UNIVERSITY OF TARTU
Institute of Computer Science
Innovation & Technology Management Curriculum

Kamran Mammadzada

Blockchain Oracles

Systematic Literature Review

Master's Thesis (20 ECTS)

Supervisor(s): Fredrik Milani

Tartu 2019

Blockchain Oracles

Abstract:

Blockchain technology has emerged as a potential disruptor of multiple industries and became an enabler for separate entities to trans-act in a secure and decentralized manner. Nevertheless, the blockchain technology in itself does not directly interact with the external data sources. External data, that is needed, is transferred by means of *oracles*. The research goal of this thesis is to explore the relationship between blockchain networks and oracles and develop a framework to help guide blockchain developers and decision makers in their blockchain projects. Few of the existing oracle projects have described similar efforts in their papers, but no systematic review has been made by authors. The framework, presented in the thesis, is developed based on Systematic Literature Review of existing blockchain projects involving oracles. It includes components such as type of information oracles collect, blockchain networks with which they interact as well as encryption of communication between the oracles and the data source. Additionally, oracle decision making, which captures how the information is passed to the oracle, along with the verification of that data and methods of integration of oracles with blockchain networks, play an important role in blockchain oracle projects. The results of the review demonstrate that blockchain oracles are complex solutions involving multiple components and aspects. They can be intangible or tangible and transport data from web or sensor devices respectively. Oracles can be used in all types of blockchain networks and integrated in different formats including custom smart contract interfaces or directly with blockchain nodes. They can be centralized or decentralized in terms of decision making and utilize various existing consensus mechanisms to decide on correctness of the data or simply trust the external data provider. These findings will help the blockchain developers demystify the potential usage or implementation of oracles in their blockchain projects and help bridge the gap between the virtual world of blockchain and the external environments.

Plokiahelatehnoloogia on osutunud paljude tööstusharude potentsiaalseks lammutajaks ning on saanud eraldiseisvate üksuste jaoks turvalise ja detsentraliseeritud toimimise võimaldajaks. Sellest hoolimata ei ole plokiahela-tehnoloogia iseenesest väliste andmeallikatega otseselt seotud. Vajalikke väliseid andmeid vahendatakse oraaklite abil. Selle magistritöö eesmärk on uurida seoseid plokiahela-võrkude ja oraaklite vahel ning töötada välja raamistik, mis aitab plokiahela-arendajaid ja otsuste langetajaid nende plokiahela-projektides millestki juhinduda. Mõnedes olemasolevates oraakliprojektides on kirjeldatud sarnaseid püüdluseid, kuid seni pole nende autorid süstemaatiliste ülevaadeteni jõudnud. Lõputöös esitatud raamistik on välja töötatud olemasolevate oraaklitega seotud plokiahela projektide süstemaatilise kirjanduse ülevaate põhjal. See hõlmab selliseid komponente nagu oraaklite poolt kogutud informatsiooni tüübid, plokiahelavõrgud, millega nad suhtlevad, ning ka oraaklite ja andmeallika vahelise suhtluse krüptimine. Lisaks mängib plokiahela-oraakli projektides olulist rolli ka oraaklite otsuste tegemine, mis kajastab teabe edastamist oraaklile, nende andmete kontrollimist ja meetodeid, kuidas oraakleid integreeritakse plokiahela-võrkudega. Läbivaatamise tulemused näitavad, et plokiahela-oraaklid on keerulised lahendused, mis hõlmavad paljusid komponente ja aspekte. Need võivad olla immateriaalsed või materiaalsed ning edastada andmeid vastavalt veebist või anduriseadmetest. Oraakleid saab kasutada igat tüüpi plokiahela-võrkudes ja integreerida erinevates formaatides, sealhulgas nutikates lepinguliidestes, või otse teiste plokiahela-sõlmedega. Neid saab otsustusprotsessides tsentraliseerida või detsentraliseerida ja nad suudavad kasutada andmete õigsuse üle otsustamiseks mitmesuguseid olemasolevaid nõuandemehhanisme või usaldada lihtsalt välist andmepakkujat. Need uurimise tulemused aitavad plokiahela arendajatel demüstifitseerida oraaklite potentsiaalset kasutamist või rakendamist oma plokiahela-projektides ning aitavad ületada lõhet plokiahela virtuaalse maailma ja väliste keskkondade vahel.

Keywords:

Blockchain, blockchain oracle

CERCS:

Table of Contents

1 Introduction

Blockchain is an emerging technology which enables individuals and groups to innovate by developing novel solutions. It is revolutionizing multiple industries and allowing entities to transact in a more secure and decentralized manner. Blockchain is a distributed, immutable and decentralized ledger where transactions are secured and verified using cryptography. To perform these transactions, blockchain contains a network of participants who agree on transactions using a consensus mechanism. Majority of blockchain technologies today operate in the domain of the virtual environment with very little exposure to the data from the outside. Blockchain solutions might require data from external sources to function properly. The data being received should be verified before being broadcasted to the blockchain as once added, it is immutable. The data interaction between blockchain solutions and its external world is facilitated by means of oracles. Conceptually oracles are entities that enable collection, verification and transmission of data from external source to the blockchain. They are important components in blockchain solutions requiring external input and there are different types of oracles. Blockchain solution developers and decision-makers will need to consider multiple variables before designing and/or embedding oracles into their blockchain solutions. Given that the blockchain oracles is a new domain, the above-mentioned actors would benefit from guidance as to which oracle technologies and their properties are available to them.

Although some have reviewed existing blockchain solutions in the industry, none of them provided such a guideline or framework. For instance, researchers behind Astraea [1], a decentralized blockchain oracle, provided a short discussion of few other oracle solutions, available on the market, but their overview only considered solutions related to theirs. Likewise, authors of PDFS [2], a practical system for data feeds from websites to the blockchain, shortly outlined some of the existing technologies used in oracles. Papers utilizing blockchain oracle with the goal of building a prediction platform have either provided an overview of some other similar projects [3] or simply used them as reference [4], [5].

Nevertheless, the papers mentioned above focus on their specific solutions while providing almost no high-level overview of the mechanisms and components necessary to build oracles or integrate them in existing blockchain based projects. The oracles reviewed in these papers were chosen arbitrarily with no systematic approach. As such there are no existing frameworks that cover the main components and mechanisms involved in either developing or integrating an existing oracle into the blockchain solutions.

Therefore, a systematic approach to reviewing existing literature on blockchain oracles with the further goal of developing an overarching framework to support decision makers and developers in delivering blockchain oracle projects is necessary. The purpose of this study is to investigate the relationship between the oracles and blockchain. Increasing adoption of blockchain solutions both in the private and public sector makes the research into the role and function of oracles in blockchain solutions increasingly important. Exploring this relationship starts with investigating the information types that oracles handle, since their primary mission is to bring external source data into the blockchain. The ways in which the oracles receive the information, processes and manages it forms a basis for oracle properties and understanding these properties would enable developers to correctly identify the blockchain oracles that would add the most value. Blockchain is immutable storage and for oracles to serve their purpose they need to ensure truthful and correct data is inserted into the blockchain. This data verification aspect forms one of the directions explored as part of the blockchain oracle interplay. At last, the information received, collected and verified needs to be injected into the blockchain but since oracles are entities foreign to blockchain, the exploration of integration methods is pivotal for completing the cycle. Lastly, it helps to put various components into context and look at real world applications and various approaches taken by blockchain developer teams at using and implementing oracles in the real world. To sum up, all of the aspects of blockchain oracles discussed above is critical to understanding the nature of this relationship which would provide greater visibility for solution developers when it comes to designing the architectures and processes of projects involving or requiring blockchain oracles.

The research aims to answer six research questions. First research question is "What are the types of information oracles provide to the blockchain?" where the thesis explores data types passed to the blockchain from external sources. Identifying properties that oracles possess is critical for identifying their impact on blockchain solutions. The thesis address this by answering "What are the necessary properties of oracles for use in blockchain solutions?" research question. Data on the blockchain is immutable, therefore the research question "How do oracles verify the data they provide to blockchain solutions?" covers various oracle data verification mechanisms existing today. "How can oracles be integrated into blockchain platforms?" aims to uncover different integration approaches used for delivering blockchain oracle projects. The last two research questions "What are the use cases for blockchain oracles?" and "In which industry cases are oracles proposed to be used?" set out to investigate industries where blockchain oracles have been implemented along with a specific use case or a problem that was targeted by an implementation.

The contribution of this thesis is presentation of a blockchain oracle framework which aims to provide guidance for blockchain project developers seeking to either develop a blockchain oracle or use an existing oracle solution to add value to their implementation. In the framework the thesis seeks to outline major components that are critical for successful blockchain oracle implementation. The use of this framework won't be limited to project managers and decision makers but will also provide high level information for software engineers developing blockchain solutions as well. The contributions of this paper are achieved by the means of systematic literature review (SLR). The author uses SLR to review literature across databases containing academic and non-academic studies.

The rest of this thesis is structured as follows. Section 2 provides background information describing major concepts discussed in the research; Section 3 describes the review protocol used to find the primary studies; Section 4 presents the results of the research; Section 5 outlines the novel blockchain oracle framework; Section 6 discusses related work, outlining existing papers or lack thereof on the topic of blockchain oracles; finally Section 7 concludes the research by summarizing the thesis effort.

2 Background

Blockchain is the technology behind popular cryptocurrencies such as Bitcoin and Ethereum. Bitcoin was first introduced by Satoshi Nakamoto in his paper in 2008 and its goal was to provide a technology for peer-to-peer electronic cash, enabling people to transact without the necessity of a third-party such as financial institution. [6] Blockchain was merely a concept within this paper. Later this concept was generalized and today Blockchain is a distributed, immutable, irreversible, decentralized and secure ledger that consists of network of nodes or participants who agree on transactions using consensus mechanisms and are fully aware of all of the transactions taking place. [7] It contains sequence of individual blocks where every next block contains the reference to the previous one. This sequence of blocks holds a complete list of transaction records just like an accounting ledger. The blockchain network does not rely on a central authority and participants interact anonymously. It enables for a more secure and persistent transactions, where information that is recorded on the blockchain is immutable and can't be tampered with. The participants of the network must sign transactions using asymmetric cryptography before broadcasting it to the network.[8] Each participant of the network holds the complete copy of the blockchain, which is important for network's security and reliability. Later the network of participants validates these transactions using one of existing consensus mechanisms.

Essentially, consensus mechanisms allow for secure update of a distributed shared state, where new information on the blockchain is recorded based on the agreement of network participants. [9] To achieve consensus many different consensus protocols have been proposed and implemented for blockchain. The two most widely implemented mechanisms are Proof of Work (PoW) and Proof of Stake (PoS). In PoW, each node in the network has to perform certain amount of work in order to add new blocks to the blockchain. [9] The mechanism was first implemented as part of Bitcoin's blockchain network. This work involves solving mathematical puzzles of various degrees of complexity. Nodes that perform these calculations are called miners. Miners usually possess huge computational power to solve these puzzles, in return for which they are rewarded. [10] Due to this PoW is known to have high electricity consumption. Proof of Stake algorithms were designed to mitigate some of the PoW's disadvantages such as high power usage. [9] PoS model on the other hand is developed based on the notion that the more stake a user has in the network, the less likely they will want to participate in a fraudulent activity. [11] In this consensus model any user wishing to participate in the blockchain network must allocate a certain investment, which often is an amount of cryptocurrency, into the system. This stake can't be spent and the likelihood of a user publishing new block is directly tied to the ratio of that investment to the overall blockchain network's stake. [11] PoW and PoS are considered "high-profile" consensus models, where "high-profile" is calculated based on the market capitalization of the cryptocurrency utilizing a specific model. [12] Apart from the models mentioned above, there many other consensus mechanisms such as Ripple Protocol Consensus Algorithm (RPCA) developed specifically for Ripple cryptocurrency in order to address latency challenges present in other mechanisms, Cardano's Ouroboros – built based on PoS with additional security measures implemented, delegated Proof of Stake (dPOS) and Proof of Importance (PoI). For the purposes of this paper, these consensus models won't be presented further.

Consensus mechanisms play a central role in maintaining a blockchain network's security, liveness and fault tolerance. Another crucial architectural aspect of blockchain is its permission model. [11] Blockchain networks can be divided into two major categories – permissionless or public and permissioned or private. In public blockchains anyone can publish a new block, read and write to the blockchain network and participate in the consensus.

Consensus mechanisms used within this type of networks usually incentivize non-malicious behavior through rewarding users with assets (e.g. cryptocurrency). Two most popular public blockchain networks are Bitcoin and Ethereum. The popularity here is measured by market capitalization. [13] Permissioned blockchain networks on the other hand enforce users to authorize to publish blocks onto the blockchain. Due to its nature, users need meet certain requirements to receive access to read data or make transactions on the network. [11] [14] Tis type of blockchain can enable companies, groups of companies or consortiums to streamline processes across an enterprise. One implementation of private blockchain is Hyperledger Fabric, which is extensively utilized by various companies to meet their technology demands. Hyperledger Fabric is an open source implementation, initiated by different companies in 2015 and is currently under the guardianship of the Linux Foundation.[14]

Various permissioned and permissionless blockchains mostly act as a distributed, decentralized secure shared storage. Smart contracts enhance the functionality of blockchains and allow for further transaction automation. The concept of smart contract was proposed by Nick Szabo in 1994. [15] Despite this the idea did not become popular till the emergence of blockchain as a distributed ledger technology. Smart contract is a computer program that is self-verifying, self-executing and tamper-resistant and it runs on the blockchain platform. Smart contracts are defined as the computer programs that digitally facilitate, verify and enforce contracts made between two or more participants on the blockchain. [16] They are usually deployed on the blockchain platform and essentially live on the network. They enforce participants to abide by the rules written in the contract without any third party controlling it with the aim of reaching agreements and solving common problems with minimal trust. [17] Smart contracts are event driven, meaning that they are intelligent agents that can be activated if a predefined condition is met. [16], [18] Bitcoin was the first blockchain implementation that provided support for smart contracts, though designing complex logic for those is not possible due to Bitcoin's scripting language limitations. Ethereum is considered to be the first blockchain platform to support development of advanced and customizable smart contracts [16] Despite its advantages in augmenting blockchain's core functionalities, smart contracts have some fundamental limitations. One major challenge is that a smart contract can only use resources available on the network and can't access or interact with the data that lives . [2] This is where blockchain oracles come into play.

Oracles are trusted entities that bring external information into the blockchain. [1][19] Blockchain as a distributed shared storage doesn't have direct access to data that lives outside of its network. In order to guarantee the security of the blockchain network smart contracts are generally executed in closed environments, preventing them from accessing external data important for their execution. [16] Thus, smart contracts require services of an oracle. The role of oracles is not limited to simply querying the information from outside of the network, but also includes verifying the authenticity and validity of that data. Some authors define this as an "Oracle Problem" [11] due to the challenges of bringing reliable data from real world into blockchain. This aspect of oracles is crucial for the sustainability and fault tolerance of the blockchain, because oracles directly interact with smart contracts on the blockchain. Smart contract execution is triggered when a certain condition or data is provided by an oracle. Some contracts may initiate a financial transaction because of the trigger or settle disputes between parties. Therefore, it is important that oracles provide reliable and valid information to ensure consistency and validity of smart contract execution, making oracles an essential part of a successful blockchain implementation.

The concept of a blockchain oracle was developed mostly by the industry and blockchain solution developers and did not exist in academia previously. It is important to note that the concept of oracles in blockchain should not be confused with random oracles in cryptography or oracle machines in complex theory. [20] Majority of resources today that discuss blockchain oracles or provide implementation examples are mostly created within blogs, websites and written as whitepapers. In academic papers the notion of blockchain oracles has been inconsistent and while some authors use the term verifier and reverse-verifier [21] others use concepts like trusted data feeds [2] [22] and validation oracles [17]. Some blockchain implementations describe components that act like blockchain oracles without specifically calling them as such, while others have built solutions that make blockchain oracles a central part of their blockchain implementation. One of the first solutions to the oracle problem was Town Crier, which is an authenticated data feed for smart contracts [22] It aims to ensure that information injected into the network comes from a reliable source and hasn’t been tampered with. To achieve this it uses „trusted software“ enclave on Intel processors. [19], [22] Few early implementations of blockchain oracles are centered around prediction markets. Project Augur, a prediction market where individuals can wager on the outcome of future events, utilizes a decentralized oracle consisting of profit-motivated reporters whose task is to simply report on the real-world outcome of an event. [23] Decentralized oracles have been introduced and utilized by projects like Augur, ChainLink [5], Astraea [1] and aim to overcome some of the shortcomings of a single oracle. Primary challenge is the centralization since single oracles must be run by a third party as a service, making them vulnerable to tampering. Decentralized oracles aim to address this challenge by using a similar to blockchain consensus mechanism across multiple oracles. These projects are examples of attempts to develop solutions to the oracle problem by introducing new use cases, some of which are already profit generating businesses. [23]

3 Review protocol

The review protocol, used to conduct the SLR study, specifies the research questions (Section 3.1), the search strategy (Section 3.2) and the inclusion and exclusion criteria (Section 3.3). In order to develop a quality study, the guidelines proposed by Kitchenham [24] were followed. SLR can be described in three main phases [24] – planning, conducting and reporting. The 1st phase includes reasoning behind the review, the definition of research questions, development and evaluation of the review protocol. The 2nd phase aims to identify business cases and studies, selection of relevant ones, quality assessment, data extraction and data synthesis. Finally, the 3rd phase considers the dissemination, formatting, and evaluation of the report.

3.1 Research Questions

The goal of this SLR is to identify published studies and papers that describe the role and function of oracles within blockchain solutions. To identify primary studies where this relationship occurs, the research objective of investigating the relationship between the oracles and blockchain is decomposed into a set of research questions.

RQ1: What types of information do oracles provide to blockchain?

In order to describe the nature of information oracles provide to blockchains and to set a basis of the relationship between oracles and blockchain, it is valuable to explore various information or data types that oracles communicate to blockchain.

RQ2: What are the necessary properties of oracles for use in blockchain solutions?

It’s important to map out the properties, oracles need to possess to be used in blockchain implementations, since this is what defines a blockchain oracle. Exploring these properties ensures understanding the characteristics the entities need to possess to inject information into blockchain, and for them to be named oracles. These properties include the oracle type, blockchain type, encryption and authentication methods used in blockchain oracles.

RQ3: How do oracles verify the data they provide to blockchain solutions?

Data verification is the bread and butter of blockchain oracles, since information recorded in blockchain can’t be deleted, which makes this topic vastly critical for the use of oracles in blockchain networks. Mapping the landscape of possible data verification mechanisms employed in blockchain oracles is crucial for their real-world implementation. Data verification here is defined as ensuring that information that is collected from external to blockchain sources matches data that is passed to the blockchain and is truthful and correct.

RQ4: How can oracles be integrated with blockchain platforms?

Oracle integration into blockchain platforms contributes to blockchain wide-spread implementation since oracles help solve the issue of bringing external data into the network. Investigating various ways oracles could be integrated provides a good basis for planning possible implementations allowing implementors to make necessary decisions when developing blockchain solutions.

RQ5: How are oracles used in blockchain solutions?

RQ6: In which industries are oracles proposed to be used?

Blockchain projects utilize oracles for different purposes and in various industries. The last pair of research questions set out to describe a current landscape of possibilities for use of oracles in blockchain implementations. They aim to uncover the various purposes and use cases for which blockchain oracles are implemented, along with associated industries. By use case the thesis describes scenarios where oracles are integral to the success of the blockchain project. As such RQ5 aims to explore blockchain oracle usage scenarios (use cases), while RQ6 investigates the industries where these use cases take place.

3.2 Search Strategy

The overall search strategy is to find a body of relevant studies. For this SLR two search strategies were used, as recommended by Okoli et al [25], Fink et al [26] and Levy et. al. [27], to secure identification of relevant studies. Accordingly, in the first step, called primary search, search strings were used to identify an initial set of papers [26]. Several electronic databases were used for this step. In the second step, a secondary search was performed by means of backward and forward tracing [25] [27]. I firstly provide a set of search strings in Section 3.2.1 and then I present search sources in Section 3.2.2.

3.2.1 Search strings

The development of the search strings, the thesis followed the guidelines suggested by Kitchenham et.al., [24]. The guidelines describe the importance of transparency and replicability of the SLR and suggests documenting the search in enough detail for readers to be able to assess the completeness of the search.

(1) The terms “blockchain” and “oracle” are used as they represent the core concepts of this thesis. We chose the term blockchain as our study is restricted to blockchain solutions. We combined the term blockchain with oracles as the study is focused on oracles.

(2) Many implementations of oracle(s) and blockchain integration today don’t refer to data input units as oracles and use the below terms without any mention of oracle. The following conclusion was done based on preliminary keyword research on the sources indicated in the SLR. Thus, to avoid missing important studies and papers the above terms will also be used:

   a. IoT
   b. Internet of Things

Throughout the SLR, the definition of oracle will include the term “internet of things” and “IoT”.

Specific search term “authenticated data feeds” and “data feeds” was also examined as to understand its use in the scientific community when referring to oracles. ‘Blockchain authenticated data feeds’ search strings were also used. The results were less than 5 hits in all the databases and the results were not relevant. Therefore, I decided to exclude them.

Based on the search terms, the following search string was formulated.

ST1: ((“blockchain” AND (“oracle” OR “internet of things” OR “IoT”))

ST2: ((“blockchain” AND “oracle”)) OR

ST3: ((“blockchain” AND (“internet of things” OR “IoT”)))

3.2.2 Search sources

In this SLR, databases summarized below in Table 1 were chosen.

Table 1
Electronic Databases used

Database	Organization	Abbr.
ACM Digital Library	ACM	ACM
IEEE Xplore	IEEE	IEEE
Scopus	Elsevier	SCP
Web of Science	Thomson Reuters	ISI
Wiley	Wiley	WL
Google Scholar	Google	GSC

The above databases were chosen to identify scientific papers in the field of computer science while google scholar was included to identify publications by companies and other non-academic organizations [24].

3.3 Selection Criteria

The importance of the selection criteria is to identify relevant studies that provide enough information to address the research questions. The criteria consisted of exclusion and inclusion criteria.

3.3.1 Inclusion Criteria

The inclusion criteria allowed us to filter out the papers relevant to the SLR. The following inclusion criteria was utilized

IC1: Is the study within the domain of blockchain?

IC2: Does the study cover integration between oracles and blockchain networks?

IC3: Does the paper describe the connection/solution of the oracle in the overall blockchain-based solution/application?

Given the nature of the research, it’s important that the paper covers the topic of blockchain and oracles (IC1). For research purposes, the study needs to describe the role and the function of oracles in blockchain implementations as well as describe that relationship, covering integration methods (IC2, IC3). Studies that mention blockchain and oracles, but do not discuss their interplay or integration mechanisms as part of the study are not included in the research, as they do not provide sufficient information to address the research questions. Some papers have focused on smart contract or blockchain but briefly cover oracles. These papers are not included in the research unless enough detail covering integration of oracles in blockchain has been presented.

3.3.2 Exclusion Criteria

The exclusion criteria help us to filter papers using more administrative approach, ensuring available and appropriate papers will be chosen. The list of exclusion criteria is the following:

EC1: Is the full-text version accessible? (I)

EC2: Is the study written in English? (I)

EC3: Is the study a duplicate? (E)

EC4: Is the study is less than 5 pages? (E)

EC5: Was the study published later than 2018? (E)

The first two exclusion criteria are defined to ensure access and understandability. If the study is not accessible or in English, it will be impossible to understand them. Papers accessible via digital libraries subscribed to by the University or available on the Internet, are considered as accessible. Papers that require payment of any kind, are considered as inaccessible. Finally, duplicates were excluded. Duplicate papers are those where papers with the same title from the same authors appear in different sources (exact duplicate). Duplicates are also studying from the same authors with approximately the same topic (version duplicate). In case of exact duplicate, only one is included and in the case of version duplicates, the most recent version is included. Since blockchain became a popular topic after 2008, following the publication of the bitcoin by Satoshi Nakamoto [6], the search only considered papers post that period.

3.4 Conducting the review

In this section, I present the steps and intermediate results that lead to selecting the final set of primary studies. The section also covers the data extraction strategy and information that was extracted from the papers. Table 2 contains the summary of number of papers processed in each step.

In the first step, I have collected the list of query results from each source. All the sources indicated in Table 1 allowed exporting the results, except Google Scholar and Wiley for which browser extension was used to scrap data from the search results[1]. At this stage, a total of 3036 papers were found from all sources. Using EC5 11 pre-2008 papers were removed, resulting in 3025 papers. Due to difference in export format across various sources, I used only three data headers:

- *source* – three letter code indicating the database. The codes were taken from Table 1.
- *title* – paper title
- *authors* – list of paper's authors

Table 2
Number of results by steps

Step	Step Name	Number of papers
	Search results	3036
Step 1	Initial list filtered by time	3025
Step 2	Filtered by duplicates	2356
Step 3	Filtered by title	571
Step 4	Filtered by abstract	70
Step 5	Full examination	21

[1] Data Scraper - Easy Web Scraping Chrome extension bit.ly/2IEVRiP

Step 6	Backward tracing	23
Final		23

In the next step, using exclusion criteria EC2 and EC3 668 duplicates and 12 non-English papers were removed. As a result of this step total of 2356 papers remained. In the next step, I have filtered the paper titles using the inclusion criteria identified in the Section 3.3.1. At this phase, there were 1785 irrelevant paper titles, resulting in 571 final papers. In the fourth step, in addition to the data points mentioned above abstracts and page numbers were also collected. Here I filtered only the papers that were accessible and had more than or equal to 5 pages (EC1 and EC4). Result of the fourth step was 70 papers chosen based on their abstract and potential content relevance. In the last step to ensure that these papers do contain the information that is needed for the SLR and using inclusion criteria used in Section 3.3.1, I further examined each of the 70 papers to assess their relevance. This resulted in total of 21 papers that were appropriate for the final inclusion. The final list of papers included whitepapers and academic papers. I then reviewed these papers and total of 2 additional papers were found as a result of backward tracing. Resulting 2 papers were considered as part of Step 5. In total 23 papers were used for analysis in this thesis.

3.5 Data Extraction Strategy

Following the identification of the final list of papers, relevant data was extracted. To ensure unbiased data extraction strategy, it has been recommended [26] [28] [29]to develop a data extraction form and strategy.

The data extraction form was developed after the screening process, allowing for utilize the insights drawn during the screening phase. Three types of data were extracted. The first relates to data about the paper. The second was data related to the context of the study and finally, the third type related to the actual process improvement. Table 2 summarizes the information extracted.

Table 3
Data Extraction Form

Data Extraction Form	
Data about the Paper	
Data	Description
Identifier	Unique id of the paper
Title	Title of the paper
Authors	Authors of the paper
Publication Year	Year of publication of the paper
Citations	Number of citations
Identifier	Unique id of the paper
Data about Context of the Paper	
Industry	Industry coverage of the paper
Study Objective	The objective of the study
Main findings	Main findings of the paper

Main limitations	Main limitations of the paper
Study context	The location context of the study (e.g. country, company)
Collaborators	Parties involved in the study or use case
Oracle Property Data	
Oracle Type	Type of oracle discussed / implemented
Data Verification Mechanism	Data verification mechanism employed in the context of blockchain integration
Encryption Method	Encryption method utilized
Authentication mechanism	Type of authentication utilized
Data verification	Type of data verification used
Blockchain type	Type of blockchain used in the relationship to an oracle
Information type	Type of information handled by an oracle
Use Case Data	
Information Availability	Information on implementation is publicly available
Presence of demo	Indicates whether any type of demo is present
Source code availability	Presence of source code for reproducibility
Performance evaluation presence	Results of performance evaluation present

The data was extracted in an iterative manner. One author extracted the data and populated the form.

3.6 Overview of studies

This section provides an overview of the studies. Blockchain oracles are relatively new topic in the academia. The Figure 1 displays the distribution of studies across years by sources through the first step performed as part of the SLR. The number of papers for recent years is significantly higher and represents the majority of papers. The figure only shows papers from 2009. It’s important to note that in the first three steps, there were 299, 286 and 34 papers respectively with no date indicated. This figure indicates that the research around blockchain oracles and IoT has increased exponentially starting from 2014-2015, proving the fact that blockchain oracles are a relatively new concept.

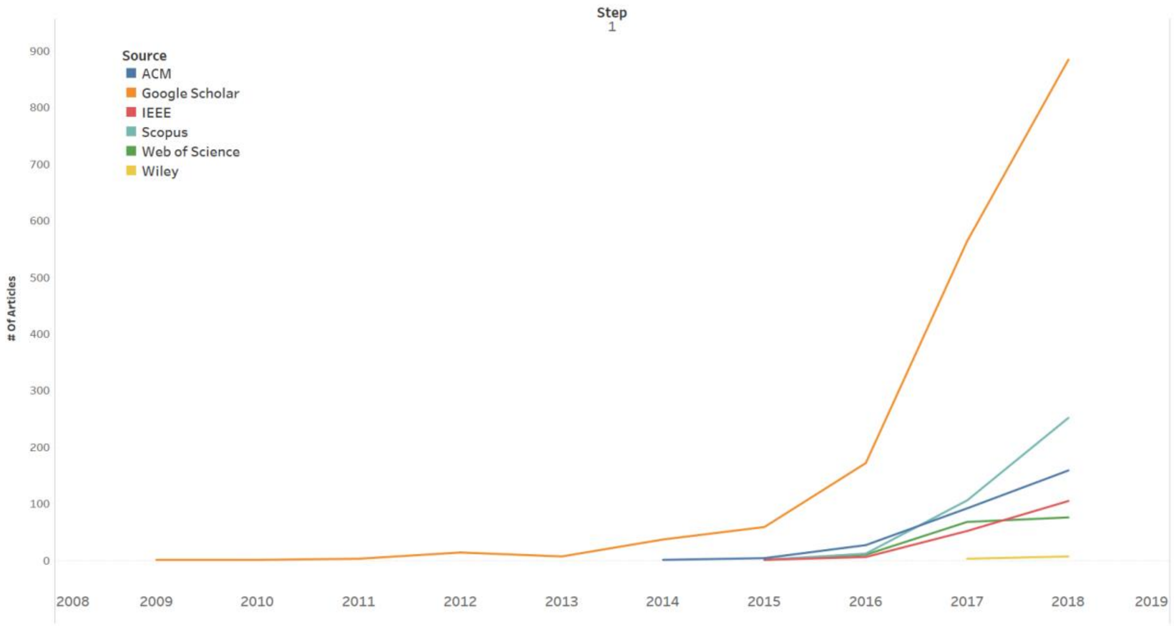

Figure 1: Paper Distribution across years by source – Results of Step 1

Figure 2 on the other hand describes the distribution of final papers by source and year. It's quite clear that most of the primary papers are from google scholar which mostly consists of whitepapers (35% of overall papers). Nevertheless, academic papers are spread across three resources (ACM, IEEE and Google Scholar) and make up 65% of the final papers. Throughout the results, one trend is clear – most of the papers are from recent years with 61% of papers from 2018 and the rest from 2015, 2016, 2017.

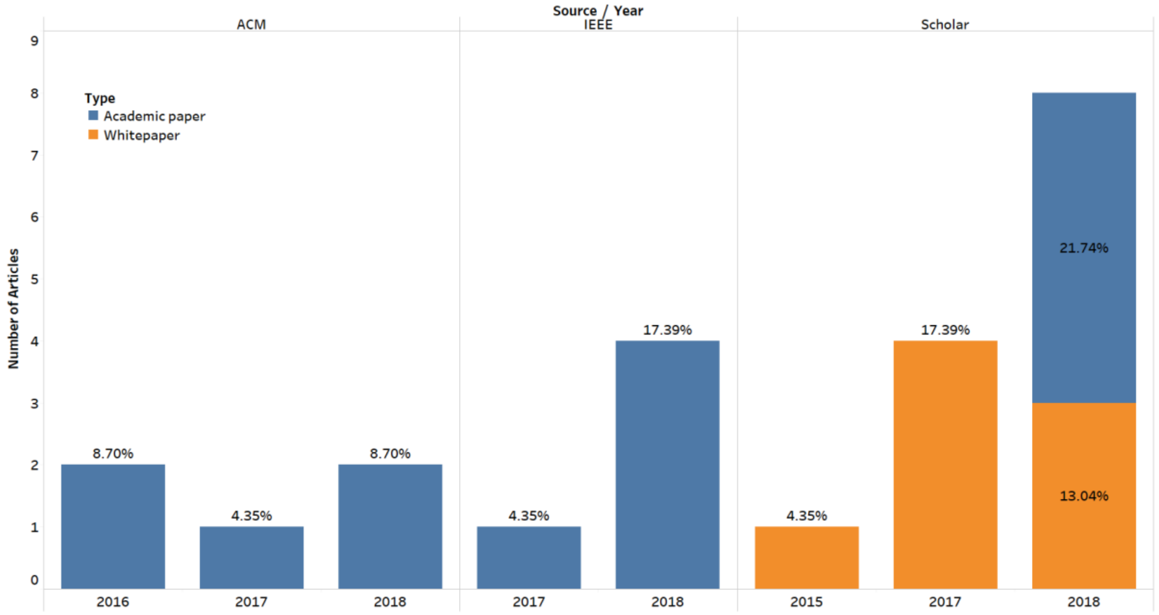

Figure 2: Distribution of final papers by source and year

4 Results

In this section the thesis presents the results obtained from the SLR study in relation to each research question.

4.1 RQ1: Information Types

In this section, the thesis aims to answer the *RQ1: What types of information do oracles provide to blockchain?*

In this thesis, papers describe implementations or frameworks where data generated by an external device (e.g. IoT, vehicle, drone and smartphone) or data that exists on the web managed by an external party is passed to the blockchain, therefore the research has identified the two major information types: *sensor data* and *web content*. While the review also tried to identify possible data sizes mentioned or discussed in the papers, very few papers mentioned size of data but only for the performance evaluation purposes [30]–[32]. These sizes ranged from 10s of kilobytes to a megabyte. Nevertheless, no discussion on limitation on the size of data transmitted via oracles were provided in papers on oracles.

Blockchain for IoT has recently become a hot topic [33] and it's not a surprise that significant among of papers found focus on blockchain and IoT integration with the discussion of devices behaving like oracles. Two papers introduce a novel blockchain based architecture for tackling smart city challenges such as congested traffic and driver safety by recording *sensor data generated by vehicles* [34], [35]. These papers propose using RSU's[2] and sensors embedded in vehicles to collect traffic-related data, road and weather conditions and potentially parking lot occupancy as oracles in blockchain based implementations. This sensor data collected from various IoT devices motivated some researchers to develop blockchain-based data trading platforms. Some papers present innovative data trading platforms aimed at enabling IoT device owners to take full control of their sensor data and sell it in a decentralized marketplace [36]–[38] while others provide an ability to efficiently manage these devices via a platform [39]. The research categorizes these papers as *IoT device readings*, simply because the focus here is on IoT sensor data generated by users and their activities (e.g. smart home devices). These solutions assume that any type of data independent of its type or provenance can be used for trading.

Nevertheless, certain implementations are focused on specific sensor data. *Health data* such as blood pressure, pulse, body temperature collected by wearable or implantable medical devices was subject of developing privacy-oriented Remote Patient Monitoring system (RPM) [40] while *biometric readings* (e.g. identity information) could be collected for the purpose of building a universal Digital ID [41]. Today data is not only generated by our body but also by our clothes, products we consume and our household. While some researchers developed unique RFID chips to collect and track clothing related data such as geolocation, product information in the supply chain [42] others focused on a more general approach of collecting sensor data such temperature, geolocation and unique identifier by combining craquelures [43] with OriginStamp [3] to address the challenge of transparency in *product tracking* in supply chain. [44] Meanwhile, *energy readings* generated by household smart meters was subject of one of the studies aimed at innovating existing smart grids with IoT and blockchain. [45] Lastly, only one study covered *visual feed* (image and video) that is sent through drones to an oracle with later injection into blockchain with the purpose of

[2] Roadside Units or Road Infrastructure Units

[3] OriginStamp is a web-based, trusted timestamping service that uses the decentralized blockchain to store anonymous, tamper-proof time stamps for any digital content [72]

building a trusted and integrated environment for drone data exchange [31]. Papers described above share a common type of information which is sensor data due to those projects aiming to bring data generated by various devices (e.g. IoT, vehicles, medical appliance, RFID chips and smartphones) into the blockchain either directly via the devices or via the combination of the device and a software module.

In contrast to *sensor data*, information that lives on the web has been one of the early elements used in various blockchain implementations aiming to solve the blockchain oracle problem [46], [47]. By content that lives on the web, this research means and includes data such as financial information, sports results, weather updates, user input (e.g. answers to 'Yes/No' questions) and this data in certain implementations is important for successful and correct execution of smart contracts. [2] *Web content* differs from *sensor data* in that it is easily available via the browser and doesn't directly originate from a physical device. This data along with other user or 3-rd party generated content (e.g. Bloomberg, Financial Times) is described in primarily studies and whitepapers that introduce a blockchain framework which have oracles as their core building block. Projects such as EdenChain, an asset tokenization platform [48], PDFS and Town Crier, an authenticated data-feed service for smart contracts, aiming to deliver trusted data input from existing web services[22] [2], utilize heavily data existing on the web to deliver their promises. Certain implementations are more focused on describing how web content can be used to safely and securely inject data into blockchain implementations by delivering various application and data transport level protocols [22], [32]. Papers mentioned above were categorized under *generic http(s) data*, because they interact with internet resources (e.g. sites, feeds) via the web.

Table 4
RQ1: Information Types

Information Type	Information Sub-type	Papers
Sensor Data		
	IoT Device Readings	[36]–[39]
	Energy Readings	[45]
	Vehicular Sensor Readings	[34], [35]
	Biometric Readings	[41]
	Health Readings	[40]
	Product Tracking Data	[42], [44]
	Visual Feed	[31]
Web Content		
	Generic HTTP(S) Data	[2], [4], [5], [22], [32], [48]
	Boolean or Scalar	[1], [3], [23], [30], [49]

Boolean propositions (Yes/No) and *Scalar* measurements (e.g. age of a person) are a subtype of web content and in the study and are used solely by specific type of blockchain solutions. It is a simple type of data that is used in mostly prediction markets [1], [3], [23], [30], [49], to check for correctness of a certain claim in order to ensure fair outcome of a bet. These solutions accept simple yes or no responses as well as discrete or continuous response [1], [23], [49]. In summary, papers presented above shared a common characteristic where they were developed with the goal of bringing information that already exists on the web into the blockchain and bridging the existing gap between web content and blockchain.

As a result, the thesis has presented two major categories of sensor data and web content along with sub-categories in Table 4. Sensor data and web content was the most logical separation of information by types due to the various challenges these papers set out to address as well as the essence of their blockchain solutions.

4.2 RQ2: Oracle Properties

Understanding blockchain oracle properties is essential in designing novel blockchain solutions, which rely on oracles. This section aims to answer the *RQ2: What are the properties of oracles for use in blockchain solutions?* and map out properties that help define blockchain oracles. Oracle properties are essential characteristics of blockchain oracle solutions, which are important to know when utilizing or building a blockchain oracle. Firstly, the review starts exploring the oracle properties from a high level by identifying blockchain types used in blockchain oracle projects. This property is further explored in the **Blockchain Types** subsection. Next, the review analyzes oracles from the perspective of its representation in the real world (e.g. device, software, etc.). Author addresses this property in the **Oracle Types** subsection. Ultimately blockchain oracle problem is a trust problem. According to some [48], this trust includes reliable data transfer from data source to oracle, reliable code execution in the oracle and reliable data transfer from oracle to the blockchain. The review explores various encryption methods to understand what mechanisms are utilized to address the challenge of reliable data transfer in the **Encryption Methods** subsection. This section also covers the discussion around reliable code execution in the oracle, which is reviewed in the **Oracle Confidentiality** subsection. Finally, ensuring authenticity while transferring information to the blockchain is critical for blockchain security and usability. The review outlines **authentication mechanisms** in the relevant subsection.

4.2.1 Oracle Type

The need for defining oracle types stems from the importance of organizing the oracles to help facilitate blockchain solution developers in quickly navigating through the landscape of potential options. Some [50], [51] have divided the blockchain oracles into the following categories:

1. Software oracles – oracles that push information available online to blockchain
2. Hardware oracles – oracles that push information from physical devices (e.g. sensors, RFID chips, etc.) into the blockchain
3. Inbound oracles – oracles that provide smart contracts with data from external world
4. Outbound oracles – oracles that send information to the outside world
5. Consensus based oracles – data passed to blockchain is treated as a result of consensus of multiple oracles

While the above categorization, can be quite useful, it is ultimately redundant since software oracles and hardware oracles can be inbound. Another challenge is lack of clear division between the functionality of oracles according to how the information is verified to ensure truthful data is injected into the blockchain. The analysis found that while there are many hardware devices that are oracles at times, those oracles consist of multiple hardware elements.

When defining oracle types, two main criteria are used: *physical attributes* and *decision-making capacity* (see Table 5). *Physical attribute* refers to whether the oracle is pushing the information from a physical device and/or the oracle itself is a physical entity or whether an oracle is an intangible entity (e.g. piece of code) collecting information from intangible sources (e.g. web sites). Two physical attributes emerge in the review of the papers and are presented in Table 5 as row headers. *Tangible* oracles include papers where information flow starts from physical device (e.g. IoT device, vehicle sensors, RFID chips etc.) and at times this device acts as an oracle or is part of a set of entities comprising an oracle. On the contrary, *intangible* oracles emerged from papers where oracles are presented as software code, running on one or many computer nodes, with the goal of bridging the gap between blockchain and external to blockchain data. *Decision making capacity* captures the mechanism behind how the decision for passing the information into the blockchain is made. Two categories representing decision making emerge in the review of the papers and are presented in Table 5 as column headers. When a single node/entity transmit data from a single source to the blockchain oracle, a *centralized* approach was taken, while when single or multiple nodes/entities were used to request information from multiple data sources a *decentralized* approach was used. Some papers were a combination of both: a collection of hardware and software components which constituted an oracle. In those cases, source of information and those papers where source of information was coming from a physical device (e.g. IoT device) were categorized as tangible oracles.

Table 5
Oracle Types

	Decision-making capacity		
Physical Attribute		*Centralized*	*Decentralized*
	Tangible	[31], [36]–[42], [44], [45]	[34], [35]
	Intangible	[2], [22], [48]	[1], [3]–[5], [23], [30], [32], [49]

Studies that discussed *tangible centralized oracles* include smart meters in an IoT aided smart grid [45], an RFID chip, used to track piece of clothing in a supply chain [42], mobile device used for biometric authentication [41], various on-body sensors, enabling tracking of vital information [40], products with a unique pattern collecting product relevant information [44] and pretty much any IoT device that collects valuable data, tradeable on a data trading platform [36]–[39], or is used to manage IoT resources within a network [39]. In all the above projects a single oracle was used to make decisions on the information. Two projects involving road side units collecting data about the environment through nearby vehicles [34], [35], built a mechanism where road side units (RSU), which acted as blockchain nodes, would also check with other nearby vehicles the presence of the vehicle which requested to join the network and provide sensor data. The vehicles would also interact with each through smart contracts to verify some of the information (e.g. vehicle identity) or request data with each other in a *decentralized* manner.

On the contrary, *intangible decentralized oracles* mainly dealt with bringing information from third parties usually via the web for various purposes such as resolving prediction market bets [3], [23], [30], [49] or leveraging human intelligence to prove the truth of propositions [1], [4]. *Intangible centralized oracles* included projects where authors built a data feed for smart contracts with trusted content providers as information sources [2], [22].

4.2.2 Blockchain Type

This section presents the summary of various blockchain types used across primary papers. Blockchain types are categorized based on the access control mechanism implemented. Public or *permissionless* blockchains allow anyone to join and participate in the network, while private or *permissioned* blockchains exercise a more tightly controlled access [11]. Some authors [36] used the combination of a permissionless and private blockchain network where, while the network was open for everyone to access, certain operations (e.g. write) were limited to a group of individuals. This characterization constituted the *semi-permissioned* category. In studies where authors emphasized that both networks would be feasible a separate category – *both* - was assigned to those papers. In the review, along with blockchain types, a blockchain network information was also extracted to provide a more granular view. Blockchain networks are specific implementations of the blockchain technology (e.g. Bitcoin, Ethereum, Hyperledger). Some authors did not use existing blockchain networks and presented a *novel* blockchain network either based on existing networks or simply leveraging blockchain technology.

Most of the primary papers (see Table 6) opted in for a *public permissionless* blockchain due to the nature and goal of their research or implementation. *Ethereum* – public permissionless blockchain network – was the choice of studies where a decentralized approach to building an oracle [1], [5], [22], [23], [38] as well as a data feed service for smart contracts was presented [2], [32]. Nevertheless, some [3], [4] decentralized oracle network protocols, and prediction market platforms [30], [49] have opted in for developing their own blockchain network with specific logic fitting to their needs and goals. Developers of self-verifying RFID chips [42] and cloud-based drone system [31] use custom made blockchain networks specifically to cater for the needs of their respective solutions. While the former authors implemented *WaltonChain* the latter blockchain developers used *DroneChain* developed for the purposes of presenting a supply chain tracking and a resilient architecture for IoT respectively. A notable mention among the papers, which used public permissionless network, is the research paper [44] prepared by researchers at the University of Constanza where they used OriginStamp – a web based, trusted timestamping service the uses decentralized blockchain to store anonymous, tamper-proof timestamps of any digital content [52] – to capture product meta information (e.g. location, temperature, noise or acceleration) collected by a mobile device. This web service allows the user to choose the type of blockchain to store the hashed timestamp. [52]

Table 6
Blockchain Types

Blockchain Type	Blockchain Network	Papers
Public permissionless		
	Ethereum	[1], [2], [5], [22], [23], [32], [38]
	OriginStamp	[44]
	Aeternity	[3]
	Witnet	[4]
	Waltonchain	[42]
	Dronechain	[31]
	Prophet	[49]

	Truthcoin	[30]
Permissioned		
	Ethereum	[39], [48]
	Hyperledger Fabric	[37]
	SpeedyChain	[35]
	Not Available	[34], [40], [45]
Both		
	BlockID	[41]
Semi-permissioned		
	ChainAnchor	[36]

Another blockchain type used in the papers was *permissioned*. EdgeChain – an edge IoT framework based on blockchain [39] – and EdenChain – a programmable economy platform [48] – built their solutions on top of private *Ethereum* blockchain, where the project owners had set up their own access mechanism and environment using Ethereum's technology stack. Datapace [37] – a decentralized data marketplace on blockchain – leveraged Hyperledger Fabric's technology to ensure digital asset management, a critical component of their project. To support the complex mechanism behind vehicular blockchain network some [35] developed a novel network called *SpeedyChain* while others did *not* make the network information *available* [34], [40], [45] and simply mentioned using a custom blockchain solution.

Blockchain-based identity management [41] and cloud based device commissioning [36] project authors have suggested using *both* or *semi-permissioned* types of blockchains and called their custom blockchain networks *BlockID* and *ChainAnchor* respectively.

4.2.3 Encryption Methods

Discussion of encryption is important to understanding methods used to ensure reliable data transfer. The *encryption methods* represents the technology or a method of cryptography used to secure the communication between two entities while the *encryption techniques* presented (see Table 7 & 8) here include protocols (e.g. TLS, TLS-N), cryptographic algorithms (e.g. ECC) as well as security schemes that support secure data transfer between entities. These techniques were either discussed in the papers or could be deduced from the context.

Today there are two widely referenced and used cryptographic encryption methods: *symmetric* (private) and *asymmetric* (public). [53] [54] The main difference between the two is that in symmetric cryptography only one key is used to encrypt and decrypt data. This key plays an important role in the process and has to be distributed between entities before the transmission. [54] The main challenge with this method of cryptography is the secure exchange of the keys prior to encryption/decryption. Asymmetric key encryption or cryptography, also known as public key encryption, solves the issue of secure exchange by using two keys: private keys, which are used for encryption, and public keys, used for decryption.

The review found that *Public Key Infrastructure (PKI)* was one of the major (see Figure 3) data encryption mechanisms utilized in blockchain solutions. PKI is a hybrid of symmetric and asymmetric encryption methods. The handshake, a process enabling the client and server to establish communication[55], uses asymmetric encryption to exchange the secret key used for symmetric encryption. Once the secret key is exchanged, the rest of the communication uses symmetric encryption. [56] Even though PKI is not just an encryption method, but rather a set of hardware, software, policies, processes and procedures to run secure communication [57] it contains both of the encryption methods discussed above.

Blockchain oracles communicate with the blockchain as well as with the external data source, thus indicating that two directions of reliable data transfer need to be addressed: *data-source to the oracle* and from *oracle to the blockchain*.

4.2.3.1 Data Source → Oracle Encryption

Protecting the information transmitted or collected from *external data source to the oracle* is critical to the integrity of blockchain solutions, due to the immutable nature of blockchain networks.

When discussing encryption of data flowing *from data sources to the oracle*, majority (see Figure 3) of the papers utilized *PKI* to encrypt communications (see Table 7), with only three papers using specifically *asymmetric*, one paper using *symmetric* encryption methods, due to it being less compute intensive, when transferring body sensor data to an oracle and the rest missing or *not discussing* the encryption method at all.

Most papers of the studies, where *PKI* was a preferred encryption method, used *TLS* as their encryption technique mostly due to the nature of those solution heavily relying on third party information from secure websites [1]–[5], [22], [23], [30], [48]. TLS or Transport Layer Security and is a cryptographic protocol that provides authentication, privacy and data integrity between communicating entities [58] and is the prevalent form of secure communication on the internet [59]. One paper presented a solution called *TLS-N*, a novel communication protocol which acts as an oracle and is built on top of TLS. [32]

While some [40] use *symmetric* cryptography, with *no discussion* of technique, to encrypt patient data coming from sensors on the human body via the mobile device others resorted to utilising *asymmetric* cryptography to protect fingerprint or identity data [41] and tracking details of a clothing item [42]. In addition, an Elliptic Curve Cryptography (ECC) was utilised to provide reliable data transfer between vehicles and road side units in the vehicular blockchain network [34].

Table 7
Encryption used for Data Source → Oracle

Encryption Method	Encryption Technique	Papers
PKI		
	TLS	[1]–[5], [22], [23], [30], [48]
	TLS-N	[32]
	Not discussed	[31], [35], [36], [49]
Symmetric Cryptography		
	Not discussed	[40]
Asymmetric Cryptography		
	ECC	[34]
	Not discussed	[41], [42]
Not covered		[37]–[39], [44], [45]

The rest of the papers [37]–[39], [44], [45] *did not explicitly cover* the encryption methods or techniques used at times simply mentioning that encryption would be used. While it is counterintuitive and dangerous to omit describing encryption from data source to the oracle, these papers were making various assumptions such as the data coming through the internet via an API, but making encryption information not explicitly available to the reader.

4.2.3.2 Oracle → Blockchain Encryption

Apart from receiving data from external sources, oracles also transfer information to the blockchain. In this scenario there are similar encryption methods and techniques used but it differs due to the implementation principles chosen by authors.

Table 8
Encryption used for Oracle → Blockchain

Encryption Method	Encryption Technique	Papers
Asymmetric		
	ECC	[3], [22], [30], [32], [34]
	ECC-TC	[48]
	Not explicitly discussed	[2], [40], [42], [44]
Not covered		[1], [4], [41], [45], [49], [5], [23], [31], [35]–[39]

Asymmetric cryptography [60]is the most widely used encryption method for blockchain networks. The review found that in the papers where this oracle to blockchain data transfer was discussed, this was the only method presented. *Elliptic curve cryptography (ECC)* is a form of asymmetric cryptography and is used in Bitcoin and Ethereum. Most of the studies which discussed an encryption technique mentioned that ECC was the method they resorted to. Truthcoin and Aeternity blockchain projects, both decentralized prediction market platforms [3], [30], both emphasize that they are building their oracles within their novel blockchain platform, thus an assumption of using ECC could be made. Only developers behind EdenChain, a programmable economy platform aiming to capitalize and trade all types of assets through programmable economy technology, have combined *ECC with threshold cryptography (TC).* Threshold cryptography is a protocol with a cooperative property. Data necessary for decryption is shared among participants so that encrypted data can be decrypted only when data of other participants is present as well as yours. [48] This in combination with ECC allows for a secure decentralized exchange of information. The rest of the papers [2], [40], [42], [44] mentioning asymmetric method did not cover the technique they used.

Majority of the studies (see Table 8) did *not cover* the encryption method or technique used when transferring information from oracle to blockchain. While it is possible to hypothesize various reasons for this choice, including the assumption encryption method that widely used by blockchain solutions – elliptic curve cryptography (ECC) [61] – would be a default choice, the review only categorized the study if an explicit mention was made or when such an assumption could be deduced from the overall content.

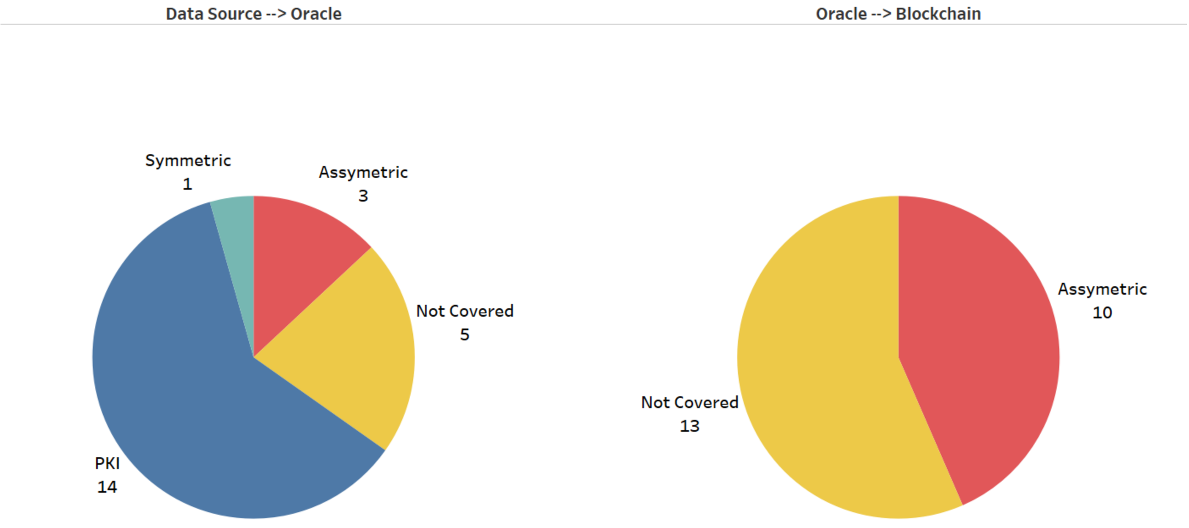

Figure 3: Encryption methods used by data transfer direction

In summary, it can be observed the reliable data transfer from oracle to the blockchain is discussed less often (see Figure 3) throughout the papers, in comparison with data transfer from external sources to the oracles. While there are multiple reasons for this, the assumption could be made that the most prevalent justification is that blockchain projects either assume the familiarity of the reader with the encryption used within blockchain or simply decide to omit this information due to the nature of their solution. Regardless, this is an important aspect to consider when developing or choosing a blockchain oracle.

4.2.4 Oracle Confidentiality

The goal of encryption, specifically in the context of blockchain oracles, is to ensure the secure data transactions between the external entities and the blockchain and to prevent tampering of the network data by an unauthorized third party. Some [1], [2], [22], [32], [38], [41], [48] papers discuss the oracle confidentiality from the perspective of the execution of logic inside the oracles as well secure transmission of sensitive query parameters [5]. Mostly whitepapers accentuated the role of protecting the confidentiality of oracles' code execution, while papers discussing tangible oracles, with the exception of one, did not discuss this as a challenge.

We found that four papers raised a concern regarding oracle confidentiality. Oracle confidentiality is concerned with ensuring privacy of data while it exists within the oracle. It is also concerned with a potential situation where an intruder could manipulate the code to modify oracle's behaviour as well as have access to sensitive information that has been requested from the external data source.

In order to address this challenge three solutions have used *Intel's Software Guard Extension (SGX)* technology in order to provide confidentiality to the data transmitted and processed. The technology is an extension to Intel Architecture which allows an application to run within a protected container called an enclave. Application code run within this enclave is protected and secured against any malicious intervention [62]. While only three papers actually used Intel SGX to protect the logic in their systems [5], [22], [48] and one solution used a similar offering from *ARM (ARM Trust Zone)* [41] three papers briefly discussed or mentioned this as a potential solution to provide higher level of protection but have indicated that trusting a central authority like Intel to ensure this type of confidentiality would defeat the purpose of decentralization and independence. [2][1][32]

4.2.5 Authentication Mechanism

In this section, various authentication mechanisms discussed in the final papers are presented. In the context of blockchain oracle solutions, the concept of *authentication* is the ability of an external information provider to prove their identity (authenticity) and ultimately ensure that data passed is coming from where it is claimed to be coming from. *Authentication mechanism* is an approach used to perform authentication.

Similar to encryption methods, authentication can be performed within two directions of communication: *data source to the oracle*, describing how data source (e.g. device, web site, etc.) is authenticated when connecting with an oracle, *oracle to the blockchain*, presenting the mechanism used by an oracle to communicate with blockchain.

When it comes to either authenticating an oracle to interact with the blockchain or authenticating a data source to interact with the oracle it's important to discuss the notion of *digitally signing the content*. Many of the challenges in constructing secure oracles arise from the fact that existing data sources don't digitally sign the data they serve. If they did, then oracles would not need to be trusted to refrain from tampering with data. A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document.[63] HTTPS, the protocol for secure web communications, does not enable data signing. It does, though, have an underlying public-key infrastructure (PKI) that requires servers to possess certificates that could in principle support data signing.[5].

HTTPS is based on TLS encrypted transport protocol and a supporting PKI composed of certificate authorities (CAs) – entities that are trusted by users' to vouch for the identity of the third party provider, specifically their web servers. CAs are trusted organizations that issue digital certificates.[64] Digital certificate is an electronic document issued by CA. It contains the public key for a digital signature and specifies the identity associated with the key, such as the name of an organization. The certificate is used to confirm that the public key belongs to the specific organization. The CA acts as the guarantor. Digital certificates must be issued by a trusted authority and are only valid for a specified time. [65] In a way these CAs attest for the authenticity of the entity and can be used to verify the origin but don't necessarily indicate the authenticity of the data they communicate. Thus, even if a data source on the internet is authentic, the data it transmits might not be. On the other hand, smart contracts on the blockchain use signed messages to verify the authenticity of the message and/or of the sender.[66] Thus in the context of authentication the thesis explored various mechanisms used to authenticate oracles or data sources.

4.2.5.1 Data Source → Oracle Authentication

Table 9 presents a summary of authentication mechanisms used when establishing communication between data source and an oracle.

In studies [31], [34]–[36], [45] where *devices* or sensors on devices acted *as data sources* authentication was not discussed. Some papers mentioned or discussed using the TLS for authentication [5], [22] or using a the TLS to perform signing of manifests (content metadata) when transferring data to the oracle [2]. Using TLS as authentication implies *trusting a certification authority* due to the underlying architecture of TLS, which was shortly discussed above.

Table 9
Data Source -> Oracle Authentication

Authentication	Papers
Not explicitly discussed	[1], [3], [4], [23], [37]–[39], [48], [49]
Device is the data source	[31], [34]–[36], [45]
Trusted Certification Authority	[2], [5], [22]
Custom method	[32], [44]
Biometric authentication	[41]
Device ID	[42]
Digital signature	[30]
Lightweight authentication protocol	[40]

TLS-N, a protocol level solution enhancing TLS to generate proofs about the content of the TLS session [32] and Craquelure, a unique pattern of cracks that develops across the surface of a paint [44] are both examples of *custom methods of authentication* which were found to be used to ensure authentication, security and, in case of craquelure, tracking of products. Researchers behind continuous patient monitoring system mentioned that *lightweight authentication protocol* would be used to authenticate sensors on human body with their mobile device but did not elaborate further [40] while Truthcoin, one of the earliest prediction market platforms discussed using private keys of individual to sign their votes on certain outcomes, generating *digital signatures* when communicating their inputs to the platform [30]. Using *biometric authentication* to enable users to manage their identities on the blockchain [41] and *device ID* to authenticate RFID trackers were other scenarios encountered by the review throughout the papers. In the rest of the papers, which constituted the majority, authentication was *not explicitly discussed*. While it would be possible to make assumptions regarding the type of authentication based on other parameters (e.g. prediction platforms using TLS for authentication [4], [23], [49]), these studies did not contain enough contextual evidence to make these assumptions.

4.2.5.2 Oracle → Blockchain Authentication

Oracles also need to authenticate to the blockchain, when communicating, to prevent unauthorized access and meddling with the network. Exceptions to this are solutions where oracles are either built-in to the existing blockchain or to novel blockchain platforms (e.g. dApps). The review found that even though an assumption of using digital signatures to authenticate oracles in communicating with blockchain could be made for the above mentioned solutions, almost all of these papers, which is the majority, *did not explicitly discuss* (see Table 10) this type of authentication. Nevertheless, *digital signatures* either in combination with digital certificate issued to a vehicle by a government authority [34] or generated on mobile device [41] or RFID reader [42] or just generated within the oracle [4] were important to ensure authentic interaction. Three papers discussed us using *Enhanced Privacy Identifier (EPID)* to digital sign data using hardware protected key. EPID is offered by Intel and is also supported in Intel SGX, thus those papers who used SGX used this functionality to ensure oracle authentication. Exception is EdenChain, where even though SGX was used, the paper did not discuss using this technology for authentication and the paper author did not assume so. Intel EPID authenticates platform identity through remote attestation using asymmetric (public and private key) cryptography.[67] Town Crier and ChainLink use EPID to digitally sign data before passing into the blockchain. [5], [22], while researchers behind ChainAnchor use it to prove the provenance of devices without relying on an external third party.[36] *Trusted Certification Authority* as discussed in the beginning of this section is a component of public key infrastructure and was explicitly mentioned to be used for authentication in PDFS, data feed service for content providers [2]. PKI uses *public, private key*, but developers at speedy chain did not specifically use PKI but rather discussed implementing public, private key in combination with location data to authenticate the vehicles sending data to the blockchain nodes. [35] A similar approach was taken by the authors of patient monitoring system on blockchain, with the exception of location data not being transmitted [40]

Table 10
Oracle -> Blockchain Authentication

Authentication	Papers
Not explicitly discussed	[1], [3], [23], [30], [37]–[39], [44], [48], [49]
Digital signature	[4], [34], [41], [42]
EPID	[5], [22], [36]
Novel method	[32], [45]
Public, private key	[35], [40]
Device ID	[31]
Trusted Certification Authority	[2]

In summary, authentication methods often were not an important part of the discussion about blockchain oracles. Nevertheless, this approach in these studies can be attributed to the fact that the focus of the studies was on presenting the solution from a high level and explicit details of authentication mechanisms were not covered. It could potentially be deduced from the papers the type of authentication given the context, but lack of contextual evidence prevented these assumptions.

4.3 RQ3: Oracle Data Verification

In this section, the paper aims to answer the *RQ3: How do oracles verify the data they provide to blockchain solutions?* In the context of blockchain oracles, according to some [48], data verification is the problem of reliability of whether the data provided by the external system is correct and true. Data verification is crucial for blockchain solutions since it aims to ensure that the data matches the original source and is not different to the original data collected and represents the truth. Simply put, data verification is ensuring that information that is collected from external source is correct and true before passing into the blockchain. This is especially important for solutions presenting intangible oracles (e.g. prediction markets). Data verification mechanism represents the specific approach taken by a study to address the challenge of data reliability.

Consensus mechanisms are one of the ways data verification was handled by the studies (see Table 11). Consensus mechanism is a core concept in blockchain that ensures one version of truth is agreed upon by all the nodes [10]. Consensus mechanism can have different approaches. Simple *majority voting* was used in most of the studies using consensus to verify the correctness of the information [1], [4], [5], [22], [49]. Majority voting uses the wisdom of the crowds to make a final decision about the truth or correctness of the information, while weighted voting uses a similar approach with each individual vote having a specific weight assigned to it.[10] Some decentralized prediction market platforms such as Augur or Truthcoin implemented *weighted voting* consensus mechanism [23], [30]. Only one paper has used a *hybrid of Proof of Work (PoW) and Proof of Stake (PoS)* consensus mechanisms to verify data passed to their built-in oracle [3].

Table 11
Data Verification Approaches

Data Verification	Verification Mechanism	Papers
Consensus		
	Majority Voting	[1], [4], [5], [22], [49]
	Weighted Voting	[23], [30]
	Hybrid of PoW & PoS	[3]
No Data Verification		
	Trusted Third Party	[2], [31], [41], [48], [32], [34]–[40]
	Not discussed	[44], [45]
Self-verification	RFID Signature Verification	[42]

Trust in the third party was the most common approach (see Figure 4) to tackling data verification at its core. Technically studies which resorted to this mechanism performed *no data verification* and simply built partnerships or made assumptions regarding trusting an external authority or data source, including trusting a device which captures information from the environment (e.g. IoT device, vehicle, sensors etc.). Vehicular blockchain networks [34], [35] had to trust the central government authority for issuance of legitimate vehicle plates, while data feed service platforms [2], [32] put that trust into the web content providers. Some [37] indicated the necessity of certified equipment when deploying IoT devices, while others involved IoT data brokers to anonymously verify device provenance[4] [36] or to mediate data exchanges [38]. Others [31], [39]–[41] simply trusted by default the IoT nodes and devices, owned by either users or organizations.

[4] According to Oxford dictionary, provenance is defined as the place of origin or earliest known history of something.

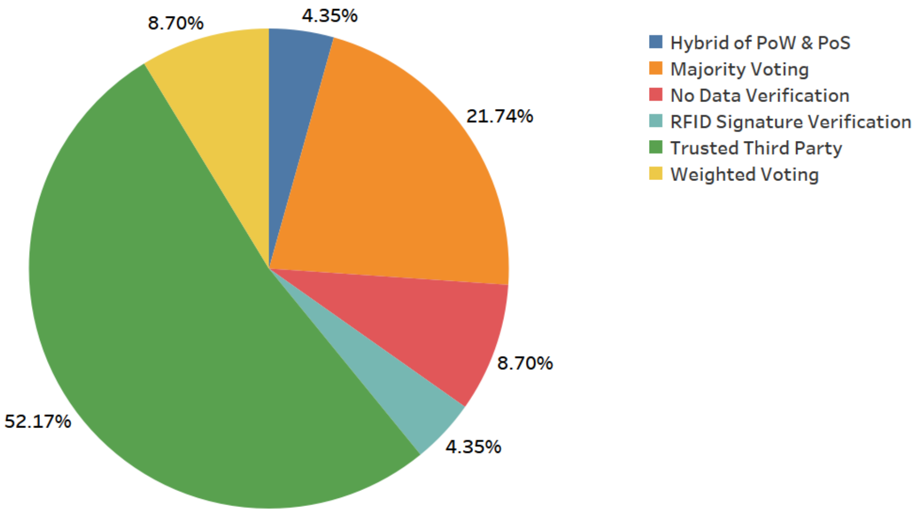

Figure 4: Data verification mechanisms used in studies

Two notable solutions where *no data verification was discussed* provided their method of dealing with data integrity and original data is not tampered with at the device level. Developers of blockchain based smart grid infrastructure elaborated on using public repositories to check the vendor, model and firmware versions to ensure smart meters are not exposed. In case of exposure the authors suggested excluding the devices in order to ensure incorrect data isn't injected into blockchain. [45] Authors of a blockchain solution that offered a novel solution for supply chain tracking, did not discuss how sensor data would be verified on the smart phone before injecting it into blockchain.[44]

Only one paper, that presented a novel solution for IoT based on blockchain, discussed using data *self-verifying RFID chips* with built-in encryption to tackle the challenge of counterfeit items[42].

4.4 RQ4: Blockchain – Oracle Integration

In this section the review tackles following RQ: *How can oracles be integrated with blockchain platforms?* Oracles, as trusted entities which aim to bridge the gap between blockchain and external sources, require an integration mechanism or approach to ensure they can add value to a blockchain solution. Integration here is defined as a method of connecting the oracle to the blockchain in a way which allows the blockchain based solution to serve its purpose. Integration method describes the blockchain oracle integration effort from a high level, while integration mechanism provides a more granular view on the integration approach (see Table 12).

The review uncovered that smart contracts and software modules play an important role in integrating oracles into blockchain solutions. Smart contract (SC) is a concept introduced by Nick Szabo in 1994 and is defined as "a computerized transaction protocol that executes the term of a contract".[68] Most studies (see Figure 5) relied on smart contract technology and had implemented a *custom smart contract interface* to bridge the gap between the oracle and the blockchain. Other studies developed *custom software modules* to address the above-mentioned gap. Software modules are logical layers that provide additional functionality in order to achieve a certain goal.[69] Few authors have implemented their own *custom* *solution* to integrate oracles into blockchain due to peculiarities and specifics of their project. The rest of the papers (see Figure 5) either simply shipped their oracles *built-in* the blockchain network or did *not explicitly discuss* any integration method. Integration mechanisms varied across papers since almost every study had to choose an approach that best fit the purpose of their project.

Custom smart contract interface is an approach used heavily by decentralized applications or dApps. dApp is a web applications built on top of open, decentralized, peer-to-peer infrastructure services and is composed of at least a smart contract on blockchain developed specifically for the dApp and a web frontend user interface [70]. Several projects [4], [5], [23], [37], [48] decided to utilize existing blockchain platforms and build a dApp. Nevertheless, not all projects presented a dApp but still benefited from a custom smart contract. While some [5], [48], [49] built two smart contracts – on-chain and off-chain– others deployed off-chain (contracts generated by an external to blockchain party) smart contracts on-chain [39] or built an on-chain SC to communicate with an external entity (e.g. server, bridge node, data cubes) [4], [22], [37], [38]. Authors of vehicular blockchain network [34]used a combination of a data storage smart contract (DSSC) and information sharing smart contract (ISSC) to communicate with vehicles acting as oracles. Two data feed service protocols – TLS-N and PDFS – used TLS identities [2] or TLS-N proofs [32] to interact with the on-chain smart contract respectively.

Table 12
Blockchain Oracle Integration Methods

Integration Method	Integration Mechanism	Papers
Custom Smart Contract Interface		
	On-chain and off-chain smart contract	[5], [48], [49]
	Off-chain smart contracts deployed on-chain	[39]
	DSSC and ISSC	[34]
	Chaincode (specialized SC)	[37]
	On-chain smart contract accessing Data Cubes	[38]
	SC able to verify TLS-N proofs	[32]
	Server + on-chain smart contract	[22]
	On-chain smart contract + Bridge node	[4]
	TLS Identities linked to Content Contract	[2]
Custom Software Module		
	RFID Reader + PC with blockchain module	[42]
	Software module (ETSE) + Adapter	[45]
	Control System + blockchain Client	[31]
	Patient Centric Agent	[40]
Custom Solution		
	Blockchain Identity bound to Government ID	[41]
	OriginStamp	[44]
Built-in		[3], [30]
Not explicitly discussed		[1], [35], [36]

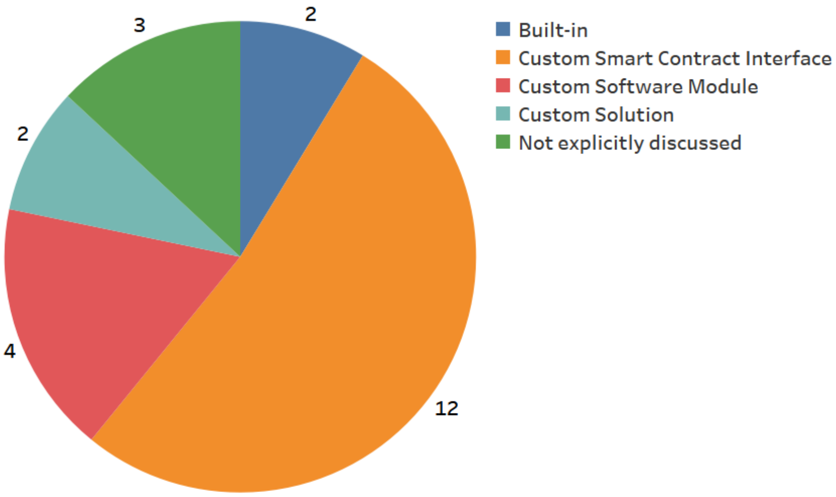

Figure 5: Papers by oracle integration methods

Custom software module approach was an important integration mechanism for projects where physical devices communicated with the blockchain via a specially developed software layer. This software layer serves as an intermediate agent which carries custom logic, manipulates the data coming from the physical device or oracle and then transmits it to the blockchain. Some [42] used an RFID reader and a PC with a blockchain node to deliver supply chain tracking solution using RFID chips with built-in encryption mechanism, others [45] used a combination of a software module and an adapter to bring electricity usage information into the blockchain. A Patient Centric Agent (PCA) – a custom software module presented by one study [40] – transmitted health information from the user's smartphone to the blockchain, performing various calculations necessary for medical professionals use later on. Nevertheless, similar to PCA, a control system for DroneChain monitored and injected the collected visual feed from the drones to the blockchain.

Among the studies, two papers presented a different approach involving a *custom solution* to integrating oracles into the blockchain. A craquelure of a product could be scanned via a mobile device and inserted into the blockchain with the help of OriginStamp, a trusted timestamping service for blockchain, to address the challenges of product counterfeiting [44]. Addressing a similar concern but for false individual identities, a custom fingerprint module using TrustZone[5] was used by authors [41] who presented a blockchain-based identity management solution.

Only two papers opted in for a *built-in* oracle, where they developed the oracle inside the blockchain network avoided developing a separate blockchain, which would act as an oracle. [3][30]

In summary, these integration methods and mechanisms show various attempts to address the challenge of bringing data collected or generated by the oracle to the blockchain.

4.5 RQ5&6: Blockchain Oracle Use Cases & Industries

In this section, the thesis aims to answer the Research Questions 5 and 6, which are:

[5] Secure memory enclave, developed to protect fingerprint data

RQ5: How are oracles used in blockchain solutions?
RQ6: In which industries are oracles proposed to be used?

To answer both of the research questions described above, Table 13 presents the *industry* for which the blockchain projects discussed in the studies developed a solution for, along with a specific *use case*, one scenario or a domain of similar scenarios, the blockchain projects aimed at addressing.

Majority of the primary papers (70%) (see Figure 6) presented solutions that do not necessarily aim to tackle challenges in a specific industry but rather discuss a platform or a service which could be used to address single or multiple challenges across industries. Among these *industry agnostic* solutions, top 3 industry agnostic use cases (see Figure 10) include *decentralized prediction markets* [3], [23], [30], [49], which uses the wisdom of the crowds in combination with reward mechanisms to ensure truthful information is injected into the blockchain, *decentralized blockchain oracle systems* [1], [4], [5], which use the input from many individuals to perform tasks such as data annotation for machine learning or information collection and agreement and *marketplaces for IoT sensor data trading* [36]–[38], which serve to democratize user data and enable individuals to sell their device sensor data. Other, less frequent (see Figure 10) industry agnostic implementations describe projects enabling IoT sensor data management [39], secure drone data collection [31] and asset tokenization platform [48].

Some [34], [35] presented a blockchain network which uses cars as oracles to provide information to the blockchain via road side units (RSUs), that act as blockchain nodes, to provide an includes intelligent management of vehicles, traffic related events (e.g. traffic jams or road accidents) in the *smart cities*. Others, discussed solutions that aim to eliminate product counterfeiting in *logistics* by implementing blockchain based tracking using craquelure and OriginStamp [44] or a self-verifying RFID chip [42].

The rest of the papers presented a remote *continuous patient monitoring system* [40], to enable *medical* professional to keep an eye on their patients and provide timely treatment when interference is necessary; a *blockchain aided IoT smart grid* [45] to enable cost-effective, autonomous *energy* transactions among peers; a *blockchain based identity management* [41] solution to allow *e-government* efforts to bind personal identities to the blockchain to correlate individuals' activities with their identities.

Table 13
Papers by industries and use cases

Industry	Use Case	Papers
Industry Agnostic		
	Decentralized Blockchain Oracle System	[1], [4], [5]
	Data Feed Service	[2], [22]
	Blockchain Oracle Communication Protocol	[32]
	Asset Tokenization Platform	[48]
	Marketplace for IoT Sensor Data Trading	[36]–[38]
	Secure Drone Data Collection	[31]
	IoT Sensor Data Management	[39]
	Decentralized Prediction Market	[3], [23], [30], [49]
Smart City		

	Efficient & Secure Data Sharing in Vehicular Networks	[34], [35]
Medical		
	Continuous Patient Monitoring System	[40]
Energy		
	Blockchain IoT aided Smart Grids	[45]
e-Government		
	Blockchain-based Identity Management	[41]
Logistics		
	Supply-chain Tracking	[42], [44]

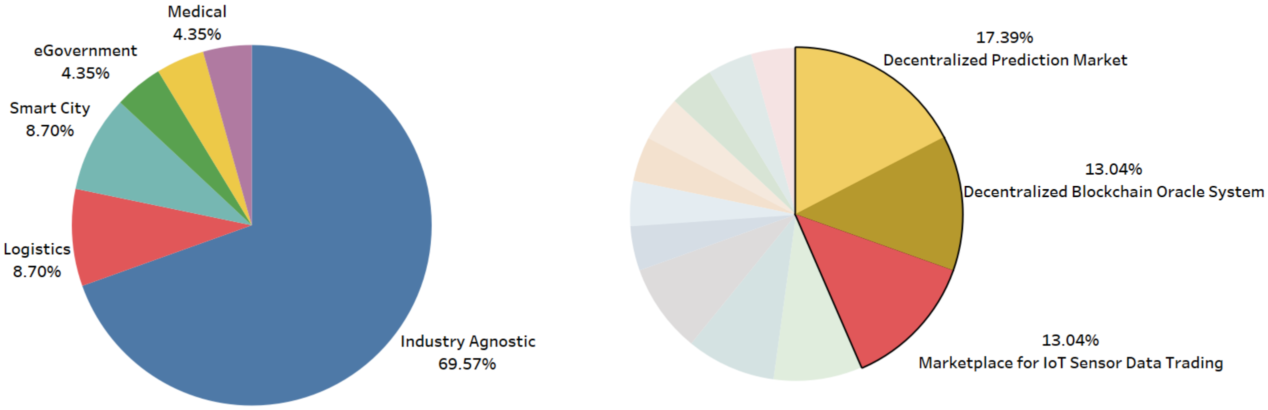

Figure 6: Left: Distribution of papers by Industry. Right: Top 3 use cases

The results presented in this section can be used to claim that majority of blockchain oracle projects are not made with a specific industry in mind but rather aim to tackle a specific challenge which could be relevant across many industries.

4.6 Framework

Blockchain oracle framework, presented in this section, aims to enable blockchain developers and/or project managers to make informed decision regarding the blockchain oracle approach or technology when implementing blockchain solutions. The goal of this framework is to summarize the results of the SLR in a clear and concise manner by describing a model representing the state of the art of blockchain oracles today. It hopes to serve blockchain teams making decisions in their blockchain projects requiring or involving blockchain oracles. The framework (see Table 14) covers the possible scenarios of combinations where certain information type passed through specific oracle types using a pre-defined decision-making mechanism and data verification approach could add value to a blockchain network. A visual representation (see Figure 11) of the framework aims to provide visual cues to the reader and communicate the data flow from left to right.

The structure of the framework is developed based on both the overall flow of data from external sources to the blockchain as well as specific blockchain project considerations need to be considered. When requiring an oracle in a blockchain solution first the authors should identify the *types of information* they need to bring into the blockchain. This decision would pave the way for other choices regarding the oracle properties. Before navigating further, discussing the scope of the project and its stakeholders would help identify the *blockchain type*. If there's a limited number of participants who either already have or will have to the blockchain, then a permissioned blockchain would be appropriate, otherwise a public permissionless network would be the best fit. Next, it's important to ensure that data is securely transported to the oracle, thus the solution developers need to identify an *encryption method* that best serves the purposes. Although there are very few methods, it's important to think about this step. Now that authors have identified the information types and how to secure their transportation, it's important to explore the level of trust necessary to handle this information by exploring *oracle types* and *decision-making approaches.* While a centralized approach can be beneficial in projects that are limited in scope or already use a permissioned blockchain, decentralized approach might be useful for efforts involving multiple actors or to augment a public permissionless blockchain network. Due to the immutable nature of the blockchain, it's critical to set up a *data verification mechanism* to ensure truthful and correct information is injected into the blockchain. Choosing to trust a third party or relying on a form of consensus mechanism will ensure trusted information is injected. Lastly, an *integration* approach is identified based on the above parameters. The framework doesn't include few parameters explored in this study due to absence of enough evidence on information regarding those. Authentication, encryption of data from oracle to blockchain as well as oracle confidentiality contain sparse results which would not add value for blockchain authors in their efforts of bringing external data to their networks.

The reader should interpret the framework (both Table and Figure) from left to right, following the natural flow of data from external sources to the blockchain. In that order, the framework contains most common types of information (see Section 4.1) passed from oracles. The most common types are web data or sensor data coming from or generated by an external source. Next is the blockchain type, which is an important aspect of a blockchain project scope (see Section 4.2.2). This is followed by common encryption methods used in protecting data coming from external sources to the oracles (see Section 4.2.3) which can be public key infrastructure, asymmetric or symmetric. Next are types of oracles (see Section 4.2.1) which receive or at times generate the data. These types are mostly tangible or intangible oracles and their type can be traced to be mostly defined by information that is brought in. This data retrieval by the oracle could be done in either a centralized or decentralized manner (see Section 4.2.1). Moving forward, the data received by the oracle is processed and verified for truthfulness and correctness (see Section 4.3). Lastly the data is inserted into a certain type of blockchain network using the communication channel built by a specific integration method (see Section 4.4). Figure 11 visually supports the table 14 and was designed with the support of blueprint architectures available in most of the study papers (see Figure 7).

As an example, consider a blockchain developer or project decision maker exploring potential approach of bringing an external data to their blockchain via an oracle. In their solution an oracle needs to pass information from an external entity to the blockchain and in deciding on how and what type of oracles to use the authors are confronted with questions such as "What type of information are they dealing with?" and "What is the level of trust they need to ensure?" as well as "How are they going to verify the data coming from external source?". For such purpose the framework would support the authors in their decision making and provide a guideline to either choosing existing oracle solutions or potentially building their own oracle based on those existing offers. To apply the framework, first the authors need to identify the information type that is fed into the blockchain and the blockchain type they are using for their solution as well as an oracle type (e.g. for web data an intangible oracle and for sensor data a tangible oracle would be best). Following this pattern, the author should read the Table 14 from left to right and understand various encryption, oracle decision-making, data verification and oracle integration methods to apply in their specific blockchain solution. As an example, one row from the table could be read in the following manner. To bring data available as a *web content* (e.g. stock price data) authors of EdenChain – a programmable economy platform on *permissioned* blockchain [48]– decided to use an *intangible oracle type,* that receives encrypted data over *public key infrastructure (PKI)* and possesses a *decentralized decision making mechanism*. This solution resorted to simply trusting *a third party* as their *data verification mechanism* to bring this content into a *permissioned blockchain type* which communicated with the oracle via a *custom developed smart contract*.

The illustration (see Figure 11) provides a visual support for the Table 14 in that the above mentioned route could be traced via the upper part of the visual all the way to the right, where information is injected into the blockchain.

Table 14

Blockchain Oracle Framework – Tabular Representation

Information Type	Oracle Type	Blockchain Type	Encryption	Decision-making	Data Verification	Oracle Integration Method	Reference
Web Content	Intangible Oracle	Permissioned	PKI	Centralized Oracle	Trusted Third Party	Custom Smart Contract Interface	[48]
		Public permissionless	PKI	Centralized Oracle	Majority Voting	Custom Smart Contract Interface	[22]
					Trusted Third Party	Custom Smart Contract Interface	[2]
				Decentralized Oracle	Hybrid of PoW & PoS	Built-in	[3]
					Majority Voting	Custom Smart Contract Interface	[4], [5], [49]
						Not explicitly discussed	[1]
					Trusted Third Party	Custom Smart Contract Interface	[32]
					Weighted Voting	Built-in	[30]
						Custom Smart Contract Interface	[23]
Sensor Data	Tangible Oracle	Both	Asymmetric	Centralized Oracle	Trusted Third Party	Custom Solution	[41]
		Permissioned	Asymmetric	Decentralized Oracle	Trusted Third Party	Custom Smart Contract Interface	[34]
			Not Covered	Centralized Oracle	No Data Verification	Custom Software Module	[45]
					Trusted Third Party	Custom Smart Contract Interface	[37], [39]
			PKI	Decentralized Oracle	Trusted Third Party	Not explicitly discussed	[35]
			Symmetric	Centralized Oracle	Trusted Third Party	Custom Software Module	[40]
		Public permissionless	Asymmetric	Centralized Oracle	RFID Signature Verification	Custom Software Module	[42]
			Not Covered	Centralized Oracle	No Data Verification	Custom Solution	[44]
					Trusted Third Party	Custom Smart Contract Interface	[38]
			PKI	Centralized Oracle	Trusted Third Party	Custom Software Module	[31]
		Semi-permissioned	PKI	Centralized Oracle	Trusted Third Party	Not explicitly discussed	[36]

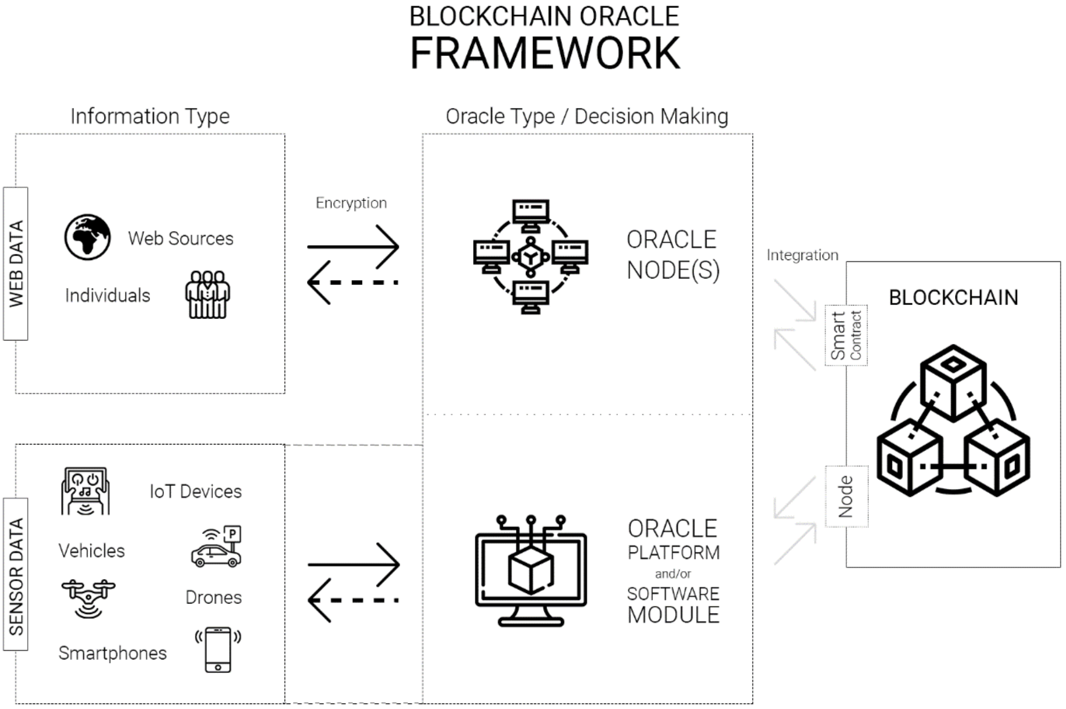

Figure 7: Blockchain Oracle Framework (Visual Representation)

The framework outlined most common scenarios the thesis came across when investigating the blockchain oracles. It can help serve as a reference guide for those either wanting to build blockchain oracles or to use existing offerings to support blockchain based projects by injecting real world data. Considering web content as input, the thesis proposes intangible oracles as software has either centralized (single node) or decentralized (multiple nodes) decision making. It's also possible to develop a blockchain solution that consists of multiple layers of logic (e.g. an oracle platform) but ultimately a certain layer of it would need to possess decision making capabilities as well as data verification, if third party is not trusted.

4.7 Threats to validity

In this section, the thesis discusses possible threats to validity (TTV) based on the mapping of threats prepared by [71]. Threats to validity that are relevant for this SLR are *restricted time span*, *bias in study selection* and *bias in data extraction*.

Restricted time span threat represents the inability of the researcher to anticipate other relevant studies outside the time span within the planning phase. Blockchain is a constantly evolving technology with more applications and technologies introduced on a daily basis. Thus, the author of the thesis could not anticipate other relevant studies simply because they appeared later on and could not have been included in the primary papers. While its challenging to account for this, all of the extracts were dated and could be reproduced if papers before this date are analysed.

Bias in study selection threat stands for the subjective conjecture which reviewers have in the process of search, resulting in them not completely using the inclusion and exclusion criteria. This bias could have been introduced in this review due to the personal knowledge and experience of the author from his experience in the studies as well as knowledge in the area of blockchain oracles. Since the field is not set and stone in terms of definitions and categories, the author could have introduced bias in selection of studies specifically concerning papers where oracles were not specifically named as blockchain oracles. To reduce this type of bias, the researcher has read and reviewed the abstract and the introduction where necessary and possible.

Another similar threat could be the *bias in data extraction.* Since certain concepts in the papers were not explicitly discussed and the author has had made assumptions regarding those, it could be possible that specific invalid or biased assumptions were made. In this case, to reduce this threat the author always indicated in the body of the text that a certain assumption was made or not.

5 Conclusions

Blockchain has become a technological enabler for many industries and innovative solutions. Majority of blockchain projects today operate in the domain of the virtual environment, due to limited exposure of the blockchain to the external to the network data sources. Oracles are entities that enable collection, verification and transmission of this data to the blockchain. The objective of the review was to explore the relationship between the oracles and the blockchain. As part of this, the research explores various types of information and blockchain oracles interact with as well the types of oracles involved in these interactions. Blockchain oracle interaction aspect such as encryption, authentication, data verification and integration were explored to understand the existing landscape of the relationship blockchain and oracles have. As a result, the review enabled the thesis author to present an overarching framework to support blockchain project decision makers and developers in their efforts. In this thesis a systematic literature review, following guidelines proposed by Kitchenham [24], was used to achieve the objective of the research. As a result, 23 papers were collected and examined in detail.

The review showed that web content and sensor data are two information types that are discussed in the blockchain oracle projects. While web content represents information that originates on an internet resource (e.g. website), sensor data is collected by a physical device. This information is handled by two types of oracles, intangible and tangible respectively. Intangible oracles are presented as software code running on one or many computer nodes, while tangible oracles pass information that originates in the physical device. When a single node/entity transmits data from a single source, a centralized oracle is being used, but in case of multiple nodes/entities request data from multiple sources a decentralized oracle is being utilized. Blockchain types used in the oracle solutions included public permissionless, private permissioned, both or semi-permissioned network types, indicating that oracles are used in all types of blockchain networks. The data transferred from external sources to these blockchains via oracles were encrypted using methods such as public key infrastructure (PKI), symmetric or asymmetric encryptions. This highlights the importance of encryption when receiving data from outside of the oracle and the network. Nevertheless, there was a lack of evidence on encryption methods used for securing data flowing from oracles to the blockchain. The review also uncovered that while authentication of external sources with the oracle as well as the oracle with the blockchain can be done using multiple methods, most of the time this was not a discussion in the oracle projects. Having received information from external sources, oracles are tasked with verifying this information for correctness and truthfulness. Consensus, a core concept in blockchain that ensures one version of truth is agreed upon by all the nodes, along with self-verification and no verification were main mechanisms employed to handle the challenge of incorrect data. Important to note that, no data verification mostly implied trusting a third party or an external data source provider in delivering truthful information.

To finally deliver this verified or trusted information into the blockchain, integrating the oracle with it was a must. The review found that common mechanisms to integrate the two were a custom smart contract interface, where developers simply build their own smart contracts and connect it to the blockchain, and custom software module, which required projects to build an additional logical layer to handle the oracle blockchain communication. Nevertheless, custom solutions, which employed their own integration approach, and built-in oracles were also a form of blockchain oracle integration, with few authors simply not covering this topic. The review also found that multiple industries have benefited from blockchain oracle implementations. The efforts to enrich blockchain networks with external sources were made in smart city, medical, energy, e-government and logistics with many authors presenting projects that could benefit multiple or any industry.

Based on the findings of the review, a blockchain oracle framework was developed. The framework consists of seven important aspects. The consumer of the framework starts with understanding the information types his/her blockchain project would deal with, simply because it defines the basic format of the relationship with the external source. Next, an oracle type would be chosen which is tightly related to the data that is the focus of the project. The focus of the project also includes its scope and defines the blockchain type that is either already used or will be used to receive and store the data received from the oracle. Ensuring secure transmission of this data from the source involves considering possible encryption mechanisms. This leads to discussing available encryption mechanisms and choosing one of them to address the challenge of secure data communication. Structurally, this communication can happen either between one node and one source or multiple nodes and multiple sources. This is the decision-making aspect of the oracle, which can be either centralized or decentralized respectively. Data verification is the next step in the framework, which aims to ensure the correctness and/or truthfulness of the data, since it has been received by the oracle already. To transfer this verified knowledge into the blockchain there needs to be a channel between the oracle and the blockchain, the presence of which is ensured by an integration mechanism. Developers should make the decision of the format of integration as a last since it covers more technical level of blockchain oracle solutions. The steps discussed above are advised to be taken in that order because they mimic the natural flow of information from a data source to the blockchain via the oracles.

This framework aims to serve as a guide and support blockchain developers, project managers and/or blockchain project decision makers in their efforts to bring external data into their new or existing networks. It would enable these teams to better understand the existing landscape of oracle offerings, but also be able to design new oracle solutions that would fit their specific needs.

Some of the findings in the review have the potential for further investigation and future research. In our findings we noted that the size of data has not been discussed. Given that the size of data to be transferred to blockchain solutions might be relevant for design choices, it merits further investigation. Additionally, oracle confidentiality might be important for specific projects where very sensitive information needs to be managed by the oracle. In our review, few papers discussed this challenge, and most did not discuss this aspect of the oracles. Thus, this topic can be a valuable avenue for further research.

6 References

[1] J. Adler, R. Berryhill, A. Veneris, Z. Poulos, N. Veira, and A. Kastania, “Astraea: A Decentralized Blockchain Oracle,” 2018.

[2] J. Guarnizo and P. Szalachowski, “PDFS: Practical Data Feed Service for Smart Contracts,” 2018.

[3] J. Pettersson, Y. Malahov, and Z. Hess, “Æternity blockchain,” 2017.

[4] A. S. de Pedro, D. Levi, and L. I. Cuende, “Witnet: A Decentralized Oracle Network Protocol,” pp. 1–58, 2017.

[5] S. Ellis, A. Juels, and S. Nazarov, “ChainLink: A Decentralized Oracle Network,” vol. 2017, no. September, pp. 1–38, 2017.

[6] N. Satoshi and S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic cash system,” *Bitcoin*, 2008.

[7] R. Chatterjee and R. Chatterjee, “An Overview of the Emerging Technology: Blockchain,” 2018.

[8] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, “An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends,” in *Proceedings - 2017 IEEE 6th International Congress on Big Data, BigData Congress 2017*, 2017.

[9] A. Baliga, “Understanding Blockchain Consensus Models,” *Whitepaper*, 2017.

[10] N. Chaudhry and M. M. Yousaf, “Consensus Algorithms in Blockchain: Comparative Analysis, Challenges and Opportunities,” 2019.

[11] D. Yaga, P. Mell, N. Roby, and K. Scarfone, “Blockchain technology overview,” 2018.

[12] L. M. Bach, B. Mihaljevic, and M. Zagar, “Comparative analysis of blockchain consensus algorithms,” in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2018 - Proceedings*, 2018.

[13] CoinMarketCap, “CoinMarketCap,” *coinmarketcap.com*, 2018. [Online]. Available: https://coinmarketcap.com/. [Accessed: 18-Mar-2019].

[14] T. Blummer *et al.*, “An Introduction to Hyperledger,” p. 33, 2018.

[15] N. Szabo, “Formalizing and securing relationships on public networks,” *First Monday*, 1997.

[16] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F.-Y. Wang, “Blockchain-Enabled Smart Contracts: Architecture, Applications, and Future Trends,” *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. PP, pp. 1–12, 2019.

[17] X. Xu *et al.*, “The blockchain as a software connector,” in *Proceedings - 2016 13th Working IEEE/IFIP Conference on Software Architecture, WICSA 2016*, 2016.

[18] B. K. Mohanta, S. S. Panda, and D. Jena, “An Overview of Smart Contract and Use Cases in Blockchain Technology,” in *2018 9th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2018*, 2018.

[19] M. E. Peck, “Blockchains: How they work and why they’ll change the world,” *IEEE Spectr.*, 2017.

[20] D. C. Sánchez, “Raziel: Private and Verifiable Smart Contracts on Blockchains,” pp. 1–58, 2018.

[21] Q. Lu, C. Pautasso, I. Weber, L. Zhu, and X. Xu, “A Pattern Collection for Blockchain-based Applications,” no. July, pp. 1–20, 2019.

[22] F. Zhang, E. Cecchetti, K. Croman, A. Juels, and E. Shi, “Town Crier: An Authenticated Data Feed for Smart Contracts,” in *roceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[23] J. Peterson, J. Krug, M. Zoltu, A. K. Williams, and S. Alexander, "Augur : a Decentralized Oracle and Prediction Market Platform," 2018.

[24] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature reviews in Software Engineering Version 2.3," *Engineering*, 2007.

[25] C. Okoli, "A guide to conducting a standalone systematic literature review," *Commun. Assoc. Inf. Syst.*, vol. 37, no. 1, pp. 879–910, 2015.

[26] A. Fink, "Conducting research literature reviews: From the Internet to paper (3rd ed.).," *Conducting research literature reviews: From the Internet to paper (3rd ed.).* 2010.

[27] Y. Levy and T. J. Ellis, "A systems approach to conduct an effective literature review in support of information systems research," *Informing Sci.*, 2006.

[28] J. J. Randolph, "A guide to writing the dissertation literature review," *Pract. Assessment, Res. Evalutaion*, 2009.

[29] P. Mistiaen, A. L. Francke, and E. Poot, "Interventions aimed at reducing problems in adult patients discharged from hospital to home: A systematic meta-review," *BMC Health Services Research*. 2007.

[30] P. Sztorc, "Truthcoin," 2015.

[31] X. Liang, J. Zhao, S. Shetty, and D. Li, "Towards data assurance and resilience in IoT using blockchain," in *Proceedings - IEEE Military Communications Conference MILCOM*, 2017.

[32] K. Wust, S. Capkun, A. Gervais, H. Ritzdorf, and G. Felley, "TLS-N: Non-repudiation over TLS Enabling Ubiquitous Content Signing," 2018.

[33] M. Conoscenti, A. Vetro, and J. C. De Martin, "Blockchain for the Internet of Things: A systematic literature review," in *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA*, 2017.

[34] J. Kang *et al.*, "Blockchain for Secure and Efficient Data Sharing in Vehicular Edge Computing and Networks," *IEEE Internet Things J.*, vol. PP, no. c, p. 1, 2018.

[35] R. A. Michelin *et al.*, "SpeedyChain: A framework for decoupling data from blockchain for smart cities," pp. 145–154, 2018.

[36] T. Hardjono and N. Smith, "Cloud-Based Commissioning of Constrained Devices using Permissioned Blockchains," pp. 29–36, 2016.

[37] D. Draskovic and G. Saleh, "Datapace - Decentralized Data Marketplace Based on Blockchain," pp. 1–16, 2017.

[38] P. Missier, S. Bajoudah, A. Capossele, A. Gaglione, and M. Nati, "Mind My Value: a decentralized infrastructure for fair and trusted IoT data trading," 2018.

[39] J. Pan, J. Wang, A. Hester, I. Alqerm, Y. Liu, and Y. Zhao, "EdgeChain: An Edge-IoT Framework and Prototype Based on Blockchain and Smart Contracts," *IEEE Internet Things J.*, vol. PP, no. c, p. 1, 2018.

[40] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian, "Continuous Patient Monitoring with a Patient Centric Agent: A Block Architecture," *IEEE Access*, vol. 6, pp. 32700–32726, 2018.

[41] Z. Gao *et al.*, "Blockchain-based Identity Management with Mobile Device," pp. 66–70, 2018.

[42] B. Mo, K. Su, S. Wei, C. Liu, and J. Guo, "A Solution for Internet of Things based on Blockchain Technology," *Proc. 2018 IEEE Int. Conf. Serv. Oper. Logist. Informatics, SOLI 2018*, pp. 112–117, 2018.

[43] S. Bucklow, "The Description of Craquelure Patterns," *Stud. Conserv.*, 2006.

[44] T. Hepp, P. Wortner, A. Schönhals, and B. Gipp, "Securing Physical Assets on the Blockchain," pp. 60–65, 2018.

[45] A. Margheri, V. Sassone, S. De Angelis, F. Lombardi, and L. Aniello, “A Blockchain-based Infrastructure for Reliable and Cost-effective IoT-aided Smart Grids,” pp. 42 (6 pp.)-42 (6 pp.), 2018.

[46] Legal Tech Blog, “The Problem of Blockchain Oracles – Interview with Alexander Egberts,” 2018. [Online]. Available: https://legal-tech-blog.de/the-problem-of-blockchain-oracles-interview-with-alexander-egberts. [Accessed: 08-May-2019].

[47] Blockonomi, “What are Oracles? Smart Contracts, Chainlink & ‘The Oracle Problem,’” 2019. [Online]. Available: https://blockonomi.com/oracles-guide/. [Accessed: 08-May-2019].

[48] J. Ahn, “EdenChain : Programmable Economy Platform,” 2018.

[49] F. Yayun, “Prophet : The Prediction Platform Based on GXChain White Paper,” pp. 0–37.

[50] Blockchainhub, “Blockchain Oracles,” 2019. [Online]. Available: https://blockchainhub.net/blockchain-oracles/. [Accessed: 15-May-2019].

[51] B. Asolo, “Blockchain Oracles Explained,” 2018. [Online]. Available: https://www.mycryptopedia.com/blockchain-oracles-explained/. [Accessed: 15-May-2019].

[52] OriginStamp, “OriginStamp Docs.” [Online]. Available: https://docs.originstamp.com/guide/#about-this-documentation. [Accessed: 01-Jul-2019].

[53] E. Thambiraja, G. Ramesh, and R. Umarani, “A Survey on Various Most Common Encryption Techniques,” *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 2, no. 7, pp. 226–233, 2012.

[54] Y. Kumar, R. Munjal, and H. Sharma, “Comparison of Symmetric and Asymmetric Cryptography with Existing Vulnerabilities and Countermeasures,” *Int. J. Comput. Sci. Manag. Stud.*, vol. 11, no. 03, pp. 2231–5268, 2011.

[55] IBM, “An overview of the SSL or TLS handshake.” [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_7.1.0/com.ibm.mq.doc/sy10660_.htm. [Accessed: 01-Jul-2019].

[56] Infosec, “Cryptography Fundamentals, Part 4 – PKI.” [Online]. Available: https://resources.infosecinstitute.com/cryptography-fundamentals-part-4-pki/#gref. [Accessed: 15-Jun-2019].

[57] Thales, “What is Public Key Infrastructure (PKI)?,” 2011. [Online]. Available: https://www.thalesesecurity.com/faq/public-key-infrastructure-pki/what-public-key-infrastructure-pki. [Accessed: 01-Jul-2019].

[58] ScienceDirect, “Transport Layer Security.” [Online]. Available: https://www.sciencedirect.com/topics/computer-science/transport-layer-security. [Accessed: 01-Jul-2019].

[59] Cloudflare, “What Is Transport Layer Security (TLS)?” [Online]. Available: https://www.cloudflare.com/learning/ssl/transport-layer-security-tls/. [Accessed: 01-Jul-2019].

[60] J. Lake, “Understanding cryptography’s role in blockchains,” 2019. [Online]. Available: https://www.comparitech.com/crypto/cryptography-blockchain/. [Accessed: 01-Jul-2019].

[61] Blockchainhub, “Cryptography & Blockchain – Part 2,” 2018. [Online]. Available: https://blockchainhub.net/blog/blog/cryptography-blockchain-bitcoin/. [Accessed: 01-Jul-2019].

[62] F. McKeen *et al.*, “Innovative instructions and software model for isolated execution,” pp. 1–1, 2013.

[63] SearchSecurity, “Definition: digital signature.” [Online]. Available: https://searchsecurity.techtarget.com/definition/digital-signature. [Accessed: 01-Jul-2019].

[64] Z. Durumeric, J. Kasten, M. Bailey, and J. A. Halderman, “Analysis of the HTTPS certificate ecosystem,” pp. 291–304, 2013.

[65] DocuSign, “Understanding digital signatures.” [Online]. Available: https://www.docusign.com/how-it-works/electronic-signature/digital-signature/digital-signature-faq. [Accessed: 01-Jul-2019].

[66] S. Marx, “Signing and Verifying Messages in Ethereum,” 2018. [Online]. Available: https://programtheblockchain.com/posts/2018/02/17/signing-and-verifying-messages-in-ethereum/. [Accessed: 01-Jul-2019].

[67] Intel, “A Cost-Effective Foundation for End-to-End IoT Security.”

[68] K. Christidis and M. Devetsikiotis, “Blockchains and Smart Contracts for the Internet of Things,” *IEEE Access*. 2016.

[69] D. L. Parnas, “A technique for software module specification with examples,” *Commun. ACM*, 1983.

[70] G. W. P. D. Andreas M. Antonopoulos, *Mastering Ethereum: Building Smart Contracts and DApps*. O’Reilly Media, 2018.

[71] X. Zhou, Y. Jin, H. Zhang, S. Li, and X. Huang, “A map of threats to validity of systematic literature reviews in software engineering,” *Proc. - Asia-Pacific Softw. Eng. Conf. APSEC*, pp. 153–160, 2017.

[72] OriginStamp, “OriginStamp,” 2019. [Online]. Available: https://originstamp.org/home. [Accessed: 08-May-2019].

Appendix

I. Glossary

Caret The bar (or other symbol) marking the active editing point.	Sisestusmärk Märk, mis märgib teksti sisestamise asukohta.
Template A gauge, pattern, or mold, commonly a thin plate or board, used as a guide to the form of the work to be executed.	Mall Näidik, muster või valuvorm, mis esitab täitmisele võetava töö struktuuri.

II. License

Non-exclusive licence to reproduce thesis and make thesis public

I, ________________Kamran Mammadzada__________________________,

(*author's name*)

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,

_______________Blockchain Oracles____,

(*title of thesis*)

supervised by ________Fredrik Payman Milani_________________.

(*supervisor's name*)

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.

Kamran Mammadzada

14/08/2019