient, and auditable processes for regulated businesses and financial institutions.

2.4 Related Work

This section discusses studies on the application of R3 Corda blockchain technology with particular attention to security risk management. Iqbal and Matulevičius [55] explored the management of security risks in the capital market post-trade matching and confirmation process using CorDapp, a blockchain-based application built on the Corda platform. The research aimed to mitigate security risks present in centralised systems and identify new risks introduced by the CorDapp. By comparing countermeasures of centralised applications and CorDapp, the authors demonstrated how CorDapp could address security challenges in the financial industry. The security risk management (SRM) domain and STRIDE threat models were employed to systematically analyse, identify, and mitigate potential security risks. The study found that CorDapp could help mitigate various security risks associated with centralised post-trade infrastructures by leveraging the benefits of blockchain technology. The findings of this study can support developers' decisions while developing blockchain-based applications and contribute to building a future security risk reference model for such applications.

The research conducted by Hajela et al. [42] focused on developing an effective privacy and security solution for healthcare data using the R3 Corda platform of blockchain technology. The study aimed to address the research gaps in healthcare privacy by implementing a blockchain-based system called ITreatU (ITU). ITU ensures private and secure treatment transactions between doctors and patients by utilising the properties of the Corda platform, a private blockchain platform allowing only the involved parties to reach a consensus on the state of the ledger. The researchers designed the architecture and flow of the ITreatU system, which involved three types of participants: doctors, patients, and hospitals. The system maintained a shared ledger containing information about medical treatments that evolved throughout the course of treatment. The notary service, represented by hospitals, validated the transactions without storing the business trans-

actions, ensuring privacy between the involved parties. In addition, the ITreatU Corda flow algorithm was employed to simulate transactions between doctors and patients. In conclusion, the research demonstrated that the doctor-patient treatment interaction could be made private using a permission-based, bilateral ledger-based blockchain platform like Corda, offering a promising healthcare data privacy and security solution.

Carare et al. [22] explored that implementing a Corda permissioned blockchain technology-based solution automates bilateral derivatives Over-The-Counter (OTC) Post Trade Confirmation financial transactions. Traditionally, OTC transactions have been carried out manually by dedicated personnel within private banking institutions, leading to potential inaccuracies and inefficiencies. The researchers suggested an inventive certified progression of workflow states for the transaction, utilising asynchronous communication between two parties through a smart contract, which was orchestrated and validated by a notary node. The proposed solution, which automates the OTC transaction process, aims to minimise human-related errors and expedite transaction closure. The system was implemented on a Corda test net, demonstrating the potential to reduce errors and save time compared to conventional manual operations. Furthermore, by integrating the blockchain-based solution with existing fintech software tools, this research offers a practical and secure approach to automating OTC financial transactions, ensuring accurate accounting and timely settlement while reducing the risk of operational and cash errors.

Minango et al. [67] concentrated their research on utilising Corda blockchain technology within the supply chain industry. Their goal was to tackle the absence of a synchronised data mechanism for various entities in supply chains, which may lead to mistakes and financial losses. By implementing the Corda blockchain and creating CorDapps as a proof of concept, they showcased its applicability in the supply chain domain. Corda was proven to facilitate secure transactions on shared data, enhancing visibility, transparency, and efficiency throughout supply chains. The researchers also evaluated the performance of the CorDapps by measuring throughput, latency, CPU usage, and memory usage. The results showed satisfactory performance when using Corda distributed ledger technology (DLT) in the supply chain area. The Corda platform provided immutability to cargo provenance, eliminated reconciliation issues across multiple parties and offered real-time visibility for track and trace analysis, risk assessment, and acceleration of physical and financial supply chains. This proof of concept can be extended to supply chains involving more actors and highlights the potential of Corda for enhancing security and efficiency in the supply chain sector.

2.5 Summary

In this chapter, the background to the thesis was outlined and question **RQ 1, What is the current state of securing auditing processes?** was answered. In order to provide a

more comprehensive answer, this question was divided into four sub-questions:

RQ 1.1: What are the auditing processes and systems? Section 2.1 examines the auditing process and systems, focusing on the objective of a financial audit, which is to obtain reasonable assurance and express an opinion on the company's financial position, performance, and cash flows [87]. The audit process follows several stages, including pre-engagement activities, risk assessment, risk response, and summarising and reporting.

Audit systems are computer programs that assist auditors in carrying out their work, ranging from small systems for small businesses to large enterprise systems for multinational organisations. They help document engagements, record and process evidence, and archive completed engagement files. Key functionalities of audit systems include engagement project management, data extraction and analysis, sampling for testing, and reporting.

RQ 1.2: What are the security risk management approaches? In thesis Section 2.2 on Security Risk Management, the author emphasises the importance of the security risk management process in ensuring financial audit systems' integrity, confidentiality, and availability. The Information Systems Security Risk Management (ISSRM) domain model is introduced as a comprehensive conceptual framework, which includes key concepts, relationships, and definitions related to security risk management [4, 34, 64, 66]. The ISSRM domain model comprises three main concepts: asset-related, risk-related, and risk treatment-related [64, 66].

The author also highlights several international and regional standards that contribute to security risk management. These include the ISO 31000 standard [102], the ISO/IEC 27000 family of standards [103], and the Estonian Information Security Standard (E-ITS) [7]. These standards provide organisations with broad principles, frameworks, and processes to effectively manage risks and ensure the security of their information assets.

RQ 1.3: What is blockchain technology that could secure auditing processes? Based on the insights presented in Section 2.3 of the thesis, blockchain technology offers a secure, decentralised, and transparent solution for managing data and transactions, which can be leveraged to secure audit processes. The five fundamental properties of blockchain technology [121]—immutability, non-repudiation, integrity, transparency, and equality—provide a foundation for establishing trust and accountability within the network. This ensures that the transaction data recorded on the blockchain is resistant to tampering and fraud, allowing for a reliable and auditable record-keeping system [107].

In particular, the Corda blockchain platform demonstrates adaptability and versatility in addressing security, risk management, and regulatory compliance challenges across various industries and applications. Its privacy features, transaction-level security, and ability to provide seamless, efficient, and auditable processes make it well-suited for regulated businesses and financial institutions [44, 70], including those seeking to enhance the security of their auditing processes. By leveraging the potential of blockchain

technology, as detailed in Section 2.3, organisations can secure their auditing processes while also benefiting from the increased trust, transparency, and efficiency offered by this innovative technology [44, 70].

RQ 1.4: What are the current solutions to mitigate security risks by applying blockchain technology? In reference to Section 2.4 of the thesis, R3 Corda blockchain technology has demonstrated its potential to address various security risk management concerns across multiple sectors, including finance, healthcare, and supply chain management. The studies discussed in this section explored the application of Corda-based solutions to mitigate security risks in centralised systems, improve data privacy, reduce human-related errors, and enhance overall operational efficiency.

For instance, the research by Iqbal and Matulevičius [55] showcased the capability of CorDapp to address security challenges in the financial industry, while Hajela et al. [42] demonstrated the effectiveness of Corda in providing a privacy and security solution for healthcare data. Similarly, Carare et al. [22] revealed the potential of Corda-based solutions in automating bilateral derivatives Over-The-Counter (OTC) Post Trade Confirmation financial transactions, and Minango et al. [67] highlighted the benefits of employing Corda blockchain technology in the supply chain sector. Thus, Section 2.4 highlights the versatility of Corda in securing various processes and provides evidence supporting its potential to secure auditing processes as well. The next chapter focuses on the security risks of audit processes.

3 E-dok-based Auditing System

From this chapter onwards, the author's contribution is given. This chapter answers the research question **RQ 2, What are the security requirements for the audit process?**. The question has been divided into three sub-questions for better answering:

RQ 2.1: What are the assets?

RQ 2.2: What are the risks when using conventional technology?

RQ 2.3: What are the means to mitigate the risks identified when using conventional technology?

This chapter introduces the case description of the thesis. The first section describes and models the scenario. The second section identifies the security risks of the deployed architecture, the security requirements and available means to mitigate these risks.

3.1 Case Description

Different auditors and networks of auditors use software of varying complexity and capability to conduct audits. The Estonian Auditors' Association has developed the software E-dok[5] for small and medium-sized audit firms, in which the functionality of receiving files from clients has been solved in a primitive way. There is one folder per client where all authorised client representatives can upload files (input folder). It is impossible to distinguish between the files uploaded for different client engagements in this folder. All the files uploaded to the input folder by a client's representative are accessible to everyone with access permission to the folder. All the files must be imported from the input folder into the engagement file during the engagement, making the job inconvenient and time-consuming. Therefore, auditors using the E-dok and simpler software often use various additional tools to obtain information from clients, the most predominant of which are e-mails and cloud servers.

The present case concerns the collection of information by the group auditor to audit the client's consolidated annual accounts. Namely, the group auditor is responsible for auditing the consolidated accounts and, therefore, for auditing the information consolidated in that report. The consolidation group in question (the group) consists of three components (see Figure 1) – the parent company located in Estonia, which is engaged in retail and small wholesale, as well as the group's procurement and management. The group also includes subsidiaries in Latvia and Lithuania engaged in retail and small wholesale. The information regarding engagements performed for the group's audit, their scope and information collection tasks are presented in Table 1.

The parent company of the Group is audited by an audit firm licensed in Estonia. The Latvian subsidiary is audited by a Latvian audit firm (subsidiary auditor). The operating volumes of the Lithuanian subsidiary are so small that it does not need to be

[5]https://www.audiitorkogu.ee/est/e-dok

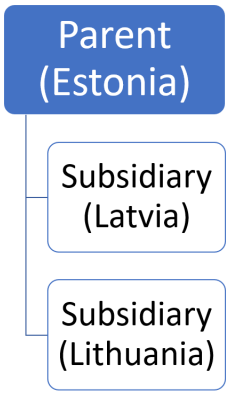

Figure 1. Structure of the client consolidation group

audited. However, the International Standards on Auditing require performing analytical procedures [90] regarding the subsidiary. Also, there are items in its financial statements that are subject to audit procedures for the audit of the consolidated financial statements. To perform the procedures, the group auditor will obtain the necessary info. This thesis deals with one of the sub-processes of the audit process – collecting information for performing an audit. This sub-process starts when the client contract for the provision of audit services has been concluded and ends when the received information is saved.

Conceptual model A conceptual model of the case as a UML class diagram is presented in Figure 2. It consists of the following concepts: Engagement, Auditor, Client, Bank, 3rdParty and SubsidiaryAuditor.

Engagement is an instance of an audit service the auditor provides to the client. Engagements are described by their characterising information (i.e., attribute *engagementDetails*, which might include the type of engagement (in the current case audit), scope (e.g., consolidated financial statements of the client), the period covered by the financial statements audited, and others). It has the client who ordered the engagement (i.e., attribute *client*) and the auditor responsible for it (i.e., attribute *auditor*). For collecting information for the engagement, an information request (i.e., attribute *infoRequest*) is issued from the auditor to the client, and the audit instructions (i.e., attribute *auditInstructions*) for the subsidiary auditor (i.e., attribute *subsidiaryAuditor*) are prepared by the auditor. During the engagement, info submitted by the client (i.e., attribute *clientInfo*), bank confirmation (i.e., attribute *bankConfirmation*), info from third parties (i.e., attribute *3rdPartyInfo*), and reporting from the auditor of the subsidiary (i.e., attribute *subsidiaryAuditorReport*) are obtained.

The auditor is an audit firm performing the engagements (i.e., attribute *engagements*). Its characterising details describe the auditor (i.e., attribute *auditorDetails*, which might

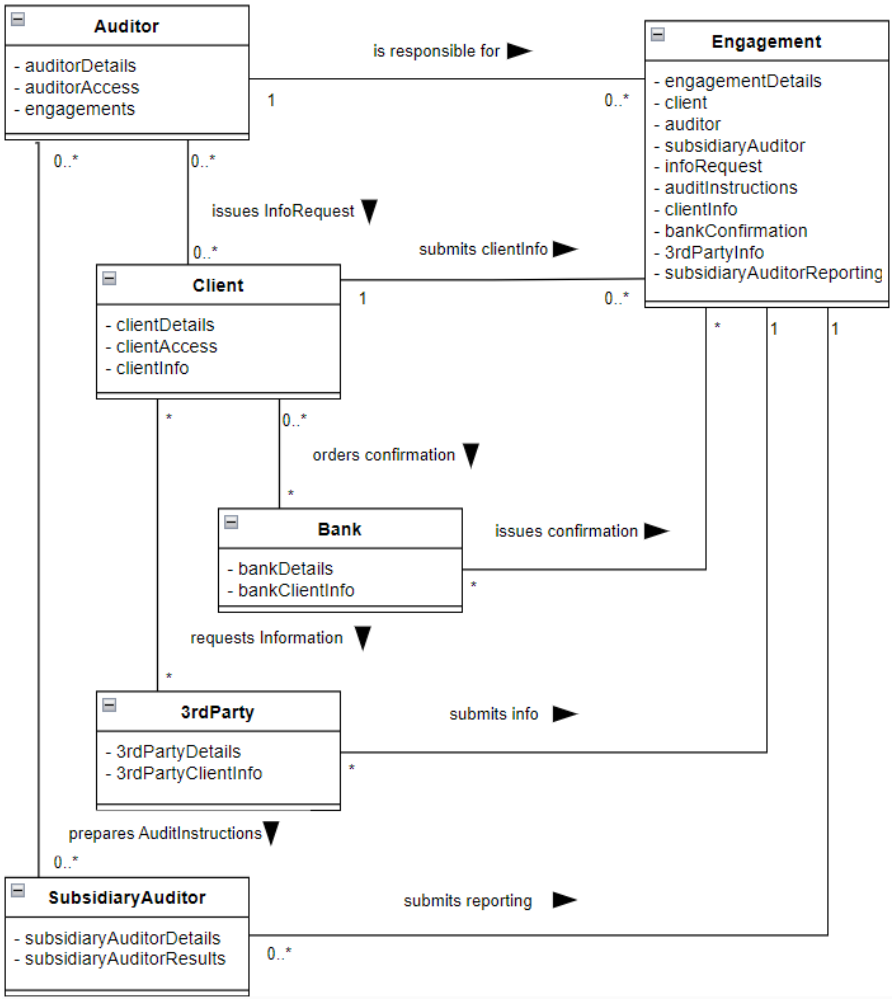

Figure 2. Conceptual model

include, e.g., name, activity license number, address and names of sworn auditors). In addition, the auditor has access (i.e., attribute *auditorAccess*) to the engagement file and input folder, opened in the E-dok environment, the first for documenting the engagement, the latter for collecting info from the client for the engagement.

The client is an economic entity specified by its characterising details (i.e., attribute *clientDetails*, which might include, e.g., name, commercial registry code and address). The entity orders the engagement from the auditor. The client has access (i.e., attribute *clientAccess*) to the E-dok input folder to submit its info (i.e., attribute *clientInfo*) requested by the auditor to perform the engagement.

Bank is the bank that serves the client, determined by the details (i.e., attribute *bankDetails*, which might include, e.g., name, commercial registry code, address and

SWIFT code/BIC). In addition, the bank records the necessary client information (i.e., attribute *bankClientInfo*, such as the persons entitled to sign the client, client banking transactions, balances and liabilities to the bank) for the audit.

3rd party is a third party, characterised by detailed information (i.e., attribute *3rdPartyDetails*, which might include the relationship with the client (e.g., lawyer, supplier or customer), name, commercial registry code and address) that has relevant information (i.e., attribute *3rdPartyClientInfo*) about the client for the audit.

SubsidiaryAuditor is the auditor of a client subsidiary, determined by the details (i.e., attribute *subsidiaryAuditorDetails*, which might include, e.g., country, name, activity license number, address and names of sworn auditors). The subsidiary auditor audits the subsidiary following the audit instructions provided by the auditor. As a result of the audit (i.e., attribute *subsidiaryAuditorResults*), the subsidiary auditor provides the reporting and information required by the audit instructions.

Business process The value chain diagram in BPMN illustrates how an organisation's business functions interrelate to achieve its objectives [35, 1, 99]. In Figure 3, an extract of the value chain of the E-dok scenario is presented. The subprocesses are as follows:

- *Prepare info request* – preparing info request and providing it to the client;
- *Prepare audit instructions* – preparing audit instructions and providing these to the auditor of the client's subsidiary;
- *Obtain info from the client* – controlling the info collection process from the client;
- *Obtain bank confirmation* – receiving and saving the bank confirmation;
- *Obtain a copy of the request 3rd party* – receiving from the client and saving the requests to third parties;
- *Obtain info from 3rd party* – receiving info from respective parties and saving it;
- *Obtain info from subsidiary auditor* – receiving and saving the reporting and info from the client's subsidiary auditor.

The process is illustrated in Figures 4 and 5. The business process model represents the process of obtaining information from different parties as follows:

- auditor,
- client,
- subsidiary auditor,

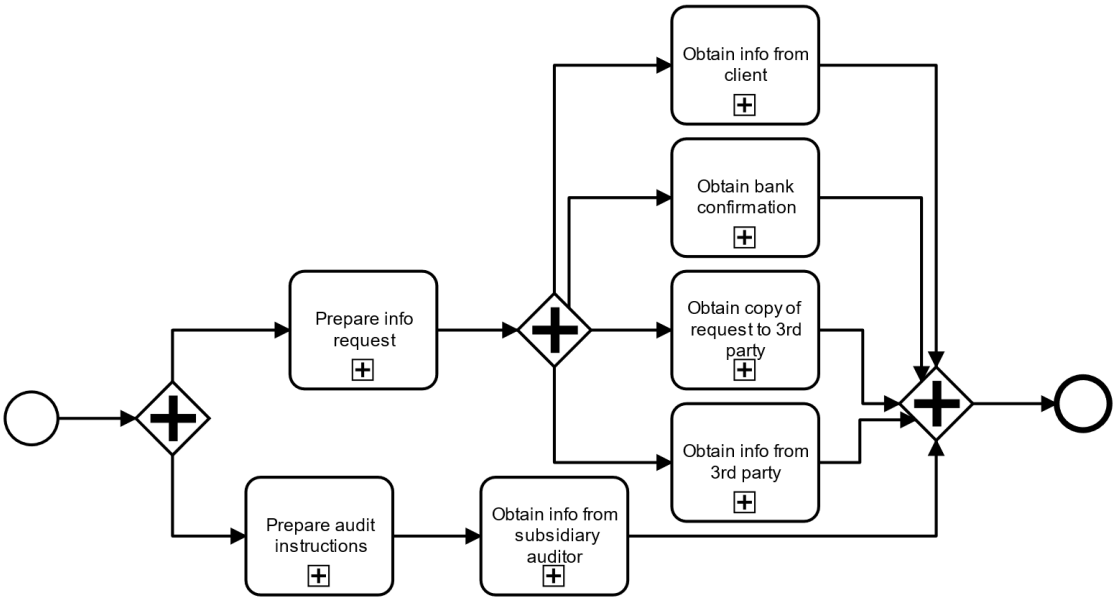

Figure 3. Business process value chain

- bank,
- third party,

expressed in pools, and E-dok is a system.

The process has business partners: *auditor* and *client*, i.e., the entities accessing the network infrastructure to communicate with *E-dok* [64].

To perform the audit, the group auditor submits a request for information necessary for the audit (information request or info request) about the group and the Estonian company and consolidation to the group's chief accountant, who is responsible for the audit of the group's consolidated financial statements. A significant portion of the requested information and documents is provided to the auditor by the chief accountant by uploading them to the E-dok input folder shared by the auditor.

The exception is external confirmations, including those from the client's banks. The client forwards requests for these confirmations to the auditor's request via e-banking or e-mail. The same applies to comparisons of balances with suppliers and customers, which according to International Standards on Auditing, must be performed under the auditor's control [89]. To this end, the client's representative (in this case, the chief accountant) sends the balance comparison inquiries by e-mail to the suppliers and customers selected by the auditor, with a copy to the auditor. Banks and representatives of suppliers and customers send the answers directly to the auditor, usually by e-mail. Foreign information providers may also do so in writing (by mail).

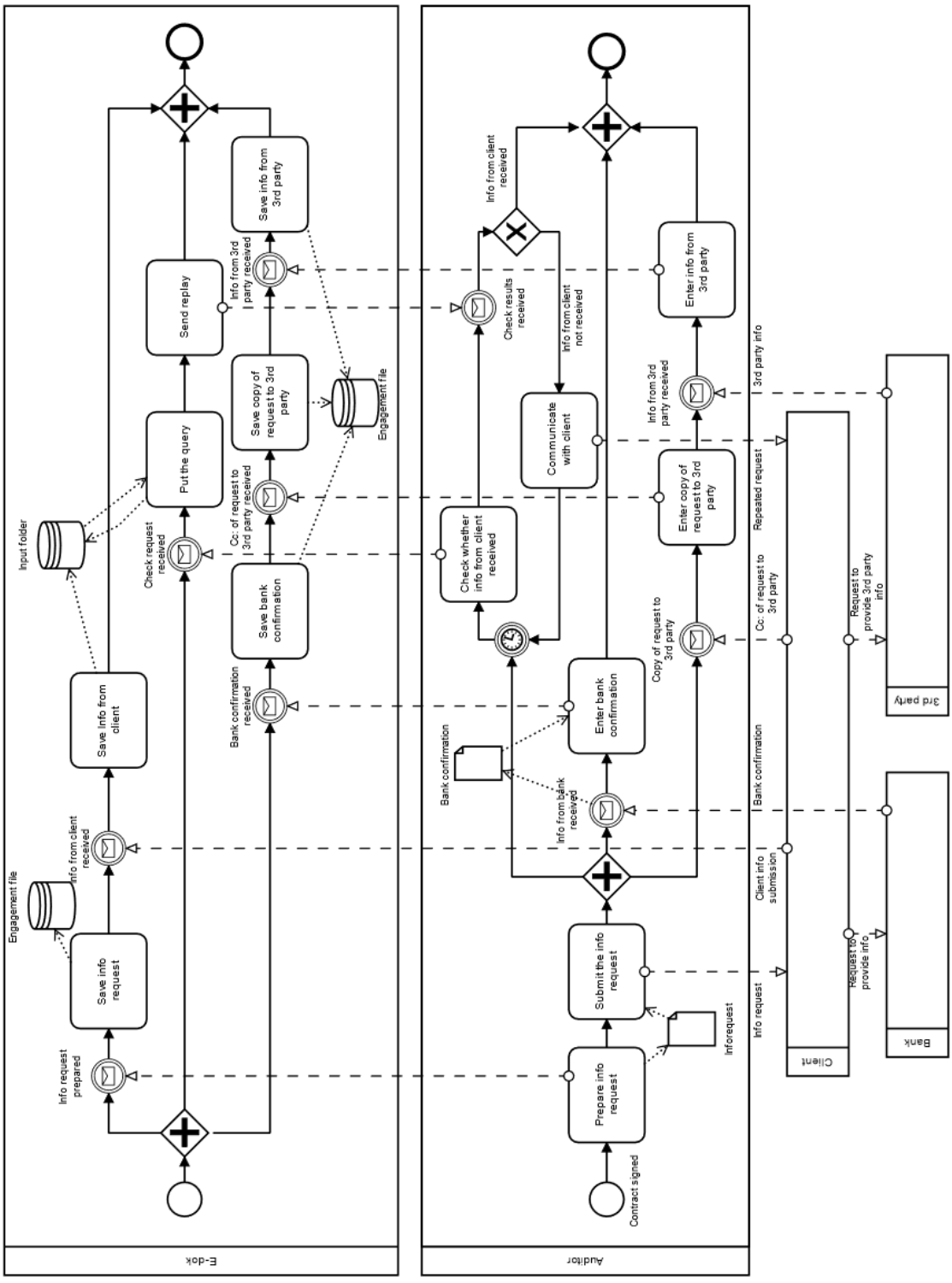

Figure 4. Main process of obtaining engagement info

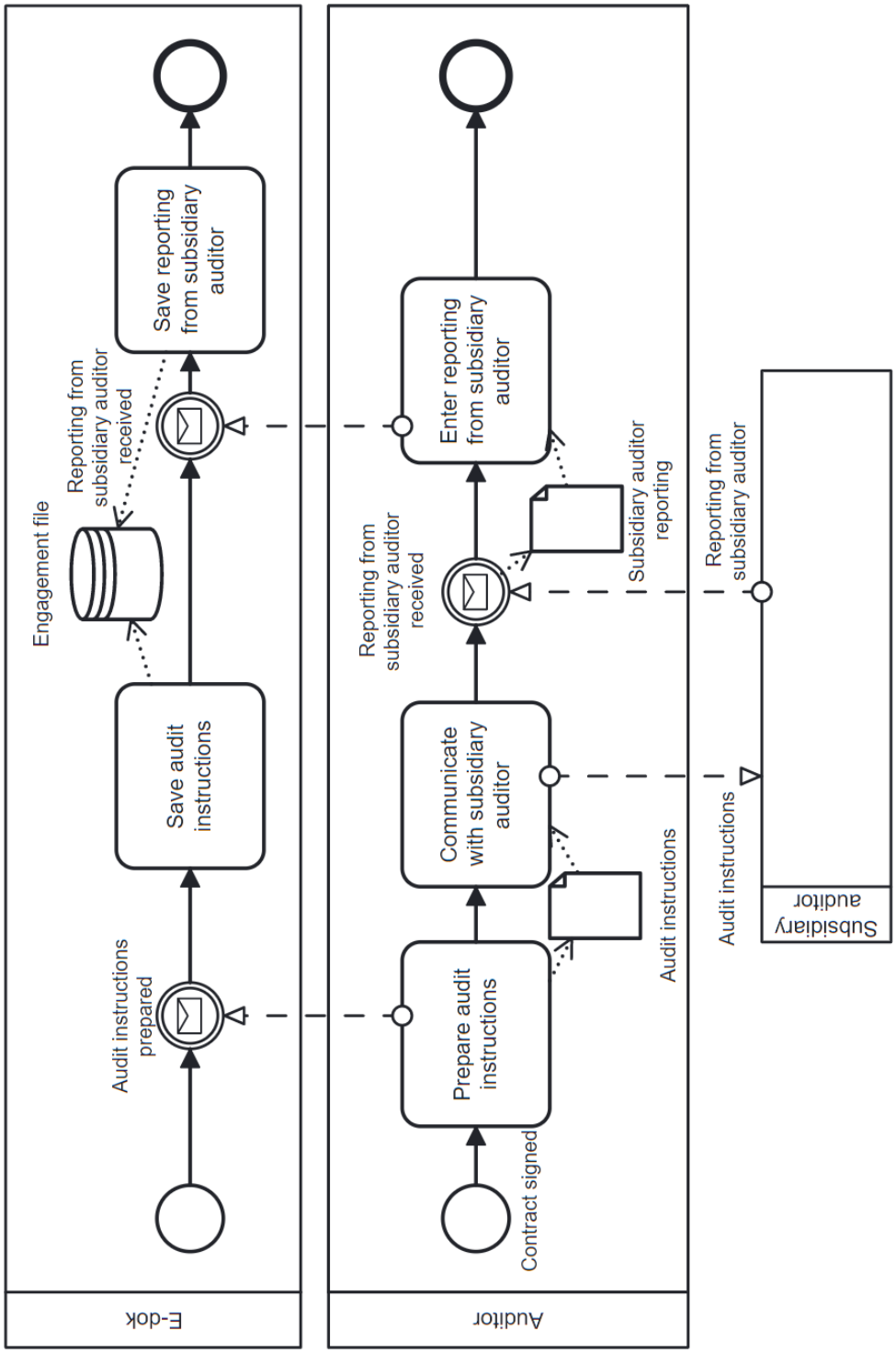

Figure 5. Process of obtaining engagement info from the subsidiary auditor

The group auditor sends the auditor of the Latvian subsidiary by e-mail the audit instructions, which contain information and instructions relevant to the group audit that the said auditor must follow in the course of its work, as well as the confirmations and inquiries to be responded as a result of its audit. Thus, it must submit the audited annual accounts of the Latvian subsidiary, the auditor's report and other documents required by the group auditor.

The operating volumes of the Lithuanian subsidiary are so small that no audit or review is required. The analytical procedures performed by the group auditor are sufficient. The accountant of the respective subsidiary provides the information and documents for these. The client accountant enters the info into the E-dok input folder. Obtaining confirmations and info from the Lithuanian subsidiary banks and third parties have the same procedure as in the case of the parent company. These are requested through the client subsidiary. Communication is performed by e-mail. The banks may provide their confirmations by regular mail. All the information the auditor receives is stored in the appropriate folders of the E-dok engagement file and used for further audit procedures.

Table 1. Consolidation group and its audit engagements – scope and information gathering

Component No	Location	Function of undertaking	Field of activity	Auditor	Scope	Required info	Info requester	Info provider	Means
1	Estonia	parent	retail and small wholesale, group procurement and management	group auditor	Audit of the consolidated annual accounts and the parent company	Info and documentation according to the InfoRequest of auditor	group auditor	client (group chief accountant)	E-dok
						confirmations from the client banks	client, based on InfoRequest	the banks	e-mail
						balance verifications/ confirmations	client, based on InfoRequest	3rd parties (selected suppliers and customers)	e-mail
2	Latvia	subsidiary	retail and small wholesale	Latvian subidiary auditor	Audit of the annual accounts of subsidiary	Documentation according to the group auditor audit instructions	group auditor	subsidiary auditor	e-mail
3	Lithuania	subsidiary	retail and small wholesale	group auditor	Analytical procedures and selected audit procedures for the group audit purposes	Info and documentation according to the info request of auditor	group auditor	client (Lithuanian accountant)	E-dok
						confirmations from the client banks	client, based on InfoRequest	the banks	e-mail or mail
						balance verifications/ confirmations	client, based on InfoRequest	3rd parties (selected suppliers and customers)	e-mail

3.2 Security Requirements Elicitation

The study will continue with security requirements elicitation from business process (SREBP). The SREBP method provides means to elicit security requirements from business processes employing security risk-oriented patterns [4, 64]. The SREBP process consists of two stages. The first one includes two steps: business asset identification and security objective determination, and the second one, security requirements elicitation, includes the steps of identifying patterns, extracting security model, and deriving security requirements [64, 94].

As a starting point for the security requirements elicitation, knowledge about the organisation's values (from the value chain in Figure 3) and business functions (from the detailed workflow of obtaining engagement info in Figures 4 and 5) have to be collected. In stage one, step one the **business assets** as follows were identified from the value chain:

1. Info request (*infoRequest*);
2. Audit instructions (*auditInstructions*);
3. Info from the client (*clientInfo*);
4. Bank confirmation (*bankConfirmation*);
5. Copy of request to a third party (*ccOfRequestTo3rdParty*);
6. Info from the third party (*3rdPartyInfo*);
7. Reporting of the subsidiary auditor (*subsidiaryAuditorInfo*).

In step two, for the business assets, the **security objectives** as follows were identified: The assets should be *confidential*, i.e., not available to unauthorised parties, *integral*, i.e., not tampered with, and *available* to the authorised business partners until the respective sections of engagement are completed.

In stage two, security requirements elicitation is performed by applying security risk-oriented patterns one by one. For each pattern, one example from the business process is analysed. Based on it, an example set of security requirements is given in the thesis. A complete list of the identified security requirements of the case is given in Appendix I.

3.2.1 Pattern One: Securing Data From Unauthorised Access

The most significant objective of the security risk-oriented pattern (see Figure 6) is to protect the *confidentiality* of the business assets when these are being manipulated by the system asset E-dok, consisting of *EngagementFile* and *InputFolder*. For example, the selected business asset is *clientInfo*, and *InputFolder* is the system asset.

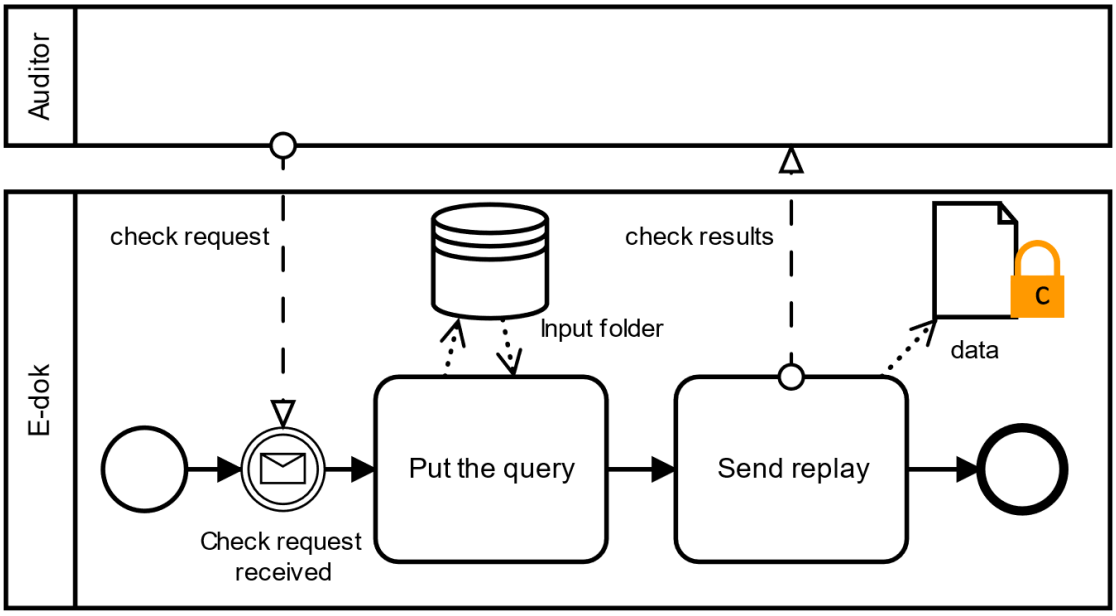

Figure 6. Securing data from unauthorised access, asset modelling

The threat (see Figure 7) arises if the system asset, characterised by its attribute *clientInfo*, is accessed by unauthorised users. The risk event would harm the reliability of E-dok, negate the confidentiality of the business asset, and lead to unintended use of the data of *clientInfo* [64]. To mitigate the security risk (see Figure 8) [64], an access control mechanism, e.g., Role-Based Access Control (RBAC), could be applied. Therefore, the model (see Figure 9) is derived from the activities as follows:

(i) Identify resources: The system assets *EngagementFile* (with its attributes *infoRequest, auditInstructions, bankConfirmation, ccOfRequestTo3rdParty, 3rdPartyInfo, subsidiaryAuditorInfo*) and *InputFolder* (with its attribute *clientInfo*) are the resources needing protection from unauthorised access. The resources are modelled in Figure 9 using the «resource» stereotype.

(ii) Identify roles: The roles that could access the resources are *Auditor* and *Client*. The roles are modelled in Figure 9 using the «role» stereotype.

(iii) Assign users: In the current case, it is impossible to derive concrete users as instances of particular roles from the business process.

(iv) Identify secured operations: saveInfoRequest, saveAuditInstructions, saveBankConfirmation, saveCcOfRequestTo3rdParty, save3rdPartyInfo, saveSubsidiaryAuditorInfo, saveClientInfo, and *queryClientInfo* are the operations, i.e., the actions that can change the state of the resources. The secured operations are modelled in Figure 9 as operations of the classes *EngagementFile* and *InputFolder*.

(v) Assign permissions: Role execution permissions determine protected security

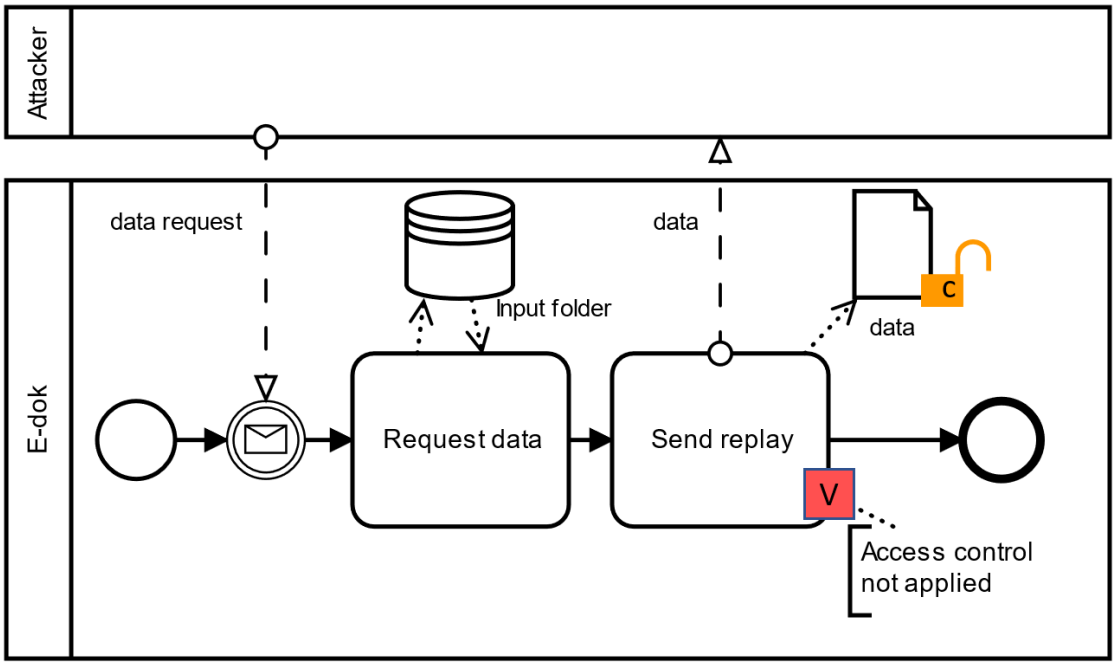

Figure 7. Risk modelling

actions which allow changing the state of the secure resource. The security actions are, in the current case, *Insert* and *Select*. The *Auditor* role has the permission to insert *EngagementFile's* attributes *infoRequest, auditInstructions, bankConfirmation, ccOfRequestTo3rdParty, 3rdPartyInfo* and *subsidiaryAuditorInfo*. It also has the role of *Auditor* permission to select *InputFolder's* attribute *clientInfo*. *Client* role has the permission to insert *InputFolder's* attribute *clientInfo*.

The authorisation constraints (AC) as follows were defined:

AC#1: Auditor prepareInfoRequest -> inserts EngagementFile.infoRequest by saveInfoRequest()

AC#2: Auditor prepareAuditInstructions -> inserts EngagementFile.auditInstructions by saveAuditInstructions()

AC#3: Auditor enterBankConfirmation -> inserts EngagementFile.bankConfirmation by saveBankConfirmation()

AC#4: Auditor enterCcOfRequestTo3rdParty -> inserts EngagementFile.ccOfRequestTo3rdParty by saveCcOfRequestTo3rdParty()

AC#5: Auditor enter3rdPartyInfo -> inserts EngagementFile.3rdPartyInfo by save3rdPartyInfo()

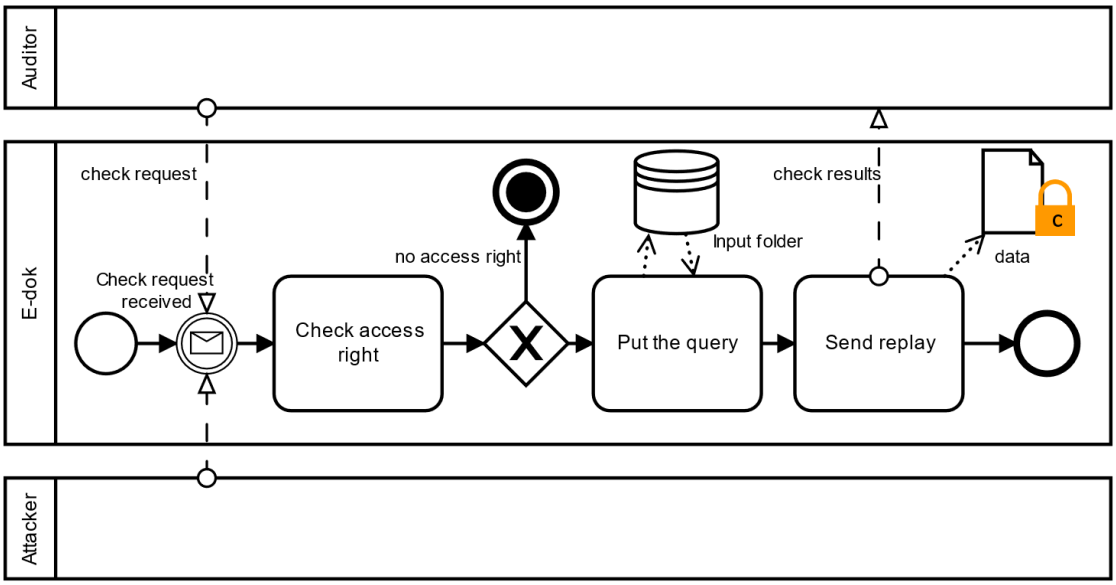

Figure 8. Risk treatment modelling

AC#6: Auditor enterSubsidiaryAuditorInfo -> inserts EngagementFile.subsidiaryAuditorInfo by saveSubsidiaryAuditorInfo()

AC#7: Auditor queryClientInfo -> selects InputFolder.clientInfo by queryClientInfo()

AC#8: Client submitClientInfo -> inserts InputFolder.clientInfo by saveClientInfo()

The derived RBAC security model (Figure 9) provides the context-specific security requirements listed in Appendix I, security requirements #1. In the example of the pattern, the auditor should be able to check the info from the client from the input folder:

SecurityRequirement#1.7: *Auditor* should be able to *query* the *clientInfo* from *InputFolder*.

3.2.2 Pattern Two: Securing Data That Flows Between Business Entities

The pattern (see Figure 10) is applied to protect data while transmitting them, i.e., the business assets, like *bankConfirmation*, between business partner, *auditor*, and the system, i.e., *E-dok* (system asset), over the untrusted channel, Internet, that is transmission medium (system asset). Here, the security criteria of *integrity* and *confidentiality* of data are addressed.

The situation has at least two vulnerabilities (see Figure 11). First, a threat agent (i.e., an attacker) could intercept the transmission channel. Second, the data could be

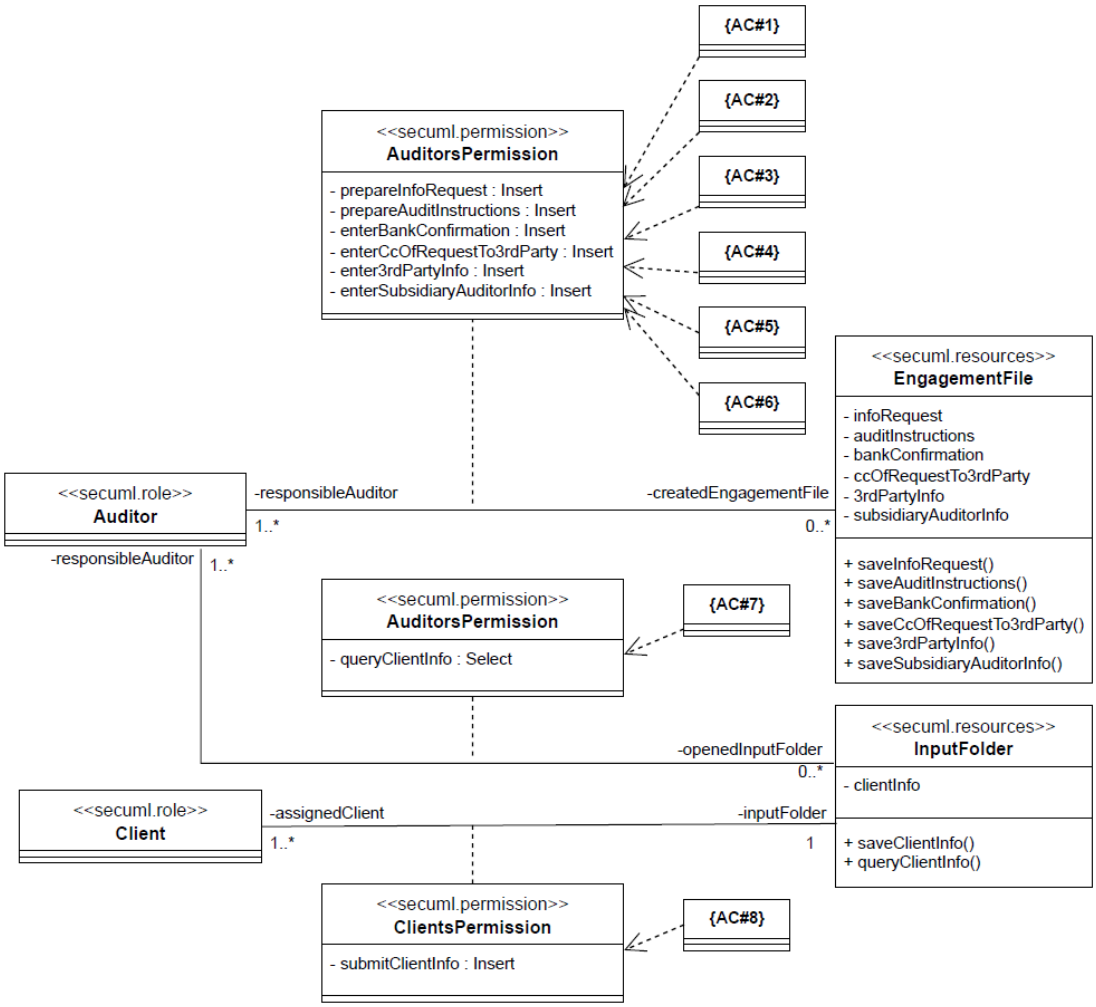

Figure 9. RBAC security model

misused (e.g., modified and sent to the E-dok) if not encrypted. The risk event would harm the data and reliability of the transmission medium and negate the integrity and confidentiality of the business assets [64]. The risk could be reduced (see Figure 12) by making data unreadable, using cryptographic algorithms, and verifying the received data, using checksum algorithms [64].

For defining security requirements, the activities as follows have to be performed:

(i) Identify communicators: The entities that transmit or receive data are the business partners *auditor* and *client*, and the system *E-dok*.

(ii) Identify data transmission: The data transmitted that needs to be protected while exchanging between the communicators is as follows: *infoRequest, auditInstructions, bankConfirmation, ccOfRequestTo3rdParty, 3rdPartyInfo, subsidiaryAuditorInfo,* and

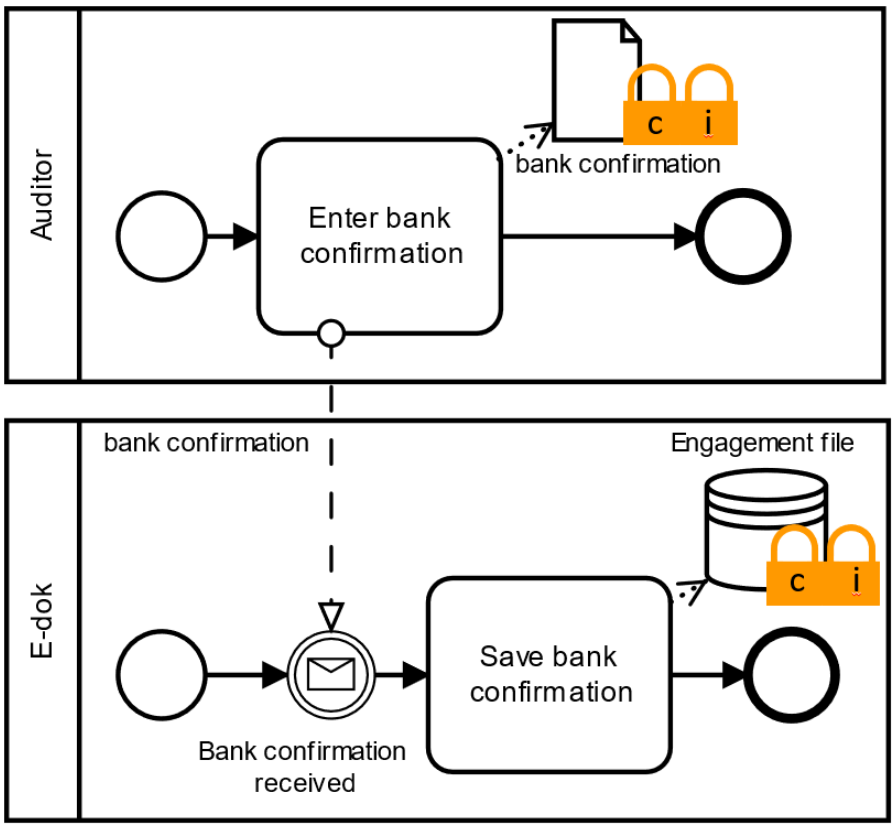

Figure 10. Securing data that flows between business entities, asset modelling

clientInfo are communicated between *auditor* and *E-dok*, and *clientInfo* is communicated between *client* and *E-dok*.

The activities result in the security requirements for the business partners mentioned, and E-dok, are listed in Appendix I, security requirements #2. As an example of the security requirements for the communication between *auditor* and *E-dok*, while data communicated is *bankConfirmation* are:

SecurityRequirement#2.1: *E-dok* should have a unique identity in the form of key pairs (public and private keys) certified by a certification authority.

SecurityRequirement#2.4: *Auditor* should encrypt and sign *bankConfirmation* to be transmitted, using keys before sending the data to *E-dok*.

The requirements could be satisfied by implementing the standard transport layer security (TLS) protocol [11, 64].

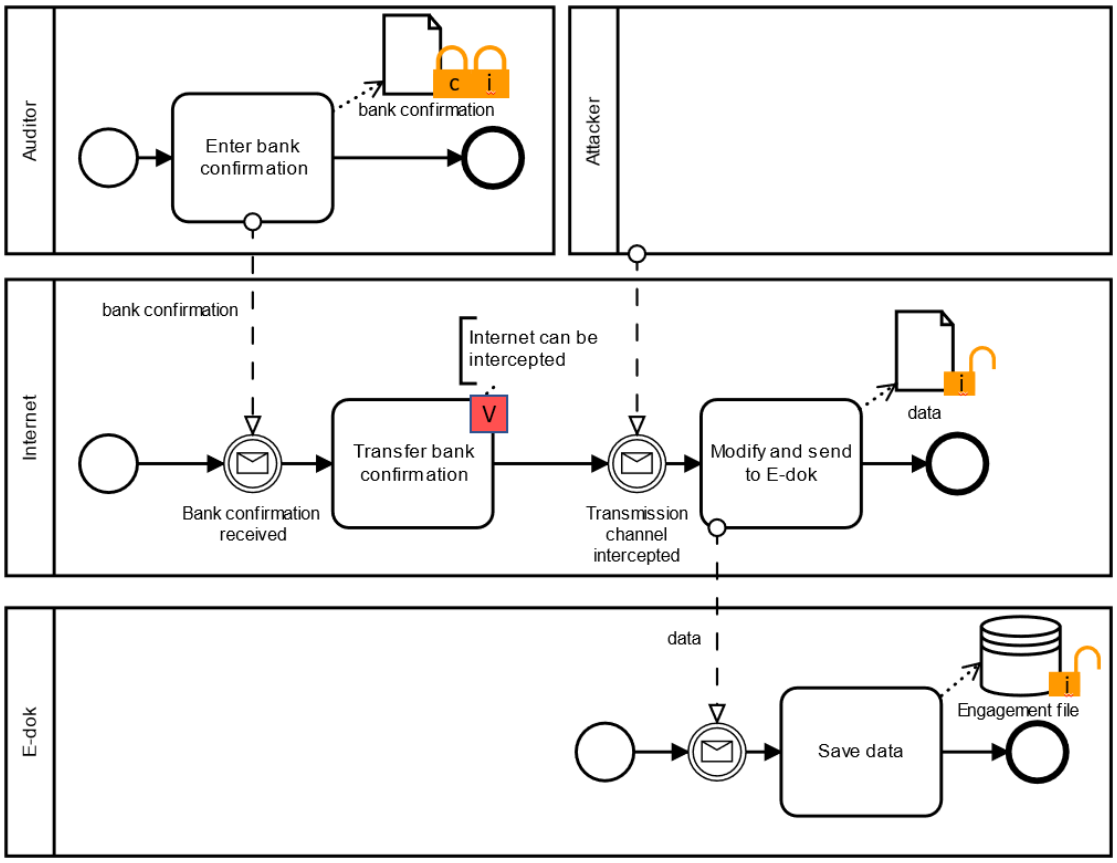

Figure 11. Risk modelling

3.2.3 Pattern Three: Securing Business Activity After Data Is Submitted

The pattern (see Figure 13) secures the business activities after data is submitted. It ensures that the data submitted by the business partners are valid by rejecting malicious data [5]. Here, the security criteria of *availability* and *integrity* of data are addressed [64]. In the area, activities, as follows, are suggested:

(i) Identify input interfaces: SaveInfoRequest, saveAuditInstructions, saveBankConfirmation, saveCcOfRequestTo3rdParty, save3rdPartyInfo, saveSubsidiaryAuditorInfo, saveClientInfo and *queryClientInfo* are considered as the input interfaces of *E-dok*, i.e., the information system activities that receive input from the business partners.

(ii) Identify input data: The data received by the input interfaces from the business partners are as follows: *infoRequest, auditInstructions, bankConfirmation, ccOfRequestTo3rdParty, 3rdPartyInfo, subsidiaryAuditorInfo,* and *clientInfo*.

The risk (see Figure 14) is that a threat agent (an attacker) could exploit the vulnerability of the input interfaces, like *queryClientInfo*, in the example, and submit the

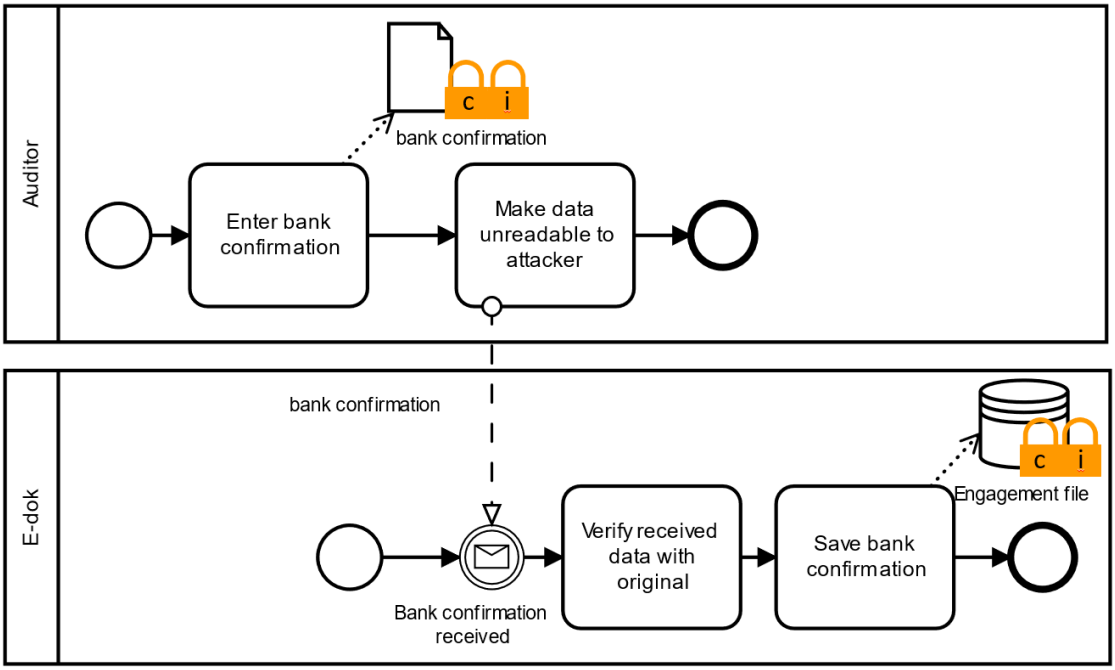

Figure 12. Risk treatment modelling

data *clientInfo*, with malicious scripts. The attacker may change the business rules or read/write the business data by running the scripts. Such an event puts data confidentiality and integrity at risk; the input interface could also be compromised, and any activity after data submission may lose its integrity or become unavailable [64].

To mitigate the risks (see Figure 15), the security requirements of filtering the incoming data could be implemented:

SecurityRequirement#3.8: *QueryClientInfo* should filter the *clientInfo* (the input).

SecurityRequirement#3.16: *QueryClientInfo* should sanitise the *clientInfo* (the input) to transform it to the required format.

SecurityRequirement#3.24: *QueryClientInfo* should canonicalise the *clientInfo* (the input) to verify against its canonical representation.

The list of security requirements of pattern three for the case is provided in Appendix I, security requirements #3. According to [25, 64] SecurityRequirement#3.8, *input filtration* checks the input data against the correct and secure syntax. SecurityRequirement#3.16, *input sanitisation* examines if the common encoding methods are used. SecurityRequirement#3.24, *input canonicalisation* validates the input against its canonical representation of the data.

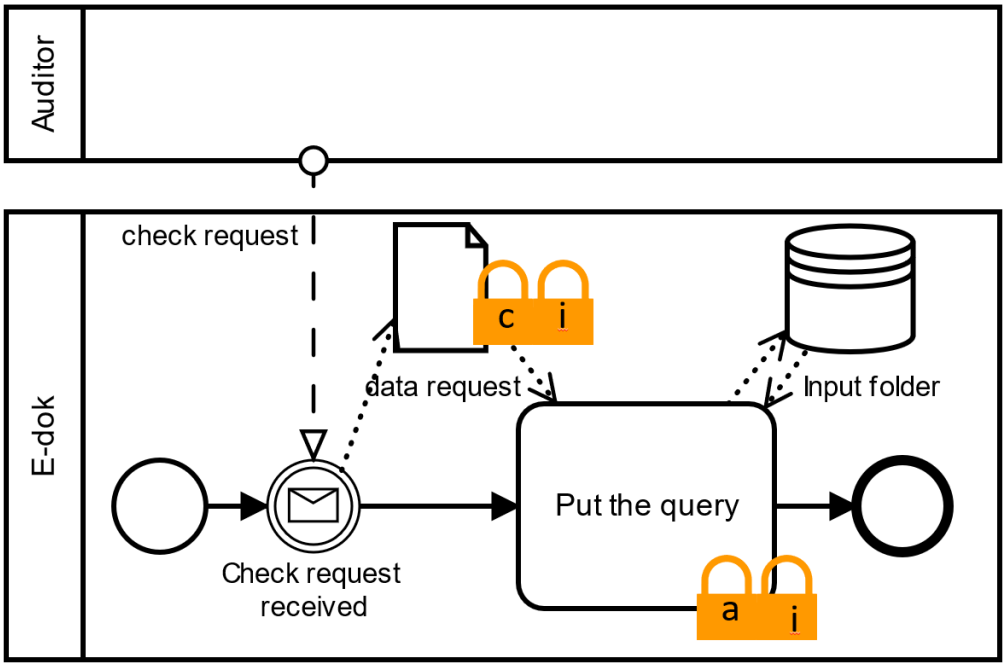

Figure 13. Securing business activity after data is submitted, asset modelling

3.2.4 Pattern Four: Securing Business Service Against DoS Attacks

The pattern intends to protect the business assets (business services) to ensure the availability of the services provided by the server [64]. This is done by protecting the information system against a denial-of-service attack [62]. The activities as follows are performed:

(i) Identify functional-unit: The E-dok system consists of the functional units: *saveInfoRequest, saveAuditInstructions, saveBankConfirmation, saveCcOfRequestTo3rdParty, save3rdPartyInfo, saveSubsidiaryAuditorInfo, saveClientInfo,* and *queryClientInfo*. These are the sub-processes or activities deployed on independent network infrastructure, connected through the Internet to provide the required functionality to the E-dok.

(ii) Identify business partner: The business partners identified are the external entities *auditor* and *client*, which can access the system, *E-dok*, to send or receive data.

For example, the *auditor*'s query is modelled (see Figure 16).

According to the pattern (see Figure 17), an attacker (i.e., a threat agent) may be able to place a request for the service from a large number of computers simultaneously. It is possible to attack the server because the protocol (e.g., DNS, TCP or ICMP) is able to handle an unlimited number of service requests [24]. The result of the risk event is that E-dok, the server, becomes incapable of handling simultaneously received multiple requests, and the business services become unavailable [64].

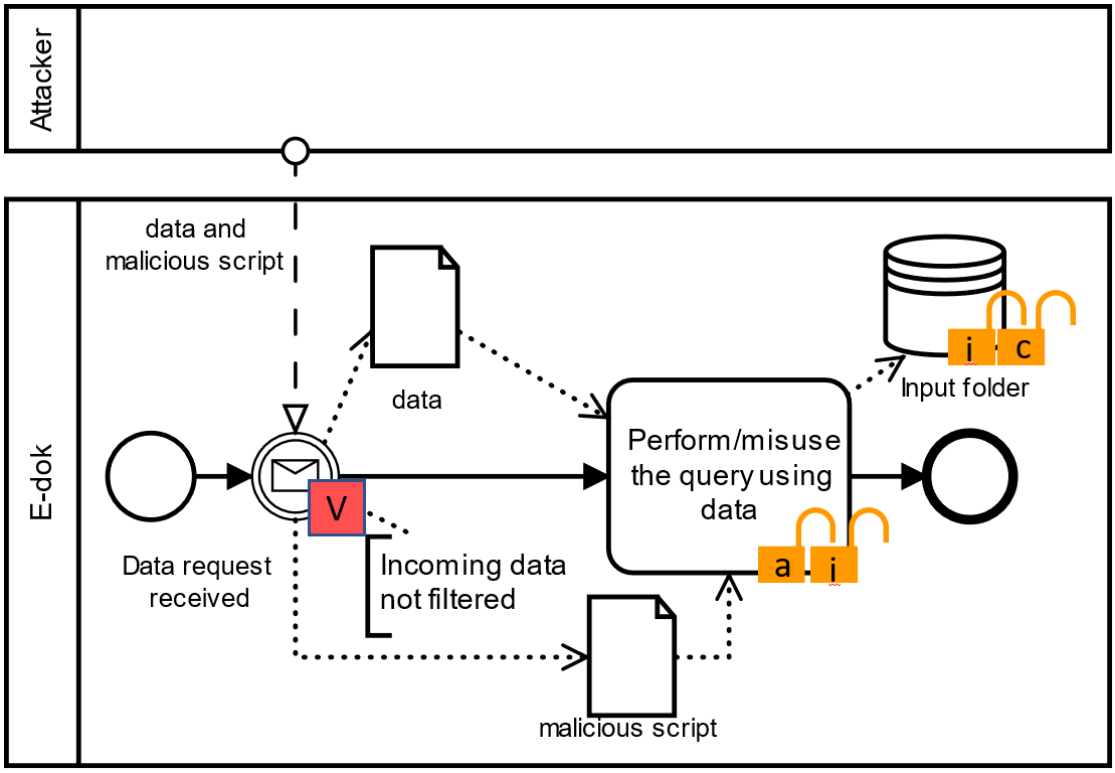

Figure 14. Risk modelling

Security requirements for detecting abnormal requests have to be implemented to reduce the probability of possible DoS attacks (see Figure 18). The security model defines the types of firewalls [96] – stateful firewall, proxy-based firewall, and packet filter firewall. In addition, it provides for mitigation of the risks the requirements for the example as follows:

SecurityRequirement#4.7: *QueryClientInfo* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.9: *Packet Filter Firewall* should filter the *auditor*'s address to determine if that is not a host used by the threat agent.

SecurityRequirement#4.17: *Proxy Based Firewall* should communicate to the proxy representing *queryClientInfo* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.19: *State Firewall* should maintain the *state table* to check the *auditor's* request for additional conditions on established communication.

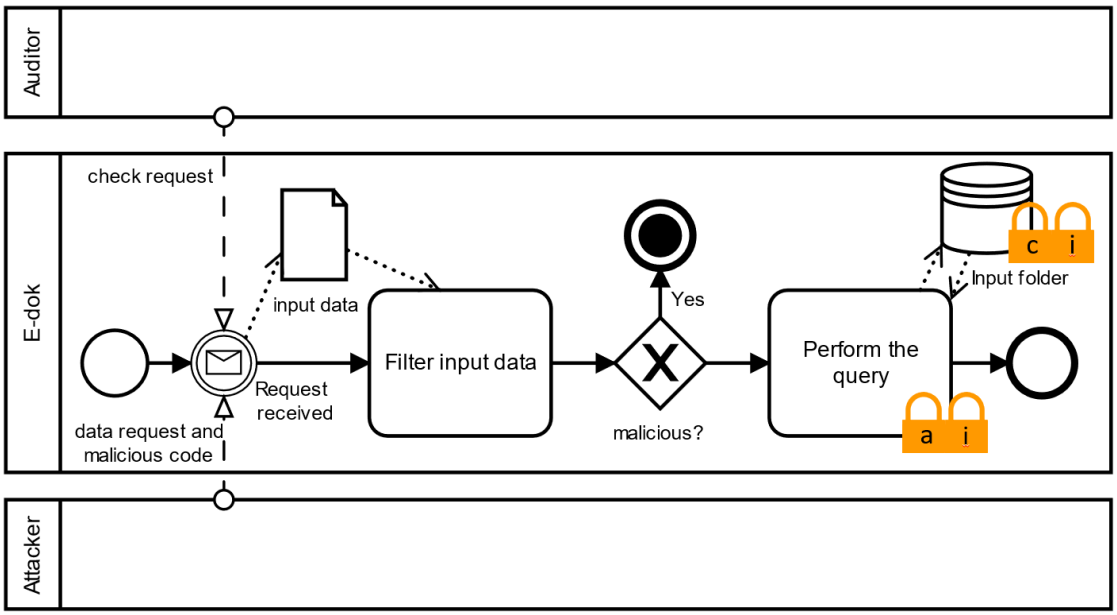

Figure 15. Risk treatment modelling

Similar requirements should be taken into account when *E-dok* sends messages back to the *Auditor*, as the communication between the *Auditor* and *E-dok* is bidirectional. The entire list of security requirements of pattern four for the scenario is provided in Appendix I, security requirements #4.

3.2.5 Pattern Five: Securing Data Stored In/Retrieved From The Data Store

The pattern aims to ensure the data privacy of the data store against insiders, e.g., administrators [5]. It is done by defining how data are stored and retrieved from data stores. It is assumed [64] existence of a storing-retrieval interface (a system asset) that helps to store the data of clients (business asset) in a data store and to retrieve them according to the need. For example, the *auditor*'s query is modelled (see Figure 19).

A malicious insider may exist, i.e., an attacker who has permission to access the data store and retrieve data from it (see Figure 20). In case data is stored and retrieved in a plain format in the storing-retrieval interface, the attacker can access and retrieve the stored data. Hence, their *confidentiality* and *integrity* will be negated [64]. This results in harm to the business assets and their supporting system assets (E-dok).

The access control model to prevent unauthorised access to the database can be determined as follows:

(i) Identify Datastore resource: EngagementFile or *InputFolder* are identified as the resources. *InfoRequest, auditInstructions, bankConfirmation, ccOfRequestTo3rdParty,*

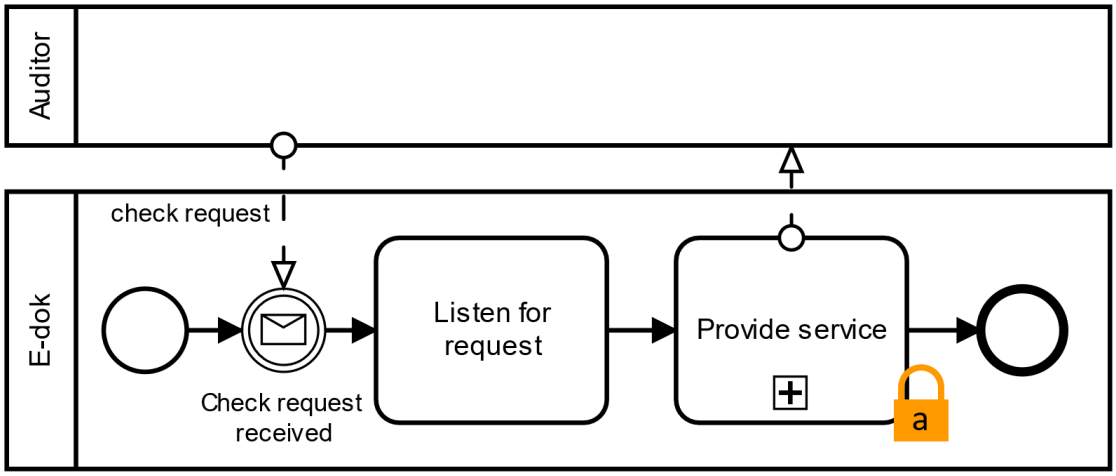

Figure 16. Securing business service against Dos attacks, asset modelling

3rdPartyInfo, and *subsidiaryAuditorInfo* are modelled as resource attributes of *EngagementFile*, and *clientInfo* as an attribute of *InputFolder*.

(ii) Identify Datastore's operations: SaveInfoRequest, saveAuditInstructions, saveBankConfirmation, saveCcOfRequestTo3rdParty, save3rdPartyInfo, saveSubsidiaryAuditorInfo are the operations of *EngagementFile, saveClientInfo,* and *queryClientInfo* are the operations of *InputFolder*.

The result of modelling resources and operations, performing the activities identify roles and assign permissions, is the security RBAC model as presented in Figure 9 for *EngagementFile* and *InputFolder*.

As soon as the access control policies are set, it is possible to establish security requirements for hiding data before storing them in the data store, making them visible after retrieving them from the data store, and monitoring and auditing (see Figure 21). Therefore, the security requirements listed in Appendix I, security requirements #5, and for this particular example, as follows should be taken into account:

SecurityRequirement#5.3/5.4: The *E-dok* should perform operations to hide/unhide data when stored/retrieved to/from the *InputFolder*.

SecurityRequirement#5.6: The *E-dok* should audit the operations after retrieving, storing or other data manipulation in the *InputFolder*.

SecurityRequirement#5.3/5.4 could be implemented using cryptographic algorithms [64]. Auditing (SecurityRequirement#5.6) in this particular context, with the support of access control rules, is the process of keeping logs and monitoring certain events and activities [77]. This is essential for identifying security violations performed against *clientInfo*,

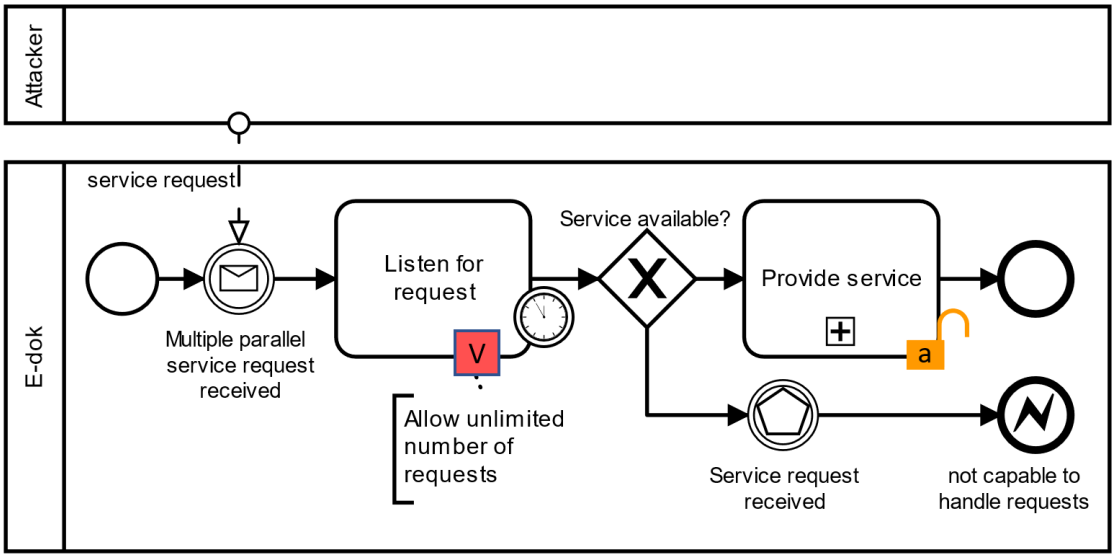

Figure 17. Risk modelling

in the case of the example, and other identified Datastore resources in the case of the scenario.

3.3 Summary

This chapter presents a case description of the collection of audit information by a financial auditor using the audit software E-dok. The question **RQ 2, What are the security requirements for the audit process?** was answered. This question was split into sub-questions:

RQ 2.1: What are the assets? Section 3.2 identifies as valuable business assets an information request, audit instructions, info from the client, bank confirmation, copy of the request to a third party, info from the third party, and reporting of the subsidiary auditor. The IS assets supporting the business assets are the engagement file and input folder.

RQ 2.2: What are the risks when using conventional technology? The following risks were identified in Section 3.2: unauthorised users could access the system asset, the transmission channel could be intercepted, data could be misused (e.g., modified and sent to the E-dok), a threat agent could submit data with malicious scripts, a Denial-of-Service (DoS) attack, and the data privacy of the data stored against insiders.

RQ 2.3: What are the means to mitigate the risks identified when using conventional technology? In the research, the following tools are recognised to mitigate the risks identified: implementing centralised access control mechanisms, implementing the

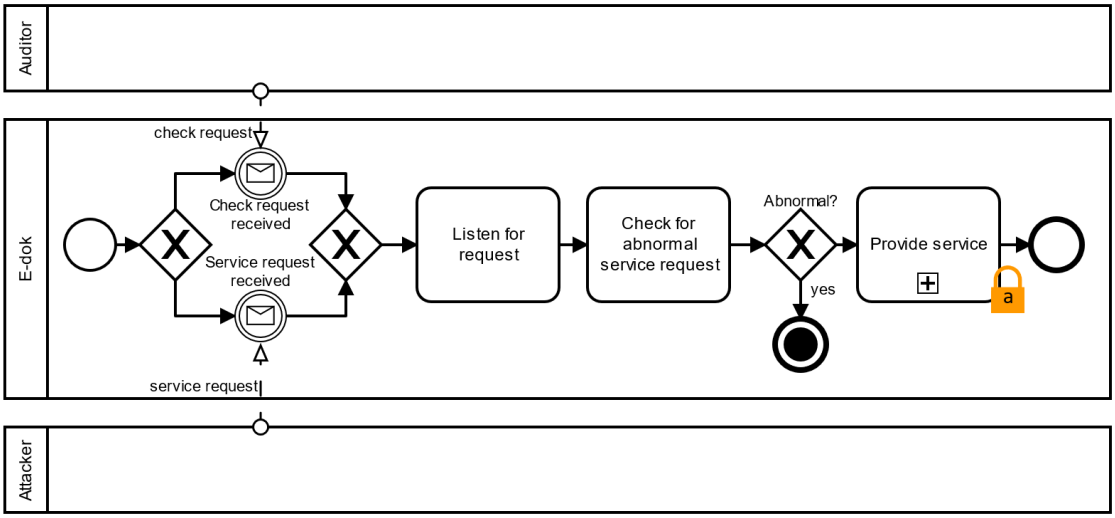

Figure 18. Risk treatment modelling

standard Transport Layer Security (TLS) protocol, filtering the incoming data, including input filtration, sanitisation and canonicalisation, installation of firewalls to monitor traffic and control abnormal requests, implementing centralised access control mechanisms, and using cryptographic algorithms.

The list of security requirements elicited in the thesis is presented in Appendix I. The results of the analysis made in this chapter can be extended to all audit processes where audit software is used.

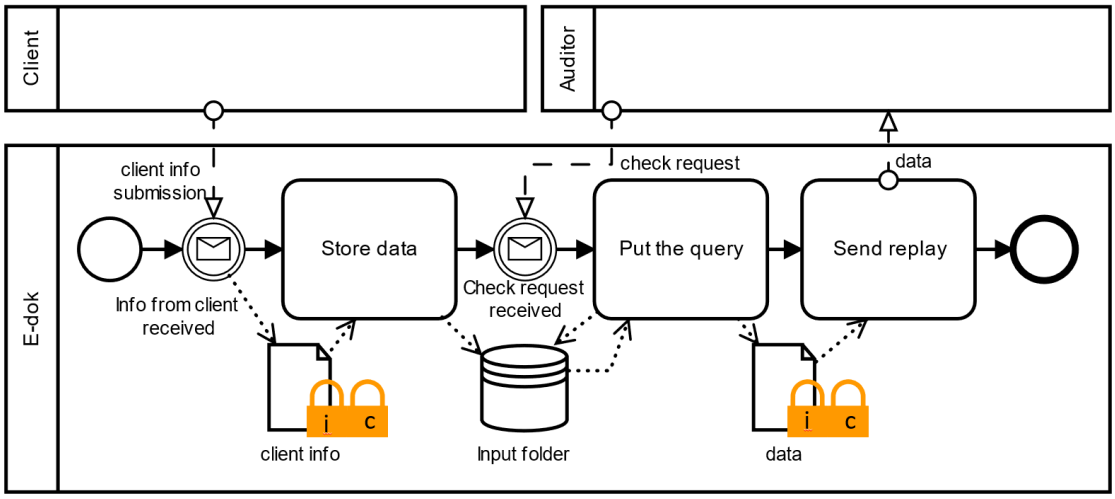

Figure 19. Securing data stored in/retrieved from the data store, asset modelling

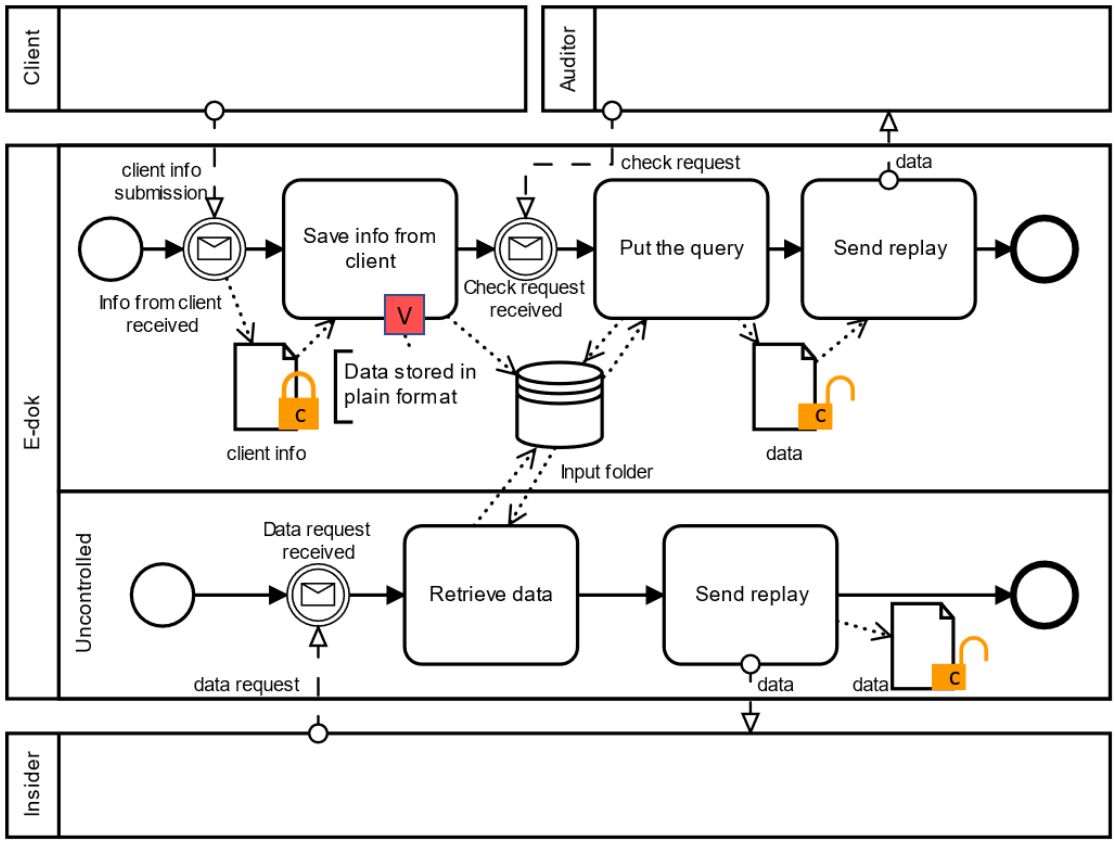

Figure 20. Risk modelling

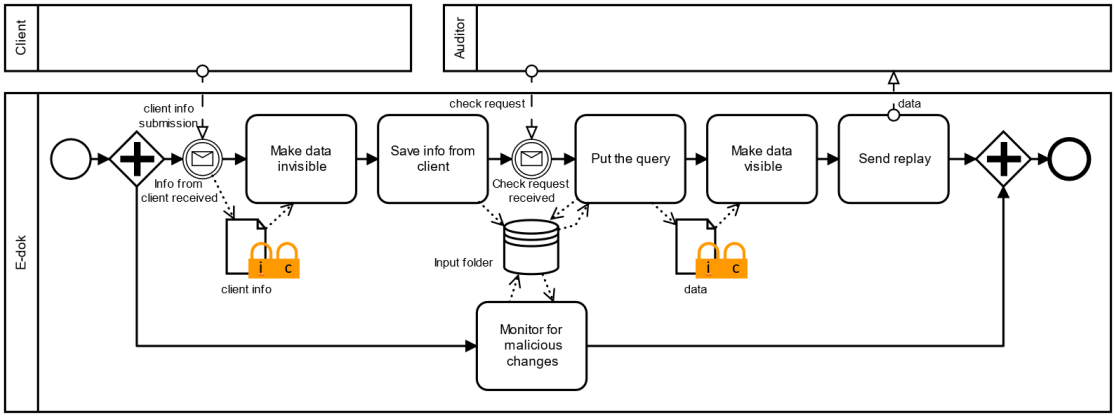

Figure 21. Risk treatment modelling

4 Blockchain-based Auditing System

This chapter deals with transferring one part of the process described earlier – the process of receiving information from the client – to the blockchain. It addresses the research question **RQ3, How does blockchain help to avoid security risks in auditing processes?**. The question has been divided into four sub-questions:

RQ 3.1: What is the suitable blockchain?

RQ 3.2: How does the blockchain mitigate the risks identified in the audit process?

RQ 3.3: What are the new risks associated with implementing blockchain technology in the audit process?

RQ 3.4: What are the means to mitigate the identified risks associated with implementing blockchain technology in the audit process?

The first section compares blockchain platforms. The second section analyses the need to use blockchain technology in the scenario. Also, the blockchain technology to be used has been selected. In the third section, the process of receiving audit information from the client is moved to the Corda platform. The fourth section has analysed the impact of the adoption of blockchain on the security risks identified in Chapter 3. It also analyses the security risks of the process on the new platform and how to mitigate them.

4.1 Blockchain Platforms

The handling of financial audit information can be considered to be similar to the processing and transmission of financial sector transaction information. While screening the different platforms that have been most commonly used for similar purposes, it can be seen that four platforms are mainly used – Corda, Hyperledger Sawtooth, Multichain and Ripple. This subsection compares, based on the literature, these four different platforms to select the most suitable platform to migrate the E-dok input folder to blockchain technology in the following section. In order to find the right platform to build the application, the essential platform characteristics have been compared in Table 2.

R3's Corda (Corda) is a blockchain application development platform that facilitates the creation of multilateral applications that promote and ensure digital trust amongst participants within regulated market environments [28]. Corda's smart contract structure relies on computer code execution, with human control and input, allowing legal enforcement whenever desired [8, 100].

Hyperledger Sawtooth, a blockchain project owned by Hyperledger, serves as a solution for developing distributed ledger applications and networks [9]. As an enterprise solution [81], Sawtooth facilitates building, deploying, and operating blockchains by

Table 2. Comparison of blockchain platforms

	Corda	Hyperledger Sawtooth	Multichain	Ripple
Purpose of the platform	Regulated industries [28]	Enterprise blockchain [50, 81]	Enterprise blockchain [73]	Money transfer network for financial services industry [91]
Public or private	Private [8]	Private, public environment possible [9]	Private or public [39]	Public blockchain [119, 46]
Security and privacy	Data shared between the parties to a transaction. The communication protocol is not visible to others [115]. Includes firewall application [8]	Anyone can create a cluster of nodes within a separate channel on the network, creating a private session specifically for those nodes [9]	The decryption key for the data is distributed to the intended users [114]	Transactions are secure since the addition of transactions to the ledger requires agreement from the majority of ledger holders for verification [91]
Access permissions	R3 members use identities to represent nodes in Corda blockchain. Each identity is issued a certificate by the network which includes the user's signature and real-world name [8]	Sawtooth nodes can form clusters with separate permissioning. Blockchain stores permission settings, including roles and identities, accessible to all network participants [50]	Permissions are created by the network administrator. Granting and revoking privileges is made using network transactions containing special metadata [23]	For members [52]
Enterprise scalability	Pluggable architecture [8]. P2P architecture enables high levels of network scalability and throughput. Transactions don't have to be sequential [115]	Parallel transaction execution. As a result, scaling up is easy for this platform [9]. Modular structure [9]	Unlimited blockchains can be deployed per server for cross-chain applications, providing owners with complete control over their open digital ledgers. [39]	XRP Ledger: Scalable: handles 1,500 transactions per second; Fast: transactions settle in 3-5 seconds; Low-cost: $0.0002 average transaction fee [119]

	Corda	Hyperledger Sawtooth	Multichain	Ripple
Sustainability	Minimal infrastructure carbon footprint by design [28]. Energy consumption per transaction 700,000 times less than Ethereum [115]		Although it is a fork of the Bitcoin blockchain, mining blocks is less expensive because it is done through delegation and not proof of work [23]	The XRP Ledger has an energy efficiency 61,000 times greater than proof-of-work blockchain systems [119]
Consensus	The blockchain technology from R3 provides multiple consensus services called notary pools. Each notary pool on the global Corda network offers distinct services, allowing users to select and use the desired notary pool [8]	Sawtooth's dynamic consensus segregates consensus processes from transaction semantics and supports diverse, interchangeable consensus engines that communicate with validators via the API, allowing for algorithm alteration post-network creation [50]	Mining is executed by a group of admins on the network using distributed consensus between identified block validators. To avoid a minority monopoly on mining, mining is restricted to a set of identifiable entities, with only one validator per block [23]	XRP Ledger Consensus Protocol [51]: 80% network approval needed for transaction or consensus changes. Ripple holds equal rights with 150+ validators, including 35+ in the default Unique Node List, of which 6 are Ripple-affiliated [38]
Distinctive features	Corda's smart contract structure creates an agreement where execution depends on computer code with human control and input, and can be legally enforced [8] SQL and Oracle databases are available for businesses [8]	Separation between the application level and the core system [50]	Scaling with off-chain data [114]	Instant cross-border payment settlement [119] Versatile exchange network. The Ripple network can, in addition to processing XRP transactions also be used for other cryptocurrencies and fiat currencies [91]

offering a modular, flexible platform for transaction-based state updates coordinated by consensus algorithms among untrusted parties [48]. Moreover, the clear separation between application and system levels simplifies development and deployment, permitting native business logic and smart contract virtual machines to coexist within a single blockchain [50].

Multichain is designed to create and operate private blockchain applications within or between organisations [23]. It employs dual-chain data storage methods, allowing for both on-chain and off-chain published information, as desired [114].

Ripple, a remittance network specifically designed for the financial services sector [91], incorporates the XRP Ledger and its native digital asset, XRP, which are optimised for exceptional speed, cost-efficiency, scalability, and environmental sustainability. These features facilitate developers and enterprises in revamping existing applications and investigating novel user experiences in domains such as DeFi, payments, currency exchange, identity, tokenisation, NFTs, and beyond [120]. By employing XRP in cross-border payments, organisations can facilitate currency conversions and ascertain transactions in local currencies on both sides, with settlement times as brief as 3 seconds [119]. Furthermore, Ripple's network is capable of processing transactions involving other cryptocurrencies and fiat currencies [91].

As Ripple is a platform primarily for creating cash payment and currency exchange applications, it cannot be considered the best choice in this case and has been excluded from further comparison. Instead, the selected characteristics of the three remaining blockchain technologies are examined below.

All the considered are **open-source** blockchain platforms [8, 50, 9, 114, 119].

Security and privacy: Corda exclusively shares data among transaction participants, rendering the communication protocol itself imperceptible to other network members [115]. Consequently, any transaction between two parties remains visible solely to them and those involved in the consensus, as they must verify it for the ledger's integrity [8]. In addition, Corda blockchain members utilise cryptographic hashes, which enable data and user identification, besides linking a transaction to its predecessor to maintain the chain [8]. Corda also incorporates a robust firewall application for enhanced security [8].

In the context of Hyperledger Sawtooth, individuals can establish a cluster of nodes within an isolated channel on the network, generating a private session explicitly designated for those nodes [9]. As for Multichain, data replication does not occur across all nodes, and the decryption key for a given dataset is disseminated only to the intended recipients who are authorised to access the information [9].

Access permissions: According to Anwar [8], Corda blockchain members use identities to represent the node. The use of digital identities is important because financial transactions need to be trustworthy. Therefore, the network issues a certificate containing the user's sign and real-world name in a Corda blockchain identity.

Sawtooth node clusters can be implemented easily with individual permissions. In a blockchain, settings that define permissions, like identities and roles, are stored in a way that all network participants can access this information [50]. The granting and revocation of multichain rights is done by network transactions containing particular metadata [23].

Scalability: Unlike many other software developments, performance is not a critical factor in the context of the case considered in this thesis, i.e. the E-dok input folder. There are not many transactions, and even if they take several minutes to complete, this does not usually cause problems, as the various parties carry out the operations (submitting or processing information or documents) according to the schedule agreed with them. Some data on the performance of different platforms are available in the literature, but the conditions in which they are measured vary. Therefore, these results are not directly comparable. Given these circumstances, the performance of the platforms has not been compared in this analysis. However, other factors affecting scalability are mentioned here.

According to Anwar [8], Corda pluggable functions help improve scalability, privacy, availability, compatibility of ledger-system and algorithmic flexibility. Byzantine's fault-tolerant consensus can be used to create any single service with equally mutually interfering parties. All uniqueness consensus services or non-validating notaries are here to determine whether any prior transactions consumed the state. They cannot validate transactions themselves on the Corda blockchain platform. Thus, these notary pools cannot see any transaction's entire content, thus preserving that transaction's privacy and scalability [8]. In addition, transactions do not need to be serial, which increases the system's overall efficiency [115].

According to the documentation [50], Sawtooth encompasses an advanced parallel scheduler segregating transactions into concurrent flows. This scheduler isolates transaction execution based on the accessed state locations while preserving contextual modifications. As a result, transactions are executed in parallel whenever feasible, effectively preventing double-spending, even in instances of multiple alterations to the same state. Consequently, parallel scheduling has the potential to yield substantial performance improvements over sequential execution [50]. Moreover, the modular architecture endows businesses with a distinct form of flexibility. This functionality allows developers the freedom to employ any consensus algorithm or desired features on the platform, exemplifying a plug-and-play scenario [9]. With Multichain, an unlimited number of blockchains per server can be used for cross-chain applications, giving IT managers complete control over their open digital ledgers [39].

Sustainability, in the context of this paragraph, refers to the size of the infrastructure's carbon footprint (energy consumption). This topic is covered more in the description of Multichain. This is because Multichain is a fork of the Bitcoin blockchain. Centieiro [23] argues that mining in the Multichain blockchain is less costly and more

environmentally friendly than in the Bitcoin blockchain because it is done through delegation and not through proof of work.

Corda has, by design, a minimal infrastructure carbon footprint [28]. The Corda web page [115] argues that Corda consumes approximately 24.6 joules of energy per transaction, whereas the corresponding consumption per Ethereum's transaction amounts to 17,222,222 joules.

Consensus: Corda provides for global consensus distribution the following three tools [8]:

- Timestamp and uniqueness services, known as Notary pools.
- Smart contracts, which allow users to transact according to pre-agreed rules.
- The Flow Framework, which facilitates the process of writing complex protocols between non-trusting users [8].

Pursuant to the documentation [50], Hyperledger Sawtooth separates consensus processes from transaction semantics. Its consensus interface supports the integration of different consensus engines that communicate with the validator via the consensus API. Notably, Sawtooth allows for consensus algorithm alterations even after establishing a blockchain network. The consensus algorithm is chosen during the initial configuration of the network and can be modified in one or two transactions for a running blockchain.

According to Centieiro [23], the mining in MultiChain is performed by a group of administrators on the network. It uses a distributed consensus among identified validators of the blocks. In order to avoid a minority monopoly in the mining process, mining is limited to a set of identified entities. There is only one validator per block. Each allowed node can create new blocks after a random timeout, given the diversity parameter, ranging from 0 to 1 [23].

Speaking of **users**, notable Corda BaaS vendors [8] include AWS Blockchain Templates (Amazon) [56, 16], Microsoft Azure [16], Accenture, and Hewlett-Packard Enterprise. In addition, users of Corda are, e.g., ING, Bank of Canada (the central bank), National Bank of Canada, Payments Canada, Royal Bank of Canada, HSBC, Bank of Montreal, CIBC, TD Bank, and Scotiabank; Monetary Authority of Singapore (acts like central banks); Hong Kong Monetary Authority and Bank of Thailand (together with nine banks); ING; and SWIFT [8]. The largest Hyperledger Sawtooth implementers are T-Mobile, State Bank of India with over 27 members on their platform and other financial institutions of India and companies of Middle East [9]. The most prominent implementer of Multichain is SAP [71].

4.2 Selecting Blockchain Platform for Auditing System

Before choosing a platform, it is necessary to analyse whether the use of blockchain technology is appropriate. Therefore, it has been done in this section. Also, a blockchain

platform has been selected to migrate the E-dok audit software input folder to the blockchain.

Using blockchain only makes sense if mutually mistrusted entities want to communicate and change the system's state and cannot use an online trusted third party [118]. In this section, the Wüst and Gervais [118] methodology has been used to analyse the need for blockchain and the type of blockchain suitable for the situation. In order to analyse whether blockchain technology would be appropriate to address the security risks described in Chapter 3, the questions in the flowchart in Figure 22 need to be analysed.

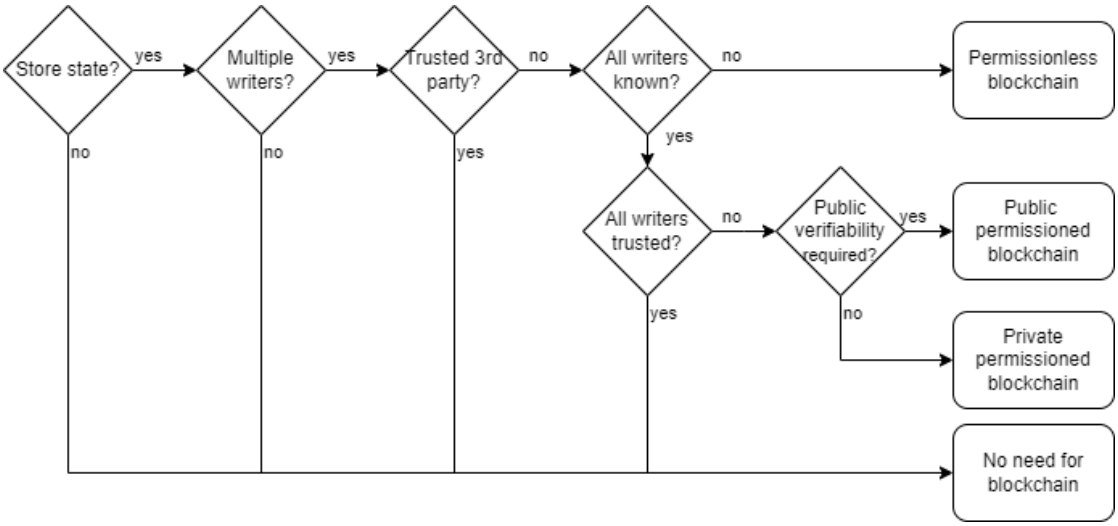

Figure 22. Flowchart to analyse the appropriateness of blockchain implementation (adapted from [118])

The analysis starts by determining whether a state needs to be stored. All of the auditor's engagements are documented, and the evidence is recorded by the auditor in the E-dok engagement file. Client representatives store information and documents for submission to the auditor in the E-dok input folder, from where it is transferred to the engagement file. The same document or a file with similar content may be submitted more than once. Where corrections are made, this may be repeated. Therefore, the auditor needs to know when and what information has been provided. Also, what is the latest version that may be used in the engagement? In addition, in some cases, client communications and engagement management may need to know when and what information was received. Hence, in the described case, we need to know the state. As can be seen in Figure 22, all different types of blockchain platforms enable storing state.

The next step is to analyse whether there are multiple writers. The writer corresponds to a write access entity in a traditional database system and a consensus participant in a blockchain system [118]. There are multiple writers – the auditor together with the

engagement team, accountants of the client, and accountants of the client's subsidiaries; other employees submitting the info on behalf of the client; other auditors auditing the subsidiaries of the client; client banks, advisors, suppliers and customers of the client. However, not all of them are submitting info and documents to the E-dok. Many of them do that usually by e-mail. In the case of the engagement file, the auditor is the writer. I.e., there is no need to replace the current setup with a blockchain. In the case of the input folder, the different client entities are the writers and this is the auditor as well. Hence, in this case, the existence of multiple writers points to the need for blockchain technology.

The servers where the input folder is located, and thus, the info and documents stored, are known. However, there is no certainty that this information is not accessible by someone to whom the auditor has not given access rights, such as administrators. It is, therefore, not possible to use an always online trusted third party, which again indicates the need for blockchain technology.

The set of writers is fixed and known for the auditor granting access to all parties, including all client entities. For example, in the structure shown in Figure 1, the parent company's chief accountant and the subsidiaries' accountants. As a result, as shown in Figure 22, permissionless blockchains are excluded from the choice of blockchain technologies, leaving only permissioned blockchain technologies.

The fact that the auditor knows all the writers does not mean they can be trusted. Therefore, the auditor has to be sceptical because the client may have made mistakes (errors) or they may have deliberately distorted information (fraud). Therefore they are audited. The application of blockchain technology is, therefore, also appropriate.

The last item to be analysed determines whether public or private permissioned blockchain technology should be implemented. In the case of auditor engagements, an engagement relationship is established between the auditor and the client. The auditor performs assurance engagements in the public interest, but the auditor assesses the sufficiency and appropriateness of the information gathered and is responsible for the work. Thus, there is no need for public verifiability in the present case. It is sufficient if the client carries out the verification as the information provider and the auditor as the recipient. Hence, according to the analysis, a private permissioned blockchain would be a technically suitable solution (see Figure 22) to replace the current technology of the E-dok input folder.

As seen from the descriptions in section 4.1, three platforms are inherently possible – Multichain, Corda and Hyperledger Sawtooth. Therefore, in this case, the choice has been made according to the primary intention of the platform. While Multichain and Hyperledger Sawtooth are designed to meet the needs of businesses in a wide range of industries. Corda is designed for *regulated* businesses and markets. Financial auditing is a regulated profession, making Corda the appropriate choice for creating applications for this sector. This selection is underpinned by Corda's security and privacy strengths, its smart contracts and consensus mechanism, and access permissions. In the

following section, the Corda platform is implemented. It also takes a closer look at the key components of Corda.

4.3 Corda for E-dok

Section 3 presents a scenario of acquiring financial auditor engagement information using the audit software E-dok. It introduces the design currently in use. The associated security risks are also discussed. The previous sections have identified the appropriateness of using blockchain in this process. It is also established that a private permissioned blockchain solution is required. Further analysis has led to the selection of Corda as the applicable platform. The objective of this section is to migrate one part of the described process, which is the acquisition of client information, to the E-dok input folder, a low-functionality folder for storing files, to decentralised infrastructure and explore the conceptual components of the platform.

Corda [18] is distributed ledger software designed for the financial industry and, more broadly, regulated industries, developed by R3. It is an open-source platform[6] that allows for the creation of decentralised applications, or "CorDapps", businesses or organisations can develop that to automate specific processes and workflows, such as the exchange of information between clients and auditors in the case of the E-dok input folder. Furthermore, Corda is unique in its focus on the privacy and security of the transaction, uses a unique consensus algorithm, and allows for selective sharing of data between participants in the network, which sets it apart from other blockchain platforms. In this way, Corda is a suitable platform to migrate the E-dok input folder and the reception of information to blockchain technology.

With the implementation of CorDapp, the business process model for information delivery will change from the traditional information delivery infrastructure architecture to the architecture of Corda nodes, as to how nodes in the Corda network operate and how information is exchanged (see Figure 23) [52]. This allows for using Corda components such as contracts, states, and vaults and flows for information delivery [55].

The parties to the process are the client, who provides the information and documents and the auditor to whom these are provided. The auditor and the audit team have been considered as one entity in this case. As shown in Figure 1, the client organisation may consist of several entities. Therefore, there may be more than one party submitting information and documents on behalf of the client. In the Corda-based business process model, Figure 23 shows the client as one pool and the auditor's E-dok input folder CorDapp as one pool.

In Corda, a network of peer-to-peer nodes facilitates the secure exchange of information between entities. The Corda network is marked in the business process model in Figure 23 as a separate pool through which information flows between entities. Each

[6]The codebase is available on GitHub https://github.com/corda/corda

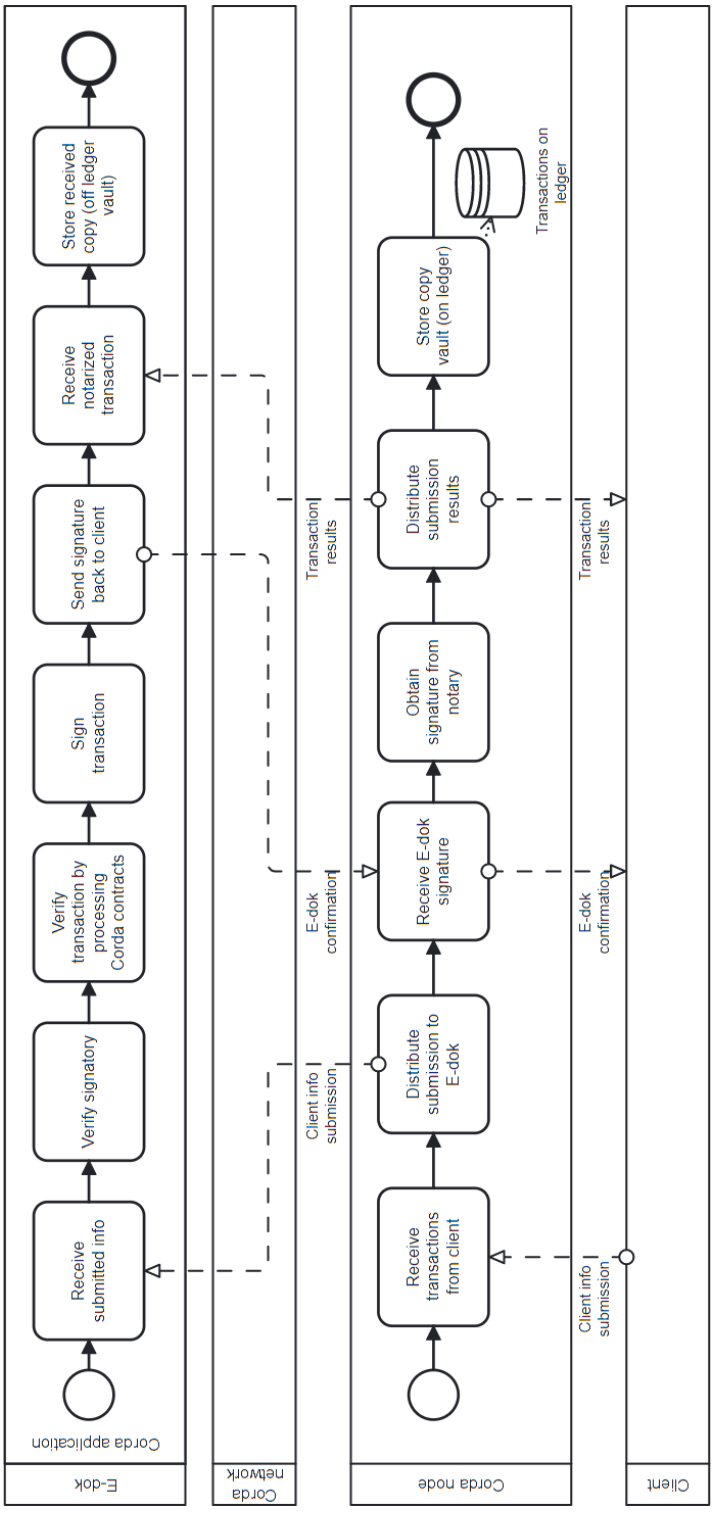

Figure 23. Process of obtaining engagement info from a client using CorDapp

node in the network represents an individual user or entity [79], such as the auditor and the client. Since the Corda nodes involved in a transaction replicate information about that transaction among themselves, which means that they have the same information about that transaction, only one Corda node is depicted in the business process model in Figure 23. An entity must first set up a Corda node on its system and join the Corda network, for which a certificate must be obtained from the network operator to become a user of Corda [78]. After that, the entity can participate in transactions, exchanging information between entities in the Corda network. For example, the submission of information by a client to the E-dok input folder for the auditor can be considered a transaction in Corda, as it involves the transfer of information from one entity to another [110].

In Corda, transactions are recorded as states, representing the current state of the information being exchanged. States can be created, updated, or consumed during transactions [110]. For example, in the E-dok input folder, a state could represent the submission of information by a customer to the auditor. The state could be created during the transaction and updated if the client submits an updated version of the same information and when the auditor reviews and approves the information. The final state would represent the approved information, securely stored in the E-dok input folder.

States in Corda are encrypted representations of information exchanged between entities in the Corda network [18, 106]. These states are stored on the nodes participating in the transaction and recorded on a distributed ledger, which provides a secure and transparent platform for exchanging information. In the business process model shown in Figure 23, storing the state in the ledger is the final task of the Corda node pool. The visibility of data on the ledger is restricted to authorised parties only [60]. It is important to note that each node in the Corda network has a different view of the ledger, as each node keeps its vault containing all of its known facts [60]. This decentralised approach ensures that sensitive information is only shared with authorised parties and provides an additional layer of security for the information being exchanged in the E-dok input folder. In the context of the present case, the rights of the different information providers may be different – the parent company's chief accountant may see the information provided to the auditor by the subsidiaries' accountants. However, the subsidiaries' information providers may not see the information provided by the chief accountant. The auditor sees the information provided to him.

There are two types of consensuses in Corda: validity consensus and uniqueness consensus [18, 52, 57]. Validity consensus ensures that the transaction being executed follows the rules set by the Corda network, such as having the required signatures, and correct input and output states, by verifying that the associated contract code executes successfully [26]. For example, in the E-dok input folder CorDapp case, the counterparties to the transaction are the validators. In the business process model shown in Figure 23, these tasks in the CorDapp pool are *Verifying signatories* and *Verifying transactions by processing Corda contracts*.

On the other hand, the uniqueness consensus ensures that a particular state can only be consumed once. Instead of organising the timeline into a chain of blocks, Corda uses notary clusters, providing transaction ordering and timestamping [44]. Notaries, a particular type of node in the Corda network, act, among others, as impartial third-party witnesses to transactions, ensuring that the transaction does not conflict with any existing states and, in case of validating notaries, also verifying the transactions' validity [80]. For example, in the case of the E-dok input folder, the notary is responsible for ensuring that transactions contain unique input states and timestamping of transactions. In the business process model shown in Figure 23, these steps are performed in the *Obtain signature from notary* task of the Corda node pool.

In Corda, smart contracts represent agreements between transaction parties, with the conditions directly incorporated into the code, executing automatically [100]. In the implementation of Corda on the E-dok input folder, the smart contracts include the terms and conditions of exchanging information between the auditor and the client. For example, the condition that the transaction must be accompanied by an attachment providing the information to the auditor. These smart contracts are written in code and are automatically executed when the specified conditions are met.

Flows in Corda are sequences of steps that define the flow of information between nodes in the network [40]. They facilitate transactions between entities in the Corda network. They are made up of a series of steps that include sending and receiving messages, updating states, and triggering the execution of smart contracts [10]. For example, in the case of the E-dok input folder, when a client provides information to the auditor by submitting it to the Corda-based E-dok input folder, the flows used would be responsible for the communication between the client and the auditor node. Also, to create a new state to represent the submitted information and the execution of the relevant smart contract to enforce the terms and conditions of the exchange of information. The use of flows in Corda provides a standardised and structured way of conducting transactions, ensuring that all parties involved in a transaction are following the same steps and that the process is transparent and secure [44].

Attachments in Corda are additional files or data that are not stored along with the transactions on the ledger but outside it [116]. These are used to store any additional information required to complete a transaction but are too large to be stored directly on the ledger [12]. The transaction contains only a hashed reference to the attachment for referencing data [12]. In the context of the Corda-based E-dok input folder, attachments are used when a client provides information to the auditor. Instead of storing the information on the chain, the client can attach the necessary documents and files to the transaction, making it easier to transfer the information. The content of the attachment is not validated or controlled by Corda. It remains the auditor's task, which will be performed during the subsequent audit.

CorDapps, or Corda decentralised applications, are software applications designed

for the Corda platform, which interact with its underlying blockchain technology [113]. A copy of CorDapp must be running on the Corda node of all parties wishing to transact in the network [29]. These applications automate various business processes by incorporating smart contracts, flows, and other components [113] that ensure secure, transparent, and auditable transactions. The E-dok input folder CorDapp is one such example. The CorDapp functions as the user interface for the client, allowing them to input and submit information to the auditor. The application then implements the necessary smart contracts and flows to validate the information and securely transfer it to the auditor via the Corda network. Moreover, the transaction is recorded immutably on the Corda ledger, providing a tamper-proof and auditable platform for exchanging information with the E-dok input folder.

In summary, the process for submitting information moved to the Corda platform is as follows: The process is initiated by the client who has done the necessary preparatory work to provide the information – (i) created the transaction by extracting the initial states, if necessary, creating the output states and adding attachments, and (ii) signed the transaction. The process in the Figure 23 input, which is the information to be presented, is submitted by the client (Client pool) through the Corda node (Corda node pool tasks *Receive transactions from client* (receiving) and *Distribute submission to E-dok* (distributing)) via Corda network (Corda network pool) to the E-dok input folder CorDapp (E-dok pool task *Receive submitted info*). Next, the transaction is validated by E-dok CorDapp by verifying signatures (E-dok pool task *Verify signatory*) and processing Corda contracts (E-dok pool task *Verify transaction by processing Corda contracts*). The transaction is then signed in E-dok CorDapp to confirm it (E-dok task *Sign transaction*). The signature is distributed to the client (E-dok pool task *Send signature back to client*, Corda network pool, Corda node pool task *Receive E-dok signature*, Client pool), after which the validation process is completed by obtaining signatures from notaries (Corda node pool task *Obtain signature from notary*). Finally, validated information is shared via a peer-to-peer network (Corda node pool task *Distribute submission results*, Corda network pool, E-dok CorDapp pool task *Receive notarised transaction*, and Client pool) with the parties and stored in the distributed Corda ledger (Corda node pool task *Store copy vault (on ledger)*).

This process change systematises the information received by the auditor. It thus affects the time spent sorting the information and identifying the correct file versions for use in the engagement. Tracking the information submitted will be simplified for the auditor and the client, allowing the client to rely on a new structure to keep track of the information submitted. There will also be a single point of reference where the files submitted are captured in an immutable form. Thus, no one can intentionally or accidentally change the information stored in the Corda ledger.

The following section analyses the impact of introducing the new model created on the Corda platform on the security risks of collecting customer information in the E-dok

input folder, as identified in Section 3.

4.4 Security Risk Management of Corda-Based E-dok System

In the previous section, the reception of customer information in the E-dok input folder has been moved to the Corda platform. The first subsection of this section analyses the impact of such a model change on the security risks of the information collecting in E-dok identified in Section 3. The second subsection discusses the security risks and mitigation options arising from introducing a new Corda-based E-dok input folder.

4.4.1 Corda Platform for Security Threat Mitigation

This subsection analyses the mitigation of the security risks identified in Section 3 of the E-dok information collection when implementing a Corda-based architecture.

The Corda platform exploits the advantages of blockchain and also introduces a number of other techniques to secure the process. The Corda-based countermeasures as follows were considered based on the literature [55]:

- Authorised nodes and distributed P2P network;
- Authorised nodes and decentralised access control mechanism;
- Secure communication protocols (mutually-authenticated transport layer security (TLS) connection and public key infrastructure (PKI));
- Notaries-based consensus mechanism;
- JVM sandbox to prevent code executing;
- Rate-limiting firewall;
- Identity Manager Service validate requests and filter malformed requests;
- A traceable, immutable, distributed ledger and distributed action logs.

A comparison of the countermeasures against the identified security threats in the traditional E-dok information collection architecture and when using Corda-based technology is presented in Table 3.

The *first* identified threat is that *unauthorised users could access the system asset*. Traditional infrastructure mitigates this threat by implementing centralised access control mechanisms. It can be subject to errors or attacks due to weak implementation of security policies, password theft or centralisation [37, 65, 108]. The CorDapp provides a built-in decentralised access control mechanism [55] that protects against this threat and prevents

Table 3. Comparison of centralised and Corda-based countermeasures implementing security requirements for mitigating threats

No	Threats	Countermeasures	
		Centralised infrastructure	**Corda-based infrastructure**
#1	Unauthorised users could access to an asset	Access control mechanism	Authorised nodes and distributed P2P network
			Authorised nodes and access control mechanism
#2	Transmission channel could be intercepted	Secure communication protocols	Authorised nodes and distributed P2P network
			Secure communication protocols
#3	Data could be misused (e.g. modified)		Notaries-based consensus mechanism
#4	Data could be submitted with malicious scripts	Filtering incoming data	Notaries-based consensus mechanism
	Business rules could be changed or business data read/written		JVM sandbox to prevent code executing
#5	Denial-of-service attack	Firewalls	Rate-limiting firewall
			Identity Manager Service validate and filter requests
#6	An insider could access and retrieve the stored data	Access control mechanism	Authorised nodes and distributed P2P network
			Authorised nodes and access control mechanism
		Cryptographic algorithms	Notaries-based consensus mechanism
		Auditing (logging and monitoring)	Distributed ledger and action logs

unauthorised access. Furthermore, only authorised nodes can join the network [55] on the Corda platform.

The *second* identified vulnerability is that *the transmission channel could be intercepted*. Furthermore, the *third* is, that *data could be misused* (e.g., modified and sent to the E-dok). Traditional infrastructures mitigate these vulnerabilities by implementing the standard Transport Layer Security (TLS) protocol [11, 64] to ensure the integrity of secure connections and data exchanges between client and server [55]. Communication protocols ensure the authentication of the data's origin and encrypt the data being transmitted. However, a weak implementation of communication protocols can be broken [19]. In such a case, an attacker may interfere with the communication and data transfer channel [19].

The Corda platform overcomes this issue by considering only authorised nodes in a

P2P network, where nodes act as both servers and clients [30]. CorDapp also incorporates PKI-based cryptographic schemes for a mutually authenticated TLS connection [55] as an additional security layer to protect the communication between nodes [55]. Against modifying the data, the Corda platform uses a decentralised consensus model based on notaries to validate the transaction and ensure integrity and authenticity [55, 44]. Even if an attacker conducts a Man-in-the-Middle (MitM) attack to alter the transaction, the notary-based consensus model safeguards and ensures the integrity of the transaction [55]. On this platform, unlike other blockchains, transaction information is not shared with all nodes but only with those involved in the transaction, which also limits this vulnerability.

The *fourth* threat identified is that a threat agent could *submit data with malicious scripts*. By running these, change the business rules or read/write the business data. The centralised infrastructure introduces filtering the incoming data [64], including *input filtration, sanitisation and canonicalisation*. The Corda platform uses a decentralised consensus model based on notaries to validate the transaction and guarantee integrity and authenticity [44]. CorDapp also uses secure communication protocols with a sandbox concept [55] to avoid unauthorised remote operations and code execution.

The *fifth* identified is a *Denial-of-Service (DoS) attack* threat, in which E-dok, the server, becomes incapable of simultaneously receiving multiple requests placed by a threat agent, and the business services become unavailable. In the case of centralised infrastructure, this threat is mitigated by the installation of firewalls to monitor traffic and control abnormal requests. CorDapp also implements request rate-limiting firewalls with an Identity Manager Service (formerly Doorman). The identity manager service validates and filters out improperly formatted requests [55]. Moreover, on the Corda platform, the TLS protocol is employed to authenticate P2P communication. This implies that an attacker cannot infiltrate the Corda network for the purpose of executing a DoS attack [55].

The *sixth* identified vulnerability is the *data privacy of the data stored against insiders*. Traditional infrastructure mitigates this vulnerability by implementing centralised access control mechanisms, also by using cryptographic algorithms [64], and by auditing, i.e. keeping logs and monitoring certain events and activities [77]. The CorDapp provides a built-in decentralised access control mechanism [55] that protects against this vulnerability. Furthermore, only authorised nodes can join the network [55] on the Corda platform. The Corda platform uses a decentralised consensus model based on notaries to validate the transaction and guarantee integrity and authenticity [44].

The security of the logs of the present infrastructure is vulnerable and subject to attack [37, 45], as controls remain with the designated authorities (E-dok OÜ under the control of the Estonian Auditors' Association; and Centre of Registers and Information Systems) and centralised storage. The system also logs sensitive information that could leak. In contrast, the Corda platform manages records using a decentralised, immutable ledger. This ensures tamper-proof transparent traceability and auditing. Corda also logs

the activity of each participating node, which is replicated over a P2P network with other authorised participating nodes [55].

In conclusion, the risks identified with the patterns have been mitigated when implementing the Corda platform.

4.4.2 Security Risks in Corda-based Input Folder and their Mitigation

The authors [53, 54] conclude that blockchain technology not only helps to reduce security risks, but also introduces new security risks. Similarly, CorDapp not only mitigates certain security risks (as discussed in section 4.4.1), but it also enables other security risks [55].

In this subsection, a security risk analysis has been carried out in order to identify which security risks may appear in the CorDapp-based E-dok input folder, and how to mitigate the risks. For this purpose, an SRM approach has been applied to identify the business assets, together with their security criteria, system assets, associated vulnerabilities, and security threats (Table 4). A risk treatment has then been performed, presenting security countermeasures against the threats (Table 4).

Table 4. CorDapp-based E-dok input folder security threats and countermeasures

No	Vulnerabilities	Business assets	System assets	Threats	Countermeasures
#1	Lack of knowledge and awareness	CorDapp service (*I, A*) Digital asset (*I*)	User, Devices, Keys	Endpoint vulnerability	Educating system users Hardware security modules
#2	Sharing content with validating notaries	Transaction content (*C*) Client data (*C*)	Consensus, Notaries	Privacy violation	Involve transaction tear-off concept
#3	Linking user account and transaction ID	Counter-parties (*C*)	Validating notary, Transaction (data)	Deanonymisation	Involve transaction tear-off concept
#4	Non-validating notaries consume state	Transaction state (*A*), Digital asset (*I*), Transaction (*I*)	State reference, Node, Ledger, Smart contract	Denial-of-state attack	Zero-knowledge proof and trusted execution environment
#5	Error-prone and faulty smart contracts	Transaction (*I*), Transaction validation (*I*), Digital asset (*I*)	Ledger, Smart contract, Non-validating notary	Smart contract attack	Code analyser for smart contracts to detect errors
#6	No quantum-resistant cryptography	Transaction (*I*)	Ledger, 2P2 network, Cryptography	Quantum computing threat	Implement quantum-resistant cryptography

Table 4 shows the security threats arising from the implementation of the CorDapp-based E-dok input folder. Also, what assets are targeted and what vulnerabilities are exploited. Furthermore, what countermeasures can be implemented to protect CorDapp against these security threats.

Vulnerability and threat #1: *Lack of security awareness and knowledge* can result in *endpoint vulnerability* [112] threat. E.g., physical access to devices, weak passwords, lost keys [55]. In the latter case, if the attacker has obtained a private key, he can use it to gain access to and ownership of the data [52]. This vulnerability could lead an attacker to steal information [15] through phishing, social engineering [2] or a corporate individual exposing secure information unintentionally [65].

As security countermeasures, to mitigate the vulnerability, governance is the starting point. Primarily key management with hardware security modules; creating and managing information security policies and guidelines; and security trainings [112]. Standards need to be set for access and identity management, data management policies, secure key management, new staff onboarding [112], end user policies and more. Organisations should provide security training for staff, including training for system users on the potential security risks when they disclose their protected information [112]. Best practices should also be promoted. Deployment of Hardware Security Modules (HSMs), which are crypto processors, to generate, protect and store keys [112].

Vulnerabilities and threats #2 - #3: With CorDapp, validating notaries see the full content of the transaction and the dependencies of the transaction, that is, its history, in order to validate the transaction; *sharing the content of the transaction with validating notaries* could result in a *breach of privacy* [57]. In CorDapp, *individuals' data can be linked*, which could cause a *de-anonymisation* threat [57, 72]. Koens et al. propose in [57] to use transaction tear-off to protect against the threats. This CorDapp solution enhances privacy by revealing the minimum amount of information that should be kept confidential about the transaction for signing, and hiding the rest of the transaction components [31, 109].

Vulnerability and threat #4: In CorDapp, a **denial-of-state attack** threat exists where a malicious attacker, either an outsider or an insider, who has access to the Corda network, and who knows a particular state reference, could create a transaction and, by considering it valid, *a non-validating notary consumes a state* [57]. Therefore, Koens et al. [57] propose the use of zero-knowledge proofs and trusted execution environments (e.g., SGX by Intel) to protect against the threat.

Vulnerability and threat #5: *Smart contracts susceptible to errors* [97], such as logical flaws, inadequate error handling, input validation, or improper use of programming language, may jeopardise valuable assets. Once smart contracts are implemented on the blockchain, they cannot be modified. Insufficient exception handling and the presence of error-prone smart contracts create opportunities for attackers to *interfere with services* and *inflict damage* on *assets* [52]. To mitigate risks, code analysers should be employed to examine the smart contract code, sanitise it prior to deployment, identify race conditions, and detect errors [13].

Vulnerability and threat #6: The advancement of quantum computing research puts currently used cryptography techniques in blockchain at risk, as they are not equipped to handle the *post-quantum* era [112]. The *quantum computing threat* is a concern [112, 122], but Corda does not have a solution to address this issue. To safeguard against the threat, quantum-resistant cryptography approaches such as multivariate, hash-based, lattice-based, symmetric key quantum resistance, or code-based methods could be adopted [52, 57, 122].

4.5 Summary

In this chapter, the collection of customer information in the E-dok input folder has been moved to the Corda platform. The question **RQ 3, How blockchain helps to avoid security risks in auditing processes?** was answered. This question was split into sub-questions:

RQ 3.1: What is a suitable blockchain? In this chapter, the author has employed the Wüst and Gervais methodology [118] to determine the appropriateness of using blockchain technology in the audit information-gathering process. Through this analysis, it was concluded that a private-permissioned blockchain is well-suited for the process. In addition, a comparison of four blockchain platforms—Hyperledger Sawtooth, Multi-Chain, Corda, and Ripple—was conducted to identify a suitable platform for the task at hand. As a result of this comparison, Corda emerged as the suitable choice due to its security qualities and suitability for use in a regulated profession that align with the audit information-gathering process requirements.

RQ 3.2: How does the blockchain mitigate the risks identified in the audit process? The blockchain technology, specifically the Corda platform, mitigates the risks identified in the audit process by offering decentralised access control, secure communication protocols, notaries-based consensus mechanisms [55], and other security features, as detailed in Section 4.4.1 of the thesis. For example, the Corda platform addresses the risk of unauthorised users accessing system assets by implementing a decentralised access control mechanism, which prevents unauthorised access and allows only authorised nodes to join the network. Additionally, the platform tackles the risk of transmission channel interception and data misuse by employing authorised nodes in a P2P network, utilising PKI-based cryptographic schemes, and incorporating a notary-based consensus model to ensure transaction integrity. Furthermore, the Corda platform addresses other risks, such as malicious script submission by using a sandbox concept to prevent unauthorised code execution. It also mitigates Denial-of-Service (DoS) attacks through request rate-limiting firewalls and an Identity Manager Service that filters incorrectly formatted requests. Furthermore, Corda ensures data privacy protection against insider threats by employing a decentralised access control mechanism and a consensus model based on notaries for transaction validation, guaranteeing integrity and authenticity. Thus, the Corda platform mitigates the risks identified in the audit process by leveraging its security features and mechanisms inherent in blockchain technology.

RQ 3.3: What new risks are associated with implementing blockchain technology in the audit process? As discussed in Section 4.4.2, implementing blockchain technology in the audit process introduces new risks [53, 54, 55] that must be addressed. One such risk is endpoint vulnerability, which stems from a lack of security awareness and knowledge [112]. This vulnerability can lead to weak passwords, loss of private keys, and unauthorised access to sensitive information through phishing and social engineering attacks. Another risk involves privacy breaches, as validating notaries in the CorDapp

platform have access to the full content and history of transactions, potentially exposing sensitive information and enabling de-anonymisation. Additional risks include denial-of-state attacks, where a malicious attacker with access to the Corda network could create invalid transactions and cause disruptions. Error-prone smart contracts, which may contain logical bugs, inadequate error handling, or input validation issues, also pose a threat to valuable assets once deployed on the blockchain [52]. Lastly, the advancement of quantum computing research presents a risk to currently used cryptographic techniques in blockchain technology, as they may not be equipped to handle the post-quantum era, leaving systems vulnerable to potential attacks [112].

RQ 3.4: What are the means to mitigate the identified risks associated with implementing blockchain technology in the audit process? To mitigate the risks associated with implementing blockchain technology in the audit process, organisations must adopt a comprehensive approach as outlined in Section 4.4.2. One critical means of addressing endpoint vulnerabilities due to a lack of security awareness and knowledge involves establishing robust governance measures [112]. This includes implementing key management with hardware security modules, creating and managing information security policies and guidelines, and providing regular security training for staff to promote best practices and increase awareness of potential threats [112].

Moreover, mitigating the risks concerning privacy breaches, denial-of-state attacks, error-prone smart contracts, and the potential influence of quantum computing threats requires employing a combination of advanced techniques and cryptographic tools. For example, transaction tear-offs can protect transaction data by revealing only the minimum necessary information. At the same time, zero-knowledge proofs and secure execution environments can defend against denial-of-state attacks [57]. Additionally, thorough analysis and sanitisation of smart contract code before deployment can minimise errors [13], and thereby, smart contract attacks. Finally, adopting quantum-resistant cryptography methods, such as multivariate, hash-based, lattice-based, symmetric key quantum resistance, or code-based methods, can safeguard against the potential impact of advancements in quantum computing research [52, 57, 122].

5 Concluding Remarks

This thesis has analysed the security of the audit process and how blockchain technology can help to improve it. It compares the security risk-oriented pattern-based approach with the blockchain-based approach and demonstrates how these ensure system security.

5.1 Limitations

Based on the literature, it can be seen that the major audit networks, in particular the four largest ones (the Big Four), develop blockchain-based software for their customers in different industries. However, historically, due to the competitive situation, it has been their practice to use their audit software only within their network rather than to license it out. As a result, information on blockchain applications in the field of financial auditing is limited. The available literature to the author is included in the work.

It is possible that, in addition to the risk identification methods used, an additional method, e.g. STRIDE [92], could have identified additional risks in the audit processes considered.

It is possible that, in addition to the identified security risk mitigation tools, there may be other effective approaches to mitigate some of the identified risks. As many blockchain platforms and their developers exist, it is possible that some important platforms worthy of attention, in this case, were not included in the comparison.

Within the scope of this thesis, the model created has not been validated, which should be the next step in the future deployment of what is created here. This thesis has not investigated the effectiveness of either the traditional, centralised or blockchain-based approach in mitigating security risks. Nor has the economic feasibility of moving to blockchain technology been examined.

5.2 Answers to Research Questions

The first introductory chapter set out the main research question: **How to manage security risks in the auditing processes?** This was divided into three sub-questions:

RQ 1: What is the current state of securing auditing processes? The author provides an overview of audit processes and systems, emphasising the importance of security risk management in maintaining financial audit systems' integrity, confidentiality, and availability. Next, the Information Systems Security Risk Management (ISSRM) domain model [4, 34, 64, 66] is introduced as a comprehensive framework for security risk management and relevant international and regional standards are discussed. The chapter then highlights the potential of blockchain technology, particularly Corda, in securing audit processes due to its adaptability, versatility, and strong privacy and transaction-level security features [44, 70]. Finally, the author reviews several studies that showcase Corda's ability to address various security risk management concerns across multiple

sectors, providing evidence for its potential application in securing auditing processes while acknowledging the unique security risks and proposed countermeasures.

RQ 2: What are the security requirements for the audit process? The author addresses the research question by examining a case of the collection of audit information using the audit software E-dok. First, the author identifies valuable business assets and their supporting information systems assets, then outlines the risks associated with conventional technology, including unauthorised access, data interception, misuse, malicious scripts, Denial-of-Service attacks, and data privacy concerns. To mitigate these risks, the author suggests implementing centralised access control mechanisms, using the standard Transport Layer Security protocol, filtering incoming data, installing firewalls, and employing cryptographic algorithms. Security requirements are derived using security risk-oriented patterns [4, 64], with a complete list provided in the thesis's Appendix I. The results of this analysis can be extended to all audit processes where audit software is used.

RQ 3: How does blockchain help to avoid security risks in auditing processes? The author explores the research question by transferring a part of the audit process—receiving information from the client—to a blockchain platform. The author determines that a private-permissioned blockchain is suitable for this purpose and selects the R3 Corda platform as the most appropriate choice based on its security qualities and suitability for use in a regulated profession. The Corda platform mitigates the security risks identified in the audit process through features such as decentralised access control, secure communication protocols, and notaries-based consensus mechanisms. However, implementing blockchain technology introduces new risks, including endpoint vulnerability, privacy breaches, denial-of-state attacks, error-prone smart contracts, and potential impacts of quantum computing threats [53, 54, 55]. To mitigate these risks, the author suggests adopting a comprehensive approach that includes robust governance measures, advanced cryptographic techniques, and thorough analysis and sanitisation of smart contract code before deployment.

5.3 Conclusion

Information in audit processes must be protected against security risks. This thesis examined the security risk management of audit processes using both traditional, centralised tools and blockchain technology. The main research question, how to manage security risks in auditing, was addressed by analysing a scenario and exploring the R3 Corda platform as a potential blockchain solution.

The case analysis focused on collecting audit information using the audit software E-dok, identifying valuable business assets and security objectives, and applying security risk-oriented patterns to identify security risks and derive security requirements. The full list of identified security requirements is provided in Appendix I of the thesis. Possible solutions to mitigate the security risks identified in the traditional, centralised design

were also discussed.

The analysis revealed that it is technologically feasible to implement private permissioned blockchain technology, specifically the R3 Corda platform, in the audit information collecting process. The Corda platform is suitable, secure, and designed for use in a regulated environment. As a result of implementing the Corda platform, the security risks of a process using current technology can be mitigated. However, it should be noted that the introduction of blockchain technology would also introduce new security risks. While existing solutions are available for most of these new risks, some remain unresolved, such as the risk posed by quantum computing. Proposed solutions to this risk have been suggested, but ready-made applications to protect the blockchain have yet to be made available. The question remains whether quantum computing will be deployed first or whether solutions to mitigate this risk will be readily developed.

This thesis contributes to understanding security risk management in auditing processes by examining both traditional, centralised tools and the potential of blockchain technology, specifically the R3 Corda platform. The findings provide insights for stakeholders to make informed decisions when considering the implementation of blockchain technology in the context of financial auditing and security risk management, offering a secure and reliable alternative to traditional centralised systems.

5.4 Future Work

The next step in the case would be to validate the model transferred to blockchain technology. In order to do this, it should be implemented in the Corda test net to validate its real-life performance.

The difference in the impact of traditional, centralised and decentralised technologies on security risks should be further compared. A feasibility study should then be carried out to determine whether implementing blockchain technology in audit processes could also be cost-effective. It means whether the incremental benefit compared to the technology currently in use is greater than the cost of introducing decentralised technology.

In this thesis, the submission of information and documents to the auditor for one of the main groups of information providers – the client – has been transferred to blockchain technology. In the future, the model should be extended to other information providers, including an analysis of whether and what changes should be made to the model, considering how information providers present information and the need to change the current behaviour. The desirable outcome of this step could be to cover the acceptance of audit information from the whole potential range of information providers.

References

[1] *About the Business Process Model and Notation Specification Version 2.0.2*. OMG. 2014. URL: https://www.omg.org/spec/BPMN (visited on 02/19/2023).

[2] Sakshi Agarwal. *Cybersecurity essentials for Capital Markets firms in the Digital age - Wipro*. wipro.com. URL: https://www.wipro.com/capital-markets/cybersecurity-essentials-for-capital-markets-firms-in-the-digital-age/ (visited on 01/01/2023).

[3] *Agora. Digital 'smart bonds' that automate the bond lifecycle*. Corda. URL: https://corda.net/digital-assets/#agora-md (visited on 04/10/2023).

[4] Naved Ahmed. "Deriving Security Requirements from Business Process Models". Accepted: 2014-11-13T14:46:46Z ISBN: 9789949327164 ISSN: 1024-4212 Journal Abbreviation: Turvanõuete tuletamine äriprotsesside mudelitest. Thesis. Nov. 13, 2014. URL: https://dspace.ut.ee/handle/10062/44267 (visited on 11/06/2021).

[5] Naved Ahmed and Raimundas Matulevičius. "Securing business processes using security risk-oriented patterns". In: *Computer Standards & Interfaces* 36.4 (June 2014), pp. 723–733. ISSN: 09205489. DOI: 10.1016/j.csi.2013.12.007. URL: https://linkinghub.elsevier.com/retrieve/pii/S0920548913001815 (visited on 02/19/2023).

[6] Ian Allison. *Deloitte, Libra, Accenture: The work of auditors in the age of Bitcoin 2.0 technology*. International Business Times UK. Section: Technology. Aug. 18, 2015. URL: https://www.ibtimes.co.uk/deloitte-libra-accenture-work-auditors-age-bitcoin-2-0-technology-1515932 (visited on 04/16/2023).

[7] Riigi Infosüsteemi Amet. *Eesti infoturbestandard E-ITS*. Eesti infoturbestandard. 2022. URL: https://eits.ria.ee/ (visited on 03/24/2023).

[8] Hasib Anwar. *Corda Blockchain: Ruler of The Financial Enterprises*. 101 Blockchains. Apr. 13, 2019. URL: https://101blockchains.com/corda-blockchain/ (visited on 06/19/2022).

[9] Hasib Anwar. *Hyperledger Sawtooth: Blockchain For Business*. 101 Blockchains. Sept. 2, 2019. URL: https://101blockchains.com/hyperledger-sawtooth/ (visited on 06/15/2022).

[10] *API: Flows*. R3 Documentation. Aug. 12, 2021. URL: https://docs.r3.com/en/platform/corda/4.10/community/api-flows.html (visited on 02/10/2023).

[11] G. Apostolopoulos, V. Peris, and D. Saha. “Transport layer security: how much does it really cost?” In: *IEEE INFOCOM ’99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320)*. IEEE INFOCOM ’99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320). Vol. 2. ISSN: 0743-166X. Mar. 1999, 717–725 vol.2. DOI: 10.1109/INFCOM.1999.751458.

[12] *Attachments*. R3 Training - R3 Training and Tutorials - R3 Docs. URL: https://training.corda.net/corda-advanced-concepts/attachments/ (visited on 02/14/2023).

[13] Nicola Atzei, Massimo Bartoletti, and Tiziana Cimoli. “A Survey of Attacks on Ethereum Smart Contracts (SoK)”. In: *Principles of Security and Trust*. Ed. by Matteo Maffei and Mark Ryan. Vol. 10204. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017, pp. 164–186. ISBN: 978-3-662-54454-9 978-3-662-54455-6. DOI: 10.1007/978-3-662-54455-6_8. URL: https://link.springer.com/10.1007/978-3-662-54455-6_8 (visited on 12/16/2022).

[14] *aXedras. Digitalize precious metals end-to-end*. Corda. URL: https://corda.net/digital-assets/#axedras-md (visited on 04/10/2023).

[15] Xavier Bellekens et al. “Pervasive eHealth services a security and privacy risk awareness survey”. In: *2016 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (CyberSA)*. 2016 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (CyberSA). London, United Kingdom: IEEE, June 2016, pp. 1–4. ISBN: 978-1-5090-0703-5. DOI: 10.1109/CyberSA.2016.7503293. URL: http://ieeexplore.ieee.org/document/7503293/ (visited on 01/01/2023).

[16] 101 Blockchains. *AWS vs AZURE vs Oracle Blockchain Solution Offering: The BaaS Comparison*. 101 Blockchains. Apr. 12, 2021. URL: https://101blockchains.com/aws-vs-azure-vs-oracle-blockchain/ (visited on 01/19/2023).

[17] Enrique Bonsón and Michaela Bednárová. “Blockchain and its implications for accounting and auditing”. In: *Meditari Accountancy Research* 27.5 (Jan. 1, 2019). Publisher: Emerald Publishing Limited, pp. 725–740. ISSN: 2049-372X. DOI: 10.1108/MEDAR-11-2018-0406. URL: https://doi.org/10.1108/MEDAR-11-2018-0406 (visited on 07/21/2021).

[18] Richard Gendal Brown. *The Corda Platform: An Introduction White Paper*. R3. May 2018. URL: https://www.r3.com/white-papers/the-corda-platform-an-introduction-whitepaper/ (visited on 07/13/2022).

[19] Chad Brubaker et al. “Using Frankencerts for Automated Adversarial Testing of Certificate Validation in SSL/TLS Implementations”. In: *2014 IEEE Symposium on Security and Privacy*. 2014 IEEE Symposium on Security and Privacy (SP). San Jose, CA: IEEE, May 2014, pp. 114–129. ISBN: 978-1-4799-4686-0. DOI: 10.1109/SP.2014.15. URL: http://ieeexplore.ieee.org/document/6956560/ (visited on 05/01/2023).

[20] Vitalik Buterin. “Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform.” In: (2014), pp. 1–36.

[21] Vitalik Buterin. *On Public and Private Blockchains*. Ethereum Foundation Blog. 2015. URL: https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains (visited on 04/09/2023).

[22] Andrei Carare et al. “Case Study: The Automation of an over the Counter Financial Derivatives Transaction Using the CORDA Blockchain”. In: *Blockchain and Applications*. Ed. by Javier Prieto et al. Vol. 320. Series Title: Lecture Notes in Networks and Systems. Cham: Springer International Publishing, 2022, pp. 128–137. ISBN: 978-3-030-86161-2 978-3-030-86162-9. DOI: 10.1007/978-3-030-86162-9_13. URL: https://link.springer.com/10.1007/978-3-030-86162-9_13 (visited on 03/25/2023).

[23] Henrique Centieiro. *What the heck is the MultiChain blockchain?* Medium. Mar. 31, 2021. URL: https://henriquecentieiro.medium.com/what-the-heck-is-the-multichain-blockchain-6b2d677785f1 (visited on 06/18/2022).

[24] Rocky KC Chang. “Defending against flooding-based distributed denial-of-service attacks: A tutorial”. In: *IEEE communications magazine* 40.10 (2002). Publisher: IEEE, pp. 42–51.

[25] Justin Clarke. *SQL injection attacks and defense*. Waltham, MA: Elsevier, 2012. 547 pp. ISBN: 978-1-59749-963-7.

[26] *Consensus*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-consensus.html (visited on 02/10/2023).

[27] *Contour. Letters of Credit end-to-end processing*. Corda. URL: https://corda.net/global-trade/#contour-md (visited on 04/10/2023).

[28] *Corda | Leading DLT Platform for Regulated Industries*. Corda. URL: https://corda.net/ (visited on 01/17/2023).

[29] *CorDapp contracts*. R3 Documentation. July 15, 2021. URL: https://docs.r3.com/en/platform/corda/4.10/enterprise/cordapps/api-contracts.html (visited on 02/13/2023).

[30] Gil Dagan. *The Actual Networking behind the Ethereum Network: How It Works*. The Orbs Blog. Aug. 21, 2018. URL: https://medium.com/orbs-network/the-actual-networking-behind-the-ethereum-network-how-it-works-6e147ca36b45 (visited on 05/01/2023).

[31] *Defining transaction tear-offs*. R3 Documentation. Jan. 12, 2023. URL: https://docs.r3.com/en/tutorials/corda/4.10/community/supplementary-tutorials/tutorial-tear-offs.html (visited on 01/29/2023).

[32] *Digital Assets*. Corda. URL: https://corda.net/digital-assets/ (visited on 04/10/2023).

[33] *DLT. Digitizing cross-border trade and supply chains*. Corda. URL: https://corda.net/global-trade/#dltledgers-md (visited on 04/10/2023).

[34] Éric Dubois et al. "A Systematic Approach to Define the Domain of Information System Security Risk Management". In: *Intentional Perspectives on Information Systems Engineering*. Ed. by Selmin Nurcan et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 289–306. ISBN: 978-3-642-12543-0 978-3-642-12544-7. DOI: 10.1007/978-3-642-12544-7_16. URL: http://link.springer.com/10.1007/978-3-642-12544-7_16 (visited on 11/07/2021).

[35] Marlon Dumas et al. *Fundamentals of Business Process Management*. Berlin, Heidelberg: Springer, 2018. ISBN: 978-3-662-56508-7 978-3-662-56509-4. DOI: 10.1007/978-3-662-56509-4. URL: http://link.springer.com/10.1007/978-3-662-56509-4 (visited on 02/19/2023).

[36] Andreas Ellervee, Raimundas Matulevi˘cius, and Nicolas Mayer. "A Comprehensive Reference Model for Blockchain-based Distributed Ledger Technology". In: *Proceedings of the ER Forum 2017 and the ER 2017 Demo Track* 1979 (2017), pp. 320–333. ISSN: 1613-0073. URL: https://ceur-ws.org/ (visited on 04/09/2023).

[37] *Extreme but plausible threats to financial services | Accenture*. WordPressBlog. 2019. URL: https://www.accenture.com/_acnmedia/pdf-100/accenture_fs_threat-report_approved.pdf (visited on 01/27/2023).

[38] *FAQ | XRPL.org*. URL: https://xrpl.org/faq.html (visited on 01/16/2023).

[39] *Five emerging technologies for rapid digital transformation*. Digital Transformation: A CXO's Guide. In collab. with Dion Hinchcliffe. June 21, 2017. URL: https://www.zdnet.com/article/five-emerging-technologies-for-rapid-digital-transformation/ (visited on 06/16/2022).

[40] *Flows*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-flows.html (visited on 02/10/2023).

[41] Rishabh Garg, ed. *Blockchain for Real World Applications*. 1st ed. Wiley, 2023. ISBN: 978-1-119-90373-4 978-1-119-90376-5. DOI: 10.1002/9781119903765. URL: https://onlinelibrary.wiley.com/doi/book/10.1002/9781119903765 (visited on 03/25/2023).

[42] Priyank Hajela, Ambika Pawar, and Shraddha Phansalkar. "ITreatU An Effective Privacy and Security Solution for Healthcare Data Using the R3 Corda Platform of Blockchain Technology". In: *Data Protection and Privacy in Healthcare: Research and Innovations*. Ed. by Ahmed Elngar, Ambika Pawar, and Prathamesh Churi. 1st ed. Boca Raton, FL : CRC Press, 2021.: CRC Press, Jan. 28, 2021, pp. 165–178. ISBN: 978-1-00-304884-8. DOI: 10.1201/9781003048848. URL: https://www.taylorfrancis.com/books/9781000349313 (visited on 03/25/2023).

[43] Felix Hasse et al. *Blockchain – an opportunity for energy producers and consumers?* Düsseldorf: PwC, 2017, p. 46. URL: https://www.pwc.com/gx/en/industries/assets/pwc-blockchain-opportunity-for-energy-producers-and-consumers.pdf (visited on 04/17/2023).

[44] Mike Hearn and Richard Gendal Brown. "Corda Technical White Paper". In: *R3* (Aug. 20, 2019), p. 73. URL: https://www.r3.com/white-papers/corda-technical-whitepaper/ (visited on 07/13/2022).

[45] Penny Hoelscher. *2017 OWASP A10 update: Insufficient logging & monitoring*. Infosec Resources. 2018. URL: https://resources.infosecinstitute.com/topic/2017-owasp-a10-update-insufficient-logging-monitoring/ (visited on 05/01/2023).

[46] *Home | XRPL.org*. URL: https://xrpl.org/ (visited on 01/17/2023).

[47] *HQLAX. Atomic delivery versus delivery (DvD) of securities*. Corda. URL: https://corda.net/digital-assets/#hqlax-md (visited on 04/10/2023).

[48] *Hyperledger Sawtooth*. URL: https://sawtooth.hyperledger.org/ (visited on 01/17/2023).

[49] *Instimatch Global. Confidential cash management platform*. Corda. URL: https://corda.net/digital-assets/#intistimatch-global-md (visited on 04/10/2023).

[50] *Introduction*. URL: https://sawtooth.hyperledger.org/docs/1.2/ (visited on 06/15/2022).

[51] *Introduction to Consensus | XRPL.org*. URL: https://xrpl.org/intro-to-consensus.html (visited on 01/16/2023).

[52] Mubashar Iqbal. "Reference framework for managing security risks using blockchain". Accepted: 2022-08-29T08:18:03Z ISBN: 9789916270073 ISSN: 2613-5906 Journal Abbreviation: Viiteraamistik turvariskide haldamiseks plokiahela abil. Thesis. Aug. 29, 2022. URL: https://dspace.ut.ee/handle/10062/83826 (visited on 11/19/2022).

[53] Mubashar Iqbal and Raimundas Matulevičius. "Blockchain-Based Application Security Risks: A Systematic Literature Review". In: *Advanced Information Systems Engineering Workshops*. Ed. by Henderik A. Proper and Janis Stirna. Vol. 349. Series Title: Lecture Notes in Business Information Processing. Cham: Springer International Publishing, 2019, pp. 176–188. ISBN: 978-3-030-20947-6 978-3-030-20948-3. DOI: 10.1007/978-3-030-20948-3_16. URL: http://link.springer.com/10.1007/978-3-030-20948-3_16 (visited on 12/17/2022).

[54] Mubashar Iqbal and Raimundas Matulevičius. "Comparison of Blockchain-Based Solutions to Mitigate Data Tampering Security Risk". In: *Business Process Management: Blockchain and Central and Eastern Europe Forum*. Ed. by Claudio Di Ciccio et al. Vol. 361. Series Title: Lecture Notes in Business Information Processing. Cham: Springer International Publishing, 2019, pp. 13–28. ISBN: 978-3-030-30428-7 978-3-030-30429-4. DOI: 10.1007/978-3-030-30429-4_2. URL: http://link.springer.com/10.1007/978-3-030-30429-4_2 (visited on 12/17/2022).

[55] Mubashar Iqbal and Raimundas Matulevičius. "Managing Security Risks in Post-Trade Matching and Confirmation Using CorDapp". In: *Databases and Information Systems*. Ed. by Tarmo Robal et al. Vol. 1243. Series Title: Communications in Computer and Information Science. Cham: Springer International Publishing, 2020, pp. 325–339. ISBN: 978-3-030-57671-4 978-3-030-57672-1. DOI: 10.1007/978-3-030-57672-1_24. URL: https://link.springer.com/10.1007/978-3-030-57672-1_24 (visited on 12/12/2021).

[56] Gwyneth Iredale. *What is Blockchain on AWS?* 101 Blockchains. Feb. 16, 2021. URL: https://101blockchains.com/aws-blockchain/ (visited on 01/19/2023).

[57] Tommy Koens et al. *Solutions for the Corda Security and Privacy Trade-off: Having Your Cake and Eating It*. ING Bank, 2019, pp. 1–10.

[58] Julia Kokina, Ruben Mancha, and Dessislava Pachamanova. "Blockchain: Emergent Industry Adoption and Implications for Accounting". In: *Journal of Emerging Technologies in Accounting* 14.2 (Sept. 1, 2017), pp. 91–100. ISSN: 1558-7940, 1554-1908. DOI: 10.2308/jeta-51911. URL: https://publications.aaahq.org/jeta/article/14/2/91/9224/Blockchain-Emergent-Industry-Adoption-and (visited on 04/17/2023).

[59] Kenneth C. Laudon and Jane P. Laudon. *Management Information Systems: Managing the Digital Firm*. 13th. Pearson, 2014. 639 pp. ISBN: 13: 978-0-273-78997-0.

[60] *Ledger*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-ledger.html (visited on 02/10/2023).

[61] Antony Lewis. *A Gentle Introduction to Blockchain Technology*. Bits on Blocks. Sept. 9, 2015. URL: https://bitsonblocks.net/2015/09/09/gentle-introduction-blockchain-technology/ (visited on 04/09/2023).

[62] Georgios Loukas and Gülay Öke. "Protection against denial of service attacks: A survey". In: *The Computer Journal* 53.7 (2010). Publisher: OUP, pp. 1020–1037.

[63] *Marco Polo Network. Automated global trade and working capital finance network*. Corda. URL: https://corda.net/global-trade/#marco-polo-network-md (visited on 04/10/2023).

[64] Raimundas Matulevičius. *Fundamentals of Secure System Modelling*. Cham: Springer International Publishing, 2017. ISBN: 978-3-319-61716-9 978-3-319-61717-6. DOI: 10.1007/978-3-319-61717-6. URL: http://link.springer.com/10.1007/978-3-319-61717-6 (visited on 11/06/2021).

[65] Tim Maurer, Ariel Levite, and George Perkovich. "Toward a global norm against manipulating the integrity of financial data". In: *Economics E-Journal*. Economics Discussion Papers 2017 (July 23, 2017), pp. 1–41. URL: http://www.economics-ejournal.org/economics/discussionpapers/2017-38/ (visited on 12/17/2022).

[66] Nicolas Mayer. "Model-based management of information system security risk". PhD thesis. University of Namur, 2009.

[67] Juan Minango et al. "Proof of Concepts of Corda Blockchain Technology Applied on the Supply Chain Area". In: *Trends in Artificial Intelligence and Computer Engineering*. Ed. by Miguel Botto-Tobar et al. Vol. 619. Series Title: Lecture Notes in Networks and Systems. Cham: Springer Nature Switzerland, 2023, pp. 619–631. ISBN: 978-3-031-25941-8 978-3-031-25942-5. DOI: 10.1007/978-3-031-25942-5_48. URL: https://link.springer.com/10.1007/978-3-031-25942-5_48 (visited on 03/25/2023).

[68] Nicola Minichiello. *Deloitte Launches Rubix, A One Stop Blockchain Software Platform*. Brave New Coin. 2015. URL: https://bravenewcoin.com/insights/deloitte-launches-rubix-a-one-stop-blockchain-software-platform (visited on 04/16/2023).

[69] Eric Minuskin. *EY launches EY OpsChain Supply Chain Manager, incorporating zero-knowledge proof technology for business operations*. 2022. URL: https://www.ey.com/en_gl/news/2022/05/ey-launches-ey-opschain-supply-chain-manager-incorporating-zero-knowledge-proof-technology-for-business-operations (visited on 04/17/2023).

[70] Debajani Mohanty. *R3 Corda for Architects and Developers: With Case Studies in Finance, Insurance, Healthcare, Travel, Telecom, and Agriculture*. Berkeley, CA: Apress, 2019. ISBN: 978-1-4842-4531-6 978-1-4842-4529-3. DOI: 10.1007/978-1-4842-4529-3. URL: http://link.springer.com/10.1007/978-1-4842-4529-3 (visited on 03/25/2023).

[71] Nicky Morris. *SAP leads Pharma Supply Chain blockchain*. Ledger Insights - blockchain for enterprise. July 7, 2018. URL: https://www.ledgerinsights.com/sap-pharma-supply-chain/ (visited on 01/16/2023).

[72] Jana Moser. *The Application and Impact of the European General Data Protection Regulation on Blockchains*. R3, Feb. 1, 2017, pp. 1–14. URL: https://www.r3.com/reports/the-application-and-impact-of-the-european-general-data-protection-regulation-on-blockchains/ (visited on 12/17/2022).

[73] *MultiChain | Enterprise blockchain platform*. URL: https://www.multichain.com/ (visited on 01/16/2023).

[74] Satoshi Nakamoto. "Bitcoin: A Peer-to-Peer Electronic Cash System". In: (2008). (Visited on 04/09/2023).

[75] *Nasdaq to collaborate with R3 on institutional grade offerings for digital assets exchanges*. R3. Apr. 29, 2020. URL: https://r3.com/press-media/nasdaq-to-collaborate-with-r3-on-institutional-grade-offerings-for-digital-assets-exchanges/ (visited on 04/10/2023).

[76] *Nasdaq. Digital asset marketplace*. Corda. URL: https://corda.net/digital-assets/#nasdaq-md (visited on 04/10/2023).

[77] Ron Ben Natan. *Implementing database security and auditing*. Elsevier, 2005.

[78] *Networks, identity, and discovery*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-ecosystem.html (visited on 02/10/2023).

[79] *Nodes*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-node.html (visited on 02/10/2023).

[80] *Notaries*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-notaries.html (visited on 02/10/2023).

[81] Kelly Olson et al. "Sawtooth: An Introduction". In: (Jan. 2018), p. 7.

[82] Marc Pilkington. *Blockchain Technology: Principles and Applications*. Rochester, NY, Sept. 18, 2015. URL: https://papers.ssrn.com/abstract=2662660 (visited on 04/09/2023).

[83] R3. "Axedras Case Study. Connecting and Digitalizing the Precious Metals Industry". In: (2021). URL: https://r3.com/wp-content/uploads/2022/09/Corda_aXedras_CS_R3_Jan2021.pdf.

[84] R3. "Contour Case Study. The Network of Networks Digitalizing Trade Finance". In: (2020). URL: https://r3.com/wp-content/uploads/2022/09/Corda_Contour_Case_Study_Jan2021.pdf.

[85] R3. "Spunta Case Study. Fast and Transparent Interbank Reconciliation Powered by Distributed Ledger Technology". In: (2020). URL: https://r3.com/wp-content/uploads/2022/09/Corda_Spunta_Case_Study_R3_Nov2020.pdf.

[86] R3. *Swiss-based fintech Instimatch Global is streamlining unsecured money markets, FX and repo trades through their cash management platform*. R3. URL: https://r3.com/case-studies/instimatch/ (visited on 04/08/2023).

[87] *Rahvusvaheline auditeerimise standard (Eesti) 200 "Sõltumatu audiitori üldised eesmärgid ja auditi läbiviimine kooskõlas rahvusvaheliste auditeerimise standarditega (Eesti)"*. URL: https://www.audiitorkogu.ee/uploads/Kutsetegevuse%20standardid%20%28Kehtib%2015.%20detsembril%202021%20v%C3%B5i%20p%C3%A4rast%20seda%20algavate%20perioodide%20finantsaruannete%20auditite%20%20kohta/ISAE%20200%20%28EE%29_ISA%20315%20kaasnevate%20muudatustega.pdf (visited on 02/22/2023).

[88] *Rahvusvaheline auditeerimise standard (Eesti) 500 "Auditi tõendusmaterjal"*. URL: https://www.audiitorkogu.ee/uploads/Kutsetegevuse%20standardid%20%28Kehtib%2015.%20detsembril%202021%20v%C3%B5i%20p%C3%A4rast%20seda%20algavate%20perioodide%20finantsaruannete%20auditite%20%20kohta/ISA%20%28EE%29%20500_ISA%20315%20kaasnevate%20muudatustega.pdf.

[89] *Rahvusvaheline auditeerimise standard (Eesti) 505 "Välised kinnitused"*. URL: https://www.audiitorkogu.ee/uploads/Standardid%20kuni%202016/isa505ee.pdf (visited on 02/22/2023).

[90] *Rahvusvaheline auditeerimise standard (Eesti) 600 "Spetsiaalselt arvesse võetavad asjaolud – grupi finantsaruannete auditid (sh komponendi audiitorite töö)"*. URL: https://www.audiitorkogu.ee/uploads/Kutsetegevuse%20standardid%20%28Kehtib%2015.%20detsembril%202021%20v%C3%B5i%20p%C3%A4rast%20seda%20algavate%20perioodide%20finantsaruannete%20auditite%20%20kohta/ISA%20%28EE%29%20600_ISA%20315%20kaasnevate%20muudatustega.pdf (visited on 02/19/2023).

[91] David Rodeck. *What Is XRP (Ripple)?* Forbes Advisor. Section: cryptocurrency. May 4, 2021. URL: https://www.forbes.com/advisor/investing/cryptocurrency/what-is-ripple-xrp/ (visited on 06/15/2022).

[92] Fabian Ruffy, Wolfgang Hommel, and Felix von Eye. "A STRIDE-based Security Architecture for Software-Defined Networking". In: *ICN 2016 : The Fifteenth International Conference on Networks* (2016), pp. 95–101.

[93] Jesus Ruiz. *Public-Permissioned blockchains as Common-Pool Resources*. LinkedIn. 2020. URL: https://www.linkedin.com/pulse/public-permissioned-blockchains-common-pool-resources-jesus-ruiz/ (visited on 04/10/2023).

[94] Kurt Sandkuhl et al. "Refining Security Requirement Elicitation from Business Processes using Method Engineering". In: BIR 2015 Workshops and Doctoral Consortium co-located with 14th International Conference on Perspectives in Business Informatics Research (BIR 2015), Tartu, Estonia, August 26-28, 2015. Vol. 1420. CEUR-WS, 2015, pp. 98–109. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:hj:diva-28616 (visited on 02/11/2022).

[95] Jana Schmitz and Giulia Leoni. "Accounting and Auditing at the Time of Blockchain Technology: A Research Agenda". In: *Australian Accounting Review* 29.2 (2019). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/auar.12286, pp. 331–342. ISSN: 1835-2561. DOI: 10.1111/auar.12286. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/auar.12286 (visited on 07/22/2021).

[96] Markus Schumacher et al. *Security Patterns: Integrating Security and Systems Engineering | Wiley*. Wiley Software Patterns Series. Chichester: John Wiley & Sons Ltd, Feb. 2006. 600 pp. ISBN: 978-0-470-85884-4. URL: https://www.wiley.com/en-us/Security+Patterns%3A+Integrating+Security+and+Systems+Engineering-p-9780470858844 (visited on 02/28/2022).

[97] *Secure coding guidelines*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/secure-coding-guidelines.html (visited on 01/29/2023).

[98] Bundesamt für Sicherheit in der Informationstechnik (BSI). *IT-Grundschutz*. Federal Office for Information Security. URL: https://www.bsi.bund.de/EN/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/IT-Grundschutz/it-grundschutz.html?nn=908032 (visited on 03/26/2023).

[99] Bruce Silver. *BPMN Method and Style: A levels-based methodology for BPM process modeling and improvement using BPMN 2.0*. Aptos, Calif: Cody-Cassidy Press, June 1, 2009. 236 pp. ISBN: 978-0-9823681-0-7.

[100] *Smart contracts*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-contracts.html (visited on 02/10/2023).

[101] *Spunta. Interbank reconciliation network*. Corda. URL: https://corda.net/reconciliation/#spunta-md (visited on 04/10/2023).

[102] International Organization for Standardization. *ISO 31000:2018(en), Risk management — Guidelines*. ISO. 2018. URL: https://www.iso.org/obp/ui/#iso:std:iso:31000:ed-2:v1:en (visited on 03/20/2023).

[103] International Organization for Standardization. *ISO/IEC 27000:2018(E), Information technology — Security techniques — Information security management systems — Overview and vocabulary*. 2018. URL: https://standards.iso.org/ittf/PubliclyAvailableStandards/c073906_ISO_IEC_27000_2018_E.zip (visited on 01/10/2022).

[104] International Organization for Standardization. *ISO/IEC 27001:2022(en), Information security, cybersecurity and privacy protection — Information security management systems — Requirements*. ISO. 2022. URL: https://www.iso.org/obp/ui#iso:std:iso-iec:27001:ed-3:v1:en (visited on 03/20/2023).

[105] International Organization for Standardization. *ISO/IEC 27005:2022(en), Information security, cybersecurity and privacy protection — Guidance on managing information security risks*. ISO. 2022. URL: https://www.iso.org/obp/ui#iso:std:iso-iec:27005:ed-4:v1:en (visited on 03/20/2023).

[106] *States*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-states.html (visited on 02/10/2023).

[107] Melanie Swan. *Blockchain: blueprint for a new economy*. First edition. OCLC: ocn898924255. Beijing : Sebastopol, CA: O'Reilly, 2015. 130 pp. ISBN: 978-1-4919-2049-7.

[108] *The latest extreme but plausible threat scenarios in financial services | Accenture*. WordPressBlog. 2021. URL: https://www.accenture.com/us-en/blogs/security/extreme-but-plausible-threats-to-financial-services (visited on 01/27/2023).

[109] *Transaction tear-offs*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-tearoffs.html (visited on 01/29/2023).

[110] *Transactions*. R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-transactions.html (visited on 02/11/2023).

[111] Alexander Tsikhilov. *Plokiahel : põhimõtted ja alused*. Äripäev, 2020. 240 pp. ISBN: 978-9949-694-19-8. URL: https://www.digar.ee/arhiiv/et/raamatud/137182 (visited on 04/10/2023).

[112] John Velissarios, Justin Herzig, and Didem Unal. *Blockchain's Potential Starts With Security | Accenture*. Accenture, Feb. 28, 2019. URL: https://www.accenture.com/us-en/insights/blockchain/potential-starts-security (visited on 12/17/2022).

[113] *What is a CorDapp?* R3 Documentation. Apr. 7, 2020. URL: https://docs.r3.com/en/platform/corda/4.10/community/cordapp-overview.html (visited on 02/13/2023).

[114] *What Is Multichain: Detailed Overview | Zebpay*. Section: Crypto. May 10, 2022. URL: https://zebpay.com/blog/what-is-multichain (visited on 01/16/2023).

[115] *Why Corda | Permissioned DLT Platform for Regulated Markets*. Corda. URL: https://corda.net/why-corda/ (visited on 01/16/2023).

[116] *Working with attachments*. R3 Documentation. Jan. 12, 2023. URL: https://docs.r3.com/en/platform/corda/4.10/community/get-started/tutorials/supplementary-tutorials/tutorial-attachments.html (visited on 02/14/2023).

[117] Kean Wu, Manlu Liu, and Jennifer Xu. "How Will Blockchain Technology Impact Auditing and Accounting: Permissionless Vs. Permissioned Blockchain". In: *Current Issues in Auditing* 13 (Aug. 29, 2019). DOI: 10.2308/ciia-52540.

[118] Karl Wüst and Arthur Gervais. *Do you need a Blockchain?* 375. 2017. URL: http://eprint.iacr.org/2017/375 (visited on 11/19/2021).

[119] *XRP - Digital Asset for Global Economic Utility | Ripple*. URL: https://ripple.com/xrp/ (visited on 01/16/2023).

[120] *XRPL Blockchain Developer Resources | Ripple*. URL: https://ripple.com/developer-resources/ (visited on 01/16/2023).

[121] Xiwei Xu, Ingo Weber, and Mark Staples. *Architecture for Blockchain Applications*. Cham: Springer International Publishing, 2019. ISBN: 978-3-030-03034-6 978-3-030-03035-3. DOI: 10.1007/978-3-030-03035-3. URL: http://link.springer.com/10.1007/978-3-030-03035-3 (visited on 04/05/2023).

[122] Wei Yin et al. "An Anti-Quantum Transaction Authentication Approach in Blockchain". In: *IEEE Access* 6 (2018). Conference Name: IEEE Access, pp. 5393–5401. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2017.2788411. URL: https://ieeexplore.ieee.org/abstract/document/8242360.

Appendices

I. Security Requirements

SecurityRequirement#1.1: *Auditor* should be able to *save* the *infoRequest* to *EngagementFile*

SecurityRequirement#1.2: *Auditor* should be able to *save* the *auditInstructions* to *EngagementFile*

SecurityRequirement#1.3: *Auditor* should be able to *save* the *bankConfirmation* to *EngagementFile*

SecurityRequirement#1.4: *Auditor* should be able to *save* the *ccOfRequestTo3rdParty* to *EngagementFile*

SecurityRequirement#1.5: *Auditor* should be able to *save* the *3rdPartyInfo* to *EngagementFile*

SecurityRequirement#1.6: *Auditor* should be able to *save* the *subsidiaryAuditorInfo* to *EngagementFile*

SecurityRequirement#1.7: *Auditor* should be able to *query* the *clientInfo* from *InputFolder*

SecurityRequirement#1.8: *Client* should be able to *save* the *clientInfo* to *InputFolder*

SecurityRequirement#2.1: *E-dok* should have a unique identity in the form of key pairs (public and private keys) certified by a certification authority

SecurityRequirement#2.2: *Auditor* should encrypt and sign the *infoRequest* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#2.3: *Auditor* should encrypt and sign the *auditInstructions* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#2.4: *Auditor* should encrypt and sign the *bankConfirmation* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#2.5: *Auditor* should encrypt and sign the *ccOfRequestTo3rdParty* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#2.6: *Auditor* should encrypt and sign the *3rdPartyInfo* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#2.7: *Auditor* should encrypt and sign the *subsidiaryAuditorInfo* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#2.8: *Auditor* should encrypt and sign the *clientInfo* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#2.9: *Client* should encrypt and sign the *clientInfo* to be transmitted using keys before sending the data to *E-dok*

SecurityRequirement#3.1: *SaveInfoRequest* should filter the *infoRequest* (the input).

SecurityRequirement#3.2: *SaveAuditInstructions* should filter the *auditInstructions* (the input).

SecurityRequirement#3.3: *SaveBankConfirmation* should filter the *bankConfirmation* (the input).

SecurityRequirement#3.4: *SaveCcOfRequestTo3rdParty* should filter the *ccOfRequestTo3rdParty* (the input).

SecurityRequirement#3.5: *Save3rdPartyInfo* should filter the *3rdPartyInfo* (the input).

SecurityRequirement#3.6: *SaveSubsidiaryAuditorInfo* should filter the *subsidiaryAuditorInfo* (the input).

SecurityRequirement#3.7: *SaveClientInfo* should filter the *clientInfo* (the input).

SecurityRequirement#3.8: *QueryClientInfo* should filter the *clientInfo* (the input).

SecurityRequirement#3.9: *SaveInfoRequest* should sanitise the *infoRequest* (the input) to transform it to the required format.

SecurityRequirement#3.10: *SaveAuditInstructions* should sanitise the *auditInstructions* (the input) to transform it to the required format.

SecurityRequirement#3.11: *SaveBankConfirmation* should sanitise the *bankConfirmation* (the input) to transform it to the required format.

SecurityRequirement#3.12: *SaveCcOfRequestTo3rdParty* should sanitise the *ccOfRequestTo3rdParty* (the input) to transform it to the required format.

SecurityRequirement#3.13: *Save3rdPartyInfo* should sanitise the *3rdPartyInfo* (the input) to transform it to the required format.

SecurityRequirement#3.14: *SaveSubsidiaryAuditorInfo* should sanitise the *subsidiaryAuditorInfo* (the input) to transform it to the required format.

SecurityRequirement#3.15: *SaveClientInfo* should sanitise the *clientInfo* (the input) to transform it to the required format.

SecurityRequirement#3.16: *QueryClientInfo* should sanitise the *clientInfo* (the input) to transform it to the required format.

SecurityRequirement#3.17: *SaveInfoRequest* should canonicalise the *infoRequest* (the input) to verify against its canonical representation.

SecurityRequirement#3.18: *SaveAuditInstructions* should canonicalise the *auditInstructions* (the input) to verify against its canonical representation.

SecurityRequirement#3.19: *SaveBankConfirmation* should canonicalise the *bankConfirmation* (the input) to verify against its canonical representation.

SecurityRequirement#3.20: *SaveCcOfRequestTo3rdParty* should canonicalise the *ccOfRequestTo3rdParty* (the input) to verify against its canonical representation.

SecurityRequirement#3.21: *Save3rdPartyInfo* should canonicalise the *3rdPartyInfo* (the input) to verify against its canonical representation.

SecurityRequirement#3.22: *SaveSubsidiaryAuditorInfo* should canonicalise the *subsidiaryAuditorInfo* (the input) to verify against its canonical representation.

SecurityRequirement#3.23: *SaveClientInfo* should canonicalise the *clientInfo* (the input) to verify against its canonical representation.

SecurityRequirement#3.24: *QueryClientInfo* should canonicalise the *clientInfo* (the input) to verify against its canonical representation.

SecurityRequirement#4.1: *SaveInfoRequest* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.2: *SaveAuditInstructions* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.3: *SaveBankConfirmation* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.4: *SaveCcOfRequestTo3rdParty* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.5: *Save3rdPartyInfo* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.6: *SaveSubsidiaryAuditorInfo* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.7: *QueryClientInfo* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *auditor*.

SecurityRequirement#4.8: *SaveClientInfo* should establish a rule base (i.e., a collection of constraints used by different firewalls) to communicate with the *client*.

SecurityRequirement#4.9: *Packet Filter Firewall* should filter the *auditor's* address to determine if that is not a host used by the threat agent.

SecurityRequirement#4.10: *Packet Filter Firewall* should filter the *client's* address to determine if that is not a host used by the threat agent.

SecurityRequirement#4.11: *Proxy Based Firewall* should communicate to the proxy representing *SaveInfoRequest* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.12: *Proxy Based Firewall* should communicate to the proxy representing *SaveAuditInstructions* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.13: *Proxy Based Firewall* should communicate to the proxy representing *SaveBankConfirmation* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.14: *Proxy Based Firewall* should communicate to the proxy representing *SaveCcOfRequestTo3rdParty* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.15: *Proxy Based Firewall* should communicate to the proxy representing *Save3rdPartyInfo* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.16: *Proxy Based Firewall* should communicate to the proxy representing *SaveSubsidiaryAuditorInfo* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.17: *Proxy Based Firewall* should communicate to the proxy representing *QueryClientInfo* to determine the validity of the request received from the *auditor*.

SecurityRequirement#4.18: *Proxy Based Firewall* should communicate to the proxy representing *SaveClientInfo* to determine the validity of the request received from the *client*.

SecurityRequirement#4.19: *State Firewall* should maintain the *state table* to check the *auditor's* request for additional conditions on established communication.

SecurityRequirement#4.20: *State Firewall* should maintain the *state table* to check the *client's* request for additional conditions on established communication.

SecurityRequirement#5.1: *E-dok* should perform operations to hide data when stored in the *EngagementFile*.

SecurityRequirement#5.2: *E-dok* should perform operations to unhide data when retrieved from the *EngagementFile*.

SecurityRequirement#5.3: *E-dok* should perform operations to hide data when stored in the *InputFolder*.

SecurityRequirement#5.4: *E-dok* should perform operations to unhide data when retrieved from the *InputFolder*.

SecurityRequirement#5.5: *E-dok* should audit the operations after retrieving, storing or other data manipulation in the *EngagementFile*.

SecurityRequirement#5.6: *E-dok* should audit the operations after retrieving, storing or other data manipulation in the *InputFolder*.

II. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Madis Valk**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Security Risk Management in Auditing Processes,

   supervised by Raimundas Matulevičius.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Madis Valk
09/05/2023