TACKLING THE CHALLENGES OF INFORMATION SECURITY INCIDENT REPORTING:

A DECENTRALIZED APPROACH

ALEXIS MICHAIL

A thesis submitted in partial fulfilment of the requirements of the University of East London for the degree of professional doctorate in Information Security

School of Architecture, Computing & Engineering

JANUARY 2020

Abstract

Information security incident under-reporting is unambiguously a business problem, as identified by a variety of sources, such as ENISA (2012), Symantec (2016), Newman (2018) and more. This research project identified the underlying issues that cause this problem and proposed a solution, in the form of an innovative artefact, which confronts a number of these issues.

This research project was conducted according to the requirements of the Design Science Research Methodology (DSRM) by Peffers et al (2007). The research question set at the beginning of this research project, probed the feasible formation of an incident reporting solution, which would increase the motivational level of users towards the reporting of incidents, by utilizing the positive features offered by existing solutions, on one hand, but also by providing added value to the users, on the other. The comprehensive literature review chapter set the stage, and identified the reasons for incident under-reporting, while also evaluating the existing solutions and determining their advantages and disadvantages. The objectives of the proposed artefact were then set, and the artefact was designed and developed. The output of this development endeavour is “IRDA”, the first decentralized incident reporting application (DApp), built on “Quorum”, a permissioned blockchain implementation of Ethereum. Its effectiveness was demonstrated, when six organizations accepted to use the developed artefact and performed a series of pre-defined actions, in order to confirm the platform’s intended functionality. The platform was also evaluated using Venable et al’s (2012) evaluation framework for DSR projects.

This research project contributes to knowledge in various ways. It investigates blockchain and incident reporting, two domains which have not been extensively examined and the available literature is rather limited. Furthermore, it also identifies, compares, and evaluates the conventional, reporting platforms, available, up to date. In line with previous findings (e.g Humphrey, 2017), it also confirms the lack of standard taxonomies for information security incidents. This work also contributes by creating a functional, practical artefact in the

blockchain domain, a domain where, according to Taylor et al (2019), most studies are either experimental proposals, or theoretical concepts, with limited practicality in solving real-world problems. Through the evaluation activity, and by conducting a series of non-parametric significance tests, it also suggests that IRDA can potentially increase the motivational level of users towards the reporting of incidents.

This thesis describes an original attempt in utilizing the newly emergent blockchain technology, and its inherent characteristics, for addressing those concerns which actively contribute to the business problem. To the best of the researcher's knowledge, there is currently no other solution offering similar benefits to users/organizations for incident reporting purposes. Through the accomplishment of this project's pre-set objectives, the developed artefact provides a positive answer to the research question. The artefact, featuring increased anonymity, availability, immutability and transparency levels, as well as an overall lower cost, has the potential to increase the motivational level of organizations towards the reporting of incidents, thus improving the currently dismaying statistics of incident under-reporting.

The structure of this document follows the flow of activities described in the DSRM by Peffers et al (2007), while also borrowing some elements out of the nominal structure of an empirical research process, including the literature review chapter, the description of the selected research methodology, as well as the "discussion and conclusion" chapter.

Table of contents

List of tables

List of figures

List of abbreviations

ABS	Azure Blockchain Service	**EBA**	European Banking Authority
API	Application Programming Interface	**ECSO**	EU Cyber Security Organization
ATC	Admin Test Case	**EKP**	Emergent Knowledge Processes
AWS	Amazon Web Services	**ENISA**	European Union Agency for Network and Information Security
BAAS	Blockchain as a Service	**ENS**	Ethereum Name Service
BFT	Byzantine Fault Tolerance	**EVM**	Ethereum Virtual Machine
BTC	Bitcoin token	**GBP**	Great Britain Pounds
CCCI	Cyprus Chambers of Commerce and Industry	**GDPR**	General Data Protection Regulation
CERT	Computer Emergency Response Team	**GUI**	Graphical User Interface
CIFAS	Credit Industry Fraud Avoidance System	**IBFT**	Istanbul Byzantine Fault Tolerance
CIO	Chief Information Officer	**IDS/IPS**	Intrusion Detection System/Intrusion Prevention System
CISO	Chief Information Security Officer	**IEC**	International Electrotechnical Commission
CISP	Cyber Security Information Sharing Partnership	**IFC**	International Finance Corporation
CPNI	Centre for the Protection of National Infrastructure	**IOD**	Institute of Directors
CPU	Central Processing Unit	**IODEF**	Incident Object Description Exchange Format
CRITS	Collaborative Research into Threats	**IOT**	Internet of Things
CSIRT	Computer Security Incident Response Team	**IP**	Internet Protocol
CTX	Cyber Threat XChange	**IPFS**	Inter Planeraty File System
CVRF	Common Vulnerability Reporting Framework	**ISAC**	Information Sharing and Analysis Centre
CYCSO	Cyprus Cyber Security Organization	**ISO**	International Organization for Standardization
DAPP	Decentralized Application	**ISP**	Internet Service Provider
DDOS	Distributed Denial of Service	**IT**	Information Technology
IRDA	Incident Reporting Decentralized Application	**JSON**	JavaScript Object Notation
DNS	Domain Name System	**NIST**	National Institute of Standards and Technology
DPOS	Delegated Proof of Stake algorithm	**OECD**	Organization Economic Cooperation Development
DSR	Design Science Research	**OTP**	One Time Password
DSRM	Design Science Research Methodology	**OTX**	Open Threat Exchange

PII	Personally Identifiable Information	**SCAP**	Security Content Automation Protocol
POA	Proof of Authority algorithm	**SEC**	US Securities & Exchanges Commission
POAD	Proof of Attack Detection algorithm	**SIEM**	Security Information and Event Management
POB	Proof of Burn algorithm	**SME**	Small and Medium-sized Enterprises
POC	Proof of Capacity algorithm		
POE	Proof of Existence algorithm	**SQUARE**	Systems and software Quality Requirements
POET	Proof of Elapsed Time algorithm	**SSL**	Secure Sockets Layer
POI	Proof of Importance algorithm	**STIX**	Structured Threat Information eXpression
POS	Proof of Stake algorithm	**TAXII**	Trusted Automated Exchange of Indicator Information
POV	Proof of Validation algorithm	**TEE**	Trusted Execution Environment
POW	Proof of Work algorithm	**TERENA**	Trans-European Research and Education Networking Association
PSD2	Payment Services Directive 2	**TLS**	Transport Layer Security
P2P	Peer to peer	**TTP**	Trusted Third Party
RPC	Remote Procedure Call	**UTC**	User Test Case
RPCA	Ripple Protocol consensus algorithm	**VPS**	Virtual Private Server
RSS	Rich Site Summary	**2FA**	Two Factor Authentication
SAAS	Software as a Service		

Acknowledgments

I would like to express my gratitude to my Director of studies, Dr. Ameer Al-Nemrat for his fruitful supervision and guidance.

I would also like to thank my family and friends, for their constant support throughout this challenging journey.

Alexis Michail

University of East London

January 2020

Dedication

To those dedicated to the endless pursuit of knowledge – you make the world a better place.

1. INTRODUCTION

Information and communication technologies are facing the trend of larger connectivity and increased integration, and although various security controls exist, and usually are in place, to protect against information security incidents, such incidents still occur (Line & Albrechtsen, 2016). Longer than a decade ago, Finn et al (2007, p.409) noticed “a substantial increase in information security incidents”, with a quasi-exponential increase in the total number of incidents, according to the report of the CERT Coordination Center. As technology continuously evolves and expands, so do cyber threats and incidents, with the numbers getting progressively worse. According to Gemalto’s Breach Level Index Report (2018), during only the first six months of 2018, 3.3 billion data records, around the globe, have already been exposed. According to the same source, that counts for a 72 percent increase in stolen, lost or compromised records, when compared to the same period, in 2017. In fact, Gemalto estimates that a total number of 15 billion records (at the time of writing) have been exposed since 2013, with an astounding number of 75 records being exposed per second. Cheng et al (2017) provide some examples of “major enterprise data leak incidents in recent years”, with the “Yahoo” breach in 2014 topping the list (500 million records stolen with an estimated cost of $350 million), followed by the 2013 “Adobe” breach (152 million records – cost of $714 million) and the 2013 “Target” breach (110 million records - $252 million). Other high-profile incidents involve “JPMorgan” (2014 - 252 million records), “Adult Friend Finder” (2016 - 412 million records) and even more recently the 2018 “Marriott” hotel chain breach, with an estimated number of 500 million records exposed (Armerding, 2018).

It is obvious that in the interconnected world we are now living in, organizations around the globe face millions of security threats on a constant basis. In order to adequately deal with these threats, many organizations have developed various security incident management procedures. A key element of these procedures is incident reporting – which occurs right after the initial incident identification and verification and usually happens through the utilization of an internal (within the organization) or external (relevant regulatory bodies and authorities) reporting platform, where incidents are recorded for further analysis and consequent actions. This approach, however, does not seem very appealing to organizations, for various reasons, which are presented in the next sections of this chapter.

1.1. Information Security Incident Reporting

Information Security Incident reporting can simply be described as the process of notifying either a user, an entity (e.g. an organization), a group of entities and/or an authoritative body, about a security incident which has occurred. According to NIST's Special Publication 800-61 (2012, p.69), a security incident can be described as "the violation of an explicit or implied security policy". The Publication provides some examples of such incidents, such as attempts for unauthorized access to systems or data, unplanned disruptions or denial of service attacks, unauthorized processing of data, or unauthorized changes to hardware and/or software.

As Gordon et al (2003) point out, "one desirable way of supplementing the technical solutions to security problems is for organizations to share information related to computer security breaches, as well as to unsuccessful breach attempts". This sharing of information - according again to the authors – is useful for preventing, detecting and correcting security breaches, by helping organizations from falling victims to security breaches experienced by other organizations. Such information helps organizations respond more quickly with focused remedies, should an actual breach occur (Gordon et al, 2003). Furthermore, according to ENISA (2013), the benefits of incident reporting are both well-known and widely supported, and include -among others - "information sharing, the dissemination of lessons learnt and experience exchange, identification of root causes and mitigation techniques, as well as data trend analysis".

Because of the presumed benefits of information sharing, various governments have initiated actions toward developing security-based information sharing organizations, such as the CERT Coordination Centres, INFRAGARD, Information Sharing Analysis Centres (ISAC), Secret Service Electron Crimes Task Forces, and Chief Security Officers Round Tables (Gordon et al, 2003). In a more recent paper, Gordon et al (2015) also demonstrated that information sharing can reduce the uncertainty associated with cybersecurity investments in private sector firms, and thus reduce the tendency to underinvest in cybersecurity activities.

In today's digital societies, responding to security incidents is becoming increasingly imperative in business, while the effects of a breach can be very destructive to an organization (Grispos et al, 2017). Line & Albrechtsen (2016) state, that Information security management is a relatively young field of both practice and research, and that an efficient incident management process – the ability to appropriately prepare for, and respond to, information security incidents – is important to maintain the functioning of systems. In fact, the European Commission (through efforts initiated by the European Union Agency for Network and Information Security - ENISA) considers incident reporting so important, that it has regulated and mandated incident reporting for various segments within Europe, such as the telecom sector (through the "Art. 13a Telecom Framework Directive"), the trust service providers (through the "Art. 19 eIDAS regulation") and the digital service providers (through the "Art. 16-4 NIS Directive"). It is therefore now illegal for these entities not to report information security incidents to the specified authorities.

According to the International Standard 27035 (ISO/IEC 27035:2011, 2011), an information security incident management process has 4 major phases: prior preparations, response to an incident, post-incident evaluations and improvements. The Standard also denotes that organizations can benefit from having an adequate incident management process, by reducing the number of incidents, improving the focus and prioritization of security activities, and improving their risk assessment efforts and overall information security level. The incident reporting procedure falls within the 2nd phase, as part of the incident response phase. Gonzalez (2005) views information security reporting as a quality improvement process that is essential to reduce incidents.

According to Sveen et al (2007), information security incidents arise from many sources, such as software, hardware and configuration errors, or inadequate physical security which allows external attackers and/or malicious insiders to attack the system. They go on mentioning that the reporting of incidents allows them to be investigated and learned from, and that this knowledge can be used to avoid such incidents in the future, by putting into place adequate technical and organizational countermeasures. Furthermore, according to NIST's Incident Handling Guide 800-61 (2012), information sharing is the most important aspect of incident response coordination, where different organizations share threat, attack, and vulnerability information with each other, so that each organization's knowledge can benefit the other. This is both necessary and mutually beneficial, since the same threats and attacks often affect a multitude of organizations.

1.2. Existing incident reporting platforms

The various, available, reporting tools, focus on incidents which are caused by faults, failures or malicious activity. Reporting platforms are being used in a variety of business domains; examples include platforms being utilized as IT helpdesk/bug tracking platforms by IT departments, as security reporting platforms by information security departments, as well as safety reporting platforms, by health and safety organizational departments. In many cases safety and security are interrelated, and there are similarities between safety and information security reporting systems, as both attempt to reduce risk by learning from incidents (Sveen et al. 2007). A search for identifying incident management/reporting platforms/software available through the web, indicates a far greater amount of available platforms/software related to reporting/managing "safety" incidents (i.e. workplace health & safety, personnel injuries, facilities maintenance disruption and generally incidents directly related to occupational health and safety agencies) rather than "security" incidents (i.e. information security incidents). It therefore comes as no surprise that both Schneier (2011) and Reed-Mohn (2007), when comparing current practices in information security reporting systems against those in the healthcare, aviation, and rail industries, concluded that the quality of practices in information security reporting systems "did not match those of their safety-critical equivalents".

Furthermore, Gonzalez (2005) examined the successful implementation of incident reporting programs in another sector – aviation - and then suggested an equivalent model for information security purposes.

Information Security incident reporting has traditionally occurred through ad hoc methods, such as email, instant messaging clients, and phone (NIST, 2012). According to the same source, this type of reporting usually relies on an individual's connections with employees in incident response teams of partner organizations, and tends to be largely unstandardized, in terms of what information is communicated and how that communication occurs. As an alternative, various reporting platforms (software) have been created and became available in the market, which can be used for reporting internally within an organization, or externally, with third parties. They are utilized by national CERTs and various Information Sharing and Analysis Centres (ISAC) globally, in both the public and private sector. As an example, the UK National CERT uses the "Threatvine information sharing platform", which is designed to "enable communication between CERTs and their Competent Authorities and to ensure resilience across the wider supply chain" (Threatvine, 2018, p.2). Other examples of platforms with similar functionality include "LogicManager's Security Incident Management Software", "D3 Security incident response platform", "Resolver's Incident Management Software", "Omnigo's Incident Reporting Software" and "OTRS' STORM software".

At this point, it is important to differentiate between "incident reporting" and "incident responding" software/platforms. Incident "responding" platforms are different from incident "reporting" platforms and are out of scope of this research project. These platforms usually utilize some sort of automated incident response software, with automatic correlation of events and alerts (from across the organizational environment) and automatic or semi-automatic triaging, investigation and remedy actions. Incident "reporting" platforms, on the other hand, are utilized purely for reporting purposes and are thus within scope.

The core functionality of reporting platforms is essentially the same; participants gain access to a centralized platform (database), where they can view and/or report incidents. Information is stored in a centralized database (of the platform provider), while the service can be installed either on premise or on a public/private cloud. Access to the reporting platforms is usually possible through a web-based interface, while a few platforms have also developed separate versions for mobile clients. The typical user interface contains a homepage (with latest incidents, alerts and news) a support page and/or forum (some platforms also utilize a community forum and/or chat functionality for participant conversations), a profile page for each member (some platforms even offer social-networking-like functionality, such as following a member, updating your status, adding skills to your profile and giving out endorsements), and the ability to create and share an incident report, and/or search/browse through the already submitted incidents. Submitted incidents are usually ranked (by users who submit them) according to their severity (or risk ranking) and their visibility (some members may restrict access to other members of the same platform – e.g. a user may submit an incident only visible within his/her own company or only visible to sector-specific institutions which are members of the platform). When creating a new report, users can input various details of the incident, such as its category (e.g. phishing, Denial of Service attack, malware etc.), the incident's details (e.g. date, scope, duration, affected systems, modus operandi and various other technical details) and also upload attachments, such as text files, videos or photographs.

1.3. The causal problem and research question

Although most organizations are compelled by various regulations to report security incidents to relevant bodies and authorities (e.g. PSD2 for payment incidents in financial institutions, NIS directive for reporting incidents from critical infrastructure providers, GDPR reporting for personal data breaches, companies reporting a cyber-crime to the Police and so on) it seems that only a small percentage of incidents are actually being reported. According to the IOD & Barclays Policy report (2016), only 28% of cyberattacks against businesses were reported to the police. The Internet Security Threat Report by Symantec (2016) mentions that the number of companies that refused to report the scope of a data breach jumped by 85 percent, compared to the previous year. The US Securities & Exchanges Commission reported that nearly 65% of affected public companies did not report cybersecurity breaches to the SEC, between 2011-2017 (Newman, 2018). Since 2011, when the S.E.C. issued its initial cyber guidance, only 106 companies have reported incidents to the S.E.C. But during that same period, there were 4,732 cyberattacks on American businesses, researchers for the Privacy Rights Clearinghouse found (Newman, 2018).

It is evident that organizations hesitate to report security incidents. This happens for a variety of reasons, as companies may not want to reveal the damage they've suffered, due to concerns about possibly scaring off potential or existing customers (reputation), damaging their stock value, incurring potential legal liabilities, because of the lack of knowledge or internal policies to properly recognize or deal with attacks, or even because employees intentionally conceal information security incidents (Kaspersky, 2017). PAC & Telefonica (2015, p.17), conducted a survey among 200 decision-makers in large organizations in the UK, France and Germany, and found out - among other – that the top organizational issue, when responding to cyber incidents, was dealing with "customer concerns (in response to media attention given to high-profile breaches)", as "protecting a firm's reputation and brand image with its customers is of primary importance, since it directly affects sales, as well as competitors' positioning". Another survey from SentinelOne (2016) revealed that although 48% of the 500 organizations surveyed, worldwide, had suffered a ransomware attack (during the past year), just 54% of respondents had reported the incident(s) to law enforcement, and just 48% notified authorities and/or regulators.

Recent history justifies the concerns over reporting. Nesmith (2018) lists some infamous top-level resignations caused by major data breaches, such as the Target's CEO in 2014, Ashley Madison's CEO in 2015, Sony Picture's co-chairman, again, in 2015 and Equifax' CEO in 2017. In short, it seems there is a tendency to point the finger at the CEO after a data breach, something which does not encourage CEOs to report cyber security incidents. As Nesmith (2018) points out, with data breaches making the news on a nearly daily basis, the days of CEOs not sharing the blame are gone - it is no longer the case that the CIO or CISO of the company is solely the one to blame. The report mentioned earlier by Ipsos MORI and the University of Portsmouth (2017), stated a few other reasons for not reporting incidents, such as businesses stating that “breach was not significant enough” (58%), that businesses were not aware of whom to report it to (16%) and that reporting “won’t make a difference” (10%).

Gonzalez (2005) notes that there should be little doubt about the need to improve reporting of cyber security data/incidents, followed by analysis and sharing of insights, and that the numerous Computer Emergency Teams (CERTs) and Computer Security Incident Response Teams (CSIRTs) around the world have established various cyber security reporting systems. However, he goes on, nearly two decades after their emergence, the CERT Coordination Centre, acknowledges that systematically collected data on cyber-attacks is not generally available. He states that this often happens because of fear for bad publicity. Even when detailed data are eventually shared, in most cases restricted use agreements hamper their availability to the research community. Even between state-members of the EU, there is still little exchange of information about breaches, between different national authorities (ENISA, 2012).

It is obvious that organizations fear about their reputation, when confronted with a security incident. Apart from reputational concerns, traditional reporting platforms and centralized databases also suffer from other concerns, such as non-constant availability (100% availability is never guaranteed) and the fact that anybody with sufficient access to the platform (e.g. a malicious system administrator) can destroy/corrupt or alter the data within. Furthermore, because of their centralized nature and sensitive content, reporting platforms require major investments for ensuring their security (both physical and electronic). This cost, along with all other costs associated with a centralized database (i.e. need for increasing storage space, disaster recovery/business continuity arrangements and other) is, of course, ultimately passed on to the platform’s subscribers.

1.3.1. Research question

The identified limitations of the traditional information security incident reporting platforms, lead to the forming of the following research question:

Is there a way to create an innovative information security incident reporting solution, which will utilize the positive features offered by existing solutions, but will also provide added value to users, in order to increase their level of motivation towards the reporting of incidents?

1.3.2. Research purpose and scope

The purpose and scope of this research can be summarized as follows:

- Identify and evaluate existing information security incident reporting schemes and solutions
- Evaluate the use of blockchain technology as a resolution towards the inherent problems of existing reporting solutions
- Design, develop and evaluate an incident reporting solution, which provides added value to users, and increases their level of motivation towards the reporting of information security incidents.

1.4. Research motivation

This research was driven by various motivations:

- **The very low numbers of information security incident reporting**

Although virtually everyone agrees that information security incident reporting is beneficial to organizations (NIST, 2012; Gordon et al, 2003; ENISA, 2013; Gordon et al, 2015; Line & Albrechtsen, 2016; Gonzalez, 2005), reporting statistics show that very few incidents are actually being reported (IOD & Barclays Policy report, 2016; Symantec, 2016; Newman, 2018; Ipsos MORI, 2017; SentinelOne, 2016; ENISA, 2012). Proposing a solution for potentially increasing the reporting numbers is a serious motivation for this research.

- **The excitement of potentially utilizing a new/recent technology for producing a solution**

Blockchain technology has the potential of impacting all sectors and layers of society, in a multitude of combined ways – it is disrupting society by enabling new kinds of disintermediated digital platforms, while also improving efficiency over existing structures, by removing the need for actively intermediated data-synchronization and concurrency control (Mattila, 2016). Both developers and researchers have become aware of the capabilities of this new technology and are exploring various applications across a vast array of sectors (Christidis and Devetsikiotis, 2016). Zhao et al (2016) argue that blockchain technology is becoming “increasingly relevant”, while a recent global business survey from IBM (2017) indicated that 33% of C-suite executives surveyed, were considering, or have already been actively engaged with blockchain (IBM, 2017).

Utilizing, therefore, blockchain technology in order to propose a potential solution to the existing problem of security incidents under-reporting, is an exciting prospect.

- **The preliminary interest of the CYCSO for utilizing such as solution**

It is evident that developing such a solution would be useless, if organizations themselves would not be interested in utilizing it. Therefore, and in order to identify potential (preliminary) interest for such a solution from the industry, a request was sent to the Cyprus Cyber Security Organization (CYCSO – not for profit), which operates under the auspices of the Cyprus Chambers of Commerce and Industry (CCCI Cyprus) and has access to thousands of Cypriot enterprises. According to its website, CyCSO is “a private initiative led by the Cyprus Chamber of Commerce and Industry (CCCI) and the participation of the Cyprus Institute of Neuroscience and Technology”. Its aims are to create a “cyber-security ecosystem to be linked to the European ecosystem, in coordination with the European Cyber Security Organisation – ECSO”, as well as to develop “an innovative and dynamic cyber-security industry in Cyprus” (CYCSO, 2018).

CYCSO agreed in including three questions for this matter, as part of a wide-ranging cyber-readiness survey they were planning to send-out to their organizational members. They also expressed their initial interest in using the platform – once it was ready – for establishing an Information Sharing and Analysis Centre (ISAC) for their organizational members. This survey acted as a pilot study, in order to identify signs of any preliminary interest from the Cypriot businesses, in utilizing such a solution.

The three questions (the survey was comprised of 16 questions, in total) included in the survey were the following:

Q14) *How would you describe your incident-response capability (i.e. to take a structured approach in handling a security-related incident, once such has occurred)?*

*“**Q15)** Recently, a National Computer Security Incident Response Team (CSIRT) has been created in Cyprus, to cater, however, only for the needs of Critical Infrastructure owners/administrators, banks and ISPs. The CSIRT’s aim*

is to ensure the existence of (at least) a minimum level of security, by implementing proactive and reactive security services to reduce the risks of network information and cyber security incidents, as well as respond to such incidents as and when they occur. Do you think a similar initiative, which would cover the needs of the whole private sector would be beneficial for your organization?"

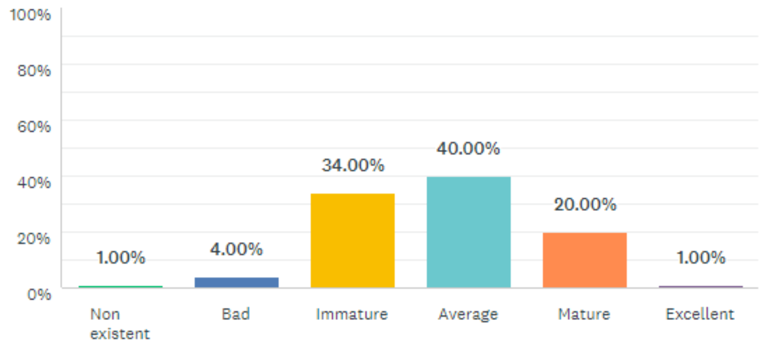
*"**Q16)** Although a great variety organization are compelled by various regulations to report security incidents to relevant bodies and authorities (i.e. PSD2 for payment incidents in financial institutions, NIS directive for reporting incidents from critical infrastructure providers, reporting because of GDPR, companies reporting a cyber-crime to the Police and so on) a very small percentage of incidents are actually reported. This happens for a variety of reasons, as companies may not want to reveal the damage they've suffered due to concerns about possibly scaring off potential or existing customers (reputation), damaging their stock value, or incurring potential legal liabilities or even because companies do not have the knowledge or internal policies to properly recognize or deal with attacks. Would you be interested in a solution which would allow your organization to submit security incidents in an anonymous fashion enabling both governmental, regulatory & supervisory authorities as well as individual organizations to have a greater picture of the attack landscape based on historical and current trends? Such a solution will not rely on a central managing authority (thus no dependence on a single platform/database/authority and no legal liabilities applicable whatsoever) and would enable your organization to have a clear, real-time view of the security incidents happening in organizations throughout Cyprus."*

The (electronic) survey was sent to about 10,000 Cypriot businesses, of all sectors and sizes, on the 27th of August of 2018, with no set deadline for responses. Due to the fact that no response deadline had been set and because of the time-limitations of this research, CYCSO was specifically asked to provide the results of the first one hundred (100) respondents. The results -

as these were provided by CYCSO - on the 15th of February of 2019 - were the following:

How would you describe your incident-response capability (i.e. to take a structured approach in handling a security-related incident, once such has occurred)?

Answered: 100

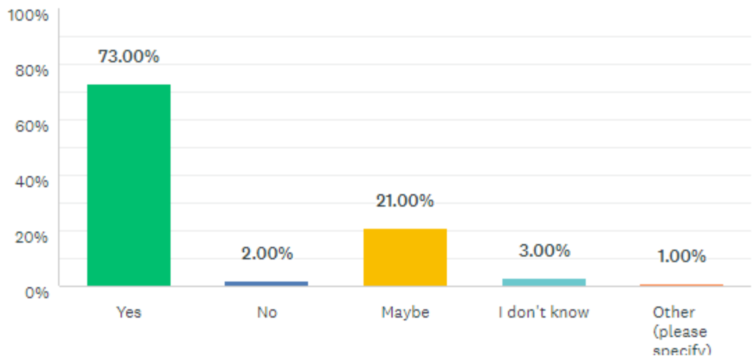


ANSWER CHOICES	RESPONSES	
Non existent	1.00%	1
Bad	4.00%	4
Immature	34.00%	34
Average	40.00%	40
Mature	20.00%	20
Excellent	1.00%	1
TOTAL		100

Figure 1.1. Q14 of CYCSO Cyber-readiness survey

Recently, a National Computer Security Incident Response Team (CSIRT) has been created in Cyprus, to cater, however, only for the needs of Critical Infrastructure owners/administrators, banks and ISPs. The CSIRT's aim is to ensure the existence of (at least) a minimum level of security, by implementing proactive and reactive security services to reduce the risks of network information and cyber security incidents, as well as respond to such incidents as and when they occur. Do you think a similar initiative, which would enable anonymized information sharing for cyber incidents and cover the needs of the whole private sector, no matter the size or segment, would be beneficial for your organization?

Answered: 100

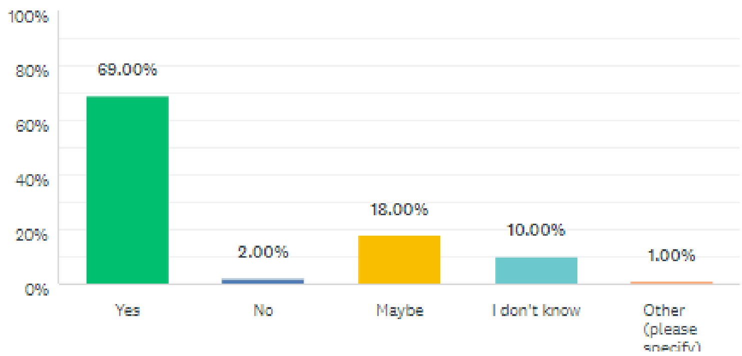


ANSWER CHOICES		RESPONSES	
Yes		73.00%	73
No		2.00%	2
Maybe		21.00%	21
I don't know		3.00%	3
Other (please specify)	Responses	1.00%	1
TOTAL			100

Figure 1.2. Q15 of CYCSO Cyber-readiness survey

Although a great variety organization are compelled by various regulations to report security incidents to relevant bodies and authorities (i.e. PSD2 for payment incidents in financial institutions, NIS directive for reporting incidents from critical infrastructure providers, reporting because of GDPR, companies reporting a cyber-crime to the Police and so on) a very small percentage of incidents are actually reported. This happens for a variety of reasons, as companies may not want to reveal the damage they've suffered due to concerns about possibly scaring off potential or existing customers (reputation), damaging their stock value, or incurring potential legal liabilities or even because companies do not have the knowledge or internal policies to properly recognize or deal with attacks. Would you be interested in a solution which would allow your organization to submit security incidents in an anonymous fashion enabling both governmental, regulatory & supervisory authorities as well as individual organizations to have a greater picture of the attack landscape based on historical and current trends? Such a solution will not rely on a central managing authority (thus no dependence on a single platform/database/authority and no legal liabilities applicable whatsoever) but facilitated/coordinated by the industry - and would enable your organization to have a clear, real-time view of the security incidents happening in organizations throughout Cyprus and beyond.

Answered: 100



ANSWER CHOICES		RESPONSES	
Yes		69.00%	69
No		2.00%	2
Maybe		18.00%	18
I don't know		10.00%	10
Other (please specify)	Responses	1.00%	1
TOTAL			100

Figure 1.3. Q16 of CYCSO Cyber-readiness survey

These preliminary results unveil some interesting assumptions about the Cypriot business entities. Just 21% of the participants rated their incident-response capability (Q14) as “mature” (20%) or “excellent” (1%), with the vast majority (74%) of the respondents describing their capability as either “average” (40%) or “immature” (34%).

However, when it comes to the question of whether the private sector would benefit from an incident-response scheme, similar to the one employed by the national CSIRT (Q15), the vast majority of respondents expressed a positive opinion (73%), while 24% of the respondents were skeptical about such an initiative (21% answered “maybe” and 3% answered “I don’t know”) and just 2% expressed a negative opinion.

Furthermore, when the issue of security incidents under-reporting was brought up and the potential of having an anonymous-reporting solution, with no reliance on a central authority (Q16), the vast majority were - again - positive (69%), with a further 18% of the participants answering “Maybe”, a 10% answering “I don’t know” and just a 2% expressing a definite negative opinion.

Although the aforementioned descriptive statistics cannot in any way lead to the forming of definite conclusions, they do provide an indication of a preliminary interest from the Cypriot businesses in utilizing such a solution. In addition, the fact that the Cyprus Cyber Security Organization has expressed its interest (as an authority) in using this decentralized platform for establishing an ISAC for its members, can also be considered as a strong motivation.

- **Academic requirement**

This research counts towards the partial fulfilment of the requirements of the University of East London, for the degree of “Professional Doctorate in Information Security”.

1.5. Research approach and structure

This research follows a methodology/framework developed by Peffers et al (2007), named “Design Science Research Methodology (DSRM)”. The framework includes 6 “activities”, which were undertaken sequentially:

Activity 1 → Problem identification and motivation

Activity 2 → Define the objectives for a solution

Activity 3 → Design and development

Activity 4 → Demonstration

Activity 5 → Evaluation

Activity 6 → Communication

The following table provides an overview of the activities undertaken as part of this research, along with a short description and concurrent association with this research's chapters:

Activity no.	Title	Description	Pertinence	Relevant Chapters
1	Problem identification and motivation	Define the specific research problem and justify the value of a solution	During this activity the problem of the traditional security reporting platforms were analysed, and the value of the decentralized solution was discussed.	1,2
2	Define the objectives for a solution	Infer the objectives of a solution from the problem definition and knowledge of what is possible and feasible	During this activity the objectives of the decentralized solution were inferred rationally from the problem specification.	4
3	Design and development	Create the artefact. Such artefacts are potentially constructs, models, methods, or instantiations or new properties of technical, social,	During this activity the reporting platform's desired functionality and architecture were determined and the platform was developed.	5

		and/or informational resources		
4	Demonstration	Demonstrate the use of the artifact to solve one or more instances of the problem	During this activity a proof of concept of the reporting platform was executed with the participation of a small number of organizations.	6
5	Evaluation	Observe and measure how well the artifact supports a solution to the problem	During this activity the actual observed results from the use of the reporting platform were compared to the initial objectives of the proposed solution. In addition, the platform was evaluated by external parties.	7
6	Communication	Communicate the problem and its importance, the artefact, its utility and novelty, the rigor of its design, and its effectiveness to researchers and other relevant audiences, such as practicing professionals, when appropriate	During this activity the aforementioned activities were documented. The structure of the document followed the flow of the DSRM activities and also utilized some elements out of the nominal structure of an empirical research process, including the “literature review”, “research methodology” and “discussion and conclusion” chapters.	1-8

Table 1.1. Research approach and structure

1.6. Related work

The comprehensive literature review (chapter two) identified a limited amount of literature directly relevant to information security incident reporting and blockchain. Graf & King (2018), used a Blockchain smart contract technique to provide an automated trusted system for incident management workflow, that allows automatic acquisition, classification and enrichment of incident data. Their work, however, is focused on developing a solution that could replace human input, by facilitating automatic cyber incident classification, in order to enable analysts to focus on other tasks. Other examples include Blockchain-based Security Information and Event Management (SIEM) systems - for storing and accessing information security events - utilized by multiple devices, within the broader concept of the Internet of Things (Mesa et al, 2019; Miloslavskaya & Tolstoy, 2019) as well as a blockchain-based risk and information system control framework, able to register risk registration data on the ledger, thus ensuring traceability and irreversibility of entries (Ma et al, 2018).

The most directly relevant work regarding incident reporting and blockchain was published by Adebayo et al (2019), who propose a theoretical framework for public information sharing, based on blockchain. This framework describes an open blockchain implementation, with no central authority, where any security-conscious organization could join as a member, and could also include various security vendors (e.g. antivirus companies) which, in-turn, could offer applicable solutions (e.g. patches) to participating organizations, via a cloud configuration, also accessible via the blockchain. As stated above, their work produced a high-level, theoretical framework and not an actual instantiation.

1.7. Contribution to knowledge

Information security incident under-reporting is unambiguously a business problem, as identified by a variety of sources, such as ENISA (2012), Symantec (2016), Newman (2018) and more. This research project identified the underlying issues that cause the problem of incident under-reporting. These issues include organizations not reporting incidents due to fears related to competition, the low chance of prosecution, reputational concerns, the increased cost related to reporting processes, possible financial penalties and reprimands, the low level of organizational IS maturity, as well as burdensome regulatory compliance procedures (Koivunen, 2010; Ahmad et al, 2015; Ruefle et al, 2014; Choo, 2011; Ahmad et al, 2012, Johnson, 2002; Metzger et al, 2011; Jaatun et al, 2009; Etzioni, 2014; Humphrey, 2017; Housen-Couriel, 2018). A solution is proposed, in the form of an innovative artefact, which confronts a number of these issues, and more specifically issues related to reputational concerns and the increased cost of reporting, by embedding specific features in the developed artefact, such as reporting anonymity, within a low-cost reporting ecosystem. The developed artefact is the first application utilizing a private blockchain for the manual reporting of incidents, through a web-accessible reporting platform. To the best of the researcher’s knowledge, there is currently no other solution offering similar benefits to users/organizations for incident reporting purposes.

In summary, this research project makes the following contributions:

- adds to the literature of two fields which have not been extensively studied (incident reporting and blockchain).
- identifies, compares, and evaluates the existing reporting schemes and solutions, with an emphasis in manual reporting platforms.
- identifies the lack of standard taxonomies for information security incidents, in line with previous findings.
- identifies the blockchain applications currently available in the areas of information security, data management and incident reporting.
- describes the process of designing, developing, and evaluating a functional, practical artefact in the blockchain domain, a domain where most studies are either experimental proposals, or theoretical concepts, with limited practicality in solving real-world problems (Taylor et al, 2019).
- indicates that the developed solution can potentially increase the motivational level of users towards reporting incidents, through a series of non-parametric significance tests.

1.8. Thesis outline

The following table provides a short summary of this research's chapters:

Chapter	Title	Summary
1	Introduction	This chapter introduces the concept of information security incident reporting, identifies the causal problem and sets out the research question. It also describes the research motivation, approach and structure while also pointing out the previously related work and the contribution to knowledge.
2	Background, literature review & reporting means evaluation	This chapter provides a comprehensive synopsis of literature and background information related to information security incident reporting and the various aspects of the blockchain technology. It also includes an evaluation of the existing incident reporting means.
3	Research methodology	This chapter includes a description of the available types of research, the research philosophy and paradigms, a description of the relevant research methods which could have been undertaken to complete this project, as well as the specific methodology (Design Science Research – DSR) which was eventually selected, along with the reasons behind this selection. The chapter also includes a synopsis of the six activities undertaken during this project (following the DSRM process model by Peffers et al (2007)), and also discusses research ethics and other research considerations.
4	The decentralized solution: Objectives	This chapter sets out the objectives of the proposed solution, which were directly derived from the research question.

5	The decentralized solution: Design and development	This chapter describes the various design and development activities that led to the creation of the artefact - a private, incident reporting platform, based on the Ethereum blockchain technology.
6	The decentralized solution: Demonstration	This chapter describes the verification and validation activities performed in order to demonstrate how the produced artefact provides a solution to the defined problem.
7	The decentralized solution: Evaluation	This chapter describes the activities performed to evaluate the developed artefact, by utilizing a DSR evaluation framework proposed by Venable et al (2012).
8	Discussion and conclusion	This chapter provides a summary of the research, identifying contribution to knowledge, along with various limitations of the proposed solution and suggestions for future research.

Table 1.2. Thesis outline

2. BACKGROUND, LITERATURE REVIEW & REPORTING MEANS EVALUATION

2.1. Introduction

This chapter presents a comprehensive overview of the background literature necessary, to examine both information security incident reporting, as a holistic process, as well as blockchain technology, in the light of a new technology, potentially capable of disrupting traditional business models (Friedlmaier et al, 2018). The two, aforementioned, domains, constitute the major themes of this research topic and the literature map in figure “2.1” portrays the structure upon which the literature analysis and evaluation are based. According to Creswell (2014), one of the first tasks of a researcher occupied with a new topic, is to organize the literature; a literature map can significantly aid this process, by breaking down the research topic into major literature topics and sub-topics, thus providing a visual summary of the available literature. According to the same author, literature maps can be organized in various ways- there is no “right” or “wrong” arrangement. Figure “2.1.” displays a hierarchical break-down of the two major topics (“Information security incident reporting” and “Blockchain technology”) unfolding into their various sub-topics:

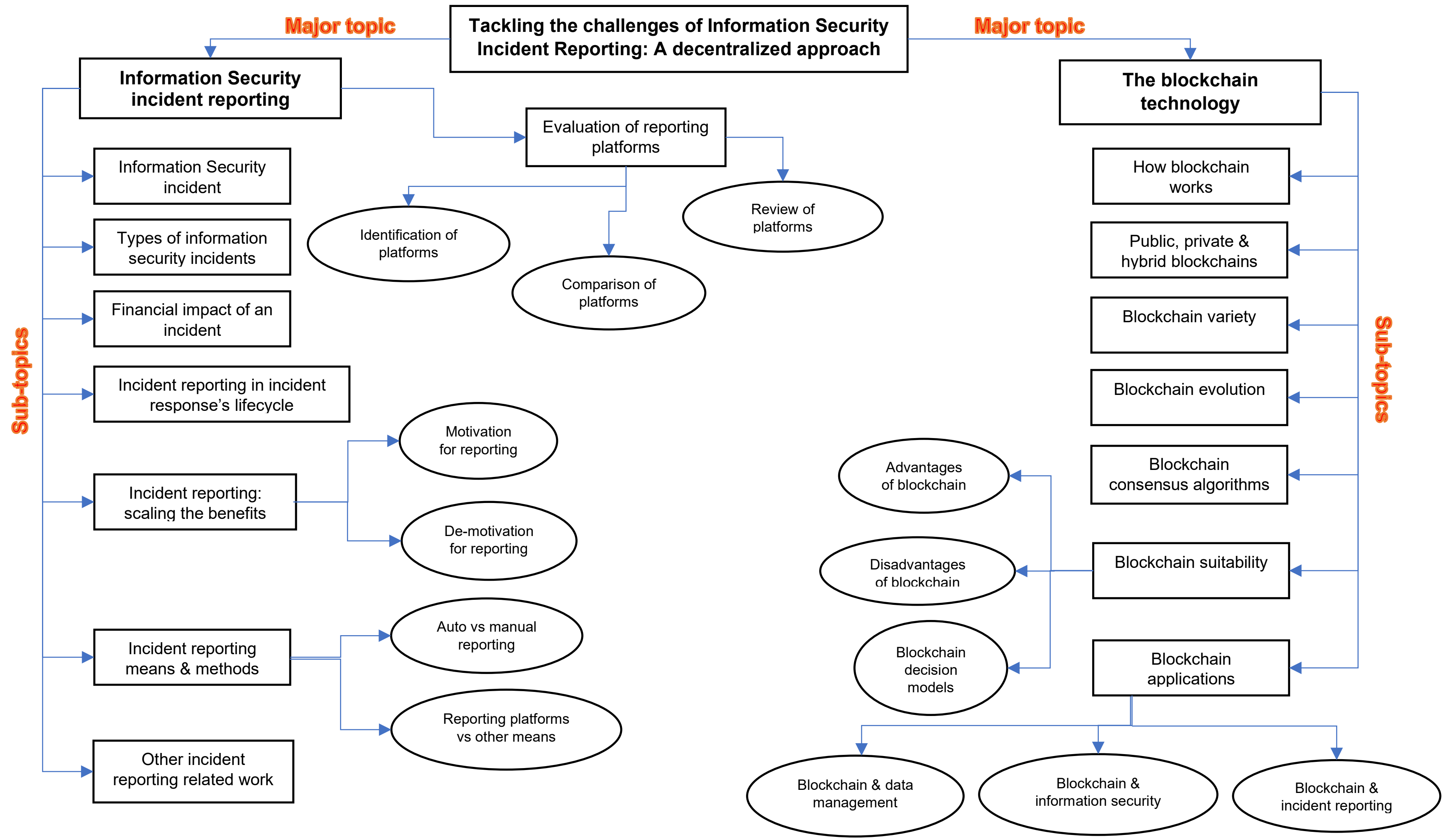


Figure 2.1. Literature map

2.2. Information Security incident reporting

Information has become the critical asset in the operation and management of virtually all modern organizations (Pipkin, 2000) and is regarded as the resource with the highest organizational value (Usmani et al, 2013). Information aids organizations to increase their operational efficiency, process automation and decision quality. It also helps in reducing response times, minimize costs and/or maximize profit (Denning, 1999; Finne, 2000; Abrahams et al, 1995). According to Sorrels et al (2008), information incidents can occur for a variety of reasons, including external attacks, malicious insiders, natural disaster, accidents, and/or equipment failure. Should an incident occur, Grimaila et al (2008), consider necessary to notify all parties whose mission is critically dependent upon the impacted information resource – and in a timely manner – in order for them to take appropriate contingency measures. As previously mentioned, Information Security Incident reporting can be described as the procedure of notifying/sharing/reporting to either a user, an entity (e.g. an organization), a group of entities/users, as well as an authoritative body about a security incident which has occurred. Incident reporting can be viewed as a quality improvement process for organizations, essential to reduce incidents (Gonzalez, 2005). The benefits of this procedure vary: sharing such information can aid organizations respond more quickly with focused remedies (Gordon et al, 2003), it can aid prevention, detection and correction of potential security breaches (Gordon et al, 2003, Sveen et al, 2007), it can enhance the identification of root causes and mitigation techniques, it can provide statistics for data trend analysis (ENISA, 2013), while it can also reduce the uncertainty associated with cybersecurity investments (Gordon et al, 2015). NIST (2012, p.45) considers incident reporting as the “most important aspect of incident response coordination”.

2.2.1. Information Security Incident

Many definitions attempt to clarify the meaning of an “information security incident”. Probably the simplest of them, is the one given by Condon & Cukier (2008, p.72), who describe a security incident as “an event that has been verified as attributable to a security failure - as opposed to a hardware failure or misinterpretation of data”. Spruit & Gerhardt (1997) describe incidents as an unintended disruption or complication which results in the disability, discontinuance or cost to an organization. Even earlier, in 1991, Wack defined a computer security incident as “any adverse event whereby some aspect of computer security could be threatened; loss of data confidentiality, disruption of data or system integrity, or disruption or denial of availability” (Wack, 1991, p.1). A security incident can also be described as “a violation (or imminent threat of violation) of computer security policies, acceptable use policies, or standard security practices” (NIST, 2012; Hansman & Hunt, 2005). Examples of such incidents are attempts for unauthorized access to systems or data, unplanned disruptions or denial of service attacks, unauthorized processing of data, or unauthorized changes to hardware and/or software (NIST, 2012; ISO/IEC 27001, 2013; ISO/IEC 27035, 2016).

A more formal definition was formed by Stephenson (2004, p.4), who described an incident as ‘‘a change of state in a bounded information system from the desired state to an undesired state, where the state change is caused by the application of a stimulus external to the system’’. In other words, an incident can be described as an event that overcomes any preventative controls of an organization and inflicts negative changes on its information systems (Baskerville et al, 2014).

At this point, it is critical to explicate the difference between an information security “event” and an “incident”. According to ISO/IEC 27035 (2016, p2), an information security event is “an occurrence indicating a possible breach of information security or failure of controls”, whereas an information security incident “is one or multiple related and identified information security events that meet established criteria and can harm an organization’s assets or compromise its operations.” Therefore, a security event does not necessarily transform into an incident. Instead, a set of pre-established criteria dictate whether an event can be classified as an incident.

Information Security incidents can be deliberate or accidental (e.g. caused by a human error or a natural phenomenon) and can be triggered by both technical and physical means (Kostina et al, 2009). Spruit (1998) points out that security incidents often occur due to a concurrence of circumstances and explains that individuals may sometimes take decisions or perform actions that initially seem correct, but eventually lead to security breaches. According to SANS' Incident Handler's Handbook (2011, p.2), "an incident is a matter of when, not if, a compromise or violation of an organization's security will occur".

Back in 1998, Howard & Longstaff, in their effort to develop a common language for security incidents, presented the following diagram, in order to explain an incident's taxonomy:

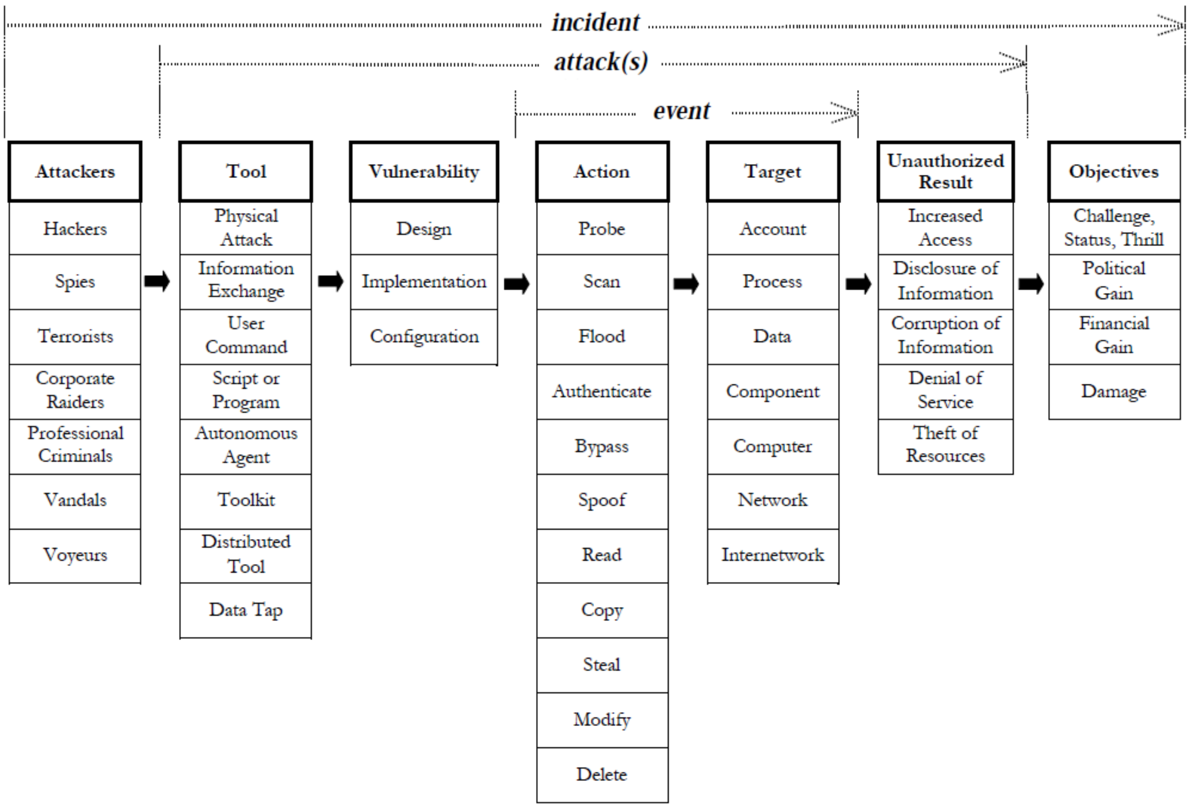


Figure 2.2. Computer & network incident taxonomy (Howard & Longstaff, 1998)

The diagram, even though developed as early as in 1998, depicts a practically valid impression of an incident's scope: from the various types of the attackers, to their tools and objectives, as well as their modus operandi. Nevertheless, some categories have since been enriched, such as the various types of attackers: organizations now face newer threats such as the “insider threat” – a very dangerous security threat posed by internal entities of an organization, such as a former or a disgruntled employee (Ambre & Shekokar, 2015), or attacks caused by “hacktivism” - a digital form of activism that often employs hacking skills and tools in order to attack governmental institutions and private organizations grounded on a particular idea/belief (Brekine et al, 2019). The attackers’ tools and methods have also evolved throughout the years, in addition to the various types of new vulnerabilities and potential targets introduced in this new era of “social media”, artificial Intelligence”, “machine learning”, “big data”, “Internet of Things” and “Blockchain”.

2.2.2. Types of information security incidents

Historically, researchers have been struggling to reach consensus upon a universally accepted list/categories of information security incidents. Icove et al (1995) and Cohen (1995 & 1997) presented incident taxonomies - in attempts to classify incidents - with terms such as: “Denial of Service attacks”, “Dumpster diving”, “Viruses & worms”, “Excess privilege” and many more appearing in their classification schemes. However, as stated by Cohen (1995), “a complete list of the things that can go wrong with information systems is impossible to create” and since “there is potentially an infinite number of different problems that can be encountered, any list can only serve a limited purpose”. Howard & Longstaff (1998) also stated, that even assuming that an exhaustive list could indeed be developed, the taxonomy would be unmanageably long and difficult to apply. According to the same authors, it is also not uncommon to find disagreements in many different definitions of security incidents (Howard & Longstaff, 1998). Even for popular terms such as “computer virus”, for example, although most agree upon the general notion, there is no universally accepted definition (Amoroso, 1994). Papers by Neumann & Parker (1989), Cheswick & Bellovin (1994), Landwehr et al (1994), Cohen (1995 & 1997), Lindqvist & Johnson (1997), Howard & Longstaff (1998), and more recent work by Kiltz et al (2007), Zhu et al (2011) and Kacha (2014), are some examples of proposed taxonomies regarding information security incidents.

The complexity of setting up (and maintaining) a universally accepted list/categories of information security incidents has not only been disconcerting individual researchers, but organizations and authoritative bodies, as well. The amount of different information security incident terms, the amount of different taxonomies and the number of different versions of the same taxonomies, have all been reported as complications by ENISA (2018), the European Union Agency for Network and Information Security. ENISA aims to create a taxonomy that can ensure that all European CSIRTS “speak the same language”, something which would further facilitate sharing across CSIRTs, as well as enable the harmonization of statistics between the CSIRT community (Mattioli & Leguesse, 2018). ENISA and the European Computer Security Incident Response Team are currently at the stage of developing such a taxonomy (ENISA, 2018), which is going to be based on the latest “eCSIRT.net mkVI” taxonomy, created by Stikvoort (2015). The eCSIRT.net mkVI” taxonomy is currently in use by many European CSIRTs (ENISA, 2018) and is using the following classification scheme for security incidents:

Incident *Classification*	**Incident *Examples***	**Description / *Explanation***
Abusive Content	Spam	or "Unsolicited Bulk Email", this means that the recipient has not granted verifiable permission for the message to be sent and that the message is sent as part of a larger collection of messages, all having a *functionally comparable* content.
	Harmful Speech [1]	Discreditation or discrimination of somebody (e.g. cyber stalking, *racism and threats against one or more individuals*)
	Child/Sexual/ Violence/...	Child Pornography, glorification of violence, ...
Malicious Code [2]	Virus	Software that is intentionally included or inserted in a system for a harmful purpose. A user interaction is normally necessary to activate the code.
	Worm	
	Trojan	
	Spyware	
	Dialer	
	Rootkit	
Information Gathering	Scanning	Attacks that send requests to a system to discover weak points. This includes also some kind of testing processes to gather information about hosts, services and accounts. Examples: fingerd, DNS querying, ICMP, SMTP (EXPN, RCPT, ...), *port scanning*.
	Sniffing	Observing and recording of network traffic (wiretapping).
	Social Engineering	Gathering information from a human being in a non-technical way (e.g. lies, tricks, bribes, or threats).

[1] *Was “harassment” – legally the term “harmful speech” is more correct, as it includes harassment, discrimination and defamation*
[2] *“Malicious code” refers to malicious software inserted into a system. The vector that caused the insertion is not apparent here. The vector can be an “intrusion” from the outside, but also a USB stick, or other internal vector.*

Intrusion Attempts [3]	Exploiting of known Vulnerabilities	An attempt to compromise a system or to disrupt any service by exploiting vulnerabilities with a standardised identifier such as CVE name (e.g. buffer overflow, backdoor, cross site scripting, etc.).
	Login attempts	Multiple login attempts (Guessing / cracking of passwords, brute force).
	New attack signature	An attempt using an unknown exploit.
Intrusions [4]	Privileged Account Compromise	A successful compromise of a system or application (service). This can have been caused remotely by a known or new vulnerability, but also by an unauthorized local access. *Also includes being part of a botnet.*
	Unprivileged Account Compromise	
	Application Compromise	
	Bot	
Availability	DoS	By this kind of an attack a system is bombarded with so many packets that the operations are delayed or the system crashes. *DoS examples are ICMP and SYN floods, Teardrop attacks and mail-bombing. DDoS often is based on DoS attacks originating from botnets, but also other scenarios exist like DNS Amplification attacks.* However, the availability also can be affected by local actions (destruction, disruption of power supply, etc.) – *or by Act of God, spontaneous failures or human error, without malice or gross neglect being involved.*
	DDoS	
	Sabotage	
	Outage (no malice)	
Information *Content* Security	Unauthorised access to information	Besides a local abuse of data and systems the information security can be endangered by a successful account or application compromise. Furthermore attacks are possible that intercept and access information during transmission (wiretapping, spoofing or hijacking). *Human/configuration/software error can also be the cause.*
	Unauthorised modification of information	

*[3] An "attempt" refers to the mechanism used to **try** and create an intrusion. The intrusion may have failed – or not.*
*[4] An "intrusion" will as rule of thumb be the result of a **successful** intrusion attempt.*

Fraud	Unauthorized use of resources	Using resources for unauthorized purposes including profit-making ventures (E.g. the use of e-mail to participate in illegal profit chain letters or pyramid schemes).
	Copyright	*Offering* or Installing copies of unlicensed commercial software or other copyright protected materials (Warez).
	Masquerade	Type of attacks in which one entity illegitimately assumes the identity of another in order to benefit from it.
	Phishing	*Masquerading as another entity in order to persuade the user to reveal a private credential.*
Vulnerable	*Open for abuse*	*Open resolvers, world readable printers, vulnerability apparent from Nessus etc scans, virus signatures not up-to-date, etc*
Other	All incidents which don't fit in one of the given categories should be put into this class.	If the number of incidents in this category increases, it is an indicator that the classification scheme must be revised.
Test	*Meant for testing*	*Meant for testing*

Table 2.1. eCSIRT.net mkVI Classification Scheme by Stickvoort (2015)

Other classification schemes, with very similar categories and characteristics, include, among other, the "Open Threat Taxonomy" by Tarala & Tarala (2015), the taxonomy developed by the "MISP threat sharing platform", with readily available machine-tags for incidents (Wagner et al, 2016), the CESNET's simplified "incident taxonomy" (Kacha, 2010) and even an incident taxonomy developed by the European Banking Authority (EBA, 2017), regarding major incident reporting for its members (i.e. financial organizations) under EU Directive 2015/2366 (PSD2).

2.2.3. The financial impact of an incident

In a broader context, Knight & Pretty (1997), while investigating organizational catastrophes, showed a direct causal relationship between organizations that effectively responded to an incident and successfully recovered from catastrophes. Shedden et al (2010) state, that the ability of organizations to effectively mitigate incidents, of all kinds, plays a major role in preventing those incidents from escalating into a catastrophe. The number of reported incidents concerning IT systems keeps rising each year (Ryba etl al, 2009), and these incidents are able to inflict staggering financial losses to organizations (Grispos et al, 2014). Information Security threats are now a major risk to organizations, and Chabinsky (2014) states that information security issues require board-level consideration, as they have the same effects with other major business issues. Information Security risk mitigation should be treated as a business issue, since it has a positive impact on the share price and market position of organizations (Von Solms & von Solms, 2005).

Zafar et al. (2012), in a study investigating the impact of an organization publicly announcing an information security breach, identified that a security breach announcement not only affects the impacted organization, but can also have an effect on the wider industry as a whole. The financial impact of security breaches has been emphasized by both academic researchers as well as practitioners, and ways to combat these increased security incidents in organizations are constantly being investigated (Glisson et al.2006). Campbell et al (2003), when examining the economic effect of information security breaches reported in newspapers on publicly traded US businesses, identified a significant negative market reaction for security breaches involving unauthorized access to confidential information – but no significant reaction when the incident did not involve confidential information. Pirounias et al (2014), when investigating the impact of security incidents on a firm’s value, identified a negative statistically significant impact of security breaches, with technology firms appearing to suffer higher costs from security breaches than non-technology firms. In the same context, when examining the impact of security incidents on the stock value of firms, Yayla & Hu (2011, p.60) identified that “pure e-commerce firms experienced higher negative market reactions than traditional firms in the event of a security breach". Furthermore, when investigating security incidents in healthcare organizations, He & Johnson (2017) identified that such incidents can have "a negative impact". In addition, in this new era of "social media" we are now living in, Rosati et al (2019, p.1) identified that "the use of social media exposure at the time of a data breach exacerbates the negative stock price to the announcement", when analyzing 87 data breaches from 73 US publicly-traded organizations. In other words, the fact that anyone can publicly share an opinion in various social networks nowadays, makes things much worse for organizations, following an information security incident.

Along the same lines, investigations by Ettredge and Richardson (2003), Garg et al. (2003), Cavusoglu et al (2004) and Acquisti et al (2006) found that information security breaches lead to significant negative market valuation in organizations. Garg et al (2003) even attempted to estimate the average cost of an incident of publicly listed companies: they found the cost to be somewhere between \$17 and \$28 million per incident, or 0.5 to 1.0 percent of the company's annual sales. On the contrary, there are some investigations that do not find any significant impact in organizations following a security breach, such as those by Hovav and D'Arcy (2003) and Kannan et al (2007). Moreover, an investigation by Rosati et al (2017) on a sample of 74 data breaches from 2005 to 2014, even identified "a positive short-term effect of data breach announcements on both bid-ask spread and trading volume", but only evidenced on the actual day of the announcement, with "market efficiency ensuring a quick return to normal market activity". A systematic review by Spanos & Angelis (2016) analyzing 45 studies on information security impact on stock prices, aligns with the above: The majority (75.6%) – but not all - of the studies, do, indeed, report statistical significance of the impact of security incidents to the stock prices of organizations (Spanos & Angelis, 2016).

It is therefore obvious, that, in most cases, financial consequences to organizations can indeed be brought on by an information security incident.

2.2.4. Incident reporting in incident response's lifecycle

Since attacks frequently compromise valuable business and personal data, it is critical for an organization to respond rapidly and efficiently when security breaches occur (NIST, 2012) .Incident response, which can also be referred to as "incident management" or "incident handling", refers to the formal, structured methods, by which organizations engage "teams" to detect and eradicate information security incidents (Wiik et al, 2005). These incident response teams are created in an effort to address information security incidents (Killcrece et al., 2003; Mitropoulos et al, 2006). The objective of a security incident response team, according to Mitropoulos et al (2006), is to minimize the damage from an incident, while allowing an organization to learn about the root cause of the incident and thus prevent its re-occurrence. Jaikumar (2002) even described these teams as 'firefighters' within organizations, devoted to the preparation, identification, analysis and recovery from security incidents.

A structured incident response procedure allows the organization and particularly those handling the incident to "know exactly what to do" (Osborne, 2001), and is thought to be one of the most important requirements for business continuity in an organization (Nowruzi et al, 2012). Whitman & Mattord (2005) note, that, in general, when considering the timeline of business continuity, incident response is the immediate action taken after a security breach (or potential security breach), while disaster recovery and business continuity are longer-term concerns. In other words, incident response can be described as the considerations and actions undertaken upon the detection of a security incident, and the immediate short-term actions taken to reduce the exposure of the organization (Shedden et al, 2010). It is a critical process, ensuring that organizations have the capability of effectively responding to, recovering from, but also learning from, security incidents (Shedden et al, 2010). A structured procedure also helps eliminating uncertainty and unnecessary panic from the human resources devoted to handling an incident (NIST, 2012). The ultimate objective of the incident response procedure is to minimize the effects of a successful attack and to ensure an expedient recovery (Wiik et al, 2005). Although incidents do not necessarily result in breaches, indeed avoiding that incidents eventually result in breaches, is the main reason for having an incident response procedure in place (Bersnmed & Tondel, 2013). Since incident response should not be just a reactive process but it should also be proactive, in nature (ISO/IEC 27035, 2016), an incident response procedure should take all appropriate measures to minimize the risk of an incident materializing in the first place (Davis et al, 2006).

There are various schemas/models describing the appropriate phases/steps an incident response procedure should follow. NIST (2012), for example, in its "Computer Security Incident Handling Guide", describes four phases in an incident's response life cycle: "Preparation, "Detection & Analysis", "Containment, Eradication & Recovery" and "Post-incident activity". Similarly, ENISA (2010), also includes four phases in its "Good Practice Guide for Incident Management": "Incident Detection", followed by "Triage", "Analysis" and finally "Incident response". Six phases are described by the SANS Institute (2011) "Incident Handler's Handbook", namely: "Preparation", "Identification", "Containment", "Eradication", "Recovery" and "Lessons learned". Another example is the "Information Security Incident management" international standard" (ISO/IEC 27035, 2016), which describes five phases in the incident response procedure: "Preparation", "Identification", "Assessment", "Response" and "Learning". Mitropoulos et al (2006), also presented a six-phase procedure for incident response, as evident in the following figure:

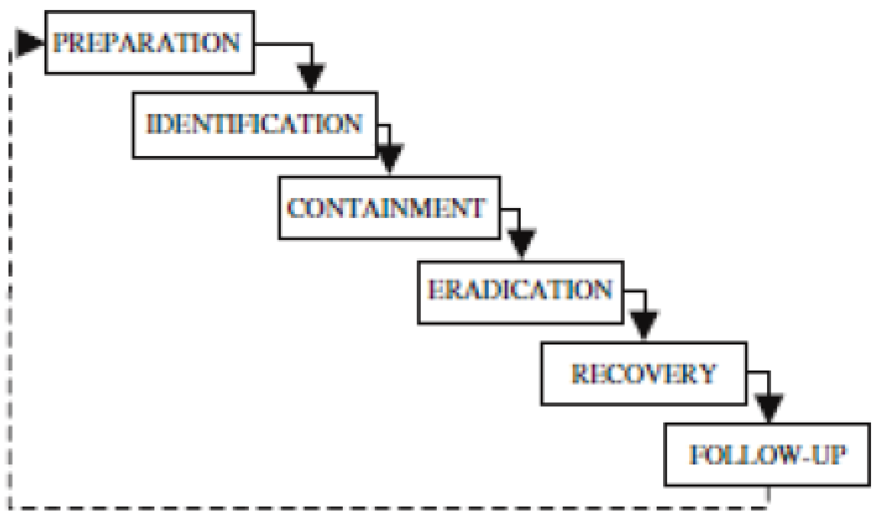


Figure 2.3. The incident response procedure (Mitropoulos et al, 2006)

It is therefore noticeable that while different organizations/bodies/researchers may use slightly different numbering and/or sequencing of phases, the overall philosophy behind an “ideal” incident response procedure, ultimately remains the same.

As stated by Line and Albrechtsen (2016), the various incident response schemas/models have a large number of similarities, with each describing a preparation phase and subsequent phases for detection analysis and incident response, while also including activities related to “lessons learned”, although not all schemas define a separate phase for this activity. Ultimately, these models perform the same functions, through a very similar procedure. It is worth noting, however, that while other models (such as SANS, NIST etc.) are either developed by single organizations or by individual researchers, the ISO/IEC standard is based on international consensus (Line & Albrechtsen, 2016). Furthermore, Tondel et al (2014), in a systematic review of empirical studies on information security incident management, compared the identified studies with the ISO standard and concluded that current practices and experiences align well with it.

The following figure depicts the five phases of the ISO/IEC 27035 standard:

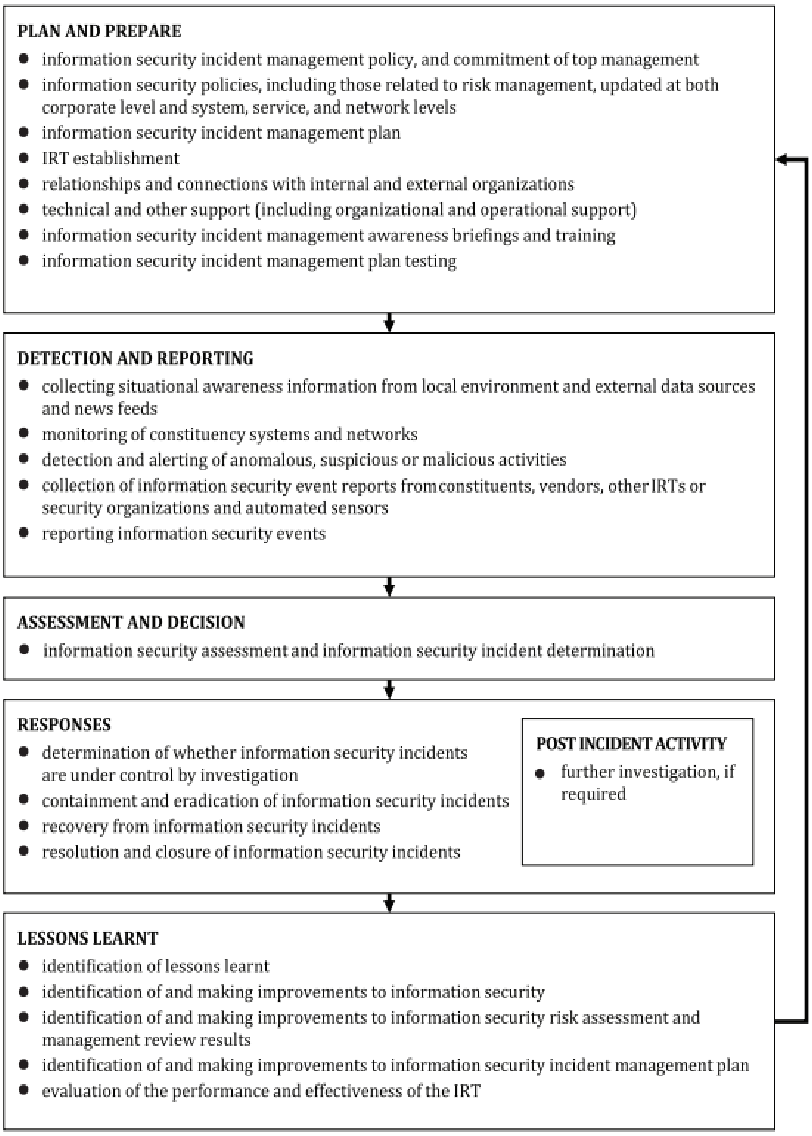


Figure 2.4. Information security incident response phases (ISO/IEC 27035, 2016)

According to the standard, the first phase (“Plan and prepare”) runs continuously, whereas the following four phases come into play upon the occurrence of an incident. “Plan and prepare” phase includes activities such as establishing a team, creating a policy and plan, as well as gaining management support and creating a culture of incident awareness, in an organization. The following phase, “Detection and Reporting”, is triggered upon the occurrence of an event, and involves collecting all necessary incident information from the internal and external environment, detecting the suspicious activity and its sources, and reporting accordingly, possibly through an incident reporting platform or through any other means (e.g. electronic mail or telephone). The “Assessment and Decision” phase is the one that follows, where the event should be assessed, and the kind of response required should be decided. During this phase, a security event could eventually escalate into a security incident. The “Responses” phase denotes the necessary actions required to contain, resolve and recover from the incident. The last phase, “Lessons Learnt”, is the phase where the organization should identify if everything eventually worked out “according to plan” and consider ways to improve their response procedure and their overall information security posture. These improvements should then be fed to the continuously running “Plan and Prepare” phase.

It is therefore critical to explicate where “information security reporting” fits within the overall incident management lifecycle. The obvious answer is, that reporting should be initiated in the “Detection & Reporting” phase - where an event, after its detection and validation, should be reported to designated internal and external systems and actors. Furthermore – and subject to an organization’s reporting protocol – reporting could also be initiated in the “Assessment and Decision” phase, after a security event has been effectively classified as a security incident. However, incident reporting, as an activity, could also be part of the “Lessons learned” phase: After an organization has concluded all necessary actions to resolve an incident – it could report a detailed description of the actual incident - to all interested parties – which could include, among other, a more accurate description of the incident, including its name, category and criticality ranking, the attacker’s modus operandi, the systems affected and any other relevant information. This report could generally re-evaluate all information submitted as part of the initial report, in order to support clarity and accuracy of information, among all information-sharing parties. This could basically be characterized as a “revisit action”, and could be important to perform, as organizations, in the initial stages of an incident, could pay less attention to precisely reporting - and describing - an incident (especially to external parties), as they would rather focus their efforts towards identifying, containing, resolving and recovering from an incident occurring within their own organization. The following figure portrays the three phases of the ISO 27035 (2016) standard where incident reporting could be incorporated:

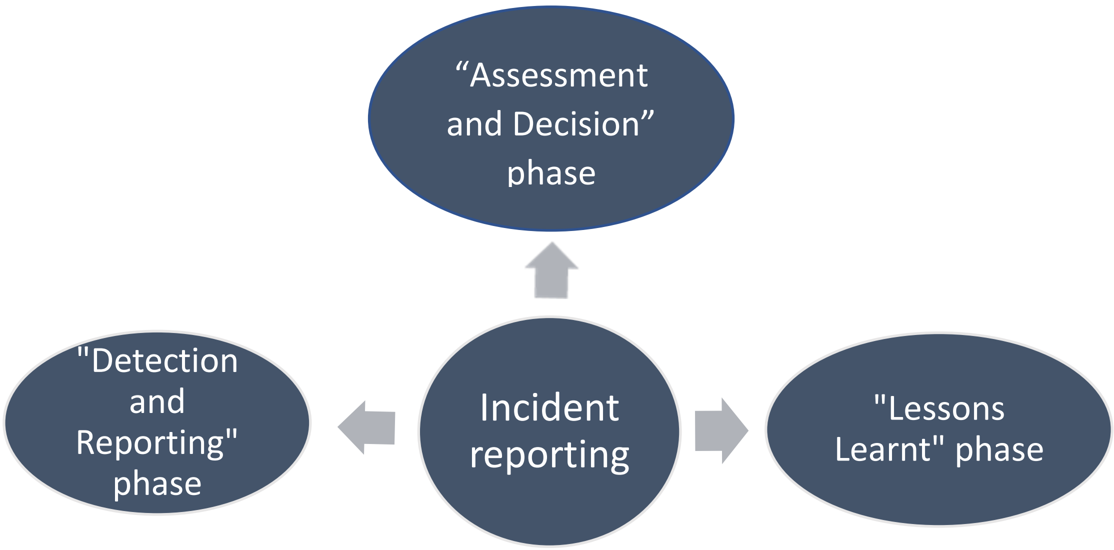


Figure 2.5. Incident reporting in the incident management lifecycle of ISO 27035

2.2.5. Incident reporting: Scaling the benefits

As already mentioned, it seems that organizations commonly find it difficult to disseminate information related to security incidents (He and Johnson, 2012; Grispos et al., 2015). A general mistrust is usually shown to any outsider who wants to obtain data on internal information security issues (Kotulic & Clark, 2004). The task of information sharing gets even harder, since organizations usually do not have a systematic or standardized way of sharing incident-related information, as identified by He et al (2014), in an empirical study aiming to present a template for structuring the organizational lessons learned from security incidents. Organizational disclosure decisions are usually shaped by various factors, such as information collection and processing costs, regulatory and litigation costs, as well as various other economic and reputational factors (Meek et al, 1995; Schwartz & Janger, 2006). Having these in mind, should organizations ultimately report or not their various security incidents? The following sections present the positive and negative aspects of each approach.

2.2.5.1. Motivation for incident reporting

According to ISO/IEC 27002 (2013), since information security incidents might surpass organizational and national boundaries, there is an increasing need to coordinate response and share information about these incidents with external organizations. Efficient reporting is considered a major factor, for effective information security management in an organization (Ma et al, 2009) and the need for information dissemination is created by every incident, both for people inside the company, as well as for outside audiences (Coombs & Holladay, 2012). Information sharing is an important aspect of incident response coordination, where different organizations share threat, attack, and vulnerability information with each other, so that each organization's knowledge can benefit the other and thus collectively reduce the potential impact of incidents (NIST Incident Handling Guide, 2012). According to the same source, this is both necessary and mutually beneficial, because the same threats and attacks often affect a multitude of organizations. Along the same lines, Reynolds & Seeger (2005, p.46) argue that disclosing incident information is an essential part of crisis communications and can "reduce and contain harm, provide specific information to stakeholders, initiate and enhance recovery, manage image and perceptions of blame and responsibility, repair legitimacy, generate support and assistance, explain and justify actions, apologize, and promote healing, learning, and change".

Information security reporting can be viewed as a quality improvement process (Gonzalez, 2005). Among the ability of recognizing an incident, effectively reporting it is of paramount value in today's digital atmosphere (Grispos et al, 2017). Researchers (Sveen et al, 2005; Khurana et al, 2009) argue that the reporting of incidents allows them to be investigated and learned from, and that this knowledge can then be used to avoid such incidents occurring in the future, by putting into place adequate technical and organizational countermeasures. Therefore, overall cyber security posture can be improved by voluntarily sharing incident information across industries (Schwartz & Janger, 2006). Furthermore, Hausken (2007) argues that when security investments become too costly for organizations, the exchange of incident-related information with other business entities can generally improve their cyber defense.

Regulatory compliance can also act as a major motivation for incident reporting (Fitzpatrick & Rubin, 1995). Regulations such as the EU General Data Protection Regulation (GDPR), the NIS Directive for critical infrastructure providers and the PSD2 Directive for financial institutions, all impose strict incident-related reporting requirements for organizations (Housen-Couriel, 2018). Among complying with regulatory requirements, organizations choose to disclose incident information also for other reasons, such as restoring reputation in the eyes of the media and value chain, or even for asking for the help of supporters (Kaufmann & Kesner, 1994). Kulikova et al (2012) argued in favor of voluntary reporting, based on the fact that the vast majority (92%) of security incidents in 2011 were discovered by a third party, and thus organizations eventually had to deal with the “public embarrassment” caused by the incident. Regarding the matter of “restoring public reputation”, Gordon et al (2010) even found statistically significant evidence that voluntary disclosures about security incidents have a positive effect on the market value of an organization, although this can be contentious, as evident in the following section.

2.2.5.2. Demotivation for incident reporting

Although it seems there are many benefits for reporting incidents (Sveen et al, 2005; Khurana et al, 2009; Schwartz & Janger, 2006; Hausken, 2007; Ma et al, 2009; Kulikova et al, 2012; Reynolds & Seeger, 2005; Gonzalez, 2005; Grispos et al, 2017 and more), the reporting statistics convey a different reality. According to the IOD & Barclays Policy report (2016), only 28% of cyberattacks against businesses were reported to the police. The Internet Security Threat Report by Symantec (2016) mentions that the number of companies that refused to report the scope of a data breach jumped by 85 percent, compared to the previous year. The US Securities & Exchanges Commission reported that nearly 65 percent of affected public companies did not report cybersecurity breaches to the SEC, between 2011-2017 (Newman, 2018). Since 2011, when the SEC issued its initial cyber guidance, only 106 companies have reported incidents to the Commission. But during that same period, there were 4,732 cyberattacks on American businesses, researchers for the Privacy Rights Clearinghouse found (Newman, 2018). Furthermore, a report by Ipsos MORI and the University of Portsmouth (2017), has identified that just over four in ten

(43%) UK businesses (survey sample: 1523 UK businesses) reported their most disruptive breach outside their organization, and most commonly this was reported only to an outsourced cyber security provider (where the reporting might be to enable appropriate aid). Only 26% of the most disruptive breaches were externally reported to anybody outside of a cyber security provider, with the most common places to report the breach being a bank, building society or credit card company, followed by the Police at just 19%. Reporting to other public sector agencies was identified as very low, with reporting to Action Fraud UK being the most common (7%), followed by a few other public sector agencies, such as the Centre for the Protection of National Infrastructure (CPNI) and the Cyber Security Information Sharing Partnerships (CISP and CIFAS). Even between state-members of the EU, there is still little exchange of information about breaches, between different national authorities (ENISA, 2012).

Sharing information about information security issues is not always as straightforward, since sensitive information is involved (Line & Albrechtsen, 2016). Organizations share a lack of willingness to disseminate incident-related information outside the organization (Jaatun et al, 2009; Hove et al, 2014), as well as a general lack of openness, when it comes to discussing security incidents (Jaatun et al, 2009). A study by Koivunen (2010) identified a substantial demand for the incident originators to retain their anonymity throughout the reporting process. Furthermore, according to Ahmad et al (2015), organizations incline towards purposely excluding ‘outsiders’ from the early stages of incident response, in order to prevent “misunderstandings” and “premature conclusions”, which may lead to embarrassment. Even within a context of a “trusted relationship”, it seems that organizations hesitate to report security incidents to other entities, for various reasons, which include negative publicity, competition and regulatory compliance (Ruefle et al, 2014). Choo (2011) states that organizations under-report due to considering most incidents as not being “serious enough”, as well due to concerns regarding adverse publicity and low chance of prosecution. According to Kopp et al (2017), even when sharing is mandated by a regulation, and government agencies actively contribute their own knowledge of cyber incidents for the mutual benefit of participants, private sector actors' participation may be less than optimal. In addition, Humphrey (2017) states that organizations may under-report due to their limited IS organizational maturity: they may focus efforts on the immediate incident, rather than on the root cause of the problem and they may be resistant to change, based on the rigidity of their core beliefs. Furthermore, they may lack corporate responsibility, while they could also exhibit a tendency towards scapegoating. They could also face dilemmas of contradictory imperatives, such as the need for "communication" versus the need for "confidentiality".

However, reporting issues may originate from within. Humphrey (2017) identified that incident under-reporting might occur in organizations because of ineffective communications of personnel, as well as due to the difficulty faced by employees in understanding complex events. Furthermore, the author states that people are often not willing to learn from negative experiences (even if it's for their own benefit), and that human alliances can lead to people "forgiving" other colleagues. Low job satisfaction and a high level of stress could also have adverse effects on the quality/quantity of reporting (Humphrey, 2017). Furthermore, Ahmad et al (2012) identified several reasons which discouraged employees from reporting security incidents within an organization. These reasons include fear for financial penalties, reprimands and reputational impact, as well as burdensome follow-up procedures applied by the regulators. Johnson (2002), when proposing barriers that must be addressed for incident reporting to be effectively applied in industries, suggested removing the fear of retribution from reporting entities, as well as creating an organizational environment which encourages reporting, while also isolating the fear of negative media publicity. Moreover, other issues also influence the overall effectiveness of the reporting process. Metzger et al. (2011) identified that some network administrators did not report incidents, either because they "did not know that they should" or because they were afraid of the incident's consequences. Moreover, according to Briggs et al (2017, p.5), employees may also hesitate to report incidents due to the fear about "being held accountable for the outcome". Hove and Tårnes (2013), while conducting a survey of employees in an organization, identified that employees were not sure which incidents to report and to whom, within the organization. Moreover, Jaatun et al (2009) identified a "deep sense of mistrust" between network administrators and process control engineers – a fact which is distressing, since incident management is collaborative in nature (Tondel et al, 2014). Cusick and Ma (2010) state that a variety of incidents may be observed but not necessarily logged, typically when the incident is considered as "non-critical". Along the same lines, Kurowski and Frings (2011) identified that in organizations with an active reporting system/platform, just 17 percent of the IT Security Managers surveyed admitted that all cases were registered in the incident reporting system/platform, with as many as 50 percent of the incidents reported just through e-mail or telephone, without being added to the platform.

Bad communications, internal or external, can cause an overall confusion about a situation among key audiences, they can initiate rumors and they can even have a negative effect on a firm's shares (Dilenschneider & Hyde, 1985). It seems likely that organizations face both internal and external contemplations in the overall incident reporting process. However, as Ruefle et al (2014) point out, incident reporting is beneficial for organizations and throughout the process of incident reporting, it is important for them to properly balance the protection of the organization's identity and providing generalized incident information. This will allow the recipients to assess their similarity to the target, perform risk analysis, and prioritize defensive actions (Ruefle et al, 2014).

2.2.6. Incident reporting: Means and methods

Incident reporting can be accomplished manually or automatically and can happen through various means such as telephone, e-mail, reporting software/platforms, or through verbal communications (Metzger et al, 2011; Grispos, 2017). Manual reporting refers to the manual registration of incidents (through any means) by human beings, while automatic reporting does not require human interference - it can be accomplished through various automated tools, such as antiviruses, firewalls and IDS/IPS systems. It is important to note, that automatic reporting is directly linked – and immediately follows - the automatic "detection" of incidents, by these various tools, whereas manual reporting depends on an individual registering the incident, with information probably combined from a variety of sources (which may also include information received from automated tools), but with the incident's detection and identification/validation stage already to have taken place (Schultz, 2007).

2.2.6.1. Automatic vs manual reporting

As previously mentioned, detection, collection, and reporting of incidents may happen manually or automatically (Tondel et al, 2014). When researching current practices in information security incident management, Tondel et al (2014) described many examples of automatic reporting mechanisms, such as reports received from security monitoring systems (IDS/IPS), antivirus software, honeypots, log monitoring systems, information security management systems and correlation engines, as well as reports occurring from network monitoring systems such as firewalls, network flow analysis, and web filtering mechanisms. Metzger et al (2011) and Grispos et al (2017) described ways for the manual reporting of incidents, which cannot happen without human interaction, and involve individuals registering incidents, such as reporting incidents through telephone and e-mail, through verbal communication and through software implementations, such as various reporting/incident tracking platforms.

In some organizations, the detection and reporting processes of incident handling are completely automated (Cusick and Ma, 2010; Metzger et al., 2011). This is a relatively new concept (Koivunen, 2010) and only became available when, in 2007, the Trans-European Research and Education Networking Association (TERENA) proposed a machine-readable format for the automated reporting of incidents: The Incident Object Description Exchange Format (IODEF), an XML-based scheme for representing information security incidents commonly exchanged between CSIRTs, currently in its second version (Danyliw, 2016). Since then, however, other standards have been proposed, by various organizations and bodies, in an attempt to standardize automated security information sharing. Examples of these standards include, among others, the Trusted Automated Exchange of Indicator Information (TAXII), proposed by the US Department of Homeland Security and utilizing the STIX language – an XML language specifically used for conveying cyberthreat information (Kampanakis, 2014), the Security Content Automation Protocol

(SCAP), proposed by the US National Institute of Standards and Technology (NIST) and the Common Vulnerability Reporting Framework (CVRF), proposed by the Industry Consortium for Advancement of Security on the Internet (Kampanakis, 2014). These standards enabled the creation of various automated information sharing platforms, such as the “Cyber Threat XChange (CTX)” platform, the “Open Threat Exchange (OTX)” platform, the “Soltra” platform and the “Collaborative Research into Threats (CRITS)” platform (Mtsweni et al, 2016). In addition, automatic reporting platforms have been developed by national CERTs, such as the “Warden” platform developed by the Czech CERT (Bodo & Kouril, 2014), the “AbuseHelper” platform developed by the Finnish, Estonian and Belgium CERTs, the “Megatron” platform developed by the Swedish CERT and the “n6” platform developed by the Polish CERT (Kijewski & Pawliński, P, 2014).

Automation in incident reporting, however, does not come trouble-free and the automated tools have their limitations (Tondel et al, 2014). Most of the proposed standards use XML, which can be considered a restricting factor for data sharing, with concerns focusing on redundancy, storage size, and processing (Kampanakis, 2014). Werlinger et al (2010) identified a lack of accuracy in tools, with high false positive rates, as a result. In addition, the automated tools’ usability is also a concern, with researchers identifying an organizational need for often customization/adjustments of these tools (Werlinger et al., 2008, 2010; Metzger et al, 2011). Furthermore, information needs to be sanitized before automated exchange can take place, while sharing all available security data could lead to performance and scaling concerns in organizations (Kampanakis, 2014). Hove and Tårnes (2013), when researching automated monitoring/reporting systems in three organizations, although they did signify the potential of tools and automation, they also highlighted the fundamental role of users in detecting and reporting abnormal and suspicious system behavior. Line (2013), when researching the power automation systems in six large distribution system operators, identified that although automated detection/reporting systems were in place, in most cases operators relied on manual detection/reporting of incidents by the employees.

It seems that manual reporting still remains the key in the reporting of incidents, despite the recent focus in automatic mechanisms (Werlinger et al, 2010; Koivunen, 2010; Metzger et al, 2011; Hove and Tårnes, 2013; Line, 2013). This is further supported by a case study conducted by Grispos et al (2015), who identified that the majority of security incidents in the organizations they surveyed were reported manually, either through e-mails or verbally. In addition, Metzger et al (2011) reached similar conclusions: they identified that even when automated systems were in place in the organizations they surveyed, the majority of incidents were manually reported, by either e-mail or telephone, through local systems and service administrators. According to the same authors, when they examined various CSIRT operations at the Leibniz Supercomputing Centre (LRZ), they identified that the manual reporting process was the one most frequently used, and ultimately, the one truly essential. Along the same lines, Hove et al (2014) concluded that in organizations, manual reporting processes were more popular than automatic ones. These findings, however, do not suggest that manual reporting is a panacea for organizations, nor that issues cannot still be caused by manual reporting processes. For example, manual reporting can still produce false positives: users may inadvertently input false data into a reporting form, or even worse, they may deliberately select to do so, given the 'right' motivation. It is therefore crucial, whatever the method (manual or automatic) an organization ultimately utilizes for its reporting purposes, that proper security controls are in place (e.g. policies, standards and procedures, proper sanitization and scrutinization of reports and other).

2.2.6.2. Incident reporting platforms vs other reporting means

Researchers have indicated a variety of methods that employees use for the manual reporting of incidents, such as e-mail, telephone and other verbal communications (Metzger et al, 2011 and Grispos et al, 2015, 2017; Hove and Tårnes, 2013; Line, 2013 and others). In addition, studies by Ahmad et al (2012) and Hove and Tårnes (2013) revealed that in some organizations, incidents were reported through existing help desk functions. Furthermore, manual incident tracking/reporting platform systems were in use by organizations surveyed in studies by Ahmad et al (2012), Cusick and Ma (2010) and Metzger et al (2011).

Communicating incidents though e-mail, telephone or other informal tools might become problematic: e-mails could be delivered to the wrong recipients (or not delivered at all), telephones might not be answered, and verbal communications might be ignored, or even deliberately neglected. The utilization of an incident reporting platform, for reporting purposes, is considered of high value to organizations: Metzger et al (2011) stated that organizations should use such a tool and recommended to collect all data related to the incidents into such a system, while Cusick and Ma (2010) praised the use of an incident reporting platform for reporting incidents.

An incident reporting platform, with a clean and easy to navigate interface, with clear submission instructions and available templates, could probably aid the incident reporting capability of an organization. It could eliminate the possibility of delivering a report to unauthorized recipients (since the platform's users would be pre-authorized), while it could also enable the possibility of extracting statistics and reports, viewing historic trends, and submitting queries in a searchable database.

2.2.7. Evaluation of existing reporting platforms

As mentioned in the "Introduction" chapter of this document, a search for identifying incident management/reporting platforms/software available through the web, indicates a far greater amount of available platforms/software related to reporting/managing "safety" incidents (i.e. workplace health & safety, personnel injuries, facilities maintenance disruption and generally incidents directly related to occupational health and safety agencies) rather than "security" incidents (i.e. information security incidents). There are, of course, similarities between safety and security platforms, since in many cases they are interrelated (Sveen et al, 2007), however "safety" reporting platforms fall out of scope, since they serve a different purpose.

Information Security incident reporting has traditionally occurred through ad hoc methods, such as email, instant messaging clients, and phone (NIST, 2012). According to the same source, this type of reporting usually relies on an individual's connections with employees in incident response teams of partner organizations and tends to be largely unstandardized, in terms of what information is communicated and how that communication occurs. As an alternative, various reporting platforms (software) have been created and became available in the market, which can be used for reporting internally, within an organization, or externally, with third parties.

2.2.7.1. Identification of existing reporting platforms

The first step towards evaluating the existing information security incident reporting platforms/software currently available, concerned their discovery and identification. Therefore, a rigorous search was conducted in both general interest databases (i.e. "Google" and "Bing" search engines) and academic databases. This rigorous search was necessary, in order to identify as many available/accessible reporting platforms as possible, so as to appropriately identify, compare and evaluate their various features and characteristics in a sufficiently representative degree. Although the search was international in scope, only texts in English language were considered. The search was purely electronic, no grey literature or hand search was applied. The following table summarizes the applicable search criteria:

Search objective	Discover and identify existing security incident reporting software/platforms
Search type	Electronic
Search strings used (in multiple combinations)	"incident", "report*", "software", "platform*", "information", "security", "tool*"
General interest databases searched	Google search engine, Bing search engine
Academic databases searched	ArXiv, CiteSeerx, DBLP, Proquest, Taylor Francis, Science Direct, Semantic Scholar, Wiley Online Library, Elsevier, Google Scholar
General interest database search details	Title and description of each result was examined. In cases of insufficient information, the electronic resource was accessed/examined

Academic databases search details	Title, abstract and keys/notes of each paper was examined. In cases where the abstract provided unclear or insufficient information, the whole paper was examined
Academic resource type	Articles and proceedings (not limited to peer-reviewed), reviews, books
Software reporting type	Pure incident reporting platform or incident response/management platform with integrated reporting functionality; Manual and/or automatic reporting functionality
De-duplication method	Manual, non-automated
Language	English
Creation date	Not specified

Table 2.2. Search criteria for identification of reporting platforms

A total number of 286 unique, reporting platforms/software were identified, which belonged to several, different, categories. The following table presents a summary of the results:

Category	Examples of platforms	Total number of software/platforms
Information Security incident reporting platforms (manual and automatic)	Warden (auto), Threatvine (manual), MISP (auto & manual), CyberCPR (manual), LogicManager (manual)	44
IT helpdesk functionality/bug tracking reporting platforms	JIRA Service Desk, Pager Duty, Victorops, Freshservice, ServiceDesk Plus	89
Safety incident reporting platforms	Quentic Incident Management Software, ProcessMAP Incident Management, Safety Dashboard, 1st Incident Reporting, andSafety Hazard and incident	136
Incident reporting platforms for specific industries	SitePatterns (construction), Alliance (healthcare), I-sight (investigations), 3tc Software (Fire & rescue services), Datix Incident reporting (healthcare)	17

Table 2.3. Incident Reporting platforms

As expected, the information security reporting platforms identified are considerably less in numbers, compared to the other categories. However, these other categories of reporting platforms are out of the scope of this project, since they are not directly relevant to the information security domain. Out of the 44 information security incident reporting platforms identified, 32 are utilized exclusively for the automatic reporting (following the automatic detection) of incidents and are also out of scope. The remaining platforms (some of which may also support automatic reporting, but also allow for the manual reporting of incidents) are presented in the following section.

2.2.7.2. Comparison of existing reporting platforms

In total, 12 manual reporting platforms were identified: Threatvine, CyberCPR, Blackthorn GRC, MISP, LogicManager, D3 Security, Resolver's Incident Management Software, OTRS' STORM software, TheHive Project, MetaIncident, Cherwell Incident Software and SureCloud platform. In order to perform a comparison between the platforms, 11 directly comparable features of the platforms were evaluated. These were extracted from each platform's published product guide/whitepaper, based on simple inclusion criteria: a feature/characteristic was included in the comparison if it appeared in two or more of the available guides/whitepapers. The results are presented in the following table:

Name	Threatvine	CyberCPR	Blackthorn GRC
Website	https://www.surevine.com/threatvine/	https://www.cybercpr.com/	https://www.blackthorn.com/
Software type	Commercial software	Commercial software	Commercial software
Deployment	Cloud or on premise	Cloud or on premise	Cloud or on premise
Guaranteed availability	99.5%	98%	Figure not disclosed
Customer support available	Yes	Yes	Yes
Web-browser interface	Yes	Yes	Yes
Mobile-friendly version available	Yes	Yes	Yes

Automated reporting features	No	No	No
Customization possible	Yes	Yes	Yes
Developer ISO/IEC 27001 certified	Yes	No	No
Supports anonymous submissions	Yes	No	No
Community forum/chat/social features available	Yes	Only chat available	No
Name	**MISP**	**LogicManager**	**D3Security**
Website	https://www.misp-project.org	https://www.logicmanager.com	https://d3security.com/
Software type	Open source	Commercial software	Commercial software
Deployment	On premise	On premise	On premise
Guaranteed availability	N/A	N/A	N/A
Customer support available	Limited – community-based support available	Yes	Yes
Web-browser interface	Yes	Yes	Yes
Mobile-friendly version available	No	Yes	Yes
Automated reporting features	Yes	Yes	Yes
Customization possible	Yes	Yes	Yes
Developer ISO/IEC 27001 certified	No	Yes	Yes
Supports anonymous submissions	No	No	No
Community forum/chat/social features available	Yes	No	Chat available
Name	**Resolver**	**OTRS' STORM**	**TheHive**
Website	https://www.resolver.com	https://otrs.com/	https://thehive-project.org/
Software type	Commercial software	Commercial software	Open source
Deployment	Cloud	On premise	On premise

Guaranteed availability	99.9%	N/A	N/A
Customer support available	Yes	Yes	Limited – community-based support available
Web-browser interface	Yes	Yes	Yes
Mobile-friendly version available	Yes	Yes	No
Automated reporting features	Yes	No	Yes
Customization possible	No	Yes	Yes
Developer ISO/IEC 27001 certified	Yes	No	No
Supports anonymous submissions	No	No	No
Community forum/chat/social features available	No	No	No
Name	**MetalIncident**	**Cherwell**	**SureCloud**
Website	https://www.metacompliance.com	https://www.cherwell.com	https://www.surecloud.com/
Software type	Commercial software	Commercial software	Commercial software
Deployment	On premise	On premise	Cloud
Guaranteed availability	N/A	N/A	95-99%
Customer support available	Yes	Yes	Yes
Web-browser interface	Yes	Yes	Yes
Mobile-friendly version available	Yes	Yes	Yes
Automated reporting features	No	No	No
Customization possible	No	Yes	Yes
Developer ISO/IEC 27001 certified	No	No	Yes
Supports anonymous submissions	No	No	No
Community forum/chat/social features available	No	No	No

Table 2.4. Comparison of Information security Incident Reporting platforms

The core functionality of reporting platforms is essentially the same; participants gain access to a platform, where they can view and/or report incidents. Information is stored in a centralized database (of the customer or of the platform's provider), while the service can be installed either on premise or on a public/private cloud, with some platforms offering both options. Regarding cloud deployments, the offered guaranteed availability ranges from 95 to 99.5 per cent, while all platforms offer customer support (with various packages/levels). Access to the reporting platforms is possible through a web-based interface, while most platforms have also developed separate versions for mobile clients. In addition to the manual submission of incidents, some platforms offer automated reporting features, which can automatically process logs from various security tools such as IDS/IPS systems, firewalls and other network monitoring mechanisms. At an extra cost, most platform providers are willing to customize their software, to fit the customer's exact environment and needs – while customization is certainly possible with the two, available, open-source, platforms. About half of the platforms' developers are certified with the ISO/IEC 27001 standard, the widely regarded standard for information security, a fact which may indicate that their product may be more secure and trust-worthy (Hsu et al, 2016).

The typical user interface contains a homepage (with latest incidents, alerts and news) a support page and/or forum (some platforms also utilize a community forum and/or chat functionality for participant conversations), a profile page for each member (some platforms even offer social-networking-like functionality, such as following a member, updating your status, adding skills to your profile and giving out endorsements), and the ability to create and share an incident report or to search/browse through the already submitted incidents. Submitted incidents are usually ranked (by users who submit them) according to their severity (or risk ranking) and their visibility (some members may restrict access to other members of the same platform – e.g. a user may submit an incident only visible within his/her own company or only visible to sector-specific institutions which are members of the platform). When creating a new report, users can input various details of the incident, such as its category (e.g. phishing, Denial of Service attack, malware etc.), the incident's details (e.g.

date, scope, duration, affected systems, modus operandi and various other technical details) and also upload attachments, such as text files, videos or photographs. Regarding anonymity, the “Threatvine” platform is the only platform which provides the option of “anonymous” submissions.

2.2.7.3. A review of existing reporting platforms

This section provides a review of the various features and characteristics of the existing information security reporting platforms:

a) Ease of use

All identified platforms are relatively easy to understand and use. Authenticated users can access the web-based platforms through a common web browser (with internet connectivity) and interact with relatively simple and straight-forward GUIs. The interfaces are clear and concise, with most platforms offering simple functionality explanations when a user hovers over the interface’s various buttons. Users can easily create, modify and delete incidents, as well as information related to their personalized profiles (where applicable). The GUIs are adequately intuitive: should an organization decide to switch to a new platform provider, its users would experience a very similar environment, with identical functionality, with maybe some features added or missing, such as user forums and social features. The GUIs of all platforms are generally consistent, across the different pages/sections of the software. Users can access the platforms from a variety of different devices (since the only prerequisite is a web browser) and most platforms offer versions of their software specifically designed for mobile use - a feature consistent with the steep increase in the overall use of mobile devices (Patel et al, 2016).

b) Support

All commercial platform providers offer initial platform functionality training for organizations buying their software, either for free, or at a cost. This is not the case for open-source platforms, however, where users are only limited to downloading software manuals. Furthermore, all commercial platforms offer technical support to their customers, through on-site visits, telephone, online chat and e-mail. The typical support package is limited to office-hours support, but in most cases, organizations have the option of upgrading (at an extra cost) to an extended-hours support package. Direct technical support is generally not an option for open-source platforms, but users of these platforms can submit issues for review by the developers and also seek answers to their questions through the community forums available. Documentation and various manuals are available for both commercial and open-source platforms.

c) Customization

Most providers can customize their reporting software according to a customer's specific environment and needs. For commercially available software this can be done at a cost, whereas for open-source platforms, software modifications can be done by the user himself. Although the standard version of software should fit most client needs, customized software might be needed for various reasons, such as expanding or decreasing the types/categories of incidents, incorporating new features (such as offline reporting or social features) and more.

d) Performance (responsiveness and reliability)

In terms of software responsiveness, the GUIs in most platforms are fast and responsive. Although no explicit testing routine was applied, the average load, wait and response times are not, in any way, cause for concern - in either cloud or on-premise installations. In terms of software reliability, no major software run-time errors (faults) were identified, in any of the platforms. Again, although no explicit testing routine was applied, the platforms exhibited a failure-free operation throughout the course of their initiated demo sessions.

e) Scalability

Due to the limitations imposed by the demo versions of the various platforms, software scalability could not be exhaustively tested. It was not possible to add multiple users to the platforms and submit multiple incidents, in order to examine how the various platforms handle increased traffic/demand, from multiple devices. It was also not possible to save a vast number of incidents in any platform's database, in order to test how the software performs under such circumstances. However, because of the centralized nature and expandable storage capabilities of the platforms' databases, these are expected to handle demand relatively effortlessly. Most platforms state that their reporting software is not CPU intensive and some provide metrics (such as CPU, Disk Storage and Memory Usage) to detect possible service degradation and take appropriate action. Nevertheless, an information security incident reporting platform – in any organization/group of organizations, regardless of size – is not theoretically expected to yield an enormous data volume, capable of deteriorating the software's performance.

f) Availability

The deployment options for the reporting platforms include cloud or on-premise installations. Some platforms offer both options to their customers. Regarding cloud deployments, providers offer up-time availability which ranges from 95 to 99.5 per cent, with different platforms mentioning various available resilience options, such as daily incremental backups, weekly full backups, real-time data mirroring at other sites and fail-over clusters. Regarding on-premise deployments, the availability figures depend on the organizational set-up and environment.

g) Transparency

All platforms support audit mechanisms - in the form of audit logs, which record user actions with relevant timestamps, read and write requests, successful and unsuccessful login attempts, logout information, IP addresses and other user identification information. Access to these logs can be permitted to specific, authenticated, users by the organization itself.

h) Security

All platforms support encryption protocols (Legacy SSL and/or TLS) both for communication between the organization and supplier networks (for cloud deployments), as well as for communication within the platform network itself. Most of the platforms support that their systems undergo a penetration test, at least on a biennial basis, while some platforms also claim that their systems employ appropriate DDOS protection controls. Concerning physical access control (regarding supplier premises), some platforms claim the presence of controls complying with the SSAE-16 / ISAE 3402 standards. About half of the suppliers claim organizational conformity with the ISO 27001 standard for information security, including secure software development and secure disposal methodologies. Regarding authentication methods, all platforms support user authentication through pairs of usernames and passwords, while some platforms also offer two-factor authentication for their users. A very few platforms offer login identity federation with existing providers (e.g. Google Apps), as an additional authentication option.

i) Anonymity

Out of the 12 identified platforms, just one (Threatvine) provides the option of anonymous submission of incidents. While a user completes an incident submission form in the platform, he/she is given the option of clicking on a checkbox:

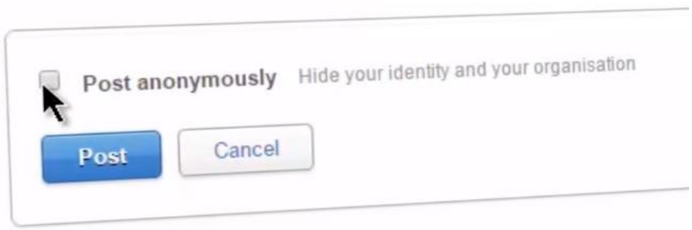


Figure 2.6. Anonymous posting checkbox of Threatvine reporting platform

No further information was, however, available in any of their technical specification guides, regarding the technical details of this anonymity feature. Therefore, direct communication was established (via Skype) in order to clarify the mechanism behind this anonymity feature. The call revealed that the anonymity feature, when selected, just removes the form fields which contain individual identifiers (e.g. name and organization) from an incident report form, but the platform owner (and/or any other actors with sufficient privileges) can nevertheless see all the incident information, including identifiers.

j) Cost

The identified platforms quote a vast range of pricing options, which are based on various factors, such as license duration, level of support, number of users, type of installation and many more. This is not applicable to open-source platforms, which are offered at no cost. It is important to note that all commercial platforms provide a free, limited-functionality demo version of their software, for evaluation purposes. In order to directly compare the various pricing options, a base configuration was established, which included licensing of the platform for twelve (12) months, for a total number of thirty (30) users, with the standard level of support and cloud deployment. Prices identified range from GBP 25,500 to GBP 150,000.

k) General comments

In general, the available, manual, information security incident reporting platforms seem to be easy to understand and use, utilizing simple and straight-forward GUIs, with a good level of support and training offered by the commercial providers. Performance seems to be smooth, in either cloud or on-premise deployments, although scalability could not be adequately tested, since the demo versions available prohibited the simulation of a resource-intensive environment, with many users and multiple submission of incidents. However, and as previously mentioned, manual information security incident reporting platforms are not theoretically expected to yield an enormous data volume, capable of deteriorating performance and efficiency. Non-constant availability is certainly an issue, but this is the case with any centralized environment. Regarding security, the encryption supported in the communication channels is certainly a major plus, while two-factor authentication offered by some platforms suggests enhanced security. However, when it comes to logging and auditing, stolen credentials could easily lead to the unauthorized modification (including erasure) of a centralized database. The absence of the option of anonymous submissions is certainly an issue, since reputational concerns are a major factor for organizations sharing information security incidents (Line & Albrechtsen, 2016; Jaatun et al, 2009; Hove et al, 2014; Ahmad et al, 2015; Ruefle et al, 2014; Koivunen, 2010; Housen-Couriel, 2018). Even Threatvine platform, which claims the option of anonymous submissions, does not essentially provide true anonymity. Furthermore, the cost of acquiring and operating a commercial reporting platform is certainly not negligible. Due to their centralized nature and sensitive content, reporting platforms require major investments for ensuring their security (both physical and electronic). This cost, along with all other costs associated with a centralized database (i.e. need for increasing storage space, disaster recovery/business continuity arrangements and other) is, of course, ultimately passed on to the platform's customers.

It therefore comes as no surprise that despite the general encouragement for information sharing related to information security incidents, organizations continue to approach it with ambivalence (Aviram & Tor, 2003). According to Housen-Couriel (2018), current reporting platforms have many drawbacks, including the problematic trust relationships among participants - who may be competitors, as well as the absence of transparency regarding both the confidentiality and efficiency of the platforms, which includes the use of the shared data by government agencies for non-information security-related purposes, as identified by Johnson et al (2016).

In addition, the exposure of organizational vulnerabilities is also thought as a drawback and so are the various costs related to incident reporting, such as operational cost, recruitment, training, and overall organizational time spent by an organization's personnel on reporting, including time devoted to examining potential "false positives" (Etzioni, 2014). Legal liability is also a major organizational concern (Housen-Couriel, 2018). These various reporting concerns need to be addressed by the reporting platforms themselves, because they have a direct impact on their own sustainability and effectiveness (Vazquez et al, 2012).

2.2.8. Other related work to incident reporting

A lot of identified literature deals with incident management and response – as a holistic process - rather than incident reporting, explicitly. This fact was also stated by Patrascu and Patriciu (2013), in a paper proposing a framework for incident response and reporting in Cloud environments; nevertheless, there seems to be a general need for more empirical studies in the incident response field (Tondel et al, 2014).

Werlinger (2010), Metzger et al (2011), Hove and Tårnes (2013) and Ahmad et al (2012 & 2015) have all partly examined incident reporting procedures, as part of a wider context of empirically examining incident response in various organizations. Line and Albrechtsen (2016) have also examined incident reporting, as part of examining information security incident management, in comparison with industrial safety management. Moreover, Sveen et al (2009) examined the role of information security incident reporting systems - in the wider context of an information security management system - and found that incident reporting is a crucial component in creating information security awareness among users.

A variety of researchers, have, however, dealt directly with incident reporting, in a multitude of ways. From a higher perspective, Settani et al (2017) proposed a collaborative cyber incident management/reporting system, utilized by European, inter-connected, critical infrastructure providers. Furthermore, Housen-Couriel (2018) analysed and compared the information sharing measures and modalities of the NIS and the IFC3, as well as some of the issues that emerge from this comparison of the two information exchange platforms. Wolff (2014) compared reporting policies in the US and the EU, and proposed templates for incident reporting, taking into consideration the nature of information, the timeline for sharing and the receiving parties. Albakri et al (2018) analyzed the risks in cyber incident information reporting, evaluating the kind of information contained in the reports and the specific risks associated with its disclosure. Moreover, Belsis et al (2005) suggested a federated information sharing model for organizations, where each organization could hold its own database of incidents, but with a centralized system in place, able to collect and correlate information stored on the various databases. Joyce et al (2016) presented various options for consideration, when creating a cyber incident reporting system, in order to foster cooperative cyber defense among participating international parties. Furthermore, Mtsweni et al (2016) proposed a semantic-enabled sharing model, for exchanging timely and relevant cybersecurity intelligence with trusted collaborators, while Kulikova et al (2012) proposed a decision-support framework for organizations, in order to help them decide on their incident reporting/disclosure strategy.

Koivunen (2010) compared the reporting recommendations provided in some information security standards, with the actual practice as observed through real-life incidents, and suggested that internet-connected organisations should adopt a rather agnostic approach to information security incident reporting. Harrison and White (2012) introduced a framework defining the information sharing requirements necessary for fast, effective, community cyber incident detection and response, and analysed a proof of concept implementation. In addition, Sveen et al (2007) examined how incident-reporting systems function, and particularly how the steady growth of high-priority incidents and the semi-exponential growth of low-priority incidents affect reporting effectiveness - while also examining how social pressures can affect incident reporting. Briggs et al (2017) also examined reporting from a social point of view: they examined message influences on incident reporting rate, and found that users were more likely to report a technical rather than a security problem, and also that users, were sometimes suspicious of messages reporting a security incident – believing that the message itself might be a cybersecurity attack. In another paper and in an attempt to standardize incident reporting, Ayres et al (2010), suggested the use of a key for the hierarchical classification of breaches, in order to improve consistency.

Research examining incident reporting in specific sectors, has also been published. For example, Gonzalez (2005) introduced a cyber security reporting system to share cyber security data based on features from Air Safety Reporting System. Lee (2017) suggested a security incident response framework for nuclear facilities, while also introducing cyber security incident reporting regulations at nuclear facilities in the Republic of Korea. Leszczyna and Wrobel (2014) proposed an approach to developing a data model for a security incident sharing platform for the smart grid (a new form of electricity network). Furthermore, Hennin (2008) suggested a standard protocol and data schema for the timely reporting of actual and potential cyber-attacks on industrial control systems. Grimaila et al (2009), suggested the key attributes of a cyber incident notification process, for use in military environments, to provide timely incident notifications, while Makori and Oenga (2010) proposed a security incident reporting model for adoption by novice users from developing countries, like Kenya. In addition, Grispos et al (2017), investigated the ability of employees in a Global Fortune 500 financial organization to recognize and report security incidents, while Chatzigeorgiou et al (2017), presented an architecture for ensuring privacy and confidentiality in incident reporting, taking into consideration the continuously increasing number of mobile devices. Moreover, there are studies examining the financial impact of incident reporting (Garg et al, 2003; Kulikova et al, 2015; Zafar et al, 2012; Spanos & Angelis, 2016 and more)

There is also literature examining automated incident reporting. Some researchers investigated various features of the automatic reporting of incidents (Cusick and Ma, 2010; Metzger et al, 2011; Koivunen, 2010; Kampanakis, 2014; Mtsweni et al, 2016) and others proposed automatic reporting systems, such as Makedon et al (2003), who presented an automatic incident classification and reporting system for information security incidents, called “SISC”. Marshall (2009) also proposed an incident modelling and reporting tool, for use in cyber incident preparedness exercises, called “CyberSMART”. Menges and Pernul (2018) proposed an incident reporting process model, based on a comparative analysis of identified, state-of-the-art, incident reporting formats, while Kacha (2014) proposed an extended taxonomies format for the automatic reporting of security events, based on previous work related to the "IDEA" taxonomy. In addition, Kurowski and Frings (2011), created a prototype for a documentation system for IT incidents, utilizing computational assistance, while Kijewski and Pawliński (2014) presented four tools and a generic analytics tool in order to sustain the process of automatic incident exchange between nodes. Moreover, Asada et al (2006), applied a semantic web approach to incident reporting, in order to enable computer processing of incident reports, while Husak and Cegan (2014), proposed a framework for the automatic detection and reporting of "phishing", a specific type of information security incident.

2.3. The blockchain technology

Blockchain is a peer-to-peer network that sits on top of the internet and was first introduced in 2008, as part of a proposal for "Bitcoin" – a virtual currency system - which is considered to be the first, ever, application of blockchain technology (Iansiti & Lakhani, 2017). Although the first applications of blockchain mainly dealt with various virtual/crypto currencies, a blockchain can serve multiple other purposes, with or without the involvement of any virtual/crypto currencies (Greenspan, 2015a). Blockchain technology is a distributed transaction database, in which different nodes operate as a system, to store sequences of bits that are encrypted as a single unit or block and then chained together (Lemieux, 2016). According to Casino et al (2019), blockchain introduced serious disruptions to traditional business processes, since applications and transactions which needed trusted third parties or centralized architectures for verification purposes, can now operate – with the same level of certainty - in a decentralized way. The impact of blockchain goes beyond the financial sector (Hughes et al, 2019), and encompasses any business that acts as or relies on an intermediary between two or more parties (Morkunas et al, 2019). The unique properties of blockchain make the technology an attractive idea for many areas of business, such as logistics, the pharmaceutical industry, record keeping, smart contracts, cyber security and more (Taylor et al, 2019). The inherent characteristics of blockchain architecture, besides decentralization, provide properties such as persistency, transparency, security, anonymity and auditability (Christidis & Devetsikiotis, 2016; Zheng et al, 2017).

2.3.1. How Blockchain works

In essence, blockchain is a chained data structure that combines blocks of data and information in a chronological order and records the blocks in encrypted form, as a distributed ledger that cannot be tampered with, or forged (Lu, 2019). Despite the many variations of blockchain networks, most of them use common core concepts (Yaga et al, 2018). According to Carlozo (2017), a blockchain database contains two types of records: transactions and blocks. The latter hold batches of transactions, which are time-stamped, linked to the previous block and cannot be retroactively altered (Carlozo, 2017). Transactions signify an “agreement” between participants, which may involve the transfer of assets, the completion of a task, or some other mutually accepted action (Casino et al, 2019). At least one participant digitally signs this transaction, and it is disseminated to neighboring blockchain nodes (Casino et al, 2019). Broadly speaking, a blockchain process is comprised of three steps: collecting new transactions and organizing them into blocks, cryptographically verifying each transaction in the block, and appending the new block to the blockchain (Prpić, 2017).

To get a better understanding of blockchain architecture, Casino et al (2019) describe blockchain as a a set of interconnected mechanisms which provide specific features to the infrastructure. At the first layer of this infrastructure, we have the signed transactions between nodes. Any entity connected to the blockchain is called a node. Nodes that verify all the blockchain rules are called full nodes and they are responsible for grouping the transactions into blocks and determining whether the transactions are valid, which is actually the purpose of the second layer, the “Consensus” layer (Casino et al, 2019). Different consensus mechanisms exist for various types of blockchains (Mingxiao et al., 2017). The “Compute Interface” layer, allows blockchains to offer more functionality, by having the ability of storing complex states, which are updated dynamically, using distributed computing. The “Governance” layer, outspreads the blockchain architecture to cover human interactions taking place in the physical world, since blockchain protocols are affected by various inputs from diverse groups of people who integrate new methods, improve the blockchain protocols and patch the system (Casino et al, 2019). The following figure depicts this architecture:

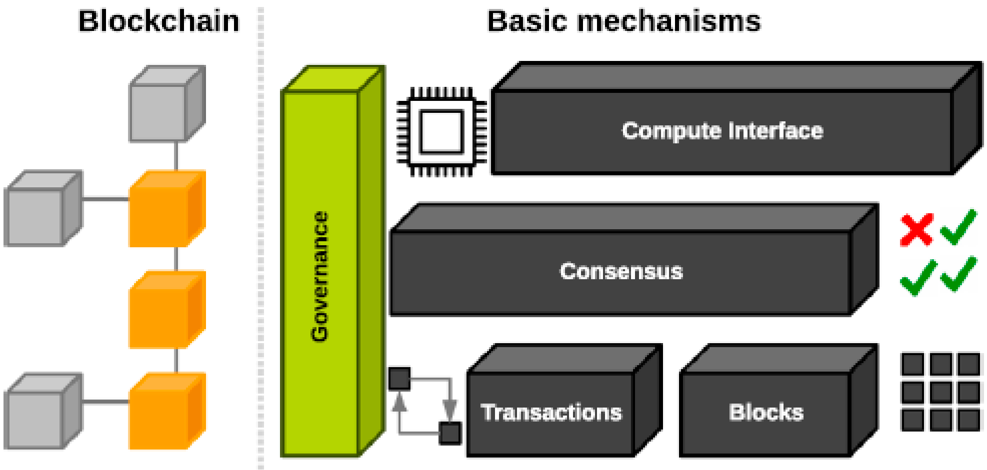


Figure 2.7. An overview of blockchain architecture by Casino et al (2019)

According to Zheng et al (2017), a block consists of the block header and the block body. The block header includes the block's version, the hash value of all the transactions in the block (Merkle), a timestamp, the threshold of a valid block hash (nBits), a nonce, and the parent block hash, a value that points to the previous block (if a previous block exists), whereas the block body includes a transaction counter and transactions (Zheng et al, 2017). The following figures depict a typical block's structure, as well as an example blockchain of a sequence of blocks:

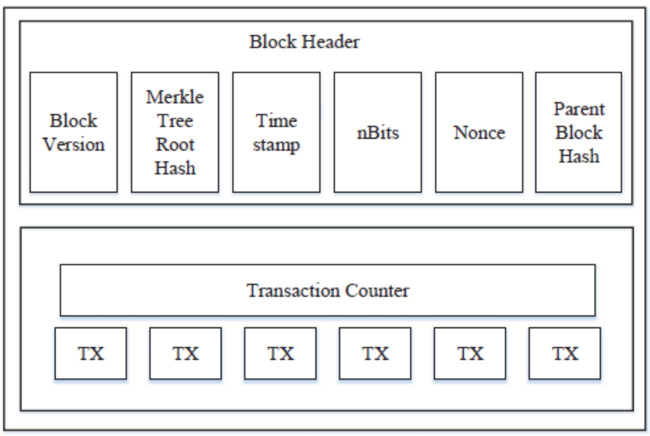


Figure 2.8. Block structure by Zheng et al (2017)

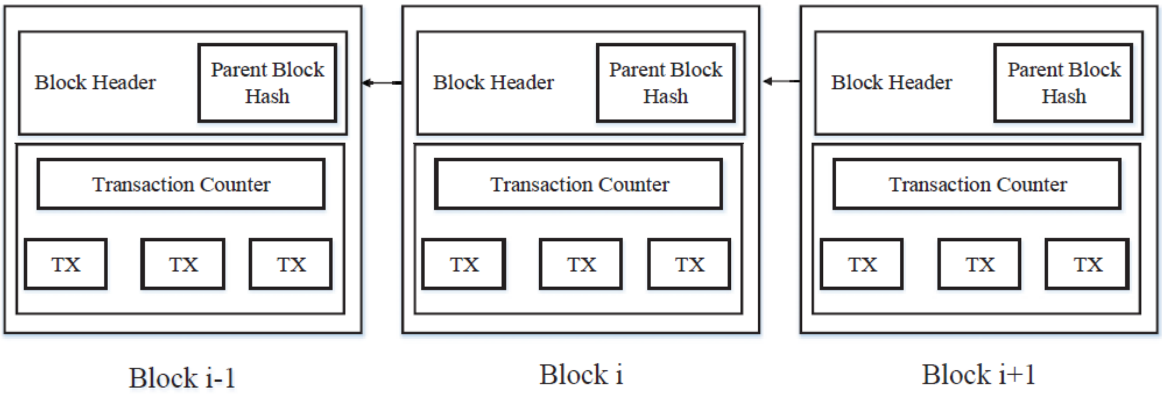


Figure 2.9. Example blockchain sequence of blocks by Zheng et al (2017)

Blockchain can also be thought as a table with three columns (where each row signifies a different transaction): the first column stores the transaction's timestamp, the second column stores the transaction's details, and the third column stores a hash of the current transaction, the transaction's details and the hash of the previous transaction (Di Pierro, 2017).

According to Casey and Vigna (2018), what makes blockchain "special", is that instead of the database being managed by a single centralized authority, transactions are stored in several copies, on several independent computers, within a decentralized network, with no single entity controlling this database and thus eliminating the need for a trusted authority (Christidis and Devetsikiotis, 2016; Tasca & Tessone, 2017). Changes can be made by any of the participating computers of the blockchain, but only by following rules dictated by a "consensus protocol," a mathematical algorithm that requires the majority of the other computers on the network to agree with the change (Casey & Vigna, 2018). Once "consensus" between the participants has been achieved, all participants update their copies of the chain simultaneously - while entries submitted without consensus are being rejected by the blockchain, thus ensuring that all copies of the distributed database are synchronized (Lipton, 2018). The following figure provides a high-level visualization of how blockchain works:

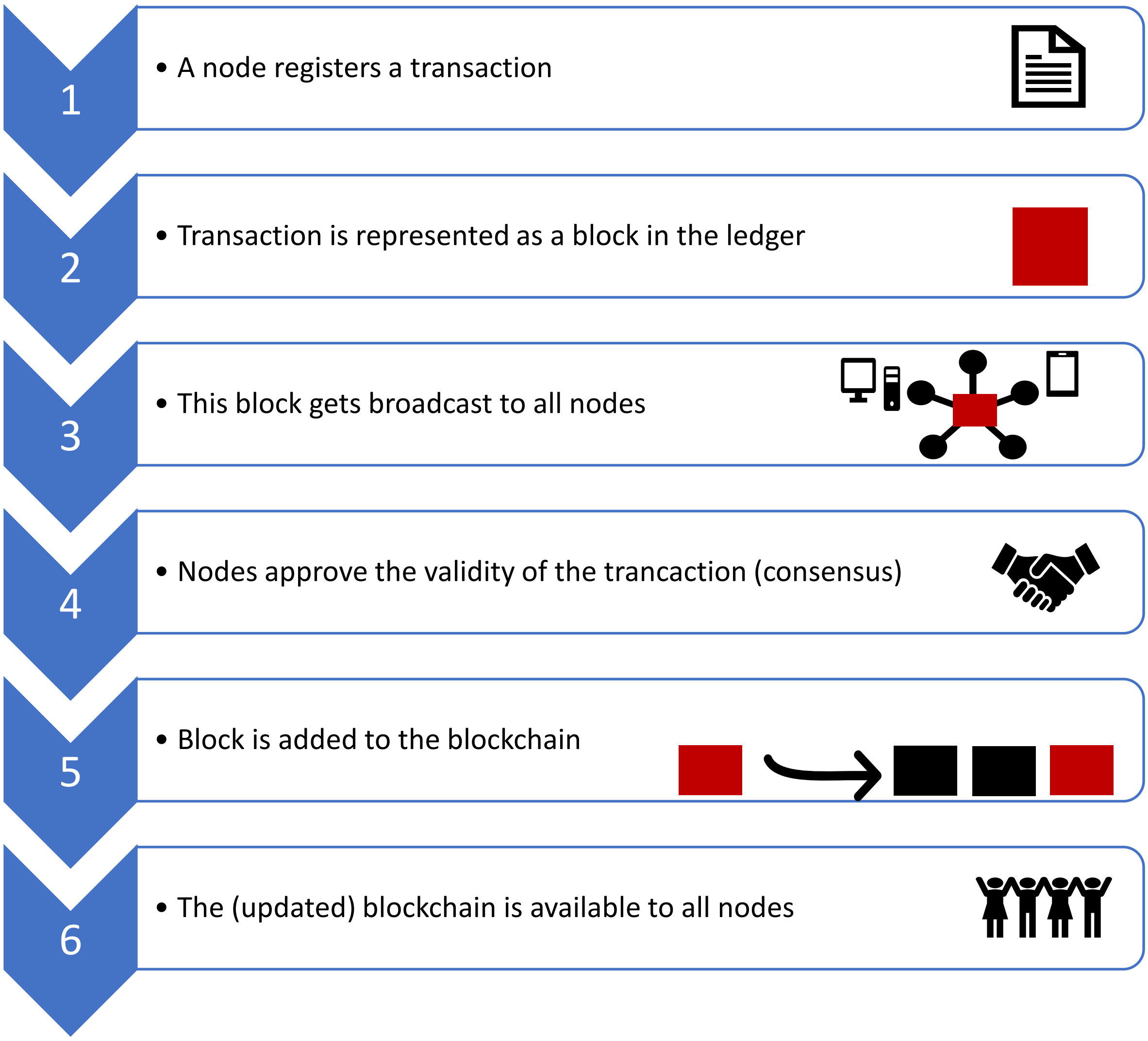


Figure 2.10. Visualization of a blockchain

Christidis & Devetsikiotis (2016) explain how the blockchain network works, in greater detail:

a) Users/nodes interact with the blockchain with a pair of private/public keys – users sign transactions with their private key, while their public key (or a hashed version of it) is used for addressing purposes on the network. According to the authors, the use of asymmetric cryptography brings authentication, integrity, and non-repudiation into the network and every signed transaction is broadcasted by a user's node to its neighboring nodes.

b) Neighboring nodes confirm the validity of the transaction (consensus) and discard invalid transactions (based on preconfigured rules of the network), whereas, eventually, the transaction is spread across the whole network.

c) Transactions collected and validated by the network, during a specific, time interval, are sorted into a time-stamped candidate block, which is then broadcasted back to the network.

d) The nodes verify that the candidate block contains valid transactions, and references (via hash) the correct previous block on their chain. If that is the case, the block is added to the chain, and its transactions applied – or else the block is discarded.

Thus, a blockchain system allows users to transact directly and securely, through public key cryptography, while also creating an immutable, publicly shared, publicly collected, and publicly verified (and verifiable) record of transactions in the process, through public key cryptography signatures (Prpić, 2017). Every transaction is time-stamped, verified, added in sequence, and made public; and every blockchain node maintains, and can access, a copy of the entire history of a Blockchain, while knowing that this history cannot be altered, except through new transactions (Reijers et al, 2016).

2.3.2. Public, private and hybrid blockchains

According to Mougayar (2016), blockchains – depending on their application – can be classified as public, private, or hybrid. According to the same author, public (or *permission-less*) blockchains are visible by anyone, have no single owner, are fully decentralized, and their consensus process/protocol is open to all nodes for participation. Some well-known implementations of public blockchains are Bitcoin, Ethereum, Litecoin and, in general, most virtual/crypto currencies (Haferkorn & Diaz, 2014). Private (or *permissioned)* blockchains, on the other hand, do have a central authority, usually include a whitelist of allowed nodes with particular characteristics and permissions over the network (Casino et al, 2019), their transactions are editable - as long as their participants have reached an agreement - and their consensus process/protocol need not be as strong as that of a public blockchain (Vranken, 2017). The various, available, consensus protocols are presented in section “2.3.5”. In private blockchains, main applications include database management, auditing and, in general, performance demanding solutions (Zheng et al, 2016). An example of a platform for building and deploying private blockchains is Multichain (Greenspan, 2015b) – although private blockchains are also supported by other platforms, such as Ethereum. A hybrid (or *consortium/federated*) blockchain is partially decentralized, where transactions are validated by a selected set of nodes (Vranken, 2017). A hybrid blockchain can be viewed as a combination of public and private blockchains solutions (Zheng et al, 2016), which allows a group of trusted nodes (entities) for the validation of actions, rather than having a single owner (Mougayar, 2016). This allows for a partially decentralized design, where “leader” nodes can grant permissions to other nodes (Casino et al, 2019). This type of blockchains are popular in the industry and banking sectors and examples include the Hyperledger project and Corda’s R3 (Casino et al, 2019).

2.3.3. Blockchain variety

No discussion of blockchain is practical without referring to “Bitcoin” (Sultan et al, 2018), since blockchain technology is the name given to the design underpinning the operation of the particular virtual currency (Ammous, 2016). In fact, Bitcoin’s creator, never explicitly mentioned “blockchain” in his paper, but rather a software design based on several current technologies, in order to create a “purely peer-to-peer version of electronic cash” (Ammous, 2016, p.1). In his paper, Nakamoto (2008) describes Bitcoin as a mean to achieve direct online payments between parties, without the need of an intermediary, such as a bank. It is crucial to note, that while Bitcoin refers to the Bitcoin protocol and the Bitcoin’s peer-to-peer network of participating nodes, people tend to use it to refer to the native token of the transaction, the Bitcoin token – “BTC” (Shermin, 2017).

Although the Bitcoin blockchain is considered to be the first application of the Blockchain technology (Iansiti & Lakhani, 2017), various alternatives have emerged since. Their philosophy and underlying core functionality, however, remains the same: a blockchain is a decentralized, peer-to-peer chain of blocks, each cryptographically linked to the previous, using a hash digest (Sultan et al, 2018). Cryptocurrencies can be considered to have emerged as the first generation of blockchain technology (Alharby & van Moorsel, 2017) and they represent a considerable percentage of the existing blockchain networks (Casino et al, 2019). Cryptocurrency can be described as “a virtual coinage system that functions much like a standard currency, enabling users to provide virtual payment for goods and services, free of a central trusted authority” (Farell, 2015). According to Alharby & van Moorsel (2017), other blockchains such as the “Ethereum” blockchain, have emerged as the second generation of blockchain, which support functionality for building complex distributed applications. This functionality is often referred to as “smart contracts”, which is basically executable code, that runs on the blockchain, in order to facilitate, execute and enforce the terms of an agreement/contract (Xu et al, 2016). Besides Bitcoin and Ethereum, other blockchain examples include “Ripple” - which although shares most of the properties of Ethereum and Bitcoin, it was specifically designed “to facilitate fast and cheap global transfer of money” (Schwartz et al, 2014), “Fabric” by “Hyperledger” - a consortium formed by the Linux foundation, and many other partners such as IBM, Intel, SAP, Cisco, Daimler, and American Express, to design and develop enterprise blockchains (Androulaki et al, 2018), R3’s “Corda” - a distributed ledger platform for recording and processing financial agreements (Brown et al, 2016) and “Multichain” - an open source blockchain platform that enables the setup, configuration, and deployment of a private, public, or hybrid blockchain (Greenspan, 2015c).

2.3.4. Blockchain evolution

Blockchain technology has undergone rapid incremental evolutions, since its debut in 2008; the technology initially was not programmable, but later versions incorporated such functionality, which consequently expanded the scope for general market decentralization (Angelis & Da Silva, 2019). The different versions of blockchain technology can be characterized by the following three stages (Swan, 2015; Zhao et al, 2016; Chen et al, 2018; Angelis & Da Silva, 2019):

a) Blockchain 1.0.

The first version of blockchain is focused on transactions, with the deployment of virtual/crypto currencies in applications related to cash management, such as transfers and digital payment systems (Swan, 2015; Luu et al, 2016; Zhao et al, 2016), with “Bitcoin” prevailing as the most well-known example (Angelis & Da Silva, 2019).

b) Blockchain 2.0.

Blockchain 2.0, an extension of Blockchain 1.0, is acknowledged as a generally programmable infrastructure, with the ability of recording computational results (Xu et al, 2016), and including features such as privacy, smart contracts, as well as the emergence of non-native, asset, blockchain tokens and capabilities (Schuster, 2018). Well-known examples of platforms with the ability of running smart contracts are Ethereum (Buterin, 2018), IBM-Maersk partnered blockchain, supporting global shipping (Kamath, 2018) and the trade finance blockchain consortium “we.trade” (Morris, 2018).

b) Blockchain 3.0.

Blockchain 3.0 further expands the technology’s capabilities with the introduction of “decentralized applications - DApps” (Angelis & Da Silva, 2019). According to the same authors, a DApp consists of back-end code that runs on a decentralized peer-to-peer network, connecting users and providers directly (via a front-end interface), and is implemented on decentralized blockchains using cryptographic tokens. According to Raval (2016), DApps should be designed to be transparent, flexible and resilient, and should have a clear structure.

In addition to the above three versions, Angelis and Da Silva (2019) indicate the emergence of a newer version/stage of blockchain technology, “Blockchain 4.0”. This new version involves the inclusion of artificial intelligence (AI) to blockchain technologies, and is particularly useful in environments in which traceability and immutability are of high importance (Angelis & Da Silva, 2019). According to the same authors, the benefits of both worlds can be combined, as Artificial Intelligence allows computers to learn from data, while blockchain provides data accuracy, which is useful for feeding data into the AI system and for recording its outputs.

2.3.5. Blockchain consensus algorithms

Blockchain is updated via the consensus protocol, which ensures a common, unambiguous ordering of blocks and transactions, while also guaranteeing the integrity and consistency of the ledger across geographically distributed nodes (Baliga, 2017). Since “consensus” literally means “agreement”, consensus algorithms are those algorithms that help a decentralized network to unanimously take a decision, whenever necessary (Sankar et al, 2017). Blockchain nodes achieve consensus by using the prior agreement of the blockchain rules and following the principle of majority dominance (Mingxiao et al, 2017). Achieving consensus in a decentralized system is a challenging task, as consensus algorithms need to be resilient to any failures of nodes, to corrupted messages, messages reaching out of order or general message delays, and to the partitioning of the network; while also capably handling “selfish” or deliberately malicious nodes (Baliga, 2017). Their features include ensuring decentralized governance, minimum structure, performance, integrity and authentication, as well as non-repudiation and byzantine fault tolerance (Seibold & Samman, 2016). Since the consensus mechanism preserves the sanctity of the blockchain’s data, a poor choice of a consensus algorithm can render the blockchain platform useless (Baliga, 2017). There are several algorithms available for a blockchain implementation project, with each algorithm making the required set of assumptions, in terms of performance, synchrony, message broadcasts and security, as well as handling of failures and malicious nodes. (Baliga, 2017). The most widely adopted algorithm is the Proof of Work (PoW), however, there are numerous others (Bach et al, 2018). A short description of the properties of the most dominant algorithms (Bach et al, 2018), both in permissioned and permission-less blockchains, follows:

a) Proof of Work (PoW)

In both Bitcoin and Ethereum implementations, PoW is the consensus algorithm in use. In both implementations, the core idea is the same: participating nodes must calculate the solution of a difficult mathematical problem - based on information obtained through the previous block - and the first participant that solves the problem can create the next block - a process also known as "mining" (Mingxiao et al, 2017). PoW algorithms have received heavy criticism due to their time-consuming processes and power-intensiveness (Baliga, 2017).

b) Proof of Stake (PoS)

PoS algorithms attempt to overcome the disadvantages of PoW algorithms, in terms of intensiveness and associated power consumption (Baliga, 2017). PoS do not utilize a mining process, but adopt a rather alternative approach, which involves a user's stake or ownership of virtual currency in the blockchain (Baliga, 2017). The concept of "coin age" is used, where the longer a node holds the coins, the more rights it can get on the blockchain (Mingxiao et al, 2017). PoS, therefore, encourages participants to hold their currencies and the blockchain is not entirely relying on a proof of work process (Baliga, 2017).

c) Ripple Protocol consensus algorithm (RPCA)

RPCA is a low-latency Byzantine agreement protocol, capable of achieving consensus without the complete agreement of participating nodes (Chase & MacBrough, 2018). Each server places all valid transactions to a public list and then votes on the veracity of each transaction, in a series of one or multiple rounds; all transactions with a minimum of 80% positive votes, are eventually recorded to the ledger (Bach et al, 2018).

d) Byzantine Fault Tolerance (BFT) and variants

In permissioned blockchains, where the environment is considered to be more confined and trusted, blockchains tend to rely on message-based consensus schemas, rather than hashing procedures, which are lighter and considerably speed up the consensus process (De Angelis et al, 2018). In these settings, Byzantine fault tolerant (BFT) algorithms, such as the Practical BFT (PBFT) and Proof of Authority (PoA), prevail (De Angelis et al, 2018). The PBFT algorithm is based on the assumption that less than one-third of the nodes are faulty (f), which means that the network should consist of at least $n = 3f + 1$ nodes to tolerate f faulty nodes (Castro & Liskov, 2002). Thus $f = [(n - 1)/3]$ and the network requires $2f + 1$ peers to agree on the block of transactions (Sukhwani et al, 2017). The PoA algorithm, differently from PBFT, has drawn attention due to the fact that it requires less message exchanges, and thus provides better performance and fault-tolerance (Dinh et al, 2017).

e) Other consensus algorithms

As previously mentioned, there is no shortage of different consensus algorithms, for both permissioned and permission-less blockchains. Based on their requirements and intended use, blockchain developers can select from a wide variety of algorithms for blockchain implementations, which include, in addition to those described above, the Delegated Proof-of-Stake (DPoS) algorithm, the Proof-of-Capacity (PoC or Proof-of-Storage), Proof-of-Existence (PoE), Proof-of-Importance (PoI), Proof-of-Burn (PoB), Proof-of-Validation (PoV), Round Robin (RR), Proof-of Elapsed Time (PoET), and various others (Mattila, 2016).

2.3.6. Blockchain suitability

Although blockchain technology is becoming increasingly relevant to real-world applications (Zhao et al, 2016), its use is not a silver bullet (Yaga et al, 2018). It is, essentially, a novel way to manage data, and it therefore competes with the traditional, established, data-management systems, such as relational databases (Peck, 2017). While Swan (2015) predicts that blockchain will lead to a great wave of disruption, as it extends to the various segments of economy, general blockchain development and acceptance is still in the early stages, and its overall impact, as a disruptive new technology, is still to be seen (Pilkington, 2016). Blockchain should therefore not be considered as panacea, or as a universally applicable solution, to every possible problem. Yaga et al (2018, p.vi) likewise argue that blockchain technology is new and should not be treated with the mindset of "how can we make our problem fit into the blockchain technology paradigm?", but rather with the mindset of "how could blockchain technology potentially benefit us?".

2.3.6.1. Advantages of blockchain technology

The use of blockchain comes with plenty of benefits, which can bring cost savings to organizations, as well as improve overall efficiency (Zheng et al, 2017):

a) Decentralization

While in a centralized database storage device are all connected to a common processor, in a distributed database, they are independent (Lipton, 2018). According to Lu (2019), information in blockchain networks is automatically shared and distributed between nodes - subject to the fulfilment of specific conditions - without any third-party intervention, while all participants could potentially join transactions and activities. For public blockchains, there is no integration point or central authority required to set rules, or approve transactions, and no single point of trust (Tasca & Tessone, 2017). In centralized environments, each transaction needs to be validated through a central trusted agency – in blockchain, consensus algorithms are utilized to maintain data consistency (Zheng et al, 2017). The failure of a blockchain node does not affect the operation of the whole network, thus ensuring the resilience, availability and reliability of applications built on blockchain, by avoiding single points of failure (Zheng et al, 2017; Chen et al, 2018; Gatteschi et al, 2018). Both public and private blockchain implementations are used to eliminate single sources of failure (Taylor et al, 2019). Nonetheless, traditional, centralized, database environments can also cope with single points of failure, through the utilization of redundancy mechanisms and fail-over systems. However, by design, blockchain keeps all nodes updated with the current content. Each node holds a single copy of the database. To achieve the same level of redundancy with traditional databases, a large number of redundancy mechanisms would have to be employed, equivalent to the number of nodes a blockchain would have.

b) Anonymity

Each user interacts with blockchain through a generated address (a public key or a hash of it), which does not reveal the explicit identity of the user (Zheng et al, 2017) - although it is pseudonymous in nature, rather than truly anonymous. This is presented as a disadvantage as well, in later sections, although solutions exist to increase user privacy (Zheng et al, 2017). Blockchain uses asymmetric encryption in the form of data encryption and digital signatures – the former ensures the security of transaction data and reduces the risk of losing or falsifying transaction data, while the latter is used to digitally sign transactions (Lu, 2019). It is unnecessary to disclose the true identity of the user associated with the node, which is a controversial feature, as it may sometimes assist illegal activities (Reid & Harrigan, 2013; Narayanan et al, 2016).

c) Transparency

Blockchain records are auditable by a predefined set of participants, albeit the set can be more (public blockchain) or less (private blockchain), open (Tasca & Tessone, 2017). The blockchain technology ensures that nodes record and transfer records on the network, and all participants can query these records, which makes information in the decentralized network both consistent and transparent (Lu, 2019). Each node can not only read the final state of the transactions, but also the history of the previous transactional states (Gatteschi et al, 2018), while each participant has the same permissions and obligations to access records, and also allow other nodes – on the same network - to access this data (Bonneau et al, 2015; Lin & Liao, 2017). Consensus mechanisms implemented in blockchain structures enable multiple writers to modify the database, and provide an authoritative transaction log, in which all nodes provably agree (Casino et al, 2019).

d) Immutability/Security

Blockchain is a shared, tamper-proof, replicated, ledger, where records are irreversible and cannot be forged, because of the use of one-way cryptographic hash functions (Tasca & Tessone, 2017; Chen et al, 2018; Gatteschi et al, 2018). A newly generated block strictly follows the linear sequence of time (Chen et al, 2018), with applied timestamps allowing the node to both keep the order of transactions, as well as to create traceable data - therefore not only guaranteeing the originality of the data, but also reducing the cost of transaction traceability (Lu, 2019). Transactions need to be reviewed by most of the nodes of the system, before they can be recorded (Lu, 2019) – but once data has been recorded in the ledger, it cannot be modified without letting the whole network know, thus permitting tamper-resistant data (Zheng et al, 2017). Blocks that contain invalid transactions can be discovered immediately (Zheng et al, 2017). Users can transfer data only if they possess a private key, which is used to generate a signature for each transaction a user sends out, which is, in turn, used to confirm both the origin and integrity of the transaction (Tasca & Tessone, 2017). Theoretically, blockchain immutability could be violated, if attackers could gather enough resources to outpace the block creation rate of the rest of the blockchain network, an attack called the “51% attack” – however this attack is not only very difficult to conduct, but is also only applicable to public blockchains, as private blockchains could remove malicious nodes from the network (Yaga et al, 2018). Furthermore, Zyskind and Nathan (2015) state, that compared to decentralized structures, centralized databases are generally more vulnerable to malicious attacks.

e) Trust

According to Casino et al (2019), blockchain avoids the use of trusted third parties - on which centralized databases rely on – and therefore enhances reliability and verifiability of contents. Unlike centralized trust we take for granted (e.g. central banks issuing currencies), the blockchain network acts as trust bearer with decentralized ledgers (Underwood, 2016). Blockchain provides trust between participants, since digital signatures ensure that every node behaves appropriately, without needing intermediaries (Gatteschi et al, 2018). Trust is a factor that plagues participants in trading, and blockchain employs hash functions and consensus protocols to solve any trust issues (Chen et al, 2018). According to Lu (2019), participants don't need to take care of mutual trust relationships in the blockchain system – the network does it for them.

f) Efficiency

According to Chen et al (2018), since all blockchain data are run automatically through pre-set procedures, blockchain technology can improve efficiency and significantly reduce the cost of labor. It can speed up the clearing and settlement of transactions, by reducing (or eliminating) the number of intermediaries involved (Chen et al, 2018), and it can also make reconciliation processes faster (Wang et al, 2016). In addition, since blockchain does not require hosting, due to its decentralized structure, it can provide significant cost reductions to organizations (Casino et al, 2019).

g) Automation

Blockchain technology allows – without the need for any human interaction – conflicting or double transactions not to be permanently written on the blockchain, since any conflicts are automatically reconciled, and valid transactions are only written once (Tasca & Tessone, 2017). Furthermore, various blockchains support the development and deployment of “smart contracts”, a series of commitments defined in digital form, which contain execution conditions and execution logic – with the logic automatically executed as soon as the condition is met (Lu, 2019). The payoff of these commitments depends on the use of algorithms, which are self-executable, self-enforceable, self-verifiable and self-constrained. (Tasca & Tessone, 2017). Smart contracts are the core technology behind the evolution of “Blockchain 2.0” (Tasca &

Tessone, 2017), and also have the capability of processing data, operating asset transactions, and managing smart assets (Luu et al, 2016).

2.3.6.2. Disadvantages of blockchain technology

According to Christidis and Devetsikiotis (2016), although blockchain technology brings many advantages to the table, it also comes with disadvantages:

a) Performance and storage

Due to its decentralized nature, a blockchain solution will generally underperform, compared to a properly configured centralized database, resulting in higher latencies and lower transaction processing throughput. (Christidis & Devetsikiotis, 2016). Performance is, of course, heavily dependent on the consensus algorithm employed by the blockchain, but for major public blockchains such as Bitcoin and Ethereum, a maximum of seven (Bitcoin) and around twenty (Ethereum) transactions per second (Peck, 2017), means adding information to the ledger is slow: creating a Bitcoin block of transactions takes from ten to sixty minutes, while Ethereum needs around 15 seconds (Gatteschi et al, 2018). Storage space on the blockchain can be used both for storage and exchange of arbitrary data structures, with the storage of data having some size limitations: the maximum block size in Bitcoin is 1MB, whereas Ethereum block size depends on the complexity of contracts being run, but is usually under 2KB in size (Zheng et al, 2017). Furthermore, with the amount of transactions increasing every day, data replication requires considerable space for blockchain nodes who have to locally store all transaction history: about 105GB for Bitcoin and 70GB for Ethereum (Gatteschi et al, 2018), and these numbers keep growing every day. All in all, if performance is an issue, or a vast amount of data needs to be saved as part of an implementation, centralized databases are considerably a better choice than blockchain implementations (Gatteschi et al, 2018).

b) True anonymity

According to Tasca and Tessone (2017), a common misunderstanding of the anonymity level provided in blockchain, is that the majority of users do not distinguish between anonymity and the pseudo-anonymity. Blockchain can, indeed, provide a certain amount of privacy for its users, since for each transaction a user conducts, only his public key – or a hash of it - is revealed (Zheng et al, 2017). However, by analyzing publicly available data (on-chain and off-chain) patterns and connections created between addresses can be identified, allowing an interested party to make informed inferences about the user's actual identity (Meiklejohn et al, 2013; Ron & Shamir, 2013; Ermilov et al, 2017; Zheng et al, 2017). Ways for increasing a user's privacy in blockchain have been proposed, such as "mixing", a kind of service which provides anonymity by transferring funds from multiple input addresses to multiple output addresses (Bonneau et al, 2014; Zheng et al, 2017), "transaction remote release", which hides the IP address of the transaction's author (ShenTu & Yu, 2015) and using "ZeroCoin" (Miers et al, 2013) or "ZeroCash" (Sasson et al, 2014) blockchain implementations, which utilize "zero-knowledge proof" and "zero-knowledge succinct non-interactive arguments of knowledge" protocols, respectively (Zheng et al, 2017).

c) Perpetuity

As far as blockchain and smart contracts are concerned, "what's done, is done". Since following their creation smart contracts become autonomous entities (Christidis & Devetsikiotis, 2016; Gatteschi et al, 2018), unless specific provisions have been included during the contract's creation, the contract can never be modified (Christidis & Devetsikiotis, 2016). Therefore, before deploying a smart contract, one should inspect it carefully and include fail-safe mechanisms (Christidis & Devetsikiotis, 2016). If code bugs are identified after deployment, new contracts have to be created by developers, and all data and pointers should be transferred from the old to the new contracts (Gatteschi et al, 2018).

d) Power-intensiveness

Due to their inherent characteristics, blockchain implementations utilizing consensus algorithms such as “Proof of Work -PoW” require expensive hardware, while most of the computing power is wasted (Gatteschi et al, 2018). Utilizing an alternative consensus algorithm in a blockchain implementation, such as “Proof of Stake – PoS” or “Practical Byzantine Fault Tolerance – PBFT” significantly reduces the power/cost involved (Lin & Liao, 2017).

e) Limited understanding and support

As evident in the next sections of this report, although more and more blockchain implementation projects/applications gradually find their way into the market, blockchain is still a very new technology. While its impact is predicted to be enormous, the process of its adoption is not expected to be sudden, but rather gradual and steady; it is expected to take decades for it to find its way into our economic and social infrastructure (Iansiti & Lakhani, 2017). It therefore comes as no surprise, that the technology is currently understood just by a few, and support is provided by even fewer. Moreover, the variation of blockchain designs and all possible configurations represent a burden for blockchain developers and software architects, making it difficult to measure and compare the performance and quality of different blockchain implementations (Tasca & Tessone, 2017).

2.3.6.3. Blockchain decision models

While the potential of blockchain technology is clear, what is not so clear, is the kind of innovation blockchain is: is it simply a new technology, a small step in the wider context of innovation, or can it be the next General-purpose Technology? (Kane, 2017). Although businesses around the world are excited about this new technology and its potential in solving real-world problems (Umeh, 2016), blockchain utilization/implementation decisions should be taken with care, as this is not a “one-size-fits-all” technology. According to Casino et al (2019), several decision-makers and developers around the globe visualize using blockchain in almost every project; what they fail to understand, however, are the fundamental reasons for using it, especially from a data management perspective. For example, blockchain is appropriate when parties require transactions between trustless sources or a permanent historical record (Casino et al, 2019); it is not appropriate in cases where just a single writer in a given system is foreseen, where a centralized database would be a far better option, particularly from a performance perspective (Greenspan, 2015b). It is, indeed, one of the most common critics raised to blockchain, that many existing blockchain applications could be better implemented using traditional technologies, such as centralized databases (Gatteschi et al, 2018).

Therefore, and in order to aid decision-makers, a number of “blockchain decision models” have been proposed, to help examine whether blockchain technology could be an appropriate fit for a project:

a) Decision model by Peck (2017)

Peck (2017) proposed a yes/no flowchart, with a total number of seven questions, with the first question prompting an individual to consider whether “traditional technologies can meet their needs” and leading to three, different, final options: “No need for a blockchain”, “might need a permissioned blockchain” or “might need a public blockchain”.

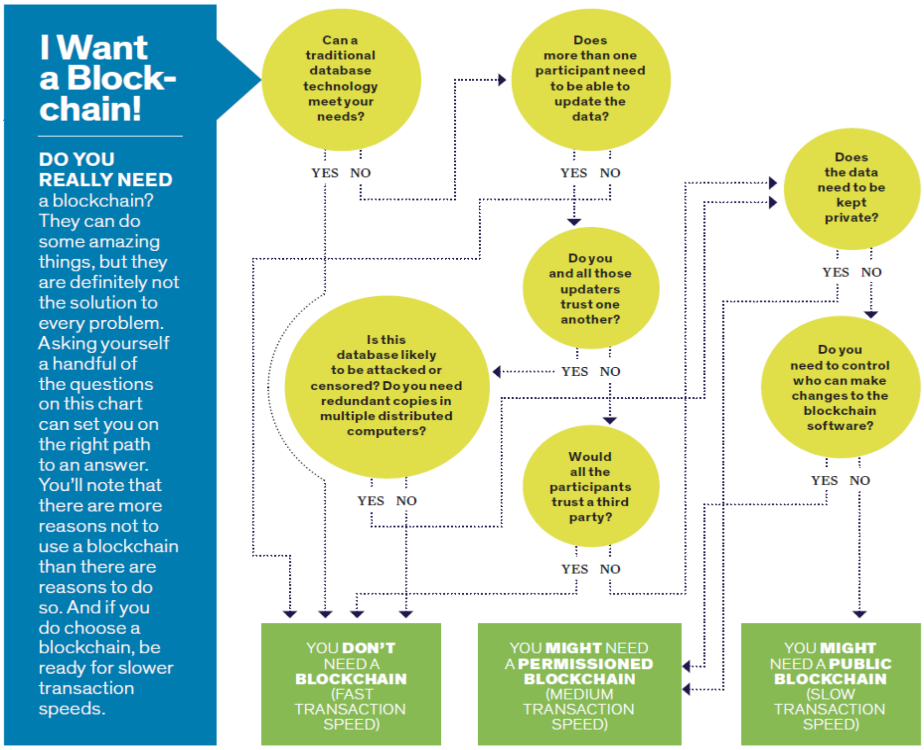


Figure 2.11. Blockchain decision model by Peck (2017)

b) Decision model by Wust and Gervais (2018)

Along the same lines, Wust and Gervais (2018) proposed a similar yes/no flowchart, with a total number of six questions, with the first question prompting an individual to consider whether "they need to store state" and leading to four, different, final options: "permission-less blockchain", "public permissioned blockchain", "private permissioned blockchain" and "don't use blockchain".

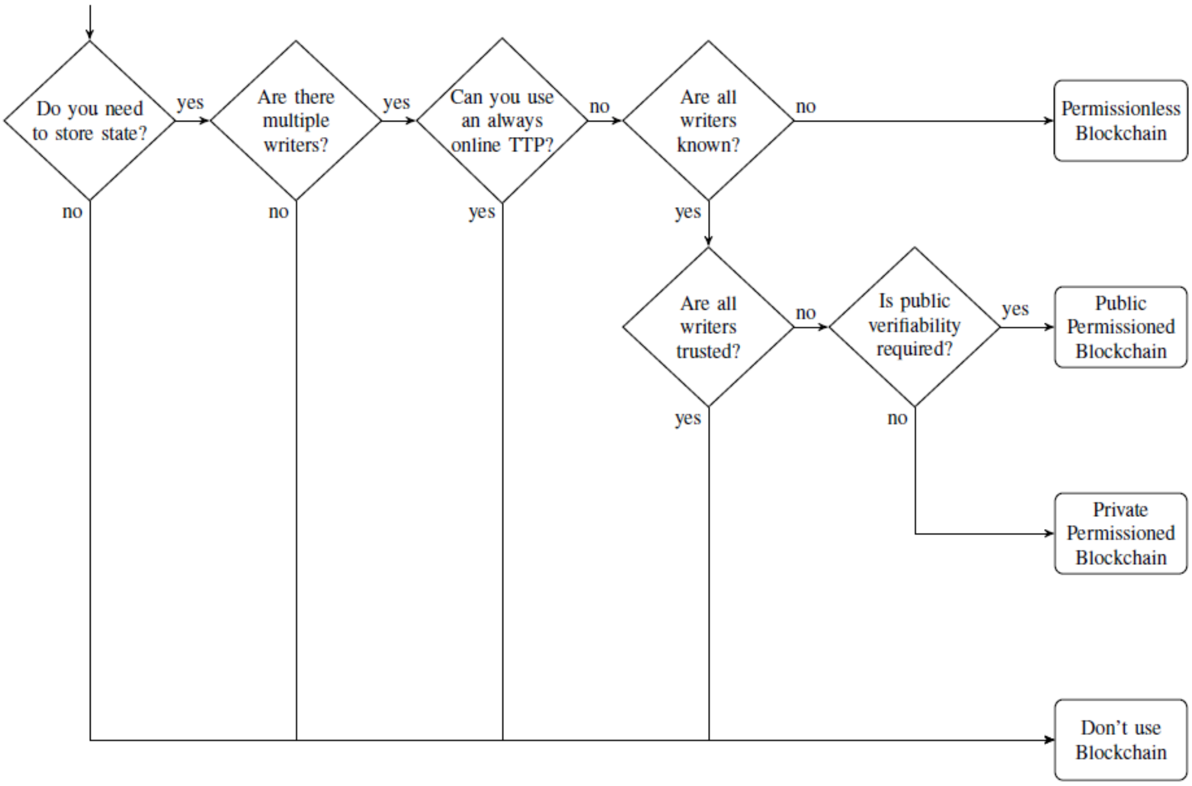


Figure 2.12. Blockchain decision model by Wust and Gervais (2018)

c) Decision model by DHS in Yaga et al (2018)

The US Department of Homeland Security (DHS) proposed a decision model which has been adopted by NIST, in their "Blockchain Technology Overview" publication, by Yaga et al (2018). This yes/no flowchart is somewhat different from the other, as it does not differentiate between the possible types of blockchain implementations, but rather generically indicates that "you may have a blockchain use case", should all outcomes to the six available questions suggest "yes", as an answer. If "no" is provided as an answer - to any stated question - suitable alternative options are suggested, such as the use of “email/spreadsheets”, “encrypted database”, “managed database” or simply, “database”.

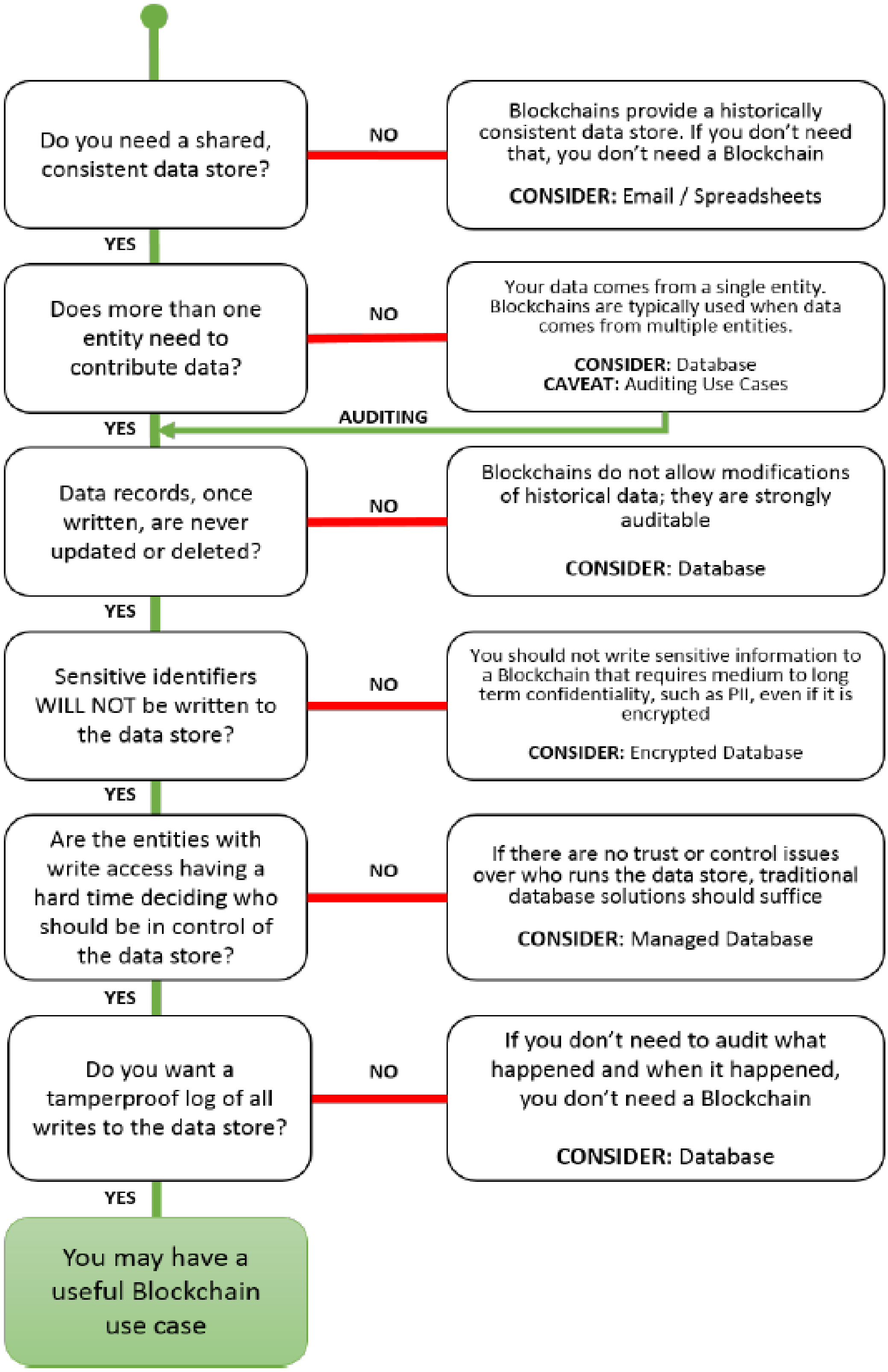


Figure 2.13. Blockchain decision model by DHS in Yaga et al (2018)

d) Other models

There is no shortage of proposed blockchain decision models. Koens and Poll (2018) literally identified thirty of them, available both in literature and on the web, before proposing their own model. The same authors identified inconsistencies between the different decision models, where same decisions could lead to different outcomes, or, conversely, similar outcomes could be reached with opposing decisions (Koens & Poll, 2018). In general, however, most of the schemes do have one thing in common: they take a critical view upon the utilization of blockchain technology for new developments, with most chart options leading to suggestions for the use of alternative/traditional technologies, rather than blockchain. According to Yaga et al (2018), this critical stand on blockchain technology is the appropriate one for organizations to take; they should first examine whether existing technology can provide a better solution to their problem. Along the same lines, it seems that most of the current advise surrounding blockchain technology is to thoroughly investigate its appropriateness, and not just use it because it is new and exciting (Yaga et al, 2018).

2.3.7. Blockchain applications

According to Yli-Huumo et al (2016), interest in Blockchain technology has been increasing since the idea was developed in 2008. In a literature review, regarding the on-going research activity on Blockchain, conducted by the aforementioned authors in 2016, they identified that the majority of the current research on Blockchain is focused on finding and identifying improvements to the current challenges and limitations in Blockchain, with a focus on security and privacy issues. Other research topics identified included wasted resources, computational power and usability, while their findings in this mapping study showed that the majority of research was conducted in the Bitcoin environment. In a paper by Gatteschi et al (2018), the existing blockchain applications identified in different business sectors are described, with various applications found in identity management, intellectual property rights, verifiable voting systems for the government, ownership of diamonds and others. The most recent literature review on multisectoral blockchain applications, is the one conducted by Casino et al (2019), identifying 260 distinct blockchain-application-related research items, over a period of four years (2014-2018), in 11 different domains: “financial applications”, “integrity verification”, “governance” (including “citizenship and user services”, “public sector”, and “voting” subsections), “Internet of Things”, “healthcare management”, “privacy and security”, “business and industrial applications”, “supply chain management”, “energy sector”, “education”, “data management” and other “miscellaneous applications”. The following figure provides an overwhelming visualization of the disparity of all the various blockchain applications:

Figure 2.14. Different types of blockchain applications in Casino et al (2019)

2.3.7.1. Blockchain applications and data management

Regarding utilizing blockchain in various data management solutions, including recording/reporting, examples include Garcia-Barriocanal et al. (2017) proposing a blockchain-based solution for metadata - supporting key functions towards the management and sustainability of digital archives. Also, Yang et al. (2018), proposed a blockchain-based, big data, sharing model, for the safe circulation of data resources. Do and Ng (2017) proposed a system that enables distributed client data management in a secure manner, using cryptographic primitives, with the owner able to grant search and read permissions of data to third parties. Moreover, Jiang et al (2017) proposed “Searchain”, a blockchain-based, keyword, search system with efficient private search of keywords, in decentralized storage. Cebe et al (2018) proposed constructing a blockchain infrastructure to provide forensic services for accident investigations, with the ledger recording data related to vehicles, such as maintenance information/history, car diagnosis reports and more. Furthermore, Goharshay et al (2018) proposed an approach for maintaining and reporting credit history records on the Blockchain, while Kavassalis et al (2017) proposed a financial risk reporting application, based on distributed computing and decentralised data management technologies. Other examples include the proposal by Lemieux (2016), who presented a blockchain-based solution for creating and preserving trustworthy digital records, for use by civil registries of births, deaths and marriages, land registries, repositories of financial transactions and others, as well as the “Vizsafe” platform, which enables users to upload incident reports about physical security threats or faulty infrastructure on a decentralized ledger, within the broader concept of a “smart city” (Mottur & Whittaker, 2018).

2.3.7.2. Blockchain applications and information security

Regarding utilizing blockchain in cyber/information security solutions, a variety of proposals can be found in literature, including a proposal by Fan et al (2017) for utilizing blockchain to enhance security and reliability in distributed networks, an anti-malware, blockchain-based, solution by Noyes (2016), a proposal for a privacy-aware public key infrastructure for protecting against single points of failure and other malicious attacks (Axon, 2015), and a proposal for using a blockchain-based protection framework for enhancing the security of power systems against cyber-attacks (Liang et al, 2018). In addition, Rodrigues et al (2017), proposed a blockchain-based architecture for a DDoS mitigation solution, while methods for improving anonymity in blockchains have also been proposed (Moser, 2013; Zheng et al, 2017). Furthermore, Schackelford & Myers (2016), analysed the potential impact of blockchain technology on advancing cybersecurity, with a particular focus on certificate authorities and the critical infrastructure context. Moreover, a recent literature review about blockchain applications in the area of cyber-security, by Taylor et al (2019), identified 30 distinct papers, in various categories, such as Internet of Things (IoT), web applications, networks and machine visualization, public key cryptography, certification schemes and the secure storage of personally identifiable information (PII). However, the authors state that a sizeable portion of the identified primary studies are either experimental proposals, or theoretical concepts, with limited practicality in solving real-world problems (Taylor et al, 2019).

2.3.7.3. Blockchain applications and incident reporting

Regarding, specifically, information security incident reporting and the blockchain, the available literature is very limited. A relevant work regarding blockchain and incident management (not reporting) was conducted by Graf & King (2018), who used a Blockchain smart contract technique to provide an automated trusted system for incident management workflow, that allows automatic acquisition, classification and enrichment of incident data. They demonstrated how their solution can be applied to support incident handling tasks, performed by security operation centres, and can assist analysts by protecting critical infrastructure against increasing cyber threats. Their work, however, is focused on developing a solution that could replace human input, by facilitating automatic cyber incident classification, in order to enable analysts to focus on other tasks. Other examples include Blockchain-based Security Information and Event Management (SIEM) systems - for storing and accessing information security events - utilized by multiple devices, within the broader concept of the Internet of Things (Mesa et al, 2019; Miloslavskaya & Tolstoy, 2019), as well as a blockchain-based risk and information system control framework, able to register risk registration data on the ledger, thus ensuring traceability and irreversibility of entries (Ma et al, 2018).

The most directly relevant work regarding incident reporting and blockchain was published very recently (April, 2019) by Adebayo et al (2019), in an article titled "Blockchain-enabled Information Sharing Framework for Cybersecurity", which was also featured as a chapter in a book by Shetty et al (2019), named "Blockchain for distributed systems". In their 10-page long article, Adebayo et al (2019) propose a theoretical framework for information sharing based on blockchain, called "BIS", which utilizes a "blockchain protocol over the public internet". The authors support that since blockchain has been successfully used in privacy-aware systems, such as Bitcoin, blockchain could also be used for a cyber-incident sharing system for organizations who highly value their anonymity. They thus propose a framework for a public, blockchain, implementation, with no central authority, where any security-conscious organization could join as a member, and could also include various security vendors (e.g. antivirus companies) which could, in-turn, offer applicable solutions (e.g. patches) to participating organizations, via a cloud configuration, also accessible via the blockchain. Although the authors explain what blockchain is by using the Bitcoin blockchain an as example, they do not explicitly examine or propose any of the various available blockchains for implementing such a solution, but provide a rather theoretical implementation framework. They also propose the theoretical utilization of a consensus algorithm called "Proof-of-Attack-Detection (PoAD)", which involves the verification and approval of transactions by all participating nodes of the public ledger.

2.4. Conclusion

Although the exchange of incident-related information with other business entities can generally improve an organization's cyber defense (Hausken, 2007), it seems that organizations commonly find it difficult to disseminate information related to security incidents (He and Johnson, 2012; Grispos et al., 2015). The fear of the incident's consequences, including negative publicity, possible financial penalties, reprimands and even possible retribution attempts (Johnson, 2002; Metzger et al, 2011; Ahmad et al, 2012) are some of the reasons which may lead to the under-reporting of information security incidents, amongst organizations. It therefore comes as no surprise that despite the general encouragement for information sharing related to information security incidents, organizations continue to approach it with ambivalence (Aviram & Tor, 2003). The utilization of incident reporting platforms, for reporting purposes, is considered of high value to organizations (Cusick & Ma, 2010; Metzger et al, 2011); however, not a great number of platforms is generally available - especially with regards to manual incident reporting - while most of the platforms require a considerable financial investment on behalf of an organization. The absence of the option of anonymity regarding incident submissions is certainly an issue, since reputational concerns are a major factor for organizations sharing information security incidents (Line & Albrechtsen, 2016; Jaatun et al, 2009; Hove et al, 2014; Ahmad et al, 2015; Ruefle et al, 2014; Koivunen, 2010; Housen-Couriel, 2018).

Even though blockchain technology initially focused on crypto/virtual currencies (Di Pierro, 2017), it has now witnessed the development of applications in a variety of fields, including data management, information security, and even incident reporting, although the available literature for the latter area is rather limited. It is important to acknowledge, that a number of these identified applications do not necessarily fulfil the criteria set by the various blockchain decisions models, as those were presented in this chapter. This may indicate, that in some cases, developers aim to force their problem fit into the blockchain technology paradigm, whereas traditional technologies might provide a better solution.

Finally, regarding the research question that this project attempts to resolve, the literature identifies various reasons which contribute to the current issue of incident under-reporting. The candidate solution, blockchain technology, can certainly not resolve all of them: it cannot increase an organization's IS maturity level, it cannot change organizations which are resistant to change, it cannot increase an organization’s level of corporate responsibility and it can certainly not compel organizations to report an incident first, and then initiate any mitigation efforts. However, blockchain, can possibly provide a resolution towards some other known issues, by confronting organizational concerns, such as negative publicity, through the inherent anonymity features that the technology offers. Furthermore, it may be able to significantly reduce the various high costs associated with reporting and its processes. These features, could create the necessary value, which along with the various positive features identified through the evaluation of the existing reporting solutions (e.g. ease of use, efficiency, security, accessibility, social features and other), could possibly create an, overall, attractive solution for the organizations to utilize towards their reporting needs, based on this new technology. Blockchain also comes with additional, inherent, characteristics, such as increased availability, immutability and transparency levels. Even though the lack of these additional characteristics was not, in any way, identified by literature as a contributor to the problem of incident under-reporting, their presence in a proposed solution, although trivial, could potentially be regarded as beneficial.

3. RESEARCH METHODOLOGY

3.1. Introduction

DePoy and Gitlin (2015, p.3) define research as “a multiple, systematic strategy, to generate knowledge about human behaviour, human experience, and human environments,” which the researcher conducts through applying and following an explicit process. While there may exist various definitions about what constitutes “research”, Hassani (2017) states that the characteristics of each research field require a particular adaptation of the research concepts, through a thorough understanding of the nature of the research at hand. In the context of research, “ research methodology”, can be described as the scientific approach which investigates, compares, contrasts, and explains the various ways that research could be conducted, as well as the various “methods” that could be utilized in the process (Hassani, 2017). The research methodology aims to both explicate the reasons for selecting a particular approach to address a research problem, as well as to explain how this approach would be implemented (Hassani, 2017). “Research methods”, on the other hand, could be described as the specific procedures and guidelines used in conducting research, which might utilize various instruments and tools (Hassani, 2017). According to the same author, the research methodology should reflect on the nature of the research and the researcher should identify the research category and research paradigm which best serve his intended research (Hassani, 2017).

Every field of science requires an adaptation of the overall research approach, in order to perform a research activity; a particular research project should adjust the generalized research approach to suit the particular problem (Hassani, 2017). However, information systems/computing research seems not to be supported by globally accepted methods - unlike most well-established science disciplines – due to both its infancy and ambiguity in its definition, as well as due to its extensive coverage and overlap with other fields (Hassani, 2017). Although, paradigmatically, computing/information systems discipline can be argued to belong in the positivism/realism paradigm, which is the main paradigm of natural and life sciences (Denicolo and Becker, 2012), computing/information systems science can be described as suffering from a "lack of identity" (as a fairly new discipline), although it combines the experience of its main roots, mathematics and engineering (Demeyer, 2011). According to Hasan (2003), to reach maturity as a discipline in its own right, the new field of computing/information systems borrows research approaches from a wide variety of older disciplines, the closest comparative fields being the engineering traditions and the design sciences.

According to Nunamaker et al (1991), some research domains are sufficiently broad to embrace an extensive range of methodologies; this is particularly true in engineering and systems, where the concept at issue is likely to be viewed for its applications value rather than for its intrinsic value. As an example, Hasan (2003) proposes that, due to its nature, information systems development can be a knowledge creating activity, when those systems relate to emergent knowledge processes (EKP) (Markus et al 2002), and that, in those cases, information systems development is a legitimate research methodology. During the process of information systems development, the author argues, not only is knowledge created about the development process itself, but also a deeper understanding about the organisational problem that the system is designed to solve. Much of information systems research demonstrates a research life cycle of the form "concept, development and impact" (Nunamaker et al, 1991). Developed systems can serve both as a proof-of-concept for the fundamental research and provide an artefact that becomes the focus of expanded and continuing research (Nunamaker et al, 1991). According to Hasan (2003), many such projects can be considered a piece of "original" research, should the requirements, design and implementation keep their "originality", and provide new knowledge, as to the ways of productively managing data in complex situations.

Mackenzie and Knipe (2006) argue that scientists are abandoning the "quantitative vs qualitative" conflict and are rather focusing on the combination which brings the most benefit to the research question in hand. Along the same lines, O'Leary (2004, p.8) argues that "what was relatively simple to define thirty or forty years ago, has become far more complex in recent times, with the number of research methods increasing dramatically”. In that sense, the rather ‘traditional’ research approaches can be substituted by alternative approaches, such as Development research, Action Science research and Design Science research, which are described in this chapter.

3.2. Types of research

According to Saunders et al (2007), there are two basic types of research, Basic (or fundamental) and Applied. Basic research can be defined as “experimental or theoretical work, undertaken primarily to acquire new knowledge of the underlying foundation of phenomena and observable facts, without any particular application or use in view” (OECD, 2015). It is undertaken predominantly in universities, as part of academic agendas (Saunders et al, 2007), its aim is purely theoretical, and it is considered successful when it discovers new phenomena or new ideas of general interest (Roll-Hansen, 2009). Applied research, on the other hand, is original investigation primarily directed towards a specific, practical aim or objective (OECD, 2015). It is directly relevant to practitioners, as it addresses issues that they see as important (Saunders et al, 2007), and produces possible applications to products, operations, methods, or systems (OECD, 2015). Applied research considers available knowledge, in order to solve actual problems, and therefore gives operational form to ideas (OECD, 2015); it may be undertaken to determine possible uses for the findings of basic research, or to determine new methods/ways of achieving pre-determined objectives. Applied research is considered successful, when an actual contribution to the solution of specific practical problems, is produced (Roll-Hansen, 2009).

Information systems can be considered to be an applied research discipline, in the sense that “theory is frequently applied from other disciplines, such as economics, computer science, and the social sciences, to solve problems at the intersection of IT and organizations” (Peffers et al, 2007, p.2). However, researchers such as Kitcher (2001), have argued that the distinction between fundamental and applied research is essentially based on the “myth” of pure science - the idea that “basic” science should be isolated - and thus no significant distinction can be made between the two types, either in the descriptive or the normative dimension. Furthermore, Nieswiadomy (2011) argues that many research studies combine elements from both types of research, and a quest starting off as basic research could, eventually, have an impact on a matter of professional practice.

Saunders et al (2007), provide a useful diagram, to illustrate the differences between the basic and applied research, although they do argue that it is possible to situate projects somewhere between the two extremes:

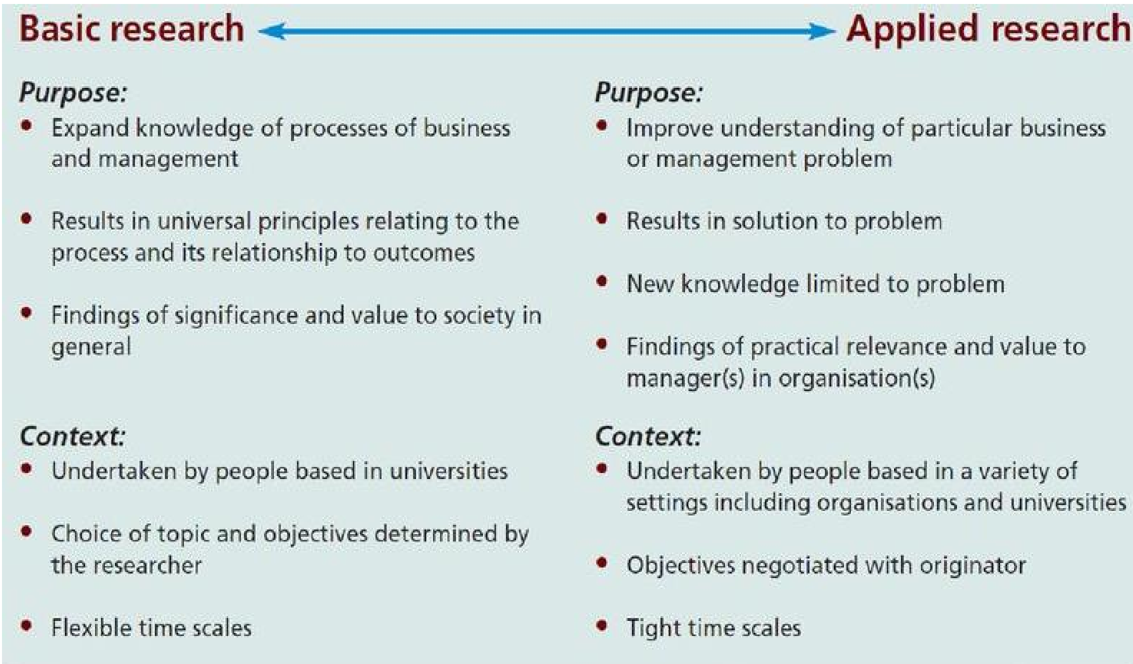


Figure 3.1. Basic vs Applied research by Saunders et al (2007)

This research project can be situated towards the applied research discipline, since it aims to provide a solution to a problem of professional practice. More particularly, it aims to provide a solution to the identified problem of incident under-reporting, by presenting an incident reporting solution based on a new technology, with distinct characteristics.

3.3. Research philosophy

There are three major ways of thinking about research: Ontology, Epistemology and Axiology (Saunders et al, 2007; Collis & Hussey, 2013; Vaishnavi et al, 2004/19). A description of each way of thinking can be found in Appendix J. The research philosophy adopted in this project will become evident in the following sections of this chapter.

3.4. Research paradigms

Paradigm may be defined as the philosophical intent or motivation for undertaking a study (Cohen & Manion, 1994) or as a "loose connection of logically related assumptions, concepts, or propositions, that orient thinking and research" (Bogdan & Biklen, 1998, p.22). Some of the most common paradigms referred to in research, include the positivist, the interpretivist, the transformative and the pragmatic paradigms (Mackenzie & Knipe, 2006). A description of these paradigms can be found in Appendix J. This research project is situated within the pragmatic paradigm, as it places the problem of incident under-reporting as central and attempts to provide a solution.

3.5. Research approaches

In general, research methods, which involve activities of design and construction, relate to grounded approaches to research – such as pragmatism – where the notion of "truth is what works in practice", prevails (Hasan, 2003). In Information Systems research, when designing and constructing a system are involved, typical methods include observation, action or participant research and prototyping (Baskerville & Wood-Harper, 1998). Hasan (2003) argues, that in terms of knowledge creation, the validity evidence of this type or research is typically referred to as "proof of concept".

A criticism among the research community, is the perceived lack of relevance of information systems research for practice (Benbasat and Zmud, 1999; Dennis, 2001; Kock et al, 2002). The argument behind this criticism, according to Cole et al (2005), is that research must contribute to both academia and practice; research should, therefore, add to existing theory - to make a scientific contribution - and should also assist in solving practical problems of the industry, anticipated or current (Cole et al, 2005). Examples of information systems research methods, which fulfill the criteria of this dual orientation, are “Development research” (Hasan, 2003), “Design Science research” (Hevner et al, 2004) and “Action research” (Davison et al, 2004), and the research community eventually seems to become more accepting of these diverse – or “untraditional” - research approaches (Boland & Lyytinen, 2004). A description of these approaches can be found in Appendix J.

3.6 Selected research approach

Although any of the above-mentioned research approaches could, potentially, have been utilized for this research project, development research was excluded, mainly because of the considerably less available resources (both general resources discussing development research in IS, as well as process models/frameworks for conducting such research) identified in literature, in comparison to both Action research and Design Science research. Between the latter two approaches, Design Science research was, eventually, selected as the research approach of choice; although CYCSO did display a vigorous, initial, interest into utilizing the reporting platform, the solution/artefact needed to be as customer-neutral as possible. Furthermore, besides the early interest of the organization, no contractual (or any other form of) agreement was pursued by either parties, nor could the organization provide any resources towards this project, such as funding, or the human resources necessary to establish the “client-researcher relationship”, a prerequisite of Action research (Iivari & Venable, 2009, p.4). Furthermore, according to Baskerville (2008, p.442), whereas Action research focuses on “problem solving through social and organizational change”, Design science research is focused “on problem solving by creating and positioning an artefact in a natural setting”. Therefore, and since for this research project an artefact would be created and consequently positioned and evaluated in a natural setting (i.e. in an organizational environment), Design Science research emerged as the most suitable research approach.

Design science research is constantly winning a wider audience (Jarvinen, 2007). It is motivated by the desire to introduce new and innovative artefacts and the processes for building these artefacts, thus improving the environment (Simon, 1996). According to Vaishnavi et al (2004/19), the metaphysical assumptions of design science research are unique, since none of the axiology, ontology or epistemology of the paradigm is derivable from any other paradigm. The following table displays the philosophical assumptions of three research perspectives, including Design science:

Basic Belief	Research Perspective		
	Positivist	**Interpretive**	**Design**
Ontology	A single reality; knowable, probabilistic	Multiple realities, socially constructed	Multiple, contextually situated alternative world-states. Socio-technologically enabled
Epistemology	Objective; dispassionate. Detached observer of truth	Subjective, i.e. values and knowledge emerge from the researcher-participant interaction.	*Knowing through making*: objectively constrained construction within a context. Iterative circumscription reveals meaning.
Methodology	Observation; quantitative, statistical	Participation; qualitative. Hermeneutical, dialectical.	Developmental. Measure artifactual impacts on the composite system.
Axiology	Truth: universal and beautiful; prediction	Understanding: situated and description	Control; creation; progress (i.e. improvement); understanding

Table 3.1. Philosophical assumption of three research perspectives by Vaishnavi et al (2004/19)

According to Vaishnavi et al (2004/19), ontologically, Design science research changes the state-of-the-world, through the introduction of novel artefacts, and therefore design science researchers are comfortable with alternative world-states. The creation of an artefact with a problem-solving functionality, the incident reporting platform, in this case, requires a natural-science-like belief in a single, fixed, grounding reality (Vaishnavi et al, 2004/19). Epistemologically, the Design science researcher acknowledges that information is factual, and further acknowledges what that information means, through the process of development (Vaishnavi et al, 2004/19). The Design science researcher is a pragmatist; an artefact - the platform - is developed, and its behaviour is the result of interactions between the various components. Descriptions of the interactions are information, and should the platform behave predictably, that information is true (Vaishnavi et al, 2004/19). Axiologically, the researcher values creative manipulation and control of the environment, over more traditional values such as the quest for truth or understanding, and he/she ought to have a higher tolerance for ambiguity (Vaishnavi et al, 2004/19).

The cognition that takes place during a Design science research cycle is evident in the following figure:

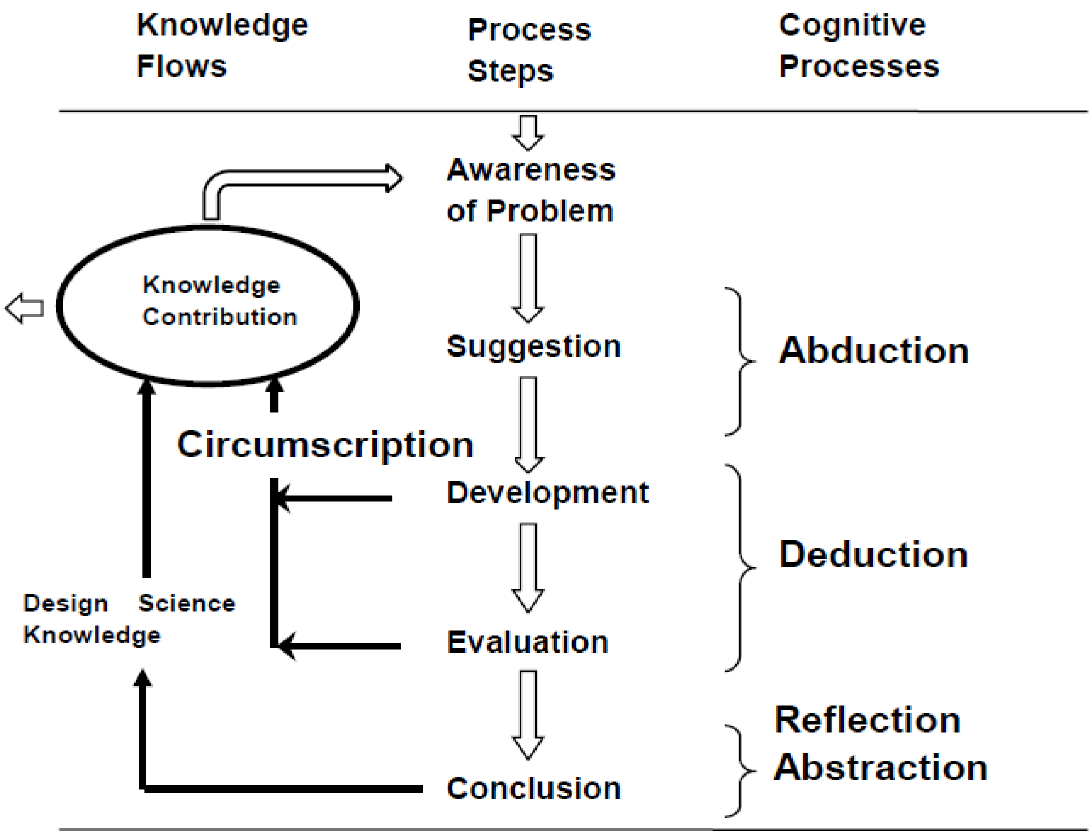


Figure 3.2. Cognition in a Design science research cycle by Vaishnavi et al (2004/19)

Research begins with awareness of the problem, the incident under-reporting, in this case. Suggestions/requirements for providing a potential solution to this problem are drawn from existing knowledge (Peirce, 1931), and using this existing knowledge, a creative solution to the problem is attempted (Vaishnavi et al, 2004/19), the blockchain-based incident reporting platform. After the platform was created, it was evaluated, through a structured procedure. The researcher's reflections of this project are evident in the conclusion chapter of this report. The following figure displays a knowledge contribution framework for Design science research, which describes the various types of knowledge contribution a research project may achieve:

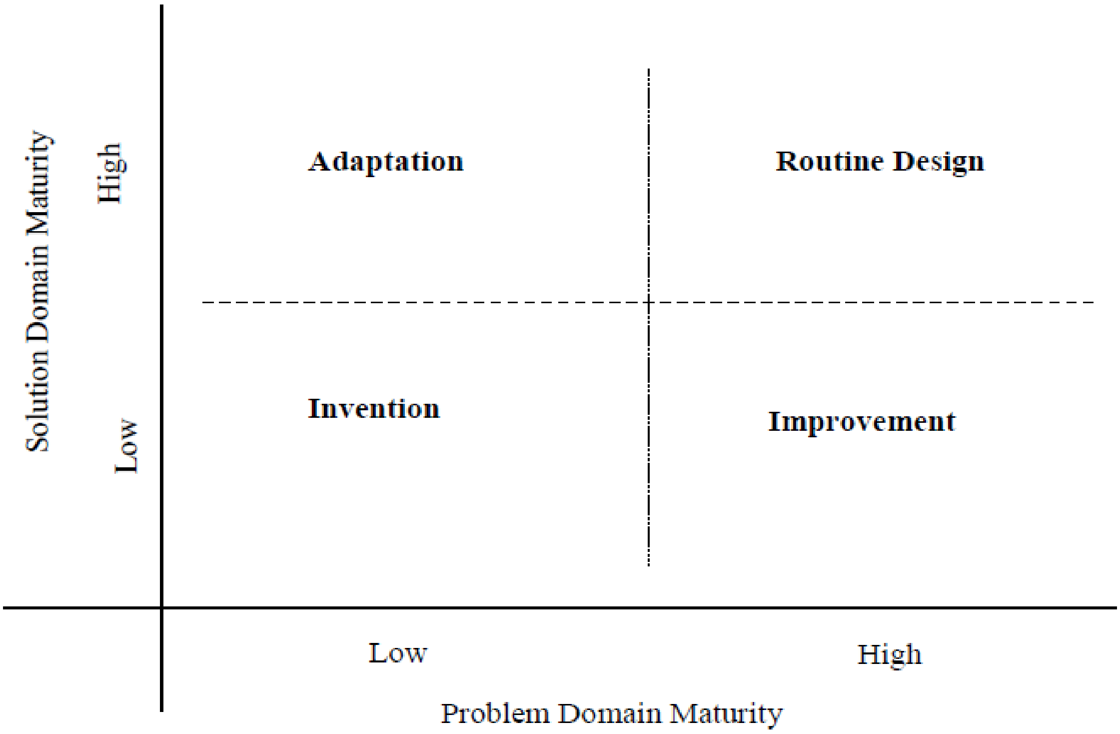


Figure 3.3. Knowledge contribution framework for Design science research by Gregor and Hevner (2013)

According to Gregor and Hevner (2013), there are four types of knowledge contribution: *Invention* is inventing new solutions/knowledge for new problems, *Improvement* is developing new solutions/knowledge for known problems, *Adaptation* concerns the innovative adaptation of known solutions/knowledge for new problems, and *Routine Design* is applying known solutions to known problems, which, by itself, would not usually be considered as a research contribution (Gregor & Hevner, 2013). According to the same authors, a research project can make more than a single type of contribution. This research project belongs in the *Improvement* segment of the framework, as it applies an innovative solution, towards the known problem of incident under-reporting.

As previously mentioned, although a generally accepted process for carrying out Design Science research does not exist (Peffers et al, 2007), there is a number of different process models/frameworks applicable towards conducting Design science research. Regarding this research project, and after careful consideration of the available options, the framework by Peffers et al (2007) was identified as the most appropriate choice, as it incorporates principles, practices and procedures necessary to conduct such research, while being consistent with prior literature (Peffers et al, 2007). In order to develop their methodology, Peffers et al (2007, p.11) looked at "influential prior research and current thought, to determine the appropriate elements, seeking to build upon what researchers have said in key prior literature about what design science researchers did or should do", and designed a methodology that could serve as "a commonly accepted framework for carrying out research based on Design science". This framework provides a nominal process model for undertaking Design science research, as well as a mental model for presenting and evaluating such research, in the Information Systems domain (Peffers et al, 2007). According to the authors, their Design Science Research Methodology (DSRM), allows researchers to present their work by referencing a commonly understood framework, instead of justifying the research paradigm on an ad-hoc basis, for every different project/paper (Peffers et al, 2007). The authors utilized a consensus building approach in creating DSRM, in the sense of including various identified common features of other proposed frameworks in their own framework, rather than focusing on the differences in views about Design science among the various researchers. According to the authors, "Archer's (1984) process for industrial design, Takeda et al.'s (1990) "design cycle" solution for intelligent computer aided design systems, Nunamaker et al's (1991) five-step methodology, Eekels and Roozenburg's (1991) process for engineering design, Walls et al.'s (1992, 2004) "components of an information system design theory," Rossi and Sein's (2003) steps, and Hevner et al.'s (2004) guidelines for the required elements of Design research, are all consistent with DSRM" (Peffers et al, 2007).

DSRM includes six, distinct, activities/steps, and it is structured in a nominally sequential order, although there is no expectation that the researcher should always proceed in sequential order; according to the authors, a researcher may begin at any step and move outward (Peffers et al, 2007). The following figure depicts the six steps of the methodology, as well as the possible research entry points, based on the researcher’s particular approach:

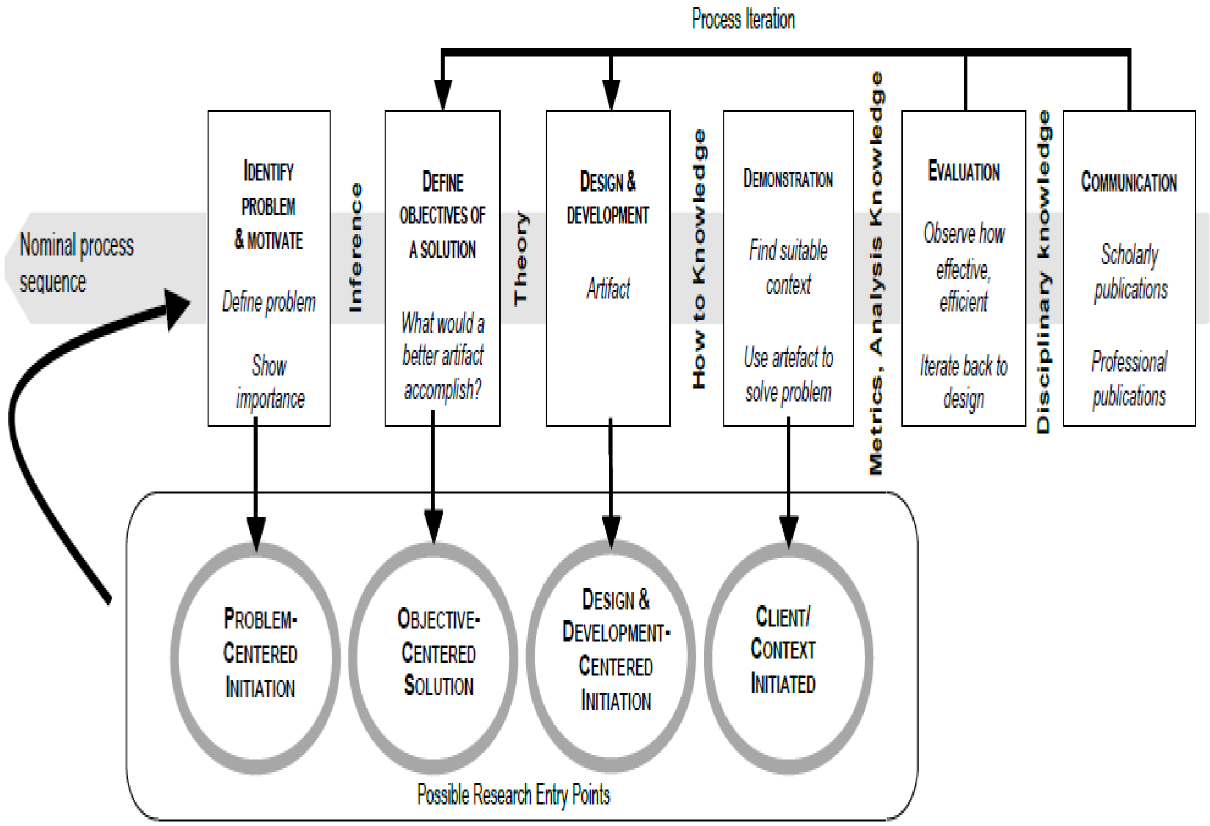


Figure 3.4. DSRM process model by Peffers et al (2007)

As the authors explain, a problem-centred research approach should begin with the first activity (“Problem identification and motivation”), and sequentially move through the other activities. An objective-centred solution, initiating effort at activity two, (“Define objectives of a solution”), could be prompted by “an industry or research need that could be addressed by developing an artefact”, whereas design and development-centred and demonstration-centred approaches, would begin with activities three and four, respectively. This research project begun at the very first activity/step of the model, as it intended to provide a solution towards the problem of incident under-reporting.

A synopsis of the six activities undertaken as part of this research project can be seen below:

Activity 1: Problem identification and motivation: "Define the specific research problem and justify the value of a solution" (Peffers et al, 2007, p.12) – During this activity, the problem of security incident under-reporting was identified and analysed, the research question was formed, research motivation was explained and a rigorous literature review was conducted, while also describing the value of the proposed solution. Evidence of this activity can be found in chapters one (Introduction) and two (Background, literature review & reporting means evaluation) of this report.

Activity 2: Define the objectives for a solution: "Infer the objectives of a solution from the problem definition and knowledge of what is possible and feasible" (Peffers et al, 2007, p.12) – During this activity, the objectives of the decentralized solution were inferred rationally from the problem specification. A justification and qualitative explanation of each stated objective was provided. Evidence of this activity can be found in chapter four (The decentralized solution: Objectives).

Activity 3: Design and development: "Create the artefact. Such artefacts are potentially constructs, models, methods, or instantiations or "new properties of technical, social, and/or informational resources" (Peffers et al, 2007, p.13) – During this activity, the solution's desired architecture, functional and non-functional requirements were determined, and the decentralized reporting platform was developed. Evidence of this activity can be found in chapter five (The decentralized solution: Design and development).

Activity 4: Demonstration. "Demonstrate the use of the artefact to solve one or more instances of the problem" (Peffers et al, 2007, p.13) – During this activity, verification and validation procedures were performed. Verification procedures ensured that the developed artefact met its predefined objectives. As part of the validation procedures, a working prototype of the reporting platform was presented to six organizations, which voluntarily tested the software, using a number of test scenarios, and provided appropriate feedback. Evidence of this activity can be found in chapter six (The decentralized solution: Demonstration).

Activity 5: Evaluation: "Observe and measure how well the artefact supports a solution to the problem" (Peffers et al, 2007, p.13) – During this activity, the Venable et al (2012) evaluation framework, applicable specifically to DSR projects, was followed. Two evaluation methods were utilized. During the first evaluation method, users who also participated in the demonstration activities of the artefact, were called to complete evaluations, by completing, two, identical, Likert-style questionnaires, initially assessing the capabilities of their current (or previously used) incident reporting platform, and then assessing the capabilities of the newly developed artefact. The results obtained from these questionnaires were tested for significance, using the non-parametric, Wilcoxon-Pratt signed ranked test. The second evaluation method aimed to assess the quality of the developed software (artefact), and included a high-level, qualitative, assessment of the developed software (performed solely by the researcher), against the requirements posed by the international standard "ISO/IEC 25010:2011 - Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE). Evidence of this activity can be found in chapter seven (The decentralized solution: Evaluation).

Activity 6: Communication: "Communicate the problem and its importance, the artefact, its utility and novelty, the rigor of its design, and its effectiveness to researchers and other relevant audiences, such as practicing professionals, when appropriate" (Peffers et al, 2007, p.14) – During this activity, the aforementioned activities were documented, as part of this report. The structure of the document followed the flow of activities described in the DSRM model, and also borrowed some elements out of the nominal structure of an empirical research process, including the literature review, the description of the selected research methodology, as well as the "discussion and conclusion" chapter. Evidence of this activity can be found in all of the chapters of this report.

3.7. Research ethics and other research considerations

According to Iivari (2007), Design science research implies an ethical transformation from “describing and explaining the existing world”, to actually, “shaping it”. According to the same author, research in the Information Systems domain, may serve the interests of particular groups, such as those of businesses and organizations, professionals, users and various others. This research project aimed to provide an alternative solution to organizations and users alike, for reporting information security incidents, using a different approach from existing solutions and by utilizing a new technology.

This research project strictly followed the University of East London’s research guidelines/procedures, as well as the Concordat (UK) for research integrity (2012), applying the “highest standards of rigour and integrity”, in all aspects of this research, and ensuring that research was conducted ethically, legally, transparently, and according to standards. Ethical approval was gained from the university’s relevant committee (Appendix D). Information confidentiality was assured throughout this project, and no organization or user/participant was named. Generic identification codes were used instead, thereby assuring no data is traceable to a particular individual or organization. All participants were adequately informed about this project’s aims and scope, and provided their input voluntarily, free from coercion.

In general, with regards to the entire work conducted as part of this research, and in particular, during the artefact’s design and development procedures, a controlled environment was utilized, in the existing lab the researcher uses for his consultancy profession, with all required health, safety and security measures and procedures in place – as specified by local and European regulations and standards.

Furthermore, since every research project assumes a particular level of inherent risk taken on by the researcher, a risk assessment was carried out, specifically for this project, prior to its commencement. A risk assessment matrix, containing the risk factors, their likelihood and potential impact, as well as the associated mitigating measures, can be found in Appendix A.

4. THE DECENTRALIZED SOLUTION: OBJECTIVES

4.1. Introduction

According to Peffers et al's (2007) Design science research (DSR) framework, the second activity in a problem-centered DSR project, involves defining the objectives of the proposed solution. According to the authors, these objectives should be inferred from the problem definition, while having adequate knowledge about what is "possible and feasible". The researcher should understand the state of the problem(s), as well as the current solution(s) and its/their efficacy (Peffers et al, 2007). Acquiring such knowledge, required identifying and evaluating existing incident reporting methods and practices, activities which are evident in the second chapter of this report. During this set of activities, it was identified that information security incidents can be reported automatically, without human intervention (e.g. through mechanisms implemented in IDS/IPS systems, firewalls and other tools), or manually, through e-mails, forms, or even verbally, through the phone. Reporting platforms can be utilized for both automatic and manual reporting. The identification and evaluation of existing incident reporting options/solutions, including their functionality and features, as well as their advantages and disadvantages, allowed the consequent establishment of the objectives of this research project. Since this project aimed to suggest a resolution towards the recognized problem of incident under-reporting, the objectives included producing a solution which would, on one hand, suggest an innovative approach towards the problem, but would also, on the other hand, incorporate all identified features and characteristics of existing approaches, which were deemed as both useful and effective.

According to Peffers et al (2007), the objectives set during this activity can be quantitative or qualitative, in nature. Quantitative objectives refer to terms in which a desirable solution would be better than existing ones, whereas qualitative objectives describe how a new artefact could support solutions to not previously addressed problems. In this chapter, the project's objectives are stated, and a justification and qualitative explanation of each objective is provided.

4.2. Objectives

The objectives of this project derive directly from the problem definition. It is therefore useful, at this point, to revisit the relevant research question:

Is there a way to create an innovative information security incident reporting solution, which will utilize the positive features offered by existing solutions, but will also provide added value to users, in order to increase their level of motivation towards the reporting of incidents?

Although some authors state that incident reporting “typically represents a situation with unknown return on investment (Briggs et al, 2017, p.9), most of the literature agrees that information security incident reporting is beneficial to organizations (NIST, 2012; Gordon et al, 2003; ENISA, 2013; Gordon et al, 2015; Line & Albrechtsen, 2016; Gonzalez, 2005). Through work evident in the previous chapters of this report, it was identified, that while information security incidents are, generally, on the rise - thus creating major implications for organizations - organizations themselves tend to under-report these incidents. Despite the current availability of a variety of incident reporting tools and methods (e-mail, verbal, platforms and various automatic and manual tools), as well as the fact that the exchange of incident-related information with other business entities can generally improve an organization’s cyber defense (Hausken, 2007), organizations approach incident reporting with ambivalence (Aviram & Tor, 2003), and find it difficult to disseminate information related to security incidents (He and Johnson, 2012; Grispos et al, 2015). Therefore, it was initially necessary to identify the reasons which prevent organizations from reporting. As identified through the available literature, these reasons include the organizational fear of the incident’s consequences, including negative publicity, legal liability, regulatory incompliance and possible financial penalties and reprimands, the exposure of organizational vulnerabilities, possible retribution attempts, the various costs related to incident reporting, such as operating costs, recruitment and training, the organization’s overall IS maturity level, as well as the overall organizational time spent by an organization’s personnel for reporting purposes (Johnson, 2002; Metzger et al, 2011; Ahmad et al, 2012; Etzioni, 2014; Ruefle et al, 2014; Humphrey, 2017; Housen-Couriel, 2018).

The aforementioned reasons, led to the forming of the first objective of this research project:

- O 1: *Create an incident reporting solution which enables and encourages the reporting of information security incidents amongst organizations, thereby reducing organizational demotivation for reporting.*

Reporting statistics confirm the under-reporting of information security incidents and indicate that very few of them are indeed being reported (IOD & Barclays Policy report, 2016; Symantec, 2016; Newman, 2018; Ipsos MORI, 2017; SentinelOne, 2016; ENISA, 2012). In order to increase the motivational level of organizations towards the reporting of incidents, a number of, current, concerns need to be tackled: the designed solution should enable organizations to anonymously report incidents, without the fear of facing any fines or reprimands from authorities, or exposing organizational vulnerabilities to non-trusted parties. It should also reduce the financial cost of reporting for organizations, by providing a, generally, non-expensive option towards the operation and maintenance of a reporting system/solution, as well as minimize any personnel training costs. The solution should be in the form of an artefact (instantiation) and should fit in the “Detection and Reporting” phase of the overall incident response procedure/lifecycle, as defined by the ISO/IEC 27035 international standard on “Information Security incident management”.

The research question, however, also dictates the incorporation of the positive aspects/features offered by existing solutions, into the designed solution. Therefore, the second objective of this research project is as follows:

- O 2: *Create an incident reporting solution which utilizes the positive features offered by existing reporting solutions.*

As identified through the evaluation of existing reporting solutions, evident in the second chapter of this report, existing solutions do have positive features: they seem to be easy to understand and use, utilizing simple and straight-forward GUIs, with a good level of support and training offered by the commercial providers. Performance seems to be smooth, in either cloud or on-premise deployments, although scalability could not be adequately tested, since the demo versions available prohibited the simulation of a resource-intensive environment, with many users and multiple submissions of incidents. However, and as previously mentioned, manual information security incident reporting platforms are not theoretically expected to yield an enormous data volume, capable of deteriorating performance and efficiency. Regarding security, the encryption supported in the communication channels is certainly a major plus, while two-factor authentication offered by some platforms suggests enhanced security. Therefore, the designed solution should offer a secure environment for participants, without any sacrifices in both efficiency and performance. It should be easy to use, widely accessible and location independent, while also offering adequate customer support. Social features identified in some existing solutions, such as forums or chatrooms, are deemed as useful, since they offer the opportunity for the immediate communication between users. They can aid in the direct coordination of actions and the immediate exchange of feedback, between participants (e.g. in the case of a same/similar threat targeting multiple participating organizations, at the same time), but they can also aid in the build-up of mutual trust, between them.

Despite the important benefits that current solutions offer, there is certainly area for improvement. The research question dictates providing “added value” to users, and thus, the third objective is as follows:

- O_3: *Create an incident reporting solution which provides added value to users, in comparison to existing solutions.*

The evaluation of existing reporting solutions acknowledged some issues, such as the lack of participants' anonymity, the non-constant availability and the limited auditability/transparency of these solutions. Providing users' and submissions' anonymity, which is also a requirement of the first objective, will certainly deliver added value to users. Moreover, non-constant availability, an inherent characteristic of centralized environments, can be tackled with a decentralized reporting solution. Regarding auditability/transparency, although current solutions offer adequate audit mechanisms, stolen credentials could easily lead to the unauthorized modification (including erasure) of transactions/submissions, in a centralized database. A decentralized solution provides an environment where successful submissions of incidents are immutable, thereby increasing overall auditability and transparency.

4.3. Implementation targets

The implementation targets (ITas) described in this section aim to enhance the overall functionality, usability and reliability of the designed solution, by dictating specific implementation tasks which were identified by the literature review (chapter two) as being the most suitable:

- ITa $_1$: *Create a <u>manual</u> incident reporting solution.*

As previously stated, automation in incident reporting, does not come trouble-free and the automated reporting tools have their limitations (Tondel et al, 2014). Werlinger et al (2010) identified a lack of accuracy in automated tools, with high false positive rates, as a result. In addition, the automated tools' usability is also a concern, with researchers identifying an organizational need for often customization/adjustments of these tools. (Werlinger et al, 2008, 2010; Metzger et al, 2011). Furthermore, information needs to be sanitized before automated exchange can take place, while sharing all available security data could lead to performance and scaling concerns in organizations (Kampanakis, 2014). Although nobody debates that automation can generally benefit organizations (Line, 2013), it seems that manual reporting methods still prevail, and organizations prefer manual reporting methods (Metzger et al, 2011; Hove et al, 2014; Grispos et al, 2015). Therefore, the designed solution should be manual, rather than automated, with physical entities (i.e. humans) submitting transactions (incidents), instead of automated tools, such as firewalls, IDS/IPS systems and/or other automated monitoring systems.

- ITa $_2$: *Create a software <u>platform</u> for the manual reporting of incidents*

As previously stated, researchers have indicated a variety of methods that employees use for the manual reporting of incidents, including e-mail, telephone, other verbal communications, help desk functions and incident reporting software/platforms (Cusick & Ma, 2010; Metzger et al, 2011; Ahmad et al, 2012; Grispos et al, 2015, 2017; Hove and Tårnes, 2013; Line, 2013). Communicating incidents though e-mail, telephone or other informal tools might become problematic: e-mails could be delivered to the wrong recipients (or not delivered at all), telephones might not be answered, and verbal communications might be ignored, or even deliberately neglected. The utilization of an incident reporting platform, for reporting purposes, is considered of high value to organizations: Metzger et al (2011) stated that organizations should use such a tool and recommended to collect all data related to the incident into such a system, while Cusick and Ma (2010) praised the use of an incident reporting platform for reporting incidents. An incident reporting platform, with an accessible, clean and easy to navigate, interface, and with clear submission instructions, can aid the incident reporting capability of an organization. It can eliminate the possibility of delivering a report to unauthorized recipients (since the platform's users would be pre-authorized), while it can also enable the possibility of extracting statistics and reports, viewing historic trends, and submitting queries in a searchable database. The designed solution should, therefore, take the form of a software reporting platform.

- ITa $_3$: *Create a <u>private</u> incident reporting platform.*

The reporting platform cannot be open for anyone to join. A trusted, central, authority is necessary both for the pre-authorization of participating members, as well as for the overall administration of the platform, by performing tasks such as removing misbehaving participants, for example. An open platform could allow the submission of irrelevant, false, or invalid incidents, while it could even permit malicious parties wasting network and storage resources (e.g. in the form of “spam” submissions), or deliberately submitting false incidents, in order to confuse participants, while preparing for a dissimilar attack. This misinformation could eventually demotivate true participants from submitting incidents, thus defying the original purpose of the designed solution, to increase incident reporting. In such an environment, mutual trust between participants (including the validity of their submissions) would be very difficult, if not impossible, to achieve. The necessity of the presence of a central authority, in the designed solution, is therefore evident. The platform, however, could allow more than one partaker to form an authority, in the form of a consortium. Two or more, designated, businesses/ organizations, for example, could act as a joint authority, and could take necessary decisions, based on mutual consensus. Nevertheless, this could add to the complexity and cost of the solution, increasing overhead, time-spent on taking decisions and overall administration effort. It could also create a sense of inequality amongst members, since the consortium participants would inevitably possess an elevated status. Therefore, it was decided that a single, commonly trusted, organization, should act as the central authority. This organization could be CYCSO for example, or could even be a designated department of management, in case the designed solution is implemented internally within an organization/business.

- ITa $_4$: *Create a familiar environment for platform users*

Apart from utilizing the positive features currently offered by existing reporting solutions (PO $_2$), the designed platform should “feel” and “look” familiar to users.

This means that the overall “experience”, structure, graphical user interface (GUI), functionality, and the typical sequence of actions of the designed platform, should match (or very much approach) the existing operational environment of the current reporting platforms. Furthermore, familiar, standardized, and widely accepted reporting templates, should be utilized for the reporting of incidents. The combination of these elements is expected to enhance the uniformity (and potentially, the overall acceptance) of the designed solution.

4.4. Aggregated table of objectives and implementation targets

The following table summarizes the objectives and implementation targets of this research project:

Research question	***Is there a way to create an innovative information security incident reporting solution, which will utilize the positive features offered by existing solutions, but will also provide added value to users, in order to increase their level of motivation towards the reporting of incidents?***	
Objectives		
No.	**Description**	**Keywords**
1	Create an incident reporting solution which enables and encourages the reporting of information security incidents amongst organizations, thereby reducing organizational demotivation for reporting.	Anonymity, Cost reduction, Artefact (instantiation)
2	Create an incident reporting solution which utilizes the positive features offered by existing reporting solutions.	Efficiency, Performance, Ease of use, Ease of understanding, Accessibility, Security, Support, Social features

3	Create an incident reporting solution which provides added value to users, in comparison to existing solutions.	Anonymity, Availability, Auditability/transparency/immutability
	Implementation targets	
No.	**Description**	**Keywords**
1	Create a manual incident reporting solution.	Manual solution
2	Create a software platform for the manual reporting of incidents	Reporting software/platform
3	Create a private incident reporting platform.	Private reporting software/platform
4	Create a familiar environment for platform users.	Familiar structure, functionality, GUI, standardized reporting templates

Table 4.1. Aggregated table of objectives and ITas

5. THE DECENTRALIZED SOLUTION: DESIGN AND DEVELOPMENT

The previous chapter set the objectives that the proposed solution should fulfil. This chapter describes the design and implementation details of the solution, which are grounded on these objectives. According to Peffers et al (2007) framework, the “Design and Development” activity involves the creation of the innovative artefact, which could be an instantiation, a construct, a method, a model, or “new properties of social, technical, and/or informational resources” (Jarvinen, 2007). The outcome of this research project is an instantiation, a working prototype of an incident reporting platform, based on the blockchain technology. According to the framework, during the “Design and Development” phase, the artefact’s desired functionality and architecture should be determined, followed by the creation of the actual artefact. The produced instantiation aims to deliver a positive response to the research question, as it attempts to create an innovative incident reporting platform, which will utilize the positive features offered by existing solutions on one hand, while providing added value to users on the other hand, and thereby motivating them towards the reporting of information security incidents.

5.1. Blockchain suitability

Even though blockchain technology initially focused on crypto/virtual currencies (Di Pierro, 2017), it has now witnessed the development of applications in a variety of fields, including data management, information security, and even incident reporting, although the available literature for the latter area is rather limited. Blockchain technology, with its decentralized structure and its various inherent characteristics, including security, anonymity and integrity (Yli-Huumo et al, 2016), could possibly provide an alternative option/solution for incident reporting to organizations. Nonetheless, and despite the fact that blockchain technology is becoming increasingly relevant to real-world applications (Zhao et al, 2016), its use is not a silver bullet (Yaga et al, 2018). Taking into consideration the project’s objectives and before instinctively utilizing blockchain as the preferred implementation technology, it would be wise to initially examine whether blockchain could, indeed, be used for fulfilling the purposes of this project. As already mentioned in chapter two, a number of “blockchain decision models” have been proposed by various authors, to help examine whether blockchain technology could be an appropriate fit for a project. The blockchain decision model by Wust and Gervais (2018), was eventually selected to evaluate the appropriateness of applying the blockchain technology for this project. The selection of this model was based on the fact that, according to the authors, “it is the first structured methodology to decide which technological solution is the most appropriate, considering the required trust assumptions, application requirements, involved parties and technical characteristics” (Wust &Gervais, 2018, p.9). It also differentiates between the possible use of a public or private blockchain and contrasts their properties against those of a traditional database model.

The outcome is presented below:

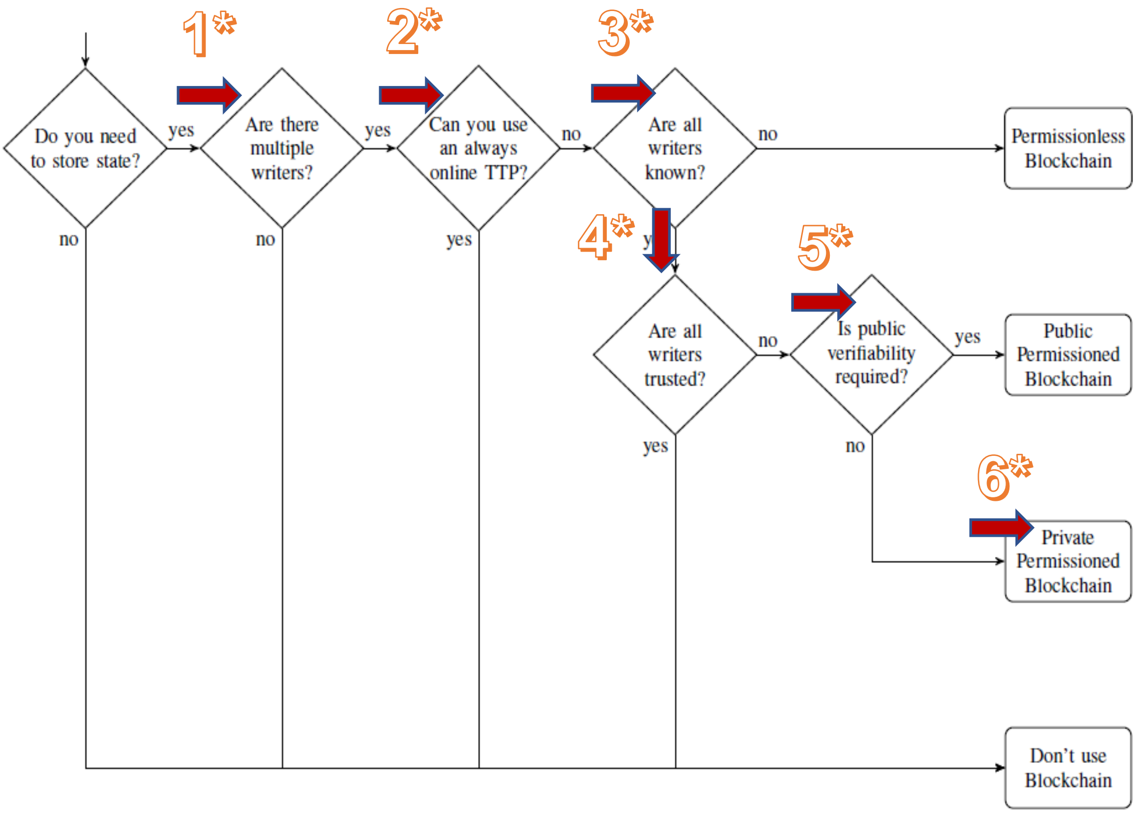


Figure 5.1. Wust and Gervais (2018) Blockchain decision model flow

Figure explanations:

- 1*: The state of the transactions/incidents submitted needs to be stored.
- 2*: There are multiple writers, as more than one users/organizations should be able to view and submit incidents.
- 3*: The trusted third party (TTP) available does not have the ability of always being online.
- 4*: All writers (users/organizations) need to be pre-approved by the TTP before being allowed to have access to the platform and to view/submit incidents.
- 5*: Although all writers are known and pre-approved (only from the TTP), they cannot be blindly trusted. Participating organizations might not essentially trust each other and in addition, an organization might misbehave, thus necessitating its removal from the platform
- 6*: Public verifiability is not required since only the pre-approved platform's users should be able to view and verify transactions.

Therefore, and according to Wust and Gervais' (2018) model, a private, permissioned, blockchain would be a suitable candidate for this research project. According to Cai et al (2018), a private blockchain implements access control and functions under a specific organization. Although a "hybrid" blockchain could also be selected for implementation – where a group of trusted entities, instead of a single owner, would have control over the blockchain (Mougayar, 2016) – the selection of a single owner/authority for the reporting platform appeared as the most attractive option, in order to both increase overall platform efficiency and also reduce complexity in taking various actions, such as approving new members, removing misbehaving entities and other administration-related tasks.

5.1.1. Blockchain of choice

Although the Bitcoin blockchain is considered to be the first application of the Blockchain technology (Iansiti & Lakhani, 2017), various alternatives have emerged since. Since the reporting platform would be based on a private blockchain operation, relevant implementation technologies were examined, in order to select a suitable blockchain for implementing this project. According to Sajana et al (2018), “Ethereum”, “Hyperledger Fabric”, “R3 Corda” and “Multichain”, are all popular options for implementing private blockchains. According to its whitepaper, Ethereum is a blockchain-based distributed, computing platform and operating system, with “smart contract” functionality, which provides the ability of building decentralized applications in situations where “rapid development time, security, and the ability of different applications to very efficiently interact, are important” (Buterin, 2014). “Fabric” by Hyperledger, is a consortium formed by the Linux foundation and many other partners, such as IBM, Intel, SAP, Cisco, Daimler, and American Express, to design and develop enterprise blockchains (Androulaki et al, 2018), whereas R3’s “Corda”, is a distributed ledger platform for recording and processing financial agreements (Brown et al, 2016). “Multichain” is an open source blockchain platform, that enables the setup, configuration, and deployment of a private, public, or hybrid blockchain (Greenspan, 2015c), and like “Corda”, it is also mainly intended for the financial industry (Cachin & Vukolic, 2017).

In theory, any of the above blockchains could be utilized for this project. However, it seems that Ethereum and Fabric are those that present themselves as “utterly independent of any specific field of application” (Valenta & Sandner, 2017, p.1), whereas Corda and Multichain appear to have been “consciously designed for the financial services industry” (Valenta & Sandner, 2017, p.7). Corda’s use cases are drawn for the financial services industry and according to Valenta and Sandner (2017), even efforts for integrating Corda into the Hyperledger project exist, thus considering Corda as complementary to Fabric, rather than a direct competitor. Along the same lines, Multichain is intended for private blockchains in the financial industry and for multi-currency exchanges in a consortium, aiming at being compatible with the Bitcoin ecosystem (Cachin & Vukolic, 2017). On the other hand, Fabric provides a modular and extendable architecture, and is applicable in various settings and industries, and so is the case with the Ethereum blockchain (Valenta & Sandner, 2017). Therefore, regarding this particular project, Fabric and Ethereum were the strongest candidates.

According to Cai et al (2018), when selecting a potential blockchain technology, one should select an implementation which is stable but also flexible. Both Fabric and Ethereum possess the above qualities, since their stability and extendibility have been tested (Cai et al, 2018). Both technologies utilize smart contracts (written in Java or Go for Fabric and Solidity, Serpent or Vyper for Ethereum) and both are open-source initiatives. However, Ethereum blockchain is considered to be the most popular blockchain, for developing smart contracts (Alharby & van Moorsel, 2017), and also boasts a significantly greater number of completed projects, as well as a bigger and more active community (Cai et al, 2018). Furthermore, “mainstream” blockchains, such as Bitcoin and Ethereum, have undergone major scrutiny throughout the past years, unlike other blockchains (Cai et al, 2018).

In conclusion, any of the aforementioned “popular” blockchains (even Corda and Multichain) could, theoretically, have been utilized for this project: they can all be used for the creation of a private chain, they all offer the possibility of creating “smart contracts”, and they all feature suitable consensus algorithms to fit the needs of this project. However, due to Ethereum’s bigger publicity, active community and overall positive reputation, it was eventually selected as the blockchain of choice regarding this project.

5.1.2. Consensus algorithm of choice

As previously mentioned (Chapter 2), Blockchain is updated via the consensus protocol, which ensures a common, unambiguous ordering of blocks and transactions, while also guaranteeing the integrity and consistency of the ledger across geographically distributed nodes (Baliga, 2017). Consensus algorithms help a decentralized network to unanimously take a decision, whenever necessary (Sankar et al, 2017) and ensure decentralized governance, minimum structure, performance, integrity and authentication, as well as non-repudiation and byzantine fault tolerance in a blockchain implementation (Seibold & Samman, 2016). Regarding the Ethereum blockchain, the public chain (which begun its operation in 2015) utilized the Proof of Work (PoW) algorithm, which is still in use, although there are plans for the chain to move to the utilization of the Proof of Stake (PoS) algorithm (Buterin & Griffith, 2017). The PoW algorithm is a “heavy” and resource-intensive algorithm, where participating nodes must calculate the solution of a difficult mathematical problem: the first participant that solves the problem can create the next block - a process also known as “mining” (Mingxiao et al, 2017). PoW algorithms have received heavy criticism due to their time-consuming processes and power-intensiveness (Baliga, 2017), and therefore, a move to the PoS algorithm seems rational. PoS does not utilize a mining process, but adopts a rather alternative approach, which involves a user’s stake or ownership of virtual currency in the blockchain (Baliga, 2017). The concept of “coin age” is used, where the longer a node holds the coins, the more rights it can get on the blockchain (Mingxiao et al, 2017). PoS, therefore, encourages participants to hold their currencies and the blockchain is not entirely relying on a proof of work process (Baliga, 2017).

However, in private blockchains, where the environment is considered to be more confined and trusted, blockchains tend to rely on message-based consensus schemas, rather than hashing procedures, which are lighter and considerably speed up the consensus process, since there is no need for mining (De Angelis et al, 2018). In these settings, Byzantine fault tolerant (BFT) algorithms, such as the Practical BFT (PBFT) and Proof of Authority (PoA) prevail (De Angelis et al, 2018). PBFT algorithm is based on the assumption that less than one-third of the nodes are faulty *(f)*, which means that the network should consist of at least $n = 3f + 1$ nodes to tolerate f faulty nodes (Castro & Liskov, 2002). Thus $f = [(n-1)/3]$ and the network requires $2f + 1$ peers to agree on the block of transactions (Sukhwani et al, 2017). The PoA algorithm, differently from PBFT, has drawn attention due to the fact that it requires less message exchanges, and thus provides better performance and fault-tolerance, while still retaining Byzantine fault tolerance (Dinh et al, 2017). According to Tasca and Tessone (2017), in a PoA implementation, some blockchain nodes are exclusively allowed to create new blocks and secure the blockchain. These nodes “sign” the new blocks with a set of private keys, thus acting as “trusted signers” and every block can be matched against this list of trusted signers (Tasca & Tessone, 2017). The PoA algorithm was proposed as part of the Ethereum ecosystem for the creation of private blockchains and was implemented in Ethereum through algorithms “Aura” and “Clique” (De Angelis et al, 2018), utilized by “Parity” and “Geth” respectively, two well-recognized clients for Ethereum private networks (Dinh et al, 2017).

Since the incident reporting platform will feature an administrative authority performing tasks, such as pre-authorizing participants and other general administration tasks, a PoA algorithm seems like a great fit. Participants are not required to “fight” for mining rewards and tokens (i.e. an example of a token is “Ether”, the native token of the Ethereum network) and they can thus focus on their core goal: to report and review incidents. The selection of a PoA algorithm for this project significantly increases the overall performance and efficiency of the platform, as well as minimizes the power/resource-intensiveness of the solution.

5.1.3. Smart contracts and development language of choice

The Ethereum blockchain, which has emerged as the second generation of blockchain, supports functionality for building complex distributed applications (Alharby & van Moorsel, 2017). This functionality is often referred to as “smart contracts”, which is basically executable code, that runs on the blockchain, in order to facilitate, execute and enforce the terms of an agreement/contract (Xu et al, 2016; Wohrer & Zdun, 2018). Ethereum was the first blockchain to offer such functionality (Hung et al, 2019); according to Bragagnolo et al (2018, p.9), “smart contracts are what embedded procedures are for databases: programs executed in the blockchain to manage and transfer digital assets”.

The development of smart contracts is necessary for creating the incident reporting platform, as they will incorporate the business logic of the application. Data received through the front-end of the application will trigger the execution of these smart contracts, which will interact with the blockchain and more specifically with the “Ethereum Virtual Machine (EVM)”, after the contract’s high-level code has been compiled into bytecode. The EVM is a network of discrete machines in constant communication, although it can be thought as a global decentralized computer, on which all smart contracts run (Wohrer & Zdun, 2018). It handles the state of contracts and computations and is built on a stack-based language, with a predefined set of instructions (opcodes) and arguments (Wood, 2014). Essentially, a contract is a series of opcodes, which the EVM executes in sequential order (Wohrer & Zdun, 2018).

Ethereum contracts can be developed in various supported languages, such as "Low-level Lisp-like Language (LLL)", "Serpent", "Vyper" and "Solidity" (Chen et al, 2017). In any case, the source of a smart contract will be complied into bytecode which will be executed by the EVM (Chen et al, 2017). According to various sources (Chen et al, 2017; Bragagnolo et al, 2018; Wohrer & Zdun, 2018; Hung et al, 2019), "Solidity" is the most popular language amongst developers for the creation of smart contracts, within the Ethereum environment. It is a Turing-complete language, with a syntax similar to common object-oriented languages (Hung et al, 2019) and, as already mentioned, is considered to be the predominant programming language for the creation of Ethereum smart contracts (Bragagnolo et al, 2018). Given Solidity's popularity, many open-source contract-code samples exist, which underwent heavy scrutiny, since a mistake (e.g. a bug) can be very costly after the contract's deployment. As mentioned in chapter two, smart contracts become autonomous entities following their creation (Christidis & Devetsikiotis, 2016; Gatteschi et al, 2018) and before deploying a smart contract, one should inspect it carefully and include fail-safe mechanisms (Christidis & Devetsikiotis, 2016). The smart contracts necessary for the incident reporting platform were developed using the Solidity programming language.

5.2. The decentralized reporting platform: functional requirements

Through the fundamentals described in the previous sections of this chapter, the designed solution has begun to take shape. Essentially, the incident reporting platform will be an application, with a backend and a front-end, accessible to users through the internet. Applications developed on the Ethereum blockchain are usually referred to as "DApps", which stands for "Decentralized Applications" (Warren & Bandeali, 2017). Consequently, this incident-reporting DApp, will be built for the purposes of reporting and reviewing information security incidents, functionalities which will be made available through Solidity smart contracts. The reporting DApp will be deployed on a private Ethereum ledger and will utilize the Proof-of-Authority consensus algorithm.

5.2.1. Basic DApp functionality

The DApp should be easy to navigate and use, and the incident reporting procedures should be straight-forward for all authorized users of the platform. There will be two kinds of users, members and administrators. At the initial stages of the platform there will only be one administrator (the designated authority), but more administrators can be added to the platform, upon need. In order to gain access to the platform, members would have to use an offline procedure to contact the authority, by submitting a registration request to the administrator, via e-mail or any other designated means, and providing all necessary documentation (participation criteria may vary between authorities) for authentication and authorization purposes. The authority would examine the request and consequently approve or deny access to the platform.

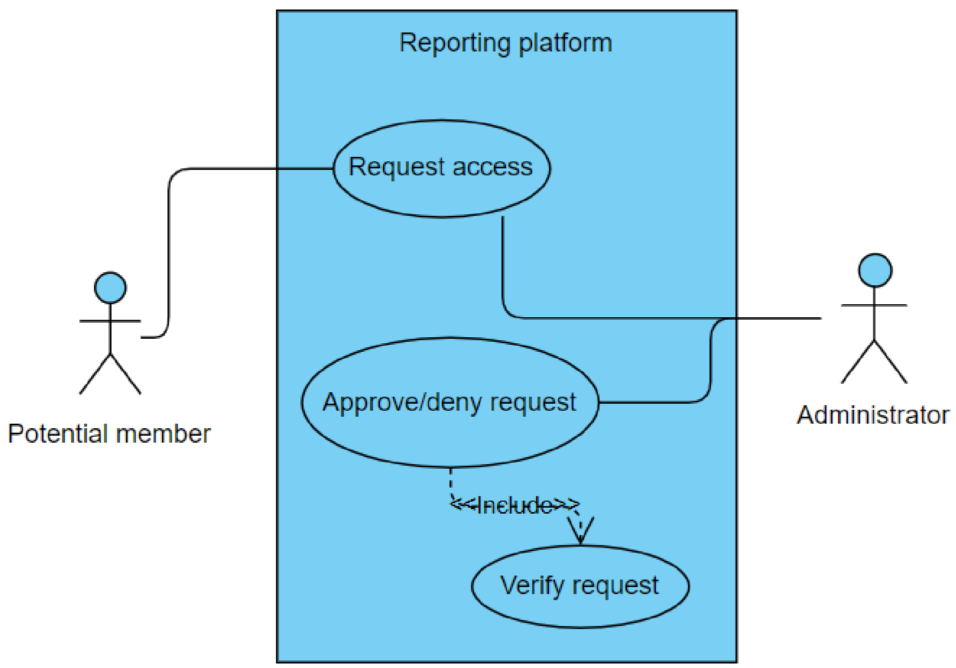


Figure 5.2. Potential member access request use case

An approved member should be able to login to the DApp (with relevant credentials) and perform tasks such as submit an incident, view submitted incidents and search through submitted incidents. The member should also be able to contact the administrator and participate in a live, anonymous, chat.

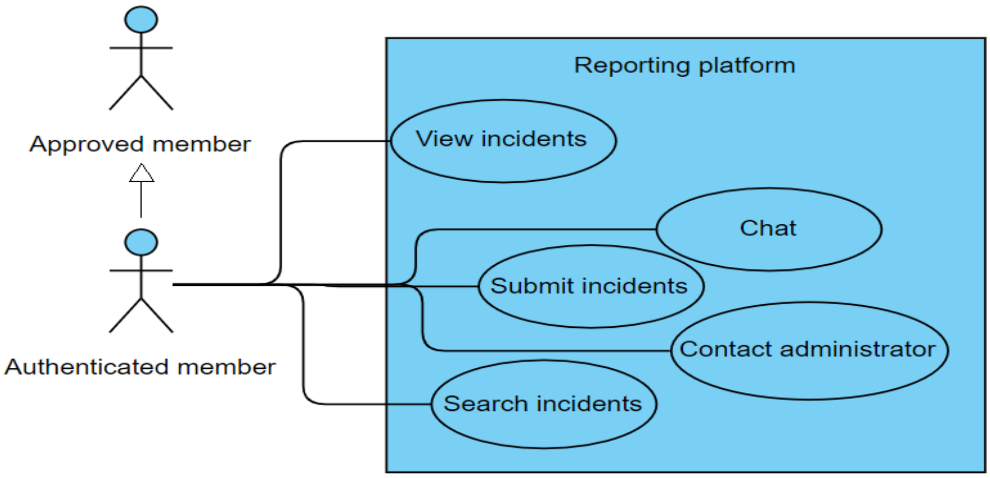


Figure 5.3. Authenticated member available actions use case

A platform administrator should also be able to chat, view, submit and search through incidents, but in addition he/she should also be able to add and remove members from the platform, as well as perform other administrative tasks, such resetting users, increasing the platforms computational capacity etc.

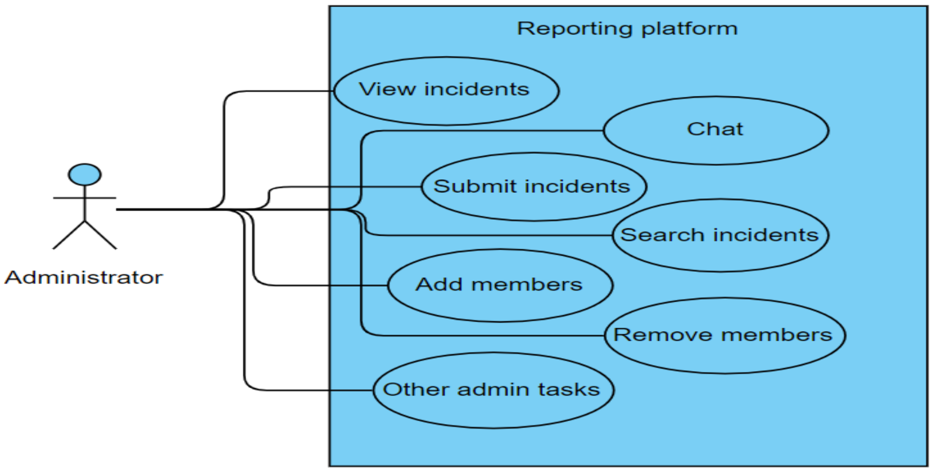


Figure 5.4. Administrator available actions use case

5.2.2. Matching objectives and implementation targets to design elements

In chapter four, the objectives and implementation targets of this research project were set. Consequently, the design procedure should aim to fulfill these objectives and targets, through a careful selection of design/implementation elements. This section lists the objectives and targets, and explains the design/implementation elements which should be utilized in order to fulfill the project's objectives.

a) Objectives

1) "Create an incident reporting solution which enables and encourages the reporting of information security incidents amongst organizations, thereby reducing organizational demotivation for reporting" (keywords: Anonymity, Cost reduction, Creation of artefact)

The proposed incident reporting solution will take the form of an artefact (instantiation). It will be a software implementation in the form of a reporting platform, and more specifically, a private, decentralized application (DApp), on the Ethereum blockchain, accessible to users through the internet. Regarding users' anonymity, each user will interact with the blockchain through a generated address (a public key or a hash of it), which should not reveal the explicit identity of the user. Although, in absolute terms, this functionality ensures pseudonymity, rather than true anonymity of the users, this design decision has been made after taking into consideration that the owner/authority of the platform should be able to identify and take appropriate action towards misbehaving participants. It is important to note that various solutions are available for ensuring the true anonymity of the participants, such as "mixing services", which utilize the grouping of several transactions into a single one (Tasca & Tessone, 2017), "secret sharing", which stores data in a decentralized manner across N parties such that any K parties can work together to reconstruct the data, but K-1 parties cannot (Tasca & Tessone, 2017), and other solutions, such as "ring signatures" (Noether & Mackenzie, 2016) and “stealth addresses” (Courtois and Mercer, 2017). However, true anonymity, in a private environment, could be potentially exploited by misbehaving parties. These parties could deliberately submit spam incidents, or even fabricated incidents, in order to confuse participants, while preparing for a dissimilar attack. This misinformation could eventually demotivate true participants from submitting incidents, thus defying the original purpose of the designed solution, to increase incident reporting. In such an environment, mutual trust between participants (including the validity of their submissions) would be very difficult, if not impossible, to achieve. True anonymity must be sacrificed, in order to create an effective solution. Nevertheless, it is important to note, that besides the central authority, no other participant will be able to identify any other members of the platform, since the only identifiable information, in an incident submission, would be the user’s hash of his public key. This provides an overall acceptable anonymity level and does not compromise the objective of providing an anonymous reporting environment for the users of the platform, since no member can explicitly identify any other member and/or their incident submissions.

Regarding cost reduction, the proposed solution should be easy to understand and use (to minimize training costs) and this can be achieved through designing straight-forward reporting procedures, as well as designing a simple, “clean” and easy to navigate, graphical user interface (GUI). Furthermore, the cost of owning and operating the platform should be significantly less for the participants, compared to existing reporting platforms. This can be achieved by selecting an appropriate development and operating environment for the platform. Details of this environment can be found in section “5.2.4.” of this report.

2) “Create an incident reporting solution which utilizes the positive features offered by existing reporting solutions” (keywords: Efficiency, Performance, Ease of use, Accessibility, Security, Support, Social features)

Regarding efficiency and performance, the selection of a private blockchain implementation - instead of a public one - significantly increases both. In a public implementation, both transactions, as well as the consensus procedure would be slow and resource-intensive. Private blockchain implementations increase the network's performance, efficiency and scalability (Cai et al, 2018) and the selection of the PoA consensus algorithm moves towards the same direction. Regarding ease of use and accessibility, the solution – as already mentioned – should be easy to navigate and use and should be widely accessible: this will be established by creating a DApp accessible from any world-wide location, through the internet.

Regarding security, Blockchain is a shared, tamper-proof replicated ledger where records are irreversible and cannot be forged thanks to the use of one-way cryptographic hash functions (Tasca & Tessone, 2017; Chen et al, 2018; Gatteschi et al, 2018). Transactions need to be reviewed by most of the nodes of the system before they can be recorded (Lu, 2019) – but once data has been recorded in the ledger, it cannot be modified without letting the whole network know, thus permitting tamper-resistance data (Zheng et al, 2017). Blocks that contain invalid transactions can be discovered immediately (Zheng et al, 2017). Users can transfer data only if they possess a private key, which is used to generate a signature for each transaction a user sends out, which is, in turn, used to confirm both the origin and integrity of the transaction (Tasca & Tessone, 2017). Furthermore, all communication channels should be encrypted, and multi-factor authentication should be utilized for authentication purposes.

Regarding user support, a button/link should be made available, through the platform's GUI, for users to contact the platform's administrator. Regarding social features, a live, anonymous, chat room should be implemented and become available to platform users, for them to instantly discuss any incidents. The "Whisper", decentralized communications protocol (Wood, 2015), seems like a good candidate for this task.

3) "Create an incident reporting solution which provides added value to users, in comparison to existing solutions" (keywords: Anonymity, Availability, Auditability/transparency/immutability)

Anonymity has already been discussed above. Regarding availability, constant availability is ensured by the inherent characteristics of the blockchain technology. The failure of a blockchain node does not affect the operation of the whole network, thus ensuring the resilience, availability and reliability of applications built on blockchain, by avoiding single points of failure (Zheng et al, 2017; Chen et al, 2018; Gatteschi et al, 2018). Both public and private blockchain implementations are used to eliminate single sources of failure (Taylor et al, 2019). Auditability, transparency and immutability are also inherent characteristics of the blockchain technology: As already mentioned, blockchain is a shared, tamper-proof replicated ledger where records are irreversible and cannot be forged thanks to the use of one-way cryptographic hash functions (Tasca & Tessone, 2017; Chen et al, 2018; Gatteschi et al, 2018). Blockchain records are auditable by a predefined set of participants, the platform's members. The blockchain technology ensures that nodes record and transfer records on the network and all participants can query these records, which makes information in the decentralized network both consistent and transparent (Lu, 2019). Each node can not only read the final state of transactions, but also the history of the previous transactional states (Gatteschi et al, 2018), while each participant has the same permissions and obligations to access records, and also allow other nodes – on the same network - to access this data (Bonneau et al, 2015; Lin & Liao, 2017). Consensus mechanisms implemented in blockchain structures enable multiple writers to modify the database and provide an authoritative transaction log in which all nodes provably agree (Casino et al, 2019). A private blockchain implementation also prevents the theoretical "51% attack" (which is applicable to public implementations, and could allow the breach of the blockchain's immutability, should attackers gather enough resources to outpace the block creation rate of the rest of the blockchain network), since misbehaving nodes can be removed from the network (Yaga et al, 2018).

b) Implementation targets

1) “Create a manual incident reporting solution” (keywords: Manual)

The proposed DApp will be designed to accept data (incidents) only through human input, through rationally designed forms, and not through automated mechanisms/tools.

2) “Create a software platform for the manual reporting of incidents” (keywords: Reporting platform)

The proposed DApp will feature a reporting platform for the registration of incidents, with an accessible, clean and easy to navigate interface, and with clear submission instructions.

3) “Create a private incident reporting platform” (keywords: Private platform)

The proposed reporting platform will not be open to all. Therefore, a private blockchain implementation will be implemented. As previously mentioned, in order to gain access to the platform, members would have to use an offline procedure to contact the authority, with a registration request. Once approved, members will be able to register their account on the platform.

4) “Create a familiar environment for platform users” (keywords: Familiar structure, functionality, GUI, standardized reporting templates)

The proposed platform will “feel” and “look” familiar to users, as the overall “experience”, structure, graphical user interface (GUI), functionality, and the typical sequence of actions of the designed platform, will match (or very much approach) the existing operational environment of the current reporting platforms. Furthermore, familiar, standardized, and widely accepted reporting templates, will be utilized for the reporting of incidents. The internationally recognized “ISO 27035:2016” incident reporting template (ISO/IEC 27035:2016, 2016) will be utilized for creating the reporting forms, with a minor alteration. This alteration entails replacing the ISO’s proposed incident categories/taxonomy, with the “eCSIRT.net mkVI” taxonomy (Stikvoort, 2015), since the latter is endorsed by ENISA, its categories are universal and practical, and it is widely used amongst European CSIRTs (ENISA, 2018). Since the CYCSO is a possible first candidate for utilizing the proposed artefact, it seems rational to use a taxonomy favourable by both ENISA and the rest of the European CSIRTS.

5.2.3. DApp GUI: content pages & forms

As previously stated, major design requirements for the reporting DApp, are simplicity, ease of use and navigation, as well as creating an overall familiar environment (both visually and in terms of functionality) for the users. When users initially access the DApp, they should be presented with an authentication page, in order to register/authenticate themselves. After successful authentication, they should have access to the homepage of the DApp, with the options of submitting an incident, viewing previously submitted incidents, asking for help, participating in a live, anonymous, chat with other authenticated members, and logging-out of the platform.

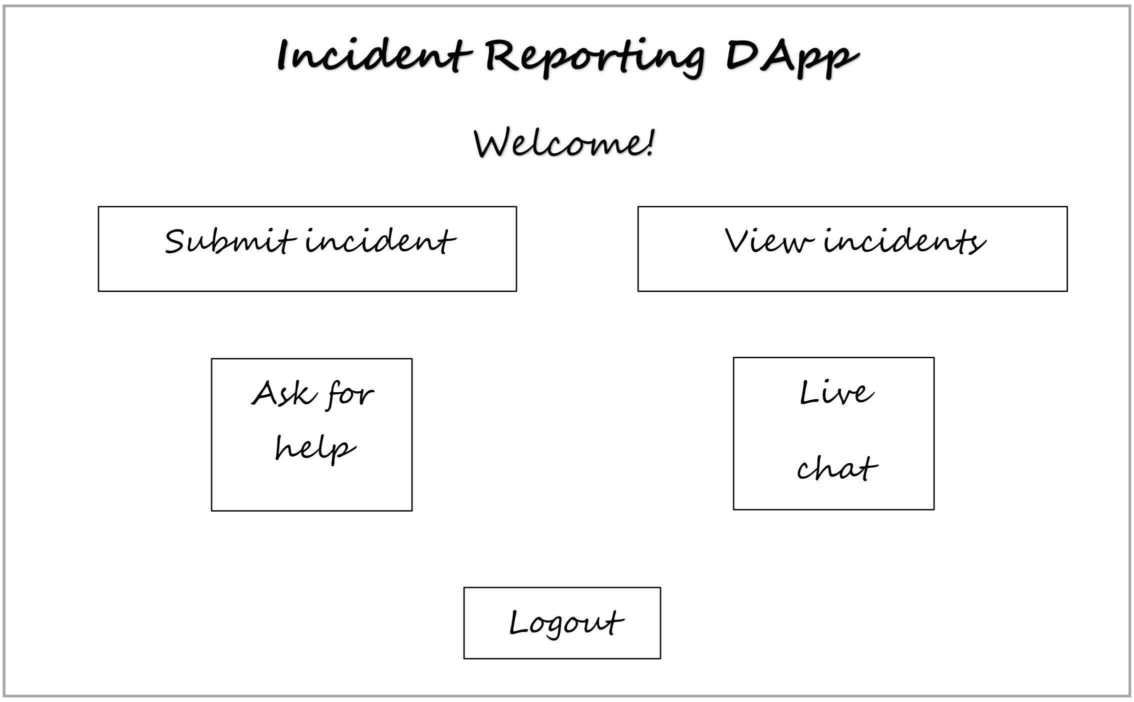


Figure 5.5. Homepage of decentralized incident reporting platform

a) "Submit incident" page

Upon clicking on the "Submit incident" button/link located on the homepage, a separate page will appear to the user. The page will include a simple form with various fields, which follows the structure of the "ISO 27035:2016" incident reporting template, with the addition of the "eCSIRT.net mkVI" taxonomy. It is important to note, that although the ISO reporting template and the "eCSIRT" taxonomy are both well-known and widely-used for incident reporting purposes, users of the platform who are unfamiliar/inexperienced with using these standards, should be adequately trained before using the platform. This training should aid in avoiding the occurrence of any misinformed and/or false positive/negative incident submissions. The fields included in the form are presented in the following table:

No.	Field name	Field type	Required field (Yes/No)	Notes
1	Title of incident	Text	Yes	
2	Incident classification	Radio button (Options: major, minor, suspected)	Yes	
3	Category of incident	Drop-down list	Yes	*See note 1
4	Date/time of incident occurrence	Date/time	Yes	
5	Date/time of incident discovery	Date/time	Yes	
6	Date/time of incident reporting	Date/time	Yes	
7	Short description of incident	Text	Yes	
8	Further description of incident	Text	No	*See note 2
9	Is the incident over	Radio button (Options: Yes, No)	Yes	
10	If yes, how long the incident has lasted?	Text	No	
11	Effect of incident	Checkbox (Options: Breach of confidentiality, breach of integrity, breach of availability, breach of non-repudiation, Destruction)	Yes	*See note 3
12	Person(s)/Perpetrator(s) involved	Radio button (Options: Person, organized group, Legally established organization/institution, accident, No perpetrator e.g. human error/disaster/failure, Other)	Yes	
13	Description of perpetrator(s)	Text	No	
14	Actual or perceived motivation	Checkbox (Options: Criminal/financial gain, political/terrorism,	No	*See note 3

		pastime/hacking, revenge, other		
15	Details of actual/perceived motivation	Text	No	
16	Actions taken to resolve incident	Text	No	
17	Actions planned to resolve incident	Text	No	
18	Other entities notified (e.g. police, regulatory authority)	Text	No	

Table 5.1. Fields of "submit incident" form in "submit incident" page

**Note 1: The available options are described in Table 2.1. "eCSIRT.net mkVI Classification Scheme by Stickvoort (2015)" in chapter two of this report.*

**Note 2: This field includes the following notice to the user: "Consider including what occurred, how it occurred, why it occurred, initial views on components/assets affected, adverse business impacts and any vulnerabilities identified".*

**Note 3: Users can select all options that apply (i.e. more than one).*

At the bottom of the page there will be two options for the user: "Preview & submit" and "Clear form".

It is important to note, that since information submitted on the blockchain through this form will be immutable, they should not contain any sensitive identifiers which may require alterations, such as Personally Identifiable Information (PII). Considering the privacy regulations applicable around the world (e.g. GDPR in Europe) and also the user anonymity requirement of this project (as stated in chapter four), the "submit incident" page has been designed in a way not to explicitly require any such data. However, since some form fields allow users to freely submit content (e.g. field 13 – "Description of perpetrator(s)"), users should be very cautious as to avoid the submission of any such identifiers.

b) “View incidents” page

Upon clicking on the “View incidents” button/link located on the homepage, a separate page will appear to the user. This page will contain a simple, searchable, array, presenting previously submitted incidents, in chronological order (newest on top), by all platform users. Users will be able to search through the array, by using appropriate keywords. Moreover, clicking on a displayed result will reveal the complete incident form. The following table presents the fields of the “view incidents” array:

Field name	**Field type**	**Notes**
Number of incident	Numeric	Each submitted incident will be assigned a unique number, in ascending order, beginning with integer number 1. Newer incidents should appear first in the array
Title of incident	Text	This field will contain the title of the incident, as submitted through the incident’s individual form
Incident classification	Text	This field will contain the classification of the incident (e.g. “major”)
Category of incident	Text	This field will contain the category of the incident (e.g. “phishing”)
Date/time of incident	Text	This field will contain the date/time of the incident
Short description of incident	Text	This field will contain the short description of the incident
Person/perpetrator involved	Text	This field will contain the perpetrator of the incident (e.g. “person”)
Actual/perceived motivation	Text	This field will contain the motivation of the incident (e.g. “criminal/financial gain”)

Table 5.2. Fields of “view incidents” array in “view incidents” page

c) “Ask for help” page

Upon clicking on the “Ask for help” button/link located on the homepage, a separate page will appear to the user. This page will contain a simple, standardized contact form, in order to enable the user to contact the platform’s administrator. Communication submitted through this form will be sent to the administrator's designated e-mail address. The following table presents the fields of the contact form:

Field name	**Field type**	**Required field (Yes/No)**
Name	Text	No
E-mail	Text	Yes
Message	Text	Yes

Table 5.3. Fields of "contact us" form in "Ask for help" page

d) "Live chat" page

Upon clicking on the "Live chat" button/link located on the homepage, a separate page will appear to the user. The "Whisper", decentralized communications protocol (Wood, 2015), will enable authenticated platform users to instantly chat between them, while on the platform. The inherent properties of this communication protocol allow users to remain anonymous while chatting.

e) Registration/login page

The registration/login page should appear to non-authenticated users. Users should be able to create an account (subject to limitations presented in 5.2.7) using their e-mail address, a password and a one-time-password/token (provided by an authentication service). Registered users should be able to login to the platform using their previously registered e-mail and password, along with a newly-generated one-time-password.

f) Administration pages

The platform's administration page should allow an administrator to view existing users of the platform and add/remove users from the platform. Through the Azure administration console, an administrator will be able to perform additional tasks, such as increasing capacity, adding/removing blockchain nodes and viewing network/utilization statistics.

5.2.4. Development environment of choice

a) Blockchain as a Service (BaaS)

The proposed artefact could have been developed and deployed in a local environment. However, the emergence of various “Blockchain as a Service (BaaS)” platforms, with similar functionality to the widely known “Software as a Services (SaaS)” model, seemed to have brought many advantages to blockchain development. According to Samaniego and Deters (2016), local blockchain installations could lack sufficient computational resources and bandwidth, as well as consume significantly more power. Utilizing a BaaS approach, means that an external provider is responsible for configuring all underlying infrastructure for a blockchain deployment, with a small (usually, monthly) fee. The provider caters for hosting requirements, while properly manages bandwidth and the allocation of resources. Furthermore, the overall level of security is enhanced, since the BaaS operator usually offers advanced security features for its infrastructure, compared to local environments. However, some organizations approach the cloud model with caution, since along with all the benefits it brings, it may also introduce new security risks (Rebollo et al, 2015), especially regarding the confidentiality of data. Thus, the forms of the incident reporting DApp have been designed in a way to minimize the risk of exposure of confidential data, should a breach occur. Since the forms do not to contain any personal/organizational identifiers (other than the hash of the public key of a participant), even if the incident data eventually end up in the wrong hands, there would be no way for the malicious party to associate the data with a particular user/organization, since the user/ key pair would only be known by the platform’s central authority and it would be stored in an offline location.

By utilizing a BaaS model, the developer can freely focus on building the core blockchain product, without having to worry about performance and other infrastructure-related issues. BaaS providers may also offer additional services, such as blockchain authentication, governance and storage features/modules. The BaaS model, with its various features, can save time and reduce cost for a developer. Many world-renowned companies offer BaaS: Microsoft, with its “Azure Blockchain-as-a-Service product” (Microsoft, 2019), Amazon, with its “Blockchain on AWS” product, (Amazon, 2019) IBM, with its “IBM Blockchain Platform” (IBM, 2019), and Oracle with its “Oracle Blockchain Platform” (Oracle, 2019) are just a few examples. The proposed reporting solution will be built and deployed on the Azure BaaS platform, since Microsoft offers support for creating private Ethereum blockchains utilizing PoA consensus algorithms, and also offers some other useful features, including a set of pre-configured Solidity smart contracts for blockchain deployment, as well as a portal for the governance of private blockchains. Regarding portability (i.e. moving to another BaaS provider, if deemed necessary), it is certainly not the easiest of tasks, since each provider offers its own, distinct, environment and operational settings. This lack of standardization is by some means expected, since the BaaS model is still in its infancy. Nevertheless, Azure allows the direct download of any previously created custom configurations, including the configuration settings of any blockchain services, nodes, virtual machines, networks and interfaces. The DApp’s incident data and smart contracts can also be traced/downloaded from the blockchain itself. Utilizing the above information and settings, an organization would be able to recreate the blockchain environment using a different BaaS provider, although seamless integration is still not possible, under the current circumstances.

The solution will be built on the Quorum blockchain, a permissioned ledger implementation of Ethereum, developed by JP Morgan (Baliga et al, 2018) and offered by the Azure BaaS. Quorum utilizes a PoA type of algorithm, called IBFT, and supports privacy and confidentiality of both transactions and smart contracts (Baliga et al, 2018).

b) Development frameworks

Microsoft's Visual Studio Code (VS Code), configured to use "Azure's Blockchain Development Kit for Ethereum" extension, will be used for creating, connecting, building, and deploying the smart contracts to the Quorum ledger. The extension includes the necessary packages for the installations of Node.js, Git, Python, Truffle and Ganache, all of which are required throughout the entire contract's lifecycle, from coding to deployment.

The application's front-end will be build with HTML, CSS and JavaScript. Firebase development platform (offered By Google) will also be used for authentication-related tasks.

c) Web3.js libraries

Web3.js is a collection of libraries facilitating connections to Ethereum nodes. Web3.js will be utilized to relay the blockchain transactions to the underlying peer-to-peer network. Furthermore, in the web interface, Web3.js will be combined with Metamask (a popular Ethereum wallet and Web3-enabled browser), to enable interaction with the Quorum network.

5.2.5. Code re-use and editing of incident submissions

Due to its inherent characteristics, the internal logic of an Ethereum smart contract cannot be altered, once it has been deployed (Warren & Bandeali, 2017). Thus, extreme care should be taken by the contract's developer(s), before deployment, to ensure that the contract code both fully satisfies the intended business logic, as well as is free from any programming bugs or security vulnerabilities. Code-reuse (as regards to code which has undergone heavy scrutiny) is therefore highly recommended in the Ethereum community, while the initial vision of the smart contract ecosystem included contract reusability (Pontiveros et al, 2018). Consequently, the practice of code re-use will be utilized as much as possible throughout the development of smart contracts for the incident reporting DApp, by using well-known and open-source, libraries and code repositories.

Along the same lines, and to increase the platform’s transparency, the users’ confidence level and their overall sense of trust towards the platform, it was decided to disallow the editing of previously submitted incidents, although such functionality was, indeed, programmatically feasible. Instead, a notice/warning, which will appear after the user clicks on the ““Preview & submit”” button of the incident submission form, will alert users towards carefully reviewing their form’s content, before proceeding to the final (and non-amendable) submission of the incident.

5.2.6. Storage considerations

Although data submitted on the blockchain will be stored in a decentralized fashion, the front-end components of the application will be stored in a traditional, centralized environment, utilizing the researcher’s hosting provider.

5.2.7 Authentication considerations

There will be four levels of user verification/authentication. Initially, the user should submit an offline verification form, which needs to be approved by the administrator. Secondly, the approved user will have to create an Ethereum wallet and forward his/her public key to the administrator. The administrator will include this key in a “white-list”. If a user’s key is not included in the authentication server’s white-list, the user will not be able to register an account on the DApp’s homepage. The user will then have to register an account, using an e-mail address, a password and a one-time-password, received from Firebase’s authentication service.

Registered users will be able to login to the platform using their previously created e-mail address and password, as well as a one-time-password.

5.2.8. Viewing blockchain transactions

Since transactions are going to be private, they will not appear into Ethereum’s main network and thus users cannot view them using popular Ethereum transaction search engines, such as “Etherscan” or “Ethplorer”. It is, however, necessary, at least for the administrator, to have access to the history of the transactions (which include details such as the transactions' hash values and the associated users' and Ethereum contract addresses), in order to detect misbehaving nodes and reprimand/remove them. A way for viewing transactions, would be to use Ethereum's "Geth" client, in order to download the blockchain's data blocks and then extract (and examine) the transactions from each block. Nevertheless, Azure supports the integration of a third-party tool, called "Epirus Azure Blockchain Service Explorer", which appears to be a better option, since it provides a convenient GUI for viewing all details of the transactions submitted on the blockchain. "Epirus" will therefore be integrated into the Azure BaaS implementation and will be used for viewing/examining the transactions submitted through the platform.

5.2.9. DApp architecture and ecosystem

The following figures depict the high-level architecture of the decentralized incident reporting platform, as well as the DApp's ecosystem:

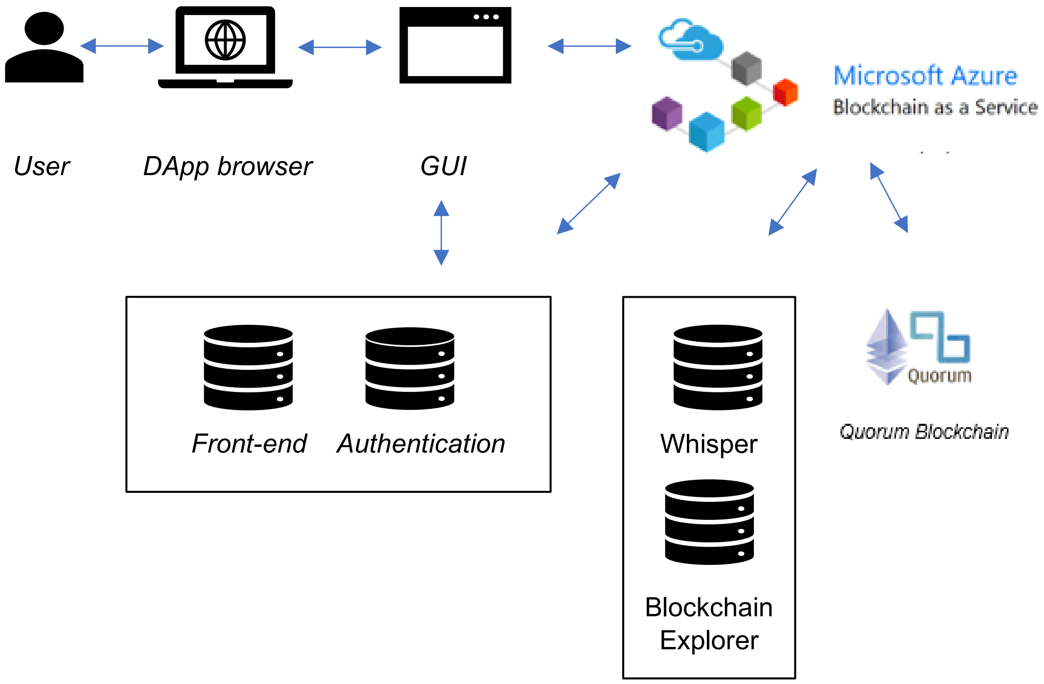


Figure 5.6. Architecture of decentralized platform

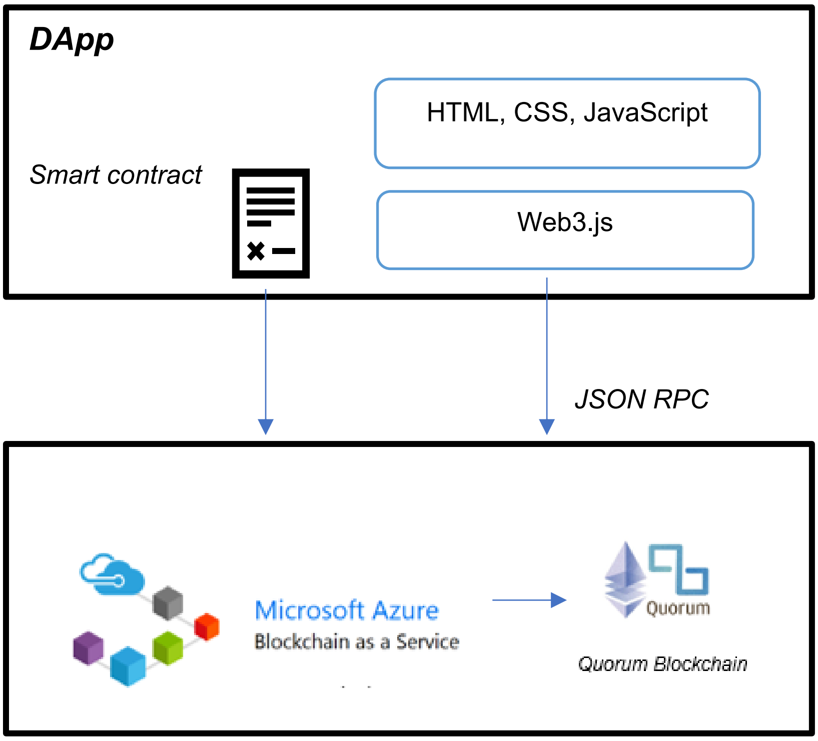


Figure 5.7. Ecosystem of decentralized platform

5.3. The decentralized reporting platform: non-functional requirements

a) Usability

The GUI will have a clean design - it will be easy to understand, use and navigate the reporting DApp. Menus will be easily identifiable and situated in noticeable locations. Text fonts will be the same throughout the DApp, while the images, buttons, background colours and notices will have a uniform look and feel across the platform.

b) Accessibility

The platform will be accessible through common web browsers (e.g. Google Chrome, Internet Explorer, Mozilla Firefox), with the appropriate Web3.0 extensions. Although every effort will be taken to cater for mobile-device accessibility, such a feature cannot be guaranteed to users of the platform.

c) Availability

The blockchain will be constantly and uninterruptedly available to users (100% availability). However, since the front-end GUI will not be stored in a decentralized fashion, its availability is guaranteed at 99.9% (provided by the researcher's hosting provider).

d) Reliability

The platform will be easily recoverable and portable, in case a change of operating environment and/or provider is deemed necessary. Adequate care will be taken to ensure that the platform and its associated software are free from programming errors (bugs) and the developed code is well-commented.

d) Performance

The platform will provide service/performance monitoring statistics and will be capable of handling an average of 100 transactions per second, with a network deployed across purely European regions.

e) Security

Communication between the users and the platform will be encrypted using adequate and up-to-date cryptographic standards. Multi-factor authentication will be required for platform registration/login purposes. Regarding development, secure coding practices will be utilized throughout the project, with a heavy investment in code-reuse. Infrastructure security shall be managed by Microsoft, which "invests over a billion dollars every year regarding security" and has "over 3500 cyber security experts" in its employment (Ben-Menahem, 2018). Furthermore, and according to Ben-Menahem (2018), Microsoft-managed networks and customers networks are isolated in Azure, while, "the network cabling, the equipment to support and secure the network, and the integration of systems for monitoring the network are managed by Microsoft". According to the same author, customer networks are also isolated from each other, through virtualization methods, while Azure also employs built-in mechanisms to protect against DDoS attacks. In addition, security controls are integrated into the firmware and hardware of Azure, while the platform offers support for both software and hardware-based Trusted Execution Environments (TEEs).

5.4. The decentralized reporting platform: Implementation

This section describes the implementation activities undertaken, in order to create the decentralized, incident reporting platform.

5.4.1. Deploying Azure Blockchain Service (ABS)

The initial activity for creating the platform, was to deploy the managed blockchain service of Azure. This was established through the Azure portal. Quorum was selected as the desired deployment blockchain, as well as a basic environment configuration, which included a validator and a transaction node, 1 vCore VPS and 5GB of storage. These resources (and all other resources deployed throughout this project) were deployed in locations described by Microsoft as “West Europe”, in an effort to keep all data within the European geographical area. By default, ABS ensures that the nodes are patched with the latest host operating system and blockchain software stack updates, while transaction nodes are secured through firewall rules and data in motion is encrypted through TLS.

A new consortium was created, with a single blockchain member. Since this is a proof-of-concept artefact, rather than a production-ready software, no other nodes were added to the consortium, due to cost considerations. In an ideal scenario, however, each member of the blockchain would have its own node. ABS provides built-in governance controls for the consortium, through pre-defined Solidity smart contracts, which allow consortium management actions, such as adding and removing members (nodes). These actions can be initiated through PowerShell (and a REST API) and therefore the administrator can programmatically manage a consortium using common interfaces, rather than through submitting smart contracts. The ABS environment also provides metrics, through the Azure Monitor Service, which provide details about the nodes’ storage usage, memory and CPU utilization, as well as blockchain network activity, such as active connections and count of transactions and blocks.

5.4.2. Creating and deploying the smart contract

In order to create and deploy the required smart contract, Microsoft’s Visual Studio Code (VS Code) was utilized, which was configured to use “Azure’s Blockchain Development Kit for Ethereum” extension. Through the development kit, a connection to the previously created consortium was established. Only creating a simple, storage, contract was required, for the needs of this project (since ABS deployment includes a number of pre-deployed, administration-related smart contracts).Thus, it was sufficiently straight-forward to identify an existing, audited, and publicly scrutinized sample, and modify its content accordingly. The sample contract used, was created by ChronoBank, and is available in their Github repository (ChronoBank, 2018).

The contract for this project was written in the Solidity language and features three basic functions:

```
function submitReport (string calldata _ttl, string calldata _reportJSON) external {
  _index++;
  _reports[_index] = Report(_ttl, block.timestamp, _reportJSON);
  emit EvtReport(_index);  }
```

The “submitReport” function displayed above, is activated when a user submits a new incident through the DApp’s GUI. The “ttl” and “reportJSON” hold the data the user has submitted, which the contract saves in “_reports”, while “emit” triggers the event.

```
 function getReport(uint256 reportID) external view returns (
   string memory ttl,
   uint256 dtsubmit,
   string memory reportJSON
 ) {
   Report memory report = _reports[reportID];
   return (
     report.ttl,
     report.dtsubmit,
     report.reportJSON
   );  }
```

The “getReport” function displayed above, returns the incident report data (the title and details of the incident, along with the timestamp) according to the report ID.

```
 function getReportsCount() external view returns (uint256 ret) {
   return _index;
 }
```

Finally, the “getReportsCount” function, simply returns the count of the submitted reports.

The entire contract is available in Appendix B.

Furthermore, Unit testing was performed, to ensure that the contract behavior was as intended, before deployment. A total number of eight tests were performed, and calls from test-users were emulated. Two tests checked the state of the chain before adding a report (pre-submit tests), three tests checked the state after adding a report (submit report), and three tests checked the state after submitting a second report (submit 2nd report). All scenarios were successful, as evident in the following figure:

```
Contract: ReportsStorage
  o Empty state checks
    ✓ initial reports count should be zero (56ms)
    ✓ get empty report data from contract
  o Submit report
    ✓ should submit report (151ms)
    ✓ reports count should be increased after submit (49ms)
    ✓ check report data from contract (94ms)
  o Submit 2nd record
    ✓ should submit report (129ms)
    ✓ reports count should be increased after second report (40ms)
    ✓ check 2nd report data from contract (123ms)


8 passing (1s)
```

Figure 5.8. Unit testing results of smart contract

The JavaScript code used for executing the tests can also be found in Appendix B.

The final step involved uploading the contract's code to Azure's Blockchain Development Kit, compiling the contract with Truffle, and deploying the contract to the blockchain, utilizing Truffle's migration scripts.

5.4.3. Creating the GUI

The application's GUI was built using HTML, CSS and JavaScript and hosted on a server which the researcher utilizes under his professional capacity. The GUI can be accessed at https://alexis-michael.eu/reporting. All requirements, as defined during the design phase of the project, were successfully implemented. The rest of this section presents a selection of implementation activities:

a) Login/register page

The authentication page was created using the Firebase development platform, owned by Google. Firebase has ready-made (boilerplate) scripts, for easily integrating authentication mechanisms (including modules for multi-factor authentication), in both web and mobile applications. The following figure depicts the GUI component created for handling user authentication:

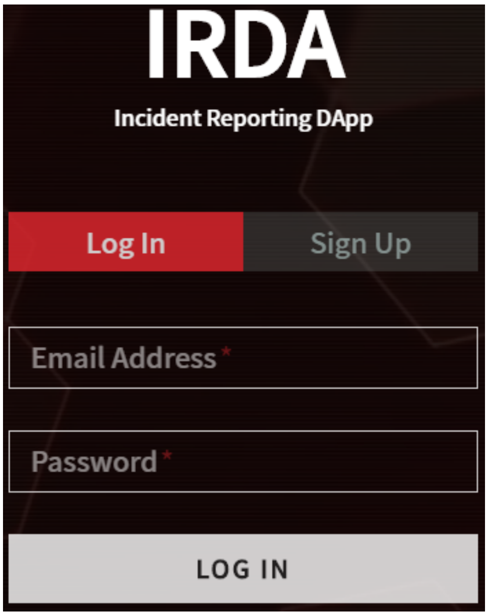


Figure 5.9. Firebase authentication component

b) Homepage

The below code segment is a snippet of the full code, which in conjunction with other scripts (e.g. CSS), were used to create the members' homepage:

```
<div id=pagehome>

  <div id=homehead>

    <h1> Incident Reporting DApp</h1>  </div>

  <div id=homeuser>

    <div id=hiuser>Welcome!</div> </div>

  <div id=homesubm><input type=button class=homebtn data-open='submit' value='Submit incident' id=opensubmit title='Click to submit a new incident report'></div>

  <div id=homelist><input type=button class=homebtn data-open='list' value='View incidents' id=openlist title='Click to see the list'></div>

  <div id=homehelp><input type=button class=homebtn data-open='help' value='Ask for help' id=openhelp title='Click to send us a message'></div>

  <div id=homechat>

    <input type=button class=homebtn data-open='chat' value='Live chat' id=openchat title='Click to chat'>

<span id=chatunread data-open='chat' class=hidden></span> </div>

  <div id=homebye> <input type=button class=homebtn data-open='bye value='Logout' id=logout title='Click to log out'></div>  </div>
```

The following page is displayed to the user after successful login:

Figure 5.10. Incident Reporting DApp's homepage

c) Submit incident page

The below code segment is a snippet of the full code, which in conjunction with other scripts (e.g. CSS), were used to create the submit incident page:

```
<div class=wincontentw>
    <div class=block>
      <div class="screen-title">
        <h2>Report new incident</h2>
      </div>  <div id=thetable class='grid2 gridtable'>
        <div class='gridline gridline1 required' data-rown='1'><span id=ttlttl>Title of incident</span>:</div>
        <div class='gridline gridline1' data-rown='1'>
          <textarea name=inpttl id=inpttl class=textareasmall required></textarea>
        </div>  <div class='gridline gridline2 required' data-rown='2'><span id=ttllevel>Incident
classification</span>:</div>
        <div class='gridline gridline2 spacebetween' data-rown='2'>
          <div><input type=radio name=inplevel id=inplevelmajor value='major'> <label for=inplevelmajor
id=vallevelmajor>Major</label></div>
          <div><input type=radio name=inplevel id=inplevelminor value='minor'> <label for=inplevelminor
id=vallevelminor>Minor</label></div>
          <div><input type=radio name=inplevel id=inplevelsuspected value='suspected'> <label
for=inplevelsuspected id=vallevelsuspected>Suspected</label></div> </div>
```

The following page is displayed to the user for submitting an incident:

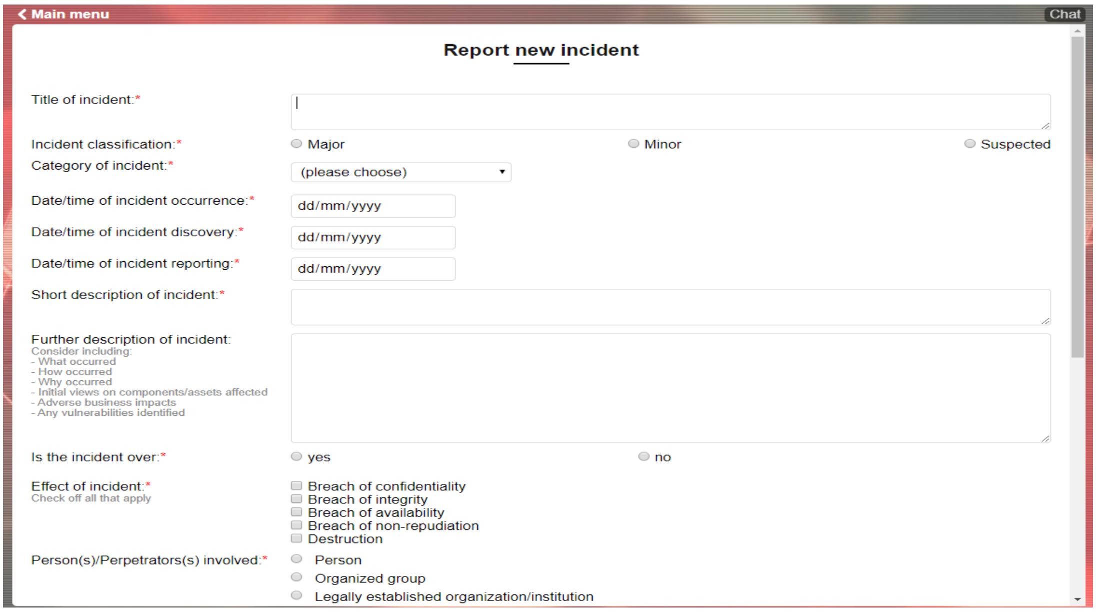


Figure 5.11. Submit incident page

d) View incident page

As per design requirements, a list of all the incidents appears to the user. The user can click on each table entry to view the full report.

14	Yet another test	Major	Malicious Code / Worm	Wed Nov 27 2019, 12:40:26 PM	Another test submitted on this platform	Person	really have no idea
13	My phone got broken	Minor	Availability / Outage (no malice)	Mon Oct 21 2019, 12:20:36 AM	I've tried to wash my phone and it just broke down	Person	I've been trying to clean it
12	No accident just kidding	Suspected	Other / Other incidents	Mon Oct 21 2019, 12:17:56 AM	Joking yo	Other	no idea
11	I've had a funny dream	Suspected	Vulnerable / Open for abuse	Mon Oct 21 2019, 12:15:01 AM	I've seen a dream about UFO	Accident	
10	A dog said moo	Minor	Fraud / Masquerade	Mon Oct 21 2019, 12:11:06 AM	A dog pretended to be a cow	Accident	
9	I've got a really weird call	Suspected	Information Gathering / Social engineering	Mon Oct 21 2019, 12:04:51 AM	I've got a phone call and there was a dog talking to me	Other	No idea
8	new incident	Suspected	Availability / DDoS	Tue Oct 08 2019, 8:56:16 PM	new incident occurred	No perpetrator (e.g. human error/natural disaster/equipment failure)	unknown
7	A car was remotely hijacked	Major	Intrusions / Privileged account compromise	Mon Oct 07 2019, 9:36:31 PM	Though I hadn't touched the dashboard, the vents in the Jeep Cherokee started blasting cold air at the maximum setting	Legally established organization/institution	It was just an experiment
6	1111			Mon Oct 07 2019, 9:11:21 PM			111
5	Somebody has eaten my cake	Suspected	Fraud / Unauthorized use of resources	Mon Oct 07 2019, 8:23:31 PM	I came home and the cake was not there	Other	They must have been hungry
4	Datacenter got flooded	Major	Availability / Outage (no malice)	Mon Oct 07 2019, 5:14:01 PM	A janitor dropped their bucket full of water when washing our ceiling and flooded it all	No perpetrator (e.g. human error/natural disaster/equipment failure)	
3	Some funny incident			Mon Oct 07 2019, 1:25:21 PM	That was just weird!		
2	Our webserver got owned	Major	Intrusions / Privileged account compromise	Mon Oct 07 2019, 11:17:06 AM	Webserver got owned via a vuln. in Exim mail server	Other	They've clearly used the server for mining; other possible aims include spam and stealing the data
1	A silly cat has eaten my hat	Minor	Information Gathering / Sniffing	Mon Oct 07 2019, 10:28:41 AM	I came home and found out that my cat has managed to eat my hat	Other	I have no idea

Figure 5.12. View incidents page

e) Chat page

A simple chat window was created for implementing the GUI of Whisper chat:

```
<h2>IRDA chat</h2> </div> <div id=chatroom> <div id=chat>

              <div class=hint>There were no messages since you've signed in</div>

            </div> <div id=chatbox> <input name=chatsend id=chatsend placeholder='Your message'
onkeyup='chatTyping(event);'>

              <input id=chatsendbtn type=image src='assets/img/sendbl.png' onclick='chatSend();' title='Send message'>
```

The following page is displayed to the user when selecting to chat:

Figure 5.13. Chat page

5.4.4. Utilizing Web3

In order to connect the web interface to the smart contract, the utilization of a lightweight, stateless, RPC protocol was necessary. JSON-RPC is such a protocol, and Web3 is the Ethereum-compatible API, built using the JSON-RPC specifications. Initially, Web3.js was installed through Node.js. Since smart contracts operate on large integer numbers, the “Bignumber” library was utilized for JavaScript compatibility reasons. The next step involved creating a Web3 instance and setting a provider, to allow users to interact with the contract from the web interface. This provider, once installed by the users of the platform, adds an Ethereum object to the browser’s main window object. If no such object is detected by the platform, the user is instructed to install Metamask (or other compatible browser). Metamask has an “enable()” function, which returns the user’s public key (after the user has approved the action), which is necessary both for authentication purposes, as well as for signing transactions. In order to allow the JavaScript front-end to communicate with the smart contract, the “ABI definition” of the contract was needed. This ABI definition file was automatically created by Truffle (in VS Code) and was then imported into the JavaScript code.

Web3 was also utilized for building Whisper, the anonymous chat of the platform. Whisper is a peer-to-peer messaging protocol for DApps and provides a simple API (called “web3.shh”) for sending and receiving messages in secrecy. The chat instance was created using sample code from “Status”, an open source messaging platform to interact with decentralized applications that run on the Ethereum Network (Status Network, 2017). The first step involved creating a new virtual machine in Azure, which would act as a Whisper node, exposing an RPC interface. The next step involved creating a Web3 connection with this newly created node. A keypair was then created, for the signing of messages to be sent. A symmetric key was also created, to encrypt messages which should be received by anyone listening to the channel. A public key was also generated, in order to identify messages that are sent over the channel. In order to send a message, the “web3.shh.post” function is used, which signs the message with the previously created keypair. In order to view messages, the “web3.shh.subscribe” function is used, which subscribes to the messages received by the symmetric key.

A final use of Web3 in this project is for authentication purposes, and more specifically, to detect whether a visitor's public key is white-listed and the visitor can therefore proceed to the registration/login procedure of the platform. The Web3 function returns the public key of the visitor's provider (e.g. Metamask). The key is then compared to the entries of the authentication database (Firebase). If a match is found, the visitor can proceed to register/login to the platform or else the visitor is informed that his/her wallet address is not white-listed.

5.4.5. Other implementation actions

a) Authentication procedure

As already mentioned, Firebase was used for implementing the various authentication features of the platform. The platform was registered to a new Firebase project, the necessary authentication JavaScript libraries were installed and "alexis-michael.eu" was listed as an authorized domain. A firebase database was also created to store the authorized users. Firebase supports a variety of authentication methods including "Email/password", "Phone", "Google", "Facebook" and many more. For the incident reporting platform, a combination of "email/password" and "phone" was selected (enabled through the Firebase console), in addition to the whitelisting of the public key. This means, that in order for a user to register/login to the platform, he/she would need to have a valid public key, a valid e-mail/password combination, as well as a valid one-time-password (in the form of a six-digit code, received through an SMS). The Firebase free account allows sending up to 500 SMS messages, per IP address, per hour, which are more than enough for the working prototype version. A "white-listed" user can register for a new account, through the DApp's GUI. He/she would need to provide an e-mail address, a password and a mobile phone number, for receiving the one-time-passwords. One-time-passwords received by the users, need to be typed in a drop-in widget, which appears to the screen, after either a successful registration request, or after an e-mail/password match (during a login attempt). All authentication credentials of a user are linked to a single Firebase user ID, which enables swift user management. Finally, Firebase's “reCAPTCHA verifier” was also implemented, to avoid misuse of authentication tokens.

b) Administration panel

Firebase was configured to provide administration-related functionality to the incident reporting platform, such as adding and removing users and administrators. The firebase database contains two levels of users: administrators and common users. As previously mentioned, all authentication credentials of a user are linked to a single Firebase user ID, including the user’s public key. This simplifies the task of removing users from the platform, since they can be removed with a single action, through Firebase’s simple GUI. However, when it comes to adding new users to the platform, through Firebase’s GUI, the task gets more complex: the administrator needs to add a new document to the database, and define four different fields (name, id, timestamp, address), as evident in the following figure:

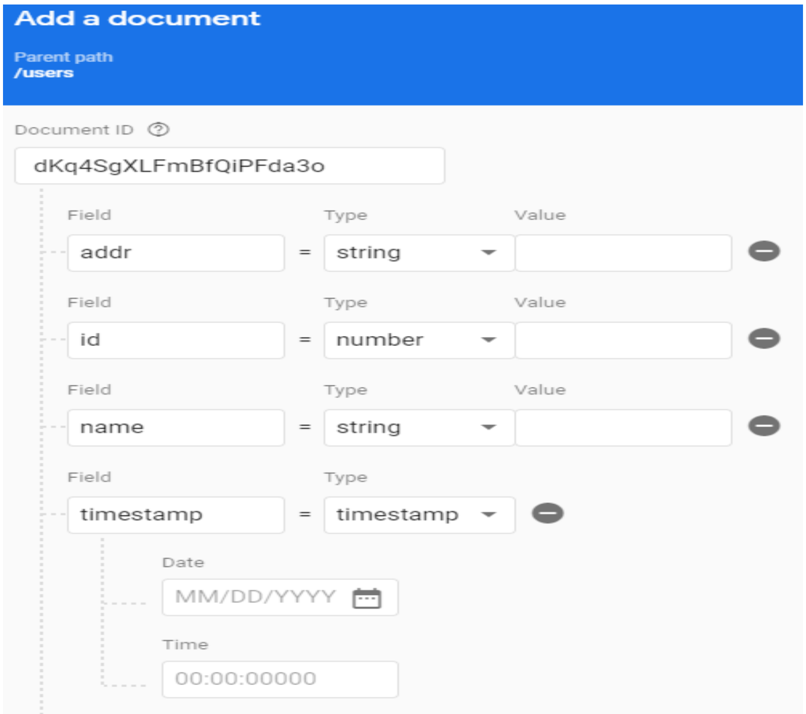


Figure 5.14. Adding new user through Firebase’s GUI

Although the process is not particularly difficult for IT literate administrators, it might seem complex for some users. Therefore, a separate administration panel was created, which is accessible at: https://alexis-michael.eu/reporting /admin. This panel, which is linked to the same Firebase database, displays a simple list of the users (name and public key) and the administrator can add or remove users, as well as search the database based on a user's public key:

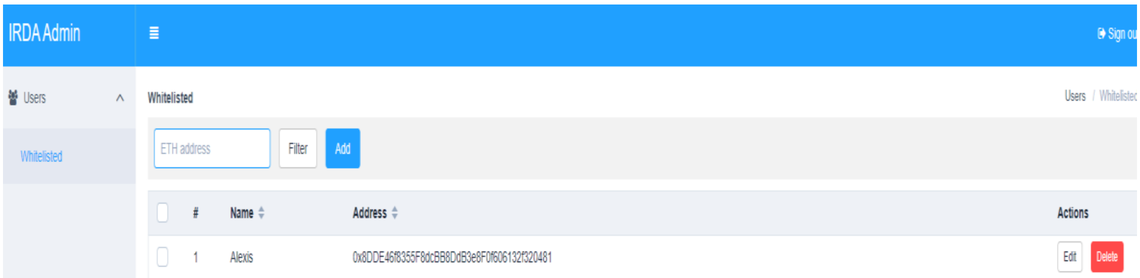


Figure 5.15. IRDA admin panel

The reporting platform's administrator can therefore simply add a new user through this panel (stating only the user's name and public key) and the Firebase database will automatically be populated with this new user. In this way, the administrator essentially "white-lists" a user. The user then needs to register through the platform's GUI (by providing an e-mail/password/mobile phone number) and he/she will be able to immediately use the platform. The administrator can also remove a user through this panel, by deleting his public key from the list. The next time a non-whitelisted (but registered) user tries to login to the incident reporting platform, a script will detect this activity and automatically remove that account from the Firebase database. Users can also be manually deleted, through the Firebase console.

c) Contact page

The incident reporting platform features a simple contact form, where users can communicate (through e-mail) with the administrator:

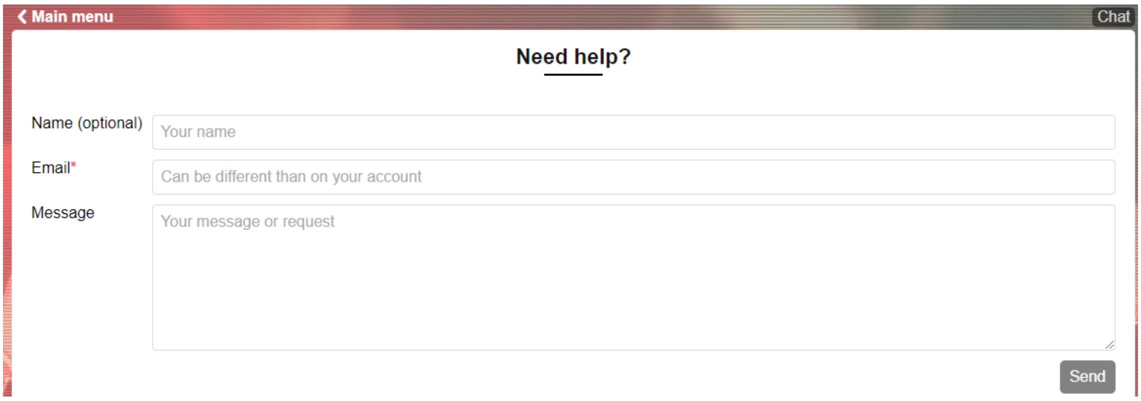


Figure 5.16. Contact/ask for help page

The form was configured to operate with “Mailgun”, an automation service offering e-mail related services for websites and applications (Nabors, 2017), since Mailgun both offers an intuitive API, as well as allows the delivery of ten thousand e-mails for free, per month.

d) Blockchain explorer

In order to view the transactions in the private blockchain, “Epirus Blockchain Service Explorer” was deployed in the Azure environment, which required a separate virtual machine. The GUI of Epirus (available at: http://epirus-8ea3d7.westeurope.cloudapp.azure.com) allows both the administrators and the users of the platform to view the transactions submitted on the blockchain.

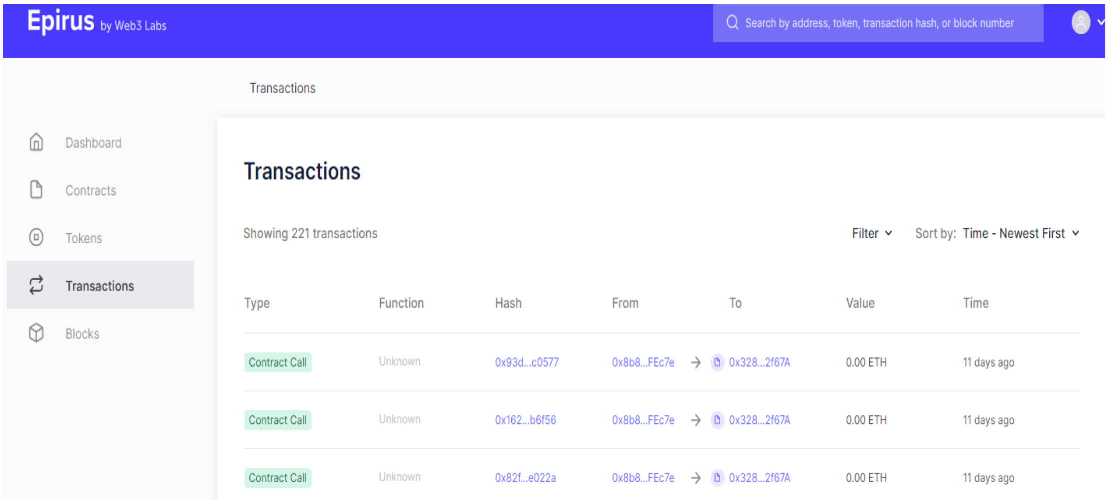


Figure 5.17. Epirus Blockchain Service Explorer with sample transactions

6. THE DECENTRALIZED SOLUTION: DEMONSTRATION

According to Peffers et al (2007) framework, the fourth activity of a design science research project involves demonstrating how the produced artefact solves one, or more, instances of the defined problem, and this could involve the artefact's use in a case study, a simulation, in experimentation, or any other appropriate activity. It is, in fact, considered as an early evaluation activity (Prat et al, 2014). As part of this activity, six organizations accepted to use the decentralized incident reporting platform and were called to perform a series of pre-defined actions, in order to confirm the platform's intended functionality. These participants were later called to evaluate the artefact, an activity described in the subsequent chapter of this report. This chapter describes the verification and validation activities performed in order to demonstrate the artefact's validity and effectiveness. All functional and non-functional requirements of the reporting platform were satisfied.

6.1. Verification

According to Geisler (2010), verification refers to the "act of demonstrating that design outputs match design inputs". Therefore, and before initiating any testing activities, it was necessary to confirm that the developed artefact met its predefined objectives. The following table presents the predefined objectives and implementation targets of this project, as those were set in Chapter four of this report, with a corresponding confirmation of actual implementation success or failure, for each objective, as well as the specific implementation details which determined the success/failure outcome:

Objectives			
No.	**Description**	**Keywords**	**Implementation result (Success/failure)** **Implementation details**
1	Create an incident reporting solution which enables and encourages the reporting of information security incidents amongst organizations, thereby reducing organizational demotivation for reporting.	Anonymity, Cost reduction, Artefact (instantiation)	**Success** - Created artefact (instantiation) in the form of a DApp - Anonymity of participants is ensured through blockchain's (Quorum) inherent characteristics. Only the public key of each participant is publicly visible and no other identifiable data. - Produced artefact is easy to understand and use, utilizing a simple GUI and standardized features and reporting templates, which minimizes any training costs. In addition, the cost of owning and operating the platform sums up to the yearly amount of approximately GBP 5.000, which is significantly lower than that of other commercially available platforms (excluding open-source platforms). The total cost involves all the services purchased through the Azure BaaS and the cost of the hosting provider for the front-end components. Furthermore, a variety of no-cost services have been utilized for this project: Mailgun for free e-mail delivery, Firebase for free database provision and multifactor authentication processes, Epirus Explorer for viewing transactions.

2	Create an incident reporting solution which utilizes the positive features offered by existing reporting solutions.	Efficiency, Performance, Ease of use, Accessibility, Security, Support, Social features	**Success** - The selection of creating a private blockchain implementation, as well as utilizing a less resource-intensive consensus algorithm (PoA/IBFT), increase the solution's performance, efficiency and scalability. - The platform is easy to navigate and use - The platform is easily accessible throughout the world (over the Internet), and only requires a Web3.0-capable browser and an Ethereum wallet. - Incidents submitted over the platform are irreversible and cannot be forged (due to one-way cryptographic hash functions) - Origin and integrity of submitted incidents are ensured through the signing of each incident (transaction) with the user's private key. - All communication channels are encrypted using up-to-date cryptographic standards - Multifactor authentication has been implemented utilizing three levels of user authentication: a) User public key white-listing b) E-mail/password combination c) One-time-password received through SMS. - The option of user support/communication with platform administrator has been implemented through the creation of an easy to use contact form - A live, anonymous, chat has been implemented as part of providing adequate social features to the platform users

3	Create an incident reporting solution which provides added value to users, in comparison to existing solutions.	Anonymity, Availability, Auditability/transparency/immutability	**Success** - Anonymity of participants is ensured through blockchain's (Quorum) inherent characteristics. Only the public key of each participant is publicly visible and no other identifiable data. - Constant blockchain availability is ensured through the inherent characteristics of the technology. Front-end availability currently stands at 99.9% - Incidents are auditable and all participants can query the submitted incidents, through the use of the Epirus Blockchain explorer. Incidents are therefore both consistent and transparent. - Incidents submitted over the platform are immutable: they and cannot be forged (due to one-way cryptographic hash functions).
Implementation targets			
1	Create a manual incident reporting solution.	Manual solution	**Success** - The produced incident reporting solution, accepts data (incidents) only through human input, through rationally designed forms, and not through automated mechanisms/tools.
2	Create a software platform for the manual reporting of incidents	Reporting software/platform	**Success** - The produced incident reporting solution is in the form of a software platform, featuring an accessible, clean and easy to navigate, interface
3	Create a private incident reporting platform.	Private reporting software/platform	**Success** - The produced incident reporting platform is only accessible by pre-authorized participants (allowed by a designated authority).

4	Create a familiar environment for platform users.	Familiar structure, functionality, GUI, standardized reporting templates	**Success** - The platform "feels" and "looks" familiar to users, as the overall "experience", structure, graphical user interface (GUI), functionality, and the typical sequence of actions of the platform, matches (or very much approaches) the existing operational environment of other reporting platforms. - The reporting of incidents is conducted through, familiar, standardized, and widely accepted reporting templates. The internationally recognized "ISO 27035:2016" incident reporting template was utilized, with a minor alteration. This alteration included replacing the ISO's proposed incident categories/taxonomy, with the "eCSIRT.net mkVI" taxonomy (Stikvoort, 2015), since the latter is endorsed by ENISA, its categories are universal and practical, and it is widely used amongst European CSIRTs (ENISA, 2018).

Table 6.1. Implementation results of set objectives

6.2. Validation Tests

According to ISO/IEC 15288 (2015), software validation activities act towards the confirmation that the requirements for an application or a specific intended use have been met, through the provision of "objective evidence". According to the same international standard, validation activities should provide confidence that a software can accomplish its pre-set objectives and intended use, while operating in its intended environment. Therefore, it was deemed necessary to seek a number of external actors, who would, in conjunction with the researcher, execute a number of pre-determined test cases/scenarios, in order to obtain this necessary "objective evidence". This "evidence" would confirm that the developed incident reporting platform behaved according to its pre-set objectives and intended use.

6.2.1. Introduction

Utilizing CYCSO's businesses database, a "call-for-participants" e-mail (Appendix C) was sent to organizations, in order to identity potential interest for testing the produced incident reporting platform. The participating organizations could be of any size and operate in any business sector. The only major eligibility prerequisite was that the participating organizations should at that point be using (or have used, at some point in the past) an existing incident reporting platform, either commercially available, or open source/free. The reason for setting this prerequisite, was that participating organizations should be familiar with using an incident reporting platform and thus be able to compare and evaluate available features and functionality. Seven organizations expressed interest in testing the platform, out of which one was disqualified, since it did not have any prior experience in using any other reporting platform.

6.2.2. Profiles of participants

Six organizations were therefore eligible to participate in testing the platform. Out of these six, one operates in the retail sector (hereinafter referred to as "Company A"), one in the higher education sector ("Company B"), three in the financial sector ("Companies "C", "D", "E") and one is a group of companies operating in both the hospitality and construction sector ("Company F"). All participants had previous experience with one, or more, commercial incident reporting platforms, although, at that point of time, only one of them was still utilizing such a platform. Furthermore, a distinct set of activities were executed by the researcher himself, who is an information security professional and professional doctorate student at the University of East London.

6.2.3. Purpose

The purpose of these tests was the assessment of the platform's intended functionality, by six, external, business actors, who were called to follow specific, predetermined, test cases, prepared by the researcher. In addition, a distinct set of activities were executed by the researcher himself, in order to test the functionality of specific, administration-related, tasks of the platform. These administrative tasks could not have been executed by an external actor, since no contractual agreement was in place with any organization to act as an administrator. If admin privileges were given to external parties, there was a risk of unintentional or intentional damage to the platform: they could have added/deleted users at will, they could have altered any platform/blockchain settings or they could have obtained copies of any platform/blockchain configurations for their own use. Therefore, and since only a single instance of the platform was developed, for the proof of concept purposes of this project, the admin test cases had to be executed by the researcher, in a controlled environment.

6.2.4. Roles & Responsibilities

The test cases had to be executed by qualified personnel of the participating organizations, who were directly involved in the operation/management of the incident reporting platform of the organization, currently (or previously) in use. The following table summarizes the roles and responsibilities of the participants:

Participant	Capacity	No. of test cases to be executed	Responsibility
Company A	CIO, CISO or similar role	5	Perform user test cases and report results (Success/Fail)
Company B	CIO, CISO or similar role	5	Perform user test cases and report results (Success/Fail)
Company C	CIO, CISO or similar position	5	Perform user test cases and report results (Success/Fail)
Company D	CIO, CISO or similar role	5	Perform user test cases and report results (Success/Fail)
Company E	CIO, CISO or similar role	5	Perform user test cases and report results (Success/Fail)

Company F	CIO, CISO or similar role	5	Perform user test cases and report results (Success/Fail)
Alexis Michail	Researcher, Developer	5	Perform admin test cases and report results (Success/Fail)

Table 6.2. Roles and responsibilities of participating organizations

6.2.5. Test Prerequisites

Before initiating any testing activities, the relevant ethical approval was obtained from the University's Ethics Committee (Appendix D). The participating organizations were informed about their required contribution to this research project and signed a consent form. They were also given written instructions for performing the test cases (Appendix E). Participants were also given two questionnaires to complete, one before commencing their test activities, and one after their completion. The results of these questionnaires are described in "chapter 7 – Evaluation".

6.2.6. Test requirements and testing schedule

Test cases by participants would be executed at their own organizational premises, in their usual operating environment, at any time between 02/12/2019 and 06/12/2019. Participants would have to strictly follow the specified test instructions and report a "success" or "fail" outcome for each, distinct, test case, through electronic mail/chat or through verbal communication, upon the tests' cessation. Should a test case present "fail" as an outcome, participants should include a detailed description of their actions, leading to this outcome, as well as any error messages presented throughout the test activity.

The following table summarizes the schedule of the various testing activities, in sequential order:

No.	Test activity	Responsibility	Date(s)
1	Design and develop test cases	Alexis Michail	01-10/11/2019
2	Call for participants	Alexis Michail	18-22/11/2019
3	Evaluate participants	Alexis Michail	23-24/11/2019
4	Review test cases and other test instructions for completeness and accuracy	Alexis Michail	25/11/2019
5	Confirm availability of participants	Alexis Michail	26/11/2019
6	Admin test cases execution by developer	Alexis Michail	27/11/2019
7	Take corrective action in case of “failed” admin test case and repeat execution by developer	Alexis Michail	27-29/11/2019
8	Provide test cases and instructions to participants	Alexis Michail	02/12/2019
9	User test cases execution by participating organizations	Company A, Company B, Company C, Company D, Company E, Company F	03-05/12/2019
10	Provide tests outcome by participating organizations	Company A, Company B, Company C, Company D, Company E, Company F	03-05/12/2019
11	Record and evaluate all tests outcome	Alexis Michail	03-05/12/2019
12	Take corrective action in case of “failed” test case and repeat execution	Alexis Michail and affected company	03-05/12/2019
13	Provide test outcome and repeat step 12 in case of “failed” test case	Alexis Michail and affected company	03-05/12/2019

14	Document test outcome	Alexis Michail	03-05/12/2019

Table 6.3. Schedule of testing activities

6.2.7. Type of testing and testing environment

Since the overall aim of this testing activity was to exhibit and validate the proper functionality of the decentralized incident reporting platform, a “black box” type of testing was deemed as the most appropriate. In a “black box” type of test, the tester solely focuses on the system output - which is generated by selected input and specific execution conditions - and does not have access to the source code or any other internal functions of the system (Liu & Tan, 2009). On the other hand, in a “white box” type of test, test cases aim to investigate the internal logic and the structure of the code (Khan & Khan, 2012). “Grey box” testing combines both “black box” and “white box” testing techniques (Khan & Khan, 2012). Since the developed artefact is a prototype, and not yet targeted towards a production environment, a black box type of test should suffice, at this stage. However, “grey box” testing should be utilized, before the developed incident reporting platform is addressed towards a greater audience.

Regarding the testing environment, all testing activities were performed in an online (internet-enabled) environment.

6.2.8. Test assumptions

In order for the participants to successfully perform the test activities, a number of assumptions needed to be satisfied:

a) Participants should have access to the Internet and operate internet-enabled devices (terminals) with suitable browsers.

b) Participants should first complete the evaluation questionnaire (presented in chapter seven) regarding their alternative incident reporting platform, before initiating any testing activities.

c) Participants should thoroughly read the applicable test instructions (Appendix E) before initiating any test activities.

d) Participants should complete the testing activities and report the subsequent results within the applicable timeframe (evident in section "6.2.6.").

e) Upon the test's cessation, participants should be available and accessible for a further period of one week, in case further clarifications are necessary.

6.2.9. Test cases

The following section describes the various test cases that both the participants (users) and the researcher (admin) were called to sequentially perform:

a) User test cases

i) User login

Test case title	User login
ID	UTC-01
Purpose	To authenticate the user of the platform and provide access to its functionality.
Preconditions	Metamask plugin, Ethereum wallet, mobile phone to receive SMS OTP, white-listed Ethereum public key, successful registration to the platform.
Description of events	1. Access the platform's webpage through suitable browser. 2. Login to Metamask wallet and allow connection. 3. Insert e-mail and password in designated fields. 4. Request token for multi-factor authentication. 5. Insert token received through SMS in designated field. 6. Press login button.
Expected result	Authentication is successful and the platform's homepage is displayed to the user.

Table 6.4. User login test case

ii) User submit incident

Test case title	User submit incident
ID	UTC-02
Purpose	To complete an incident reporting form and submit an incident.
Preconditions	Successful execution of UTC-01.
Description of events	1. After successful login and while on platform’s homepage, click on the “submit incident” button. 2. Fill in all available fields of the form with details of a mock incident. 3. Click “Preview & submit” button. 4. Review content of form and if satisfied with content click “submit” button, otherwise click “edit form” button. 5. Sign transaction with Metamask plugin or similar. 6. View transaction ID on your screen.
Expected result	Incident is successfully submitted and stored on the blockchain and the transaction ID is displayed to the user.

Table 6.5. User submit incident test case

iii) User view incidents

Test case title	User view incident
ID	UTC-03
Purpose	To view the incident previously submitted (following the successful execution of UTC-02) recorded in the “view incidents” table.
Preconditions	Successful execution of UTC-01 and UTC-02.

Description of events	1. While on platform's homepage, click on the "view incidents" button. 2. The incident submitted through UTC-02 should be evident in the relevant table and located at the top of the list. 3. Click on any field of the specific row of the incident. 4. Confirm the data displayed is identical to the data submitted as part of UTC-02 execution.
Expected result	Incident is successfully displayed in "view incidents" table.

Table 6.6. User view incident test case

iv) User ask for help

Test case title	**User ask for help**
ID	UTC-04
Purpose	To submit a message to the administrator through the "ask for help" form.
Preconditions	Successful execution of UTC-01.
Description of events	1. While on platform's homepage, click on the "ask for help" button. 2. Fill-in the details required by the form. 3. Click on the "submit" button. 4. Wait for admin confirmation regarding message receival.
Expected result	"Ask for help" form submitted and successfully received from the administrator.

Table 6.7. User ask for help test case

v) User chat

Test case title	User chat
ID	UTC-05
Purpose	To successfully connect to the anonymous chat room of the platform and submit a mock message.
Preconditions	Successful execution of UTC-01.
Description of events	1. While on platform's homepage, click on the "live chat" button. 2. Notify researcher to join the chat session. 3. After the researcher has joined the chat and sent an acknowledgment message, type a message with content: "Hello fellow anonymous!" 4. Click on the "submit" button. 5. The researcher should reply with a second acknowledgment message.
Expected result	Messages from researcher and user successfully displayed in chat window.

Table 6.8. User chat test case

b) Admin test cases

i) Admin add new user

Test case title	Admin add new user
ID	ATC-01
Purpose	To add a new user to the incident reporting DApp.
Preconditions	Successful login to the administration console of the DApp (alexis-michael.eu/reporting/admin)

Description of events	1. Enter user's public key in database/white-list. 2. Click the add button. 3. Login to Metamask account 3. Go to the platform's homepage (select sign up option). 4. Add registration details of new user (e-mail, password, mobile phone) 5. Insert token received through SMS in designated field. 6. Press "Get started" button.
Expected result	New user is added to the incident reporting DApp.

Table 6.9. Admin add user test case

ii) Admin remove user

Test case title	Admin remove user
ID	ATC-02
Purpose	To remove a user from having access to the incident reporting DApp
Preconditions	Successful login to the administration console of the DApp (alexis-michael.eu/reporting/admin)
Description of events	1. Search for database row containing the user's public key or name. 2. Click on "delete" button in relevant row. 3. Login to Firebase database and delete matching user credentials (*removal method 1*). 4. Repeat execution of ATC-01. 5. Repeat execution steps 1,2 of ATC-02 6. Login to the platform's main application and confirm that functionality is now disabled for the user. Confirm that user credentials have been automatically removed from Firebase database.

Expected result	Selected user is removed from having access to the incident reporting DApp.

Table 6.10. Admin remove user test case

iii) Admin submit incident

Test case title	Admin submit incident
ID	ATC-03
Purpose	To complete an incident reporting form, submit an incident and view the incident through Epirus Explorer.
Preconditions	Admin successfully logged in, Metamask plugin for signing the transaction, access to Epirus Explorer.
Description of events	1. After successful login and while on platform's homepage, click on the "submit incident" button. 2. Fill in all available fields of the form with details of a mock incident. 3. Click "Preview & submit" button. 4. Review content of form and if satisfied with content click "submit" button, otherwise click "edit form" button. 5. Sign transaction with Metamask plugin or similar. 6. View transaction ID on your screen. 7. Trace transaction through Epirus Explorer
Expected result	Incident is successfully submitted and stored on the blockchain and the transaction ID is displayed to the user. The submitted transaction is successfully viewed through Epirus Explorer.

Table 6.11. Admin submit incident test case

iv) Admin view incident

Test case title	Admin view incidents
ID	ATC-04
Purpose	To view the incident previously submitted (following the successful execution of ATC-03) recorded in the "view incidents" table.
Preconditions	Successful execution of ATC-03.
Description of events	1. While on platform's homepage, click on the "view incidents" button 2. The incident submitted through ATC-03 should be evident in the relevant table and located at the top of the list. 3. Click on any field of the specific row of the incident. 4. Confirm the data displayed is identical to the data submitted as part of ATC-03 execution.
Expected result	Incident is successfully displayed in "view incidents" table.

Table 6.12. Admin view incident test case

v) Admin chat

Test case title	Admin chat
ID	ATC-05
Purpose	To successfully connect to the anonymous chat room of the platform and submit a mock message.
Preconditions	Admin successfully logged in, separate user account logged in.
Description of events	1. While on platform's homepage, click on the "live chat" button. 2. Establish a separate connection to the chat session with a different user.

	3. Type a message with content: "Can you see me?" from the admin account and click on the "submit" button. 4. Through the other user account, acknowledge the admin message and reply with a message stating "Yes I can".
Expected result	Messages should be successfully displayed in both sessions.

Table 6.13. Admin chat test case

6.2.10. Test cases execution results

The following two tables summarize the results of the test cases execution by both the participants (users) and the researcher (admin):

a) Admin test cases execution results

Admin test cases execution results					
Test case title	Add new user	Remove user	Submit incident	View incident	Chat
Test case ID	ATC-01	ATC-02	ATC-03	ATC-04	ATC-05
Execution date	27/11/2019	27/11/2019	27/11/2019	27/11/2019	27/11/2019
Reporting date	27/11/2019	27/11/2019	27/11/2019	27/11/2019	27/11/2019
Execution result	**Success**	**Success**	**Success**	**Success**	**Success**

Table 6.14. Admin test cases execution results

b) User test cases execution results

User test cases execution results					
Test case title	Login	Submit incident	View incident	Ask for help	Chat
Test case ID	UTC-01	UTC-02	UTC-03	UTC-04	UTC-05

Execution date					
Company A	04/12/2019	04/12/2019	04/12/2019	04/12/2019	04/12/2019
Company B	04/12/2019	04/12/2019	04/12/2019	04/12/2019	04/12/2019
Company C	05/12/2019	05/12/2019	05/12/2019	05/12/2019	05/12/2019
Company D	03/12/2019	03/12/2019	03/12/2019	03/12/2019	03/12/2019
Company E	05/12/2019	05/12/2019	05/12/2019	05/12/2019	05/12/2019
Company F	05/12/2019	05/12/2019	05/12/2019	05/12/2019	05/12/2019
Reporting date					
Company A	04/12/2019	04/12/2019	04/12/2019	04/12/2019	04/12/2019
Company B	04/12/2019	04/12/2019	04/12/2019	04/12/2019	04/12/2019
Company C	05/12/2019	05/12/2019	05/12/2019	05/12/2019	05/12/2019
Company D	03/12/2019	03/12/2019	03/12/2019	03/12/2019	03/12/2019
Company E	05/12/2019	05/12/2019	05/12/2019	05/12/2019	05/12/2019
Company F	05/12/2019	05/12/2019	05/12/2019	05/12/2019	05/12/2019
Execution result					
Company A	**Success**	**Success**	**Success**	**Success**	**Success**
Company B	**Success**	**Success**	**Success**	**Success**	**Success**
Company C	**Success**	**Success**	**Success**	**Success**	**Success**
Company D	**Success**	**Success**	**Success**	**Success**	**Success**
Company E	**Success**	**Success**	**Success**	**Success**	**Success**
Company F	**Success**	**Success**	**Success**	**Success**	**Success**

Table 6.15. User test cases execution results

6.2.11. Acceptance and acknowledgments

All test cases were successfully executed by both the researcher and the participants and the results were as expected. No errors or any other execution issues were reported by the participants. An illustrative example of a user logging in, submitting an incident/transaction, viewing the list of incidents and tracing the incident/transaction through Epirus explorer, can be found in Appendix F.

7. THE DECENTRALIZED SOLUTION: EVALUATION

According to Peffers et al (2007) framework, the fifth activity of a design science research project involves evaluating the developed artefact. The authors state that evaluation could take many forms, and could include quantitative performance measures, such as simulations, budgets, or items produced, client feedback, the results of satisfaction surveys, as well as other quantifiable measures of system performance, such as the system's response time or availability. The authors also state that, conceptually, the evaluation activity could include "any appropriate empirical evidence or logical proof", and that at the end of this activity, the researcher has the option to iterate to previous phases of the framework, if improvements are deemed necessary, or continue to the communication activity (Peffers et al, 2007, p.13).

Although there is a common agreement, amongst researchers, that evaluation is an essential activity in conducting rigorous Design Science Research (Venable et al, 2012), the available literature on this topic seems to be rather limited, while the applicable evaluation criteria and methods are presented in a rather fragmented manner (Venable et al, 2012; Prat et al, 2014). A number of evaluation methods has been proposed, however, and this includes, among others, methods by March and Smith (1995), Hevner et al (2004), Pries-Heje et al (2008), Venable et al (2012), and Sonnenberg and vom Brocke (2012). According to Venable et al (2012), the DSR literature provides very little guidance as to the actual design of the evaluation component of a DSR project, with the exception of the work by Pries-Heje et al (2008), who propose a framework of strategies for DSR evaluation. Based on this framework, Venable et al (2012) developed their own extended framework and method, which is utilized for evaluating this research project.

7.1. Applying the Venable et al (2012) framework and method

Venable et al (2012) propose a four-step method (or process) for designing the evaluation process of a DSR project:

a) Requirements analysis of evaluation process

b) Mapping requirements to one or more of the dimensions and quadrants in applicable framework

c) Selecting an appropriate evaluation method or methods that align with the chosen strategy quadrant(s), and

d) Designing the evaluation in more detail.

The details of this four-step process, as described by the authors, can be found in Appendix G.

7.1.1. Requirements analysis of evaluation process

This process will perform a technical evaluation of a product. More particularly, this process will evaluate the artefact (instantiation) developed by the researcher, a decentralized incident reporting platform. The aspects that will be evaluated are drawn from the pre-determined objectives of the artefact, as those were set in “Chapter four – Objectives” of this report.

7.1.2. Mapping requirements to quadrants

According to Venable et al (2012), in order for the researcher to initiate the evaluation activities, the evaluation requirements should be matched to one or more DSR evaluation strategies, which include selecting whether “ex ante” (prior to artefact development) or “ex post” evaluation (after artefact development) is required, and in which setting, naturalistic (i.e. field setting) or artificial (i.e. laboratory setting).

A researcher should begin the evaluation process by understanding the context of the required DSR evaluation and then map that understanding to the criteria shown in the following figure:

DSR Evaluation Strategy Selection Framework		**Ex Ante**	**Ex Post**
		•Formative •Lower build cost •Faster •Evaluate design, partial prototype, or full prototype •Less risk to participants (during evaluation) •Higher risk of false positive	•Summative •Higher build cost •Slower •Evaluate instantiation •Higher risk to participants (during evaluation) •Lower risk of false positive
Naturalistic	•Many diverse stakeholders •Substantial conflict •Socio-technical artifacts •Higher cost •Longer time - slower •Organizational access needed •Artifact effectiveness evaluation •Desired Rigor: "Proof of the Pudding" •Higher risk to participants •Lower risk of false positive – safety critical systems	•Real users, real problem, and somewhat unreal system •Low-medium cost •Medium speed •Low risk to participants •Higher risk of false positive	•Real users, real problem, and real system •Highest Cost •Highest risk to participants •Best evaluation of effectiveness •Identification of side effects •Lowest risk of false positive – safety critical systems
Artificial	•Few similar stakeholders •Little or no conflict •Purely technical artifacts •Lower cost •Less time - faster •Desired Rigor: Control of Variables •Artifact efficacy evaluation •Less risk during evaluation •Higher risk of false positive	•Unreal Users, Problem, and/or System •Lowest Cost •Fastest •Lowest risk to participants •Highest risk of false positive re. effectiveness	•Real system, unreal problem and possibly unreal users •Medium-high cost •Medium speed •Low-medium risk to participants

Figure 7.1. DSR Evaluation Strategy Selection by Venable et al (2012)

The researcher should select an evaluation strategy (or combination of strategies) based on which rows, columns and cells in the above figure are most relevant to his/hers DSR project. Regarding this particular project, it is initially obvious that an "ex post" evaluation is required, since the artefact has already been instantiated. A naturalistic approach is also necessary, since a real problem does exist (under-reporting of incidents), as well as a real system (the developed artefact) and real users. The evaluation activity should also opt towards the evaluation of the artefact's effectiveness, with a low risk of false positives, despite the possibility of having an increased cost. Therefore, the fourth cell/quadrant (from left to right) of the second row of the framework appears more relevant to this project, as indicated by the orange arrow in figure 7.1.

7.1.3. Selecting appropriate evaluation methods

The following stage involves selecting appropriate evaluation methods, based on the quadrant selected previously, in figure 7.1. The available evaluation methods are evident in the following figure:

DSR Evaluation Method Selection Framework	**Ex Ante**	**Ex Post**
Naturalistic	•Action Research •Focus Group	•Action Research •Case Study •Focus Group •Participant Observation •Ethnography •Phenomenology •Survey (qualitative or quantitative)
Artificial	•Mathematical or Logical Proof •Criteria-Based Evaluation •Lab Experiment •Computer Simulation	•Mathematical or Logical Proof •Lab Experiment •Role Playing Simulation •Computer Simulation •Field Experiment

Figure 7.2. DSR Evaluation Method Selection by Venable et al (2012)

The available evaluation methods for this research project are evident in the third cell/quadrant (from left to right) of the second row, as indicated by the orange arrow in figure 7.2., which includes various options. The approach eventually selected for this project includes utilizing a survey for evaluating the produced artefact. More specifically, the users who participated in the testing activities (described in the “Demonstration” chapter) of the artefact were also called to complete evaluations, by completing, two, identical, Likert-style questionnaires, initially assessing the capabilities of their current (or previously used) incident reporting platform, and then assessing the capabilities of the newly developed artefact. The results obtained from these questionnaires will be used as input for the evaluating the artefact.

7.1.4. Designing the evaluation in more detail

The following stage involves designing the evaluation activity in more detail. The following table provides the finer details of the evaluation method:

IRDA evaluation method	
Purpose	The purpose of the evaluation method is to perform a comparison between the features/characteristics of the participants’ current (or previously used) incident reporting platform, and the features/characteristics of the newly developed, decentralized, incident reporting platform. This comparison (and the subsequent analysis of results) will allow the researcher to determine whether participants consider that the developed artefact has improved specific aspects of the incident reporting process.
Evaluation method	Survey/Questionnaire (quantitative).
Evaluation method details	Participants will be called to complete two, identical, questionnaires (Likert scale, 10-point), which include questions deriving from the project’s objectives. Participants will need to first complete questionnaire “A”, then successfully perform the various test activities on the newly developed platform (described in chapter six), and finally complete questionnaire “B”. Both questionnaires are available in Appendix H.

Participants	Six organizations – their profiles are described in section "6.2.2." of this report.
Roles and responsibilities	The questionnaires will be completed by the same individuals that performed the test activities described in chapter six.
Timeframe	03/12/2019 - 06/12/2019
Prerequisites	Participants will receive instructions for both executing the test cases, as well as for completing the questionnaires. Participants will be provided with the relevant information sheets and consent forms prior to the commencement of any activities.
Assumptions	o Questionnaires will be distributed by the researcher (in hard copies) to the participants, in their organizational premises. o Participants will complete the questionnaires in a voluntary fashion and without any bias, whatsoever. o Participants should complete the activities within the given timeframe. The hard copies will be collected by the researcher on the last day of the agreed timeframe. o Participants should be available and accessible for a further period of one week, in case further clarifications are necessary.
Expected outcome	By 06/12/2019 the researcher should possess six pairs of completed questionnaires.
Data analysis method of results	Non-parametric significance tests (Wilcoxon-Pratt Signed-Ranked test)

Table 7.1. Evaluation method details

7.2. Evaluation method: Results and analysis

7.2.1. Results

The questionnaires were collected by the researcher on 06/12/2019. The following tables describe the results obtained from the participants for both questionnaires:

Questionnaire "A" – Platform currently in use/previously used							
No.	Question	Company					
		A	B	C	D	E	F
1	How would you rate the level of user anonymity the platform provides?	1	2	2	1	3	1
2	How would you rate the overall cost of purchasing, operating, managing, and maintaining the platform (including any staff training costs)?	4	3	2	2	3	4
3	How would you rate the ease of understanding the platform's features and overall functionality?	8	8	8	9	9	9
4	How would you rate the overall ease of using the platform (including GUI design and simplicity in the reporting processes)?	7	8	9	9	8	8
5	How would you rate the level of customer support offered by the platform's provider?	8	8	9	10	9	9
6	How would you rate the overall level of performance and efficiency of the platform?	9	9	8	9	8	8
7	How would you rate the overall level of security of the platform?	7	8	8	9	8	9
8	How would you rate the overall level of accessibility of the platform?	9	10	9	10	8	9
9	How would you rate the social features (e.g. chat, forum etc.) offered by the platform (if, any)?	6	8	7	8	7	6
10	How would you rate the platform's availability (i.e. uptime) level?	9	8	9	9	8	8
11	How would you rate the overall platform's transparency features including the presence of any auditability mechanisms?	6	6	5	5	4	5

Table 7.2. Results of questionnaire "A"

Questionnaire “B” – Incident reporting Decentralized App (IRDA)							
No.	Question	Company					
		A	B	C	D	E	F
1	How would you rate the level of user anonymity the platform provides?	9	9	10	9	8	10
2	How would you rate the overall cost of purchasing, operating, managing, and maintaining the platform (including any staff training costs)?	8	7	9	8	7	10
3	How would you rate the ease of understanding the platform’s features and overall functionality?	9	9	8	9	10	10
4	How would you rate the overall ease of using the platform (including GUI design and simplicity in the reporting processes)?	8	9	10	9	7	7
5	How would you rate the level of customer support offered by the platform’s provider?	7	8	8	9	10	8
6	How would you rate the overall level of performance and efficiency of the platform?	7	8	9	8	8	7
7	How would you rate the overall level of security of the platform?	8	9	8	7	9	7
8	How would you rate the overall level of accessibility of the platform?	7	8	9	8	9	8
9	How would you rate the social features (e.g. chat, forum etc.) offered by the platform (if, any)?	8	8	9	7	8	7
10	How would you rate the platform’s availability (i.e. uptime) level?	10	10	10	10	10	10
11	How would you rate the overall platform’s transparency features including the presence of any auditability mechanisms?	8	9	6	8	7	7

Table 7.3. Results of questionnaire “B”

7.2.2. Analysis of results

As previously mentioned, the questions included in the questionnaires were inferred from the research project's objectives (as stated in chapter four), which were, in turn, inferred from the main research question. In order to evaluate the accomplishment of these objectives, it was necessary to compare and analyze the participants' answers before (questionnaire "A") and after (questionnaire "B") using the developed artefact. Since each question (objective) examined distinct characteristics of the platforms, it was deemed necessary to perform eleven statistical significance tests, each for every distinct question (objective).

To begin with, the questionnaires included eleven Likert scale questions. According to Derrick and White (2017, p.1), a Likert scale question is a "forced choice ordinal question, which captures the intensity of opinion, or degree of assessment, in survey respondents". According to the same authors, a Likert item is historically comprised of five or seven points, although utilizing more or less points is a usual practice. The questionnaires in scope utilized a 10-point Likert scale, with participants having to provide an answer ranging from points 1 (lowest) to 10 (highest), for each question. Likert scales are considered to be ordinal in nature, and thus subject to non-parametric tests, since although the response categories do have a rank order, the intervals between points cannot be assumed equal (Jamieson, 2004). However, researchers frequently presume that they are equal (Blaikie, 2003) and that in particular methodological and practical aspects, Likert responses may approximate interval level data, and thus become eligible for parametric tests (Norman, 2010). This practice, however, is highly controversial amongst researchers (Knapp, 1990; Meek et al, 2007; Derrick & White, 2017).

According to Jamieson (2004), before the assumption of interval data applies, researchers should consider the sample size and distribution of the responses, before applying any parametric tests. Derrick and White (2017) suggest, that if sample sizes are large, both parametric and non-parametric tests are likely to have sufficient power. In this case, however, a sample size of only six pairs of observations is available, and no assumptions about the normality of the distribution can safely be made. Chou (1989), Berenson et al (2004), Keller (2005), Bowerman and O'Connell (2007) and Doane and Seward (2007), all agreed that for paired samples, where the sample size is small, the distribution is non-normal and the measurement is ordinal, the t-test (parametric) is not appropriate and the Wilcoxon signed-rank test (non-parametric) should be used instead. According to Blair and Higgins (1985), the Wilcoxon tests for both paired and unpaired samples are never significantly less powerful that t-tests, and when normality cannot be assumed for the distribution (for ordinal or interval measurements), they can even be much more powerful.

Therefore, executing non-parametric tests appeared as the most suitable option for analysing the results of the questionnaires. Since the samples are dependant/paired (same participants for both questionnaires), data is ordinal in nature, normality cannot be assumed, and the sample size is small, the Wilcoxon signed-rank test initially emerged as the prevalent option. "Sign test", a non-parametric, binomial test, used to test for trends in a series of ordinal measurements (Conover, 1999) was also considered, but it was eventually discarded as an option, since it is considered to be a lot weaker in comparison to Wilcoxon's test, when the detection of consistent differences is required (Demsar, 2006).

To perform a Wilcoxon signed-rank test, the first step is to form the null and alternative hypotheses. The researcher can then follow the procedure illustrated by Couch et al (2018):

The function calculating the Wilcoxon test statistic is formalized in Algorithm $\mathcal{W}$. Given a database $\mathbf{x}$ containing sets of pairs (u_i, v_i), the test computes the difference d_i of each pair, drops any with $d_i = 0$, and then ranks them by magnitude. (If magnitudes are equal for several differences, all are given a rank equal to the average rank for that set.) If $s_i = \pm 1$ is the sign of d_i and r_i is its rank, then $w = \sum_i s_i r_i$.

Algorithm $\mathcal{W}$ **:** Wilcoxon Test Statistic Calculation

Input: $\mathbf{x}$
begin
  for *row i of* $\mathbf{x}$ **do**
    $d_i \longleftarrow |v_i - u_i|$
    $s_i \longleftarrow \mathrm{Sign}(v_i - u_i)$
  Order the terms from lowest to highest d_i
  Drop any $d_i = 0$
  for *row i of* $\mathbf{x}$ **do**
    $r_i \longleftarrow$ rank of row i
  $w \longleftarrow \sum_i s_i r_i$
Output: w

Figure 7.3. Wilcoxon signed-rank test method (in Couch et al, 2018)

The researcher can then compare the "W" output to a critical value from a reference table (applicable for small sample numbers), in order to reject or accept the null hypothesis, or even proceed to the calculation of the p-value (with a continuity correction).

As evident, the Wilcoxon test discards pairs with zero difference. In a Likert-scale questionnaire, with a limited range of answer options, it is certainly expected to have pairs of answers with zero difference. Since a very small number of participants was made available (n = 6), discarding any pairs could further diminish the power of the test. In addition, Pratt (1959) claimed that ignoring zeros could produce paradoxical probabilities. He therefore suggested a modified version of the test, which specified ranking the differences (including the zeros), then dropping the zeros (when summing the negative and positive ranks), and finally using the tables of probabilities for the total number of observations (including the zeros) (Hoffman, 2015). Another method of handling zero-differences was suggested by Putter (1955), but resulted in a loss of efficiency (Conover, 1973). When testing the efficiency of t-test, Wilcoxon test and Pratt's test, in various scenarios, Derrick and White (2017) acknowledged that there is little practical difference in the conclusions drawn when the sample size is large; it becomes more a theoretical question about what test to use. However, when the sample size is small and the correlation between the paired groups is strong, they indicated that the Pratt's test outperforms the two others.

Therefore, the Wilcoxon Signed Ranked test modified by Pratt (or Pratt's test) was eventually selected for the analysis of the results. All the tests could have been conducted manually (i.e. through hand calculations, since sample size is small), or by using a suitable statistical analysis software, such as "SPSS", "R", "STATA", "SAS" or similar. It was eventually decided to utilize "R" software (version 3.6.1.) for executing the tests. The standard version of the software did not include Pratt's test, so the "Coin package v. 1.3-1" (Hothorn, 2019) was installed, along with the "Paired data" package, for drawing boxplots for paired data.

The “R” script that was used to execute the tests is available in Appendix I. The following table presents the results of executing the eleven significance tests. The details of each specific test can be found in appendix K.

Assumptions applicable to all eleven tests:

- *Samples are dependant (paired) and occur from the same population*
- *Paired observations are independently and randomly drawn*
- *Paired observations are of ordinal scale*
- *Normal distribution cannot be assumed*

Significance tests (two-tailed, n=6, α = 0.05)								
Q	**Description**	**Connected objectives and ITas**	**QA results**		**QB results**		**p-value**	**Verdict**
			M	IQR	M	IQR		
1	Anonymity	**O_1, O_3** These objectives mandated the creation of a reporting solution supporting the anonymity of submissions	1,5	1	9	0,75	0,03	Evidence to suggest that level of user anonymity is improved with the proposed platform
2	Cost	**O_1** This objective mandated the creation of a reporting solution with low cost	3	1,5	8	1,5	0,03	Evidence to suggest that level of cost of proposed platform is lower
3	Ease of understanding	**O_2, ITa_4** This objective mandated the creation of a reporting solution which would be easy to understand. The implementation target required the creation of a familiar environment for the users (thus	8,5	1	9	0,75	0,13	No evidence to suggest that level of ease of understanding is different between platforms

		contributing to the overall ease of understanding)						
4	Ease of use	**O_2** This objective mandated the creation of a reporting solution which would be easy to use	8	0,75	8,5	1,75	1	No evidence to suggest that level of ease of use is different between platforms
5	Customer support	**O_2** This objective mandated the creation of a reporting solution which would provide features enabling the provision of customer support	9	0,75	8	0,75	0,38	No evidence to suggest that level of customer support is different between platforms
6	Performance and efficiency	**O_2** This objective mandated the creation of a reporting solution which would behave adequately in terms of performance and efficiency	8,5	1	8	0,75	0,31	No evidence to suggest that level of performanc e and efficiency is different between platforms
7	Security	**O_2** This objective mandated the creation of a reporting solution which would behave in a secure way	8	0,75	8	1,5	1	No evidence to suggest that level of security is different between platforms
8	Accessibility	**O_2** This objective mandated the creation of a reporting solution which would be widely accessible	9	0,75	8	0,75	0,19	No evidence to suggest that level of accessibilit y is different between platforms

9	Social features	**O_2** This objective mandated the creation of a reporting solution which would include social features	7	1,5	8	0,75	0,25	No evidence to suggest that level of social features is different between platforms
10	Availability	**O_3** This objective mandated the creation of a reporting solution which would provide a greater availability level than that of conventional solutions	8,5	1	10	0	0,03	Evidence to suggest that level of availability is improved with the proposed platform
11	Transparency	**O_3** This objective mandated the creation of a reporting solution which would provide a greater transparency level than that of conventional solutions	5	0,75	7,5	1	0,03	Evidence to suggest that level of transparency is improved with the proposed platform

Table 7.5. Results of significance tests

7.3. Complimentary evaluation method

In order to complement the artefact's main evaluation method, a separate evaluation method will attempt to assess the quality of the developed software. This includes a high-level assessment of the developed software (performed solely by the researcher) against the requirements of the international standard "ISO/IEC 25010:2011 - Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models". The complimentary evaluation can be found in Appendix L.

7.4. Concluding remarks

As part of the artefact's evaluation method, eleven, non-parametric, significance tests were performed, on feedback data received from six, distinct, organizations, through Likert-scale questionnaires. The results of these tests indicated that objectives O_1, O_2 and O_3 of this research project were achieved, as well as implementation target ITa_4 .The rest of this project's implementation targets were achieved through specific implementation actions, which are evident (and discussed) in both chapters six (Demonstration) and eight (Discussion and conclusion) of this report.

8. DISCUSSION AND CONCLUSION

8.1. Thesis summary

The undeniable fact, that in the interconnected world we are now living in, organizations around the globe, face millions of security threats, on a constant basis, should have encouraged the vigorous reporting of information security incidents, should such incidents occur. Although virtually everyone agrees that information security incident reporting is beneficial to organizations (NIST, 2012; Gordon et al, 2003; ENISA, 2013; Gordon et al, 2015; Line & Albrechtsen, 2016; Gonzalez, 2005), reporting statistics show that very few incidents are actually being reported (IOD & Barclays Policy report, 2016; Symantec, 2016; Newman, 2018; Ipsos MORI, 2017; SentinelOne, 2016; ENISA, 2012). It seems that organizations find it difficult to disseminate information related to security incidents (He and Johnson, 2012; Grispos et al., 2015), due to a variety of reasons, such as the fear of the incident's consequences, including negative publicity, legal liability and possible financial penalties and reprimands, the exposure of organizational vulnerabilities, possible retribution attempts, the various costs related to incident reporting, such as operating costs, recruitment, training, as well as the overall organizational time spent by an organization's personnel for reporting purposes (Johnson, 2002; Metzger et al, 2011; Ahmad et al, 2012; Etzioni, 2014; Ruefle et al, 2014; Housen-Couriel, 2018).

This research project began with a comprehensive literature review of the two major themes of the research topic, incident reporting and blockchain. The various types of information security incidents were discussed, as well as other topics, such as their financial impact to organizations, the reasons why organizations choose to report (or not) incidents, the stance of incident reporting in the overall incident management lifecycle, as well as the various means and methods organizations utilize for incident reporting purposes. Furthermore, existing incident reporting platforms were identified and evaluated, in order to both detect features which make these platforms prevalent to use, as well as to identify potential areas for improvement, based on the given fact of incident under-reporting. The blockchain technology was also discussed, as a potential candidate for hosting the proposed solution. The technology's core concepts were explained, followed by a presentation of the various types of blockchains, the available consensus algorithms, blockchain's advantages and disadvantages, decision models, and blockchain applications in related areas, such as data management, information security and incident reporting. The literature review confirmed that even though blockchain development was mainly focused towards the area of crypto/virtual currencies (Di Pierro, 2017), it has gradually witnessed the development of applications in a variety of other fields, including data management, information security, and even incident reporting, although the available literature for the latter area was rather limited. It was, however, confirmed, that blockchain technology, with its decentralized structure and its various inherent characteristics, including security, anonymity and integrity (Yli-Huumo et al, 2016), could act as the underlying technology, in order to develop an alternative option/solution for incident reporting purposes.

Methodologically, this research project was situated in the applied research discipline (as it aimed to provide a solution to a problem of professional practice) and within the pragmatic paradigm, as it placed the problem of incident under-reporting as central and attempted to provide a solution. The Design Science Research was eventually selected as the research approach of choice, out of a few alternative approaches, such as Development research and Action research. Although, according to Peffers et al (2007), a generally accepted process for carrying out Design Science research does not exist, a number of different process models/frameworks were available for conducting Design science research, and out of these, the framework by Peffers et al (2007) was identified as the most appropriate choice, as it incorporates principles, practices and procedures necessary to conduct such research, while being consistent with prior literature (Peffers et al, 2007). The framework included six, distinct, activities/steps, and was followed in a nominally sequential order, beginning at activity/step one, since the researcher's aim was to provide a solution to a known problem (under-reporting) and a problem-centred, research approach was therefore required.

The solution’s objectives were derived directly from the research question. Three objectives and four implementation targets were set, and were successfully achieved, as evident in the following table:

Research question	***Is there a way to create an innovative information security incident reporting solution, which will utilize the positive features offered by existing solutions, but will also provide added value to users, in order to increase their level of motivation towards the reporting of incidents?***			
Objectives				
No.	**Description**	**Key elements**	**Objective included in evaluation activity**	**Objective Achieved**
1	Create an incident reporting solution which enables and encourages the reporting of information security incidents amongst organizations, thereby reducing organizational demotivation for reporting.	Anonymity, Cost reduction, Artefact (instantiation)	**Yes** - artefact’s anonymity and cost were both included in the evaluation activity.	**Yes** – verification and validation details for anonymity and cost elements are available in chapter 6 and the evaluation details in chapter 7. The remaining element of creating an “artefact” is evidently fulfilled through the creation of the IRDA software.
2	Create an incident reporting solution which utilizes the positive features offered by existing reporting solutions.	Efficiency, Performance, Ease of use, Ease of understanding, Accessibility, Security,	**Yes** – all key elements of this objective were included in the evaluation activity.	**Yes** – verification and validation details are available in chapter 6, evaluation details in chapter 7.

		Support, Social features		
3	Create an incident reporting solution which provides added value to users, in comparison to existing solutions.	Anonymity, Availability, Auditability/transparency/immutability	**Yes** – all key elements of this objective were included in the evaluation activity.	**Yes** – verification and validation details are available in chapter 6, evaluation details in chapter 7.
Implementation targets				
No.	**Description**	**Key elements**	**Target included in evaluation activity**	**Target Achieved**
1	Create a manual incident reporting solution.	Manual solution	**No**	**Yes** – The IRDA software developed is evidently an incident reporting tool enabling the manual submission of incidents.
2	Create a software platform for the manual reporting of incidents	Reporting software/platform	**No**	**Yes** - The IRDA software developed is evidently an incident reporting platform.
3	Create a private incident reporting platform.	Private reporting software/platform	**No**	**Yes** - The IRDA software developed is evidently a private incident reporting platform, as it only allows pre-authorized, registered, users to join the platform and use its various features

4	Create a familiar environment for platform users.	Familiar structure, functionality, GUI, standardized reporting templates	**Yes** – all key elements of this target were included in the evaluation activity.	**Yes** – verification and validation details for these elements are available in chapter 6 and the evaluation details in chapter 7.

Table 8.1. Achievement of set objectives and implementation targets

The “Design and development” chapter, reports all those explicit design and development details that transformed the general concept of a decentralized incident reporting platform, into a fully working prototype. The “Demonstration” chapter illustrates the verification details behind the fulfillment of each set objective, as well as all the validation tests that were performed for testing the functionality of the software, by six, independent, organizations and the researcher. The “Evaluation” chapter describes the evaluation activities, which were conducted according to the Venable et al (2012) four-step process, for a DSR project. As part of the evaluation method, eleven, non-parametric, significance tests were performed, on feedback data received from six organizations.

Moreover, it is important to note, that this entire report/thesis satisfies the sixth and final activity of the Peffers et al (2007) DSRM framework, named “Communication”, where, the problem in scope, its importance, details of the developed artefact and its effectiveness, amongst other elements, should be appropriately conveyed to researchers and other relevant audiences, such as practicing professionals. The structure of the thesis followed the flow of activities described in the DSRM model, and also borrowed some elements out of the nominal structure of an empirical research process, including the literature review, the description of the selected research methodology, as well as the “discussion and conclusion” chapter. The present thesis can therefore act as a communicator to appropriate audiences, explaining how IRDA can be used to tackle the issue of incident under-reporting.

8.2. Contribution summary

The overall objective of a professional doctorate thesis is to propose answers/solutions towards a known business problem. Professional doctorates contribute to the overall knowledge economy and aim in improving the workplace and/or professional practice, in innovative and flexible ways (Usher, 2002). Information security incident under-reporting is unambiguously a business problem, as identified by a variety of sources, such as ENISA (2012), Symantec (2016), Newman (2018) and many more. This thesis identified the underlying issues that cause this problem of incident under-reporting, and proposed a solution, in the form of an innovative artefact.

This thesis describes an original attempt in utilizing the newly emergent blockchain technology, and its inherent characteristics, for addressing those concerns which actively contribute to the business problem. The research question set at the beginning of this research quest, probed the feasible formation of an incident reporting solution, which would increase the motivation of users towards the reporting of incidents, by utilizing the positive features offered by existing solutions, on one hand, but also by providing added value to the users, on the other. Following a structured procedure, the various demotivators affecting incident reporting were identified, and so were the current means and methods for reporting. The creation of a manual, reporting platform was recognized, through available literature, as the ideal reporting solution. The various demotivators include the organizational fear of the incident's consequences, such as negative publicity, legal liability, regulatory incompliance and possible financial penalties and reprimands, the exposure of organizational vulnerabilities, possible retribution attempts, the various costs related to incident reporting, such as operating costs, recruitment and training, the organization's overall IS maturity level, as well as the overall organizational time spent by an organization's personnel for reporting purposes (Johnson, 2002; Metzger et al, 2011; Ahmad et al, 2012; Etzioni, 2014; Ruefle et al, 2014;

Humphrey, 2017; Housen-Couriel, 2018). A number of these demotivators, such as fears for negative publicity and increased reporting cost (Koivunen, 2010; Ahmad et al, 2015; Ruefle et al, 2014; Choo, 2011; Ahmad et al, 2012, Johnson, 2002; Metzger et al, 2011; Jaatun et al, 2009; Etzioni, 2014; Housen-Couriel, 2018), were treated with embedding innovative features in the developed artefact, such as reporting anonymity, within a low-cost reporting ecosystem. Performance, efficiency, security, accessibility, the presence of social features, as well as the solution's ease of use and understanding, were all positive features, which were identified through the evaluation of existing solutions, and were also incorporated in the developed artefact. The increased availability, immutability and transparency levels of IRDA can be regarded as further benefits of the solution. The developed artefact suggests that there, indeed, exists, a way, for the creation of an innovative reporting solution, which utilizes the positive features offered by existing solutions, but also provides that necessary added value, which may ultimately increase the motivational level of users towards the reporting of incidents.

Revisiting Gregor and Hevner (2013) and their DSR knowledge contribution framework, there are four possible types of knowledge contribution in Design Science research, and a DSR project can make more than a single type of contribution: Invention is inventing new solutions/knowledge for new problems, Improvement is developing new solutions/knowledge for known problems, Adaptation concerns the innovative adaptation of known solutions/knowledge for new problems and Routine Design is applying known solutions to known problems, which, by itself, would not usually be considered as a research contribution (Gregor & Hevner, 2013). This research project belongs in the Improvement segment of the framework, as it generates an innovative solution towards the known problem of incident under-reporting. The developed artefact is the first application utilizing blockchain for manual incident reporting purposes. To the best of the researcher's knowledge, there is currently no other solution offering similar benefits to users/organizations for incident reporting purposes. The most directly relevant/similar work is the theoretical framework for information sharing, proposed by Adebayo et al (2019), which was described in the literature review chapter of this thesis. The authors of this framework, however, propose a public, rather than a private, blockchain implementation, where any security-conscious organization could join as a member, and could also include various security vendors (e.g. antivirus companies) which, in-turn, could offer applicable solutions (e.g. patches) to participating organizations, via a cloud configuration, also accessible via the blockchain. As previously mentioned, their work produced a high-level, theoretical framework and not an actual instantiation.

The literature review chapter comprehensively examined the previous work related to the application of blockchain in domains relative to this project, such as data management, information security and incident reporting, in particular. In line with authors who have utilized blockchain as data management systems/repositories for various tasks (such as Lemieux, 2016, who presented a blockchain-based solution for creating and preserving digital records, for use by civil registries of births, deaths and marriages; Garcia-Barriocanal et al, 2017, who utilized blockchain for the storage of metadata of digital archives; Cebe et al, 2018, who constructed a blockchain system for storing forensic evidence for accident investigations; Goharshay et al, 2018, who proposed an approach for maintaining credit history records on blockchain and others), this project indicates that blockchain can, indeed, be used for data storage/management purposes in the field of information security incident reporting. As also identified through this project's literature review chapter, the previous work related directly to blockchain applications in the incident reporting domain is fairly limited. It seems that researchers in this domain have been focusing in automating tasks utilizing the blockchain technology, such as developing a solution that could replace human input, by facilitating automatic cyber incident classification (Graf & King, 2018) or by developing Blockchain-based SIEM systems - for storing and accessing information security events - utilized by multiple devices, within the broader concept of the Internet of Things (Mesa et al, 2019; Miloslavskaya & Tolstoy, 2019). Although introducing some intelligent mechanisms in the developed platform is listed as a potential future task (section "8.5"), this project mainly focused in the creation of a manual, blockchain-based, reporting solution (which involves direst human involvement and supervision), since despite the recent focus in automatic mechanisms, it seems that manual reporting is still favoured in organizations (Werlinger et al, 2010; Koivunen, 2010; Metzger et al, 2011; Hove and Tårnes, 2013; Line, 2013).

Nevertheless, it seems that the interest of the research community regarding the potential applications of blockchain in information security incident reporting is swiftly intensifying. During the ending of this research project, and within the period of September 2019 and January 2020, four different research papers have emerged. Moreno et al (2020) proposed an incident response process utilizing a private blockchain network, for the recording of incidents occurring in a big data ecosystem. Furthermore, in order to increase resistance to a Sybil attack, Gong and Lee (2020) proposed a cyber threat intelligence framework which utilizes smart contracts and stores metadata of attacks on blockchain. Also, Riesco et al (2020) suggested a blockchain-based incentive model, to encourage the exchange of cyber threat and risk information, along with an Ethereum smart contract marketplace, to incentivize the sharing of information among different parties. Finally, Putz et al (2019) presented a private, blockchain-based, model for preserving the integrity of computer log records, in order for them to be successfully presented in courts. It seems that the research community is eagerly beginning to explore the interesting opportunities that blockchain unwraps in the area of incident reporting.

This research project contributes to knowledge in various ways. To begin with, it investigated subjects (incident reporting and blockchain) where the available literature is rather limited. Although incident response/management has received attention from researchers, It was identified that incident reporting, as a distinct process, has not been extensively studied, in line with previous findings (e.g. Patrascu & Patriciu, 2013; Tondel et al, 2014; Humphrey, 2017). On the other hand, the available literature in blockchain is quite limited as well, although this could be due to its infancy, as a relatively new technology. Thus, this project adds to the literature of two fields which have not been extensively studied. In addition, this work contributed towards the evaluation of existing reporting schemes and solutions, with an emphasis in manual reporting platforms. It has identified the – currently available – reporting platforms, tested their use, compared and evaluated their features, and also identified their positive and negative aspects. It has also identified the lack of standard taxonomies for information security incidents, in line with previous findings (e.g. Humphrey, 2017). Furthermore, this research project identified the blockchain applications currently available in the areas of information security, data management and incident reporting. This work also contributed to research by creating a functional, practical artefact in the blockchain domain, a domain where most studies are either experimental proposals, or theoretical concepts, with limited practicality in solving real-world problems (Taylor et al, 2019). Through this work, the first information security incident reporting “DApp” was designed, developed, and evaluated. Lastly, through the evaluation activity, and by conducting a series of non-parametric significance tests, it was identified that the developed solution could potentially increase the motivational level of users towards reporting incidents, although larger confirmatory studies are required, as discussed in sections “8.3.” and “8.4.” of this chapter.

In general, according to Morkunas et al (2019), customers are interested in purchasing a “solution” to get a job done, rather than simply purchasing “products”. Johnson et al (2008) state that the value derived by the customer increases proportionally to the importance that the customer places on the job to be done, as well as by the satisfaction level related to the current options to complete the job, the availability of other options, and their pertinent cost. The developed artefact offers an alternative option to customers (users/organizations), to perform the explicitly important job of information security incident reporting. As identified through available literature, “customers” express concerns related to their currently available reporting options, which ultimately lead to the fundamental problem of incident under-reporting. IRDA aims to put these concerns at ease, as well as significantly reduce the cost associated to incident reporting. IRDA is addressed towards a range of potential customers, including authorities and businesses, which can use the product both internally (i.e. reporting within the various departments of a single organization) or externally (i.e. reporting within a group of businesses, under a designated authority). Furthermore, the platform could be of particular interest to the various CSIRTs and CERTs around the world (and especially within EU), which could evaluate its use over their current reporting solutions, built with conventional technologies. More particularly, the early assumption that European CSIRTs/CERTs could potentially be both customers and evaluators of the decentralized platform, led to the integration and utilization of the "eCSIRT.net mkVI" incident taxonomy, since this taxonomy is endorsed by ENISA, its categories are universal and practical, and it is currently widely used amongst European CSIRTs (ENISA, 2018). This decision was taken in order to both create a familiar reporting environment for users, as well as to facilitate the transition process from another CSIRT solution to the IRDA platform.

Morkunas et al (2019) state that blockchain interest is currently focused on financial services, with "very little discussion about non-financial services and how blockchain technology may affect organizations, their business models, and how they create and deliver value". This project demonstrates that Blockchain can, indeed, be used for non-financial applications, possibly encouraging others to explore the various capabilities blockchain has to offer.

Through the accomplishment of this project's pre-set objectives and implementation targets, the developed artefact provides a positive answer to the research question: There, indeed, exists a way to create an innovative incident reporting solution, which builds upon the positive features of the existing solutions, but also provides essential added value. Hopefully, this developed software, featuring increased anonymity, availability, immutability and transparency levels, as well as an overall lower cost, will increase organizational motivation towards the reporting of incidents. IRDA successfully confronted a number of issues, identified by literature, as demotivators for incident reporting; whether it can ultimately change the dismaying statistics of incident under-reporting, remains to be seen.

8.3. Limitations

This research project was affected by various limitations. To begin with, the literature review chapter utilized resources (of both academic and professional nature) which were made available only in the English and/or Greek language.

Thus, a possibility exists, that a number of - relevant to this project – resources, in any other language, were not identified and therefore not taken into consideration.

Methodologically, this research project utilized the Design Science Research framework by Peffers et al (2007). Although DSR "has been slow to diffuse into the mainstream of Information Systems research" (Peffers et al, 2007, p.2), its legitimacy is now widely acknowledged within the academic community, and several researchers have been successful in making the case for its value and validity, through the integration of "design" as a major component of research (Peffers et al, 2007). A limitation, however, of this methodology, is that a generally accepted process for carrying out Design Science research does not exist. Nevertheless, various models/frameworks have been made available, and the framework by Peffers et al (2007) was identified as the most appropriate choice for this project, as it incorporates principles, practices and procedures necessary to conduct such research, while being consistent with prior literature (Peffers et al, 2007). Another limitation of DSR, is that although there is a common agreement, amongst researchers, that evaluation is an essential activity in conducting rigorous Design Science Research (Venable et al, 2012), the available literature on this topic seems to be rather limited, while the applicable evaluation criteria and methods are presented in a rather fragmented manner (Venable et al, 2012; Prat et al, 2014). Thus – and as is the case with conducting DSR, in general - no generally accepted process for carrying out the evaluation activity of a DSR project exists. This limitation was confronted by selecting an evaluation framework which provided extensive and appropriate guidance, and, at the same time, was consistent with the prior literature, the Venable et al (2012) DSR evaluation framework.

The financial cost of this research project was also a serious limitation. Since no financial aid was pursued, the researcher had to utilize his own, personal, resources, to fund the development of the artefact. Nevertheless, it is important to note, that not securing any external financial aid was a strategic decision, taken at the very beginning of the project's development activities. Since the researcher had the ability to secure the minimum budget for this project using his own resources, this decision was taken in order to avoid potential complications with any third parties, which could hinder the project's successful and/or timely completion. Such complications could include disagreements regarding the project's scope and overall objectives, since the sponsoring organization might have wanted to specifically tailor the requirements according to its own operational environment and needs. This could create an adverse effect regarding the produced solution's applicability, generalizability, and universality. Furthermore, the involvement of a third party could potentially delay the implementation of the platform: for example, the third party could be reviewing and approving milestones at a very slow pace, thereby creating implementation delays. The sponsoring organization could have also not been releasing funds according to schedule, thus creating further delays. Further complications could also include a sudden halt of funding, a shift in the sponsoring organization's priorities, or even a total withdrawal, following, for example, a lack of resources (both in available personnel and funds) due to a force majeure event. Therefore, and since a preliminary interest from organizations in utilizing such a solution was already suggested (through the pilot study, conducted prior to the initiation of this project), it was best decided not to pursue any sponsoring opportunities.

The first consequence of this decision, however, was that the decentralized platform had to be hosted on a single blockchain node, on the Azure environment, rather than five or six, distinct nodes. Utilizing more than a single node, would require an additional investment, and while this cost might not appear prohibitive for an organization, it was deemed as an unnecessary expense, for a single researcher, attempting to produce a proof-of-concept (and not a production-ready), artefact. Another consequence related to the cost limitations of this project, was that the researcher was only available to experience and test the trial/limited versions of other incident reporting platforms, since the cost of purchasing/renting the full version of the platforms was prohibitive. However, this was not a major issue, since most features of the platforms were also made available through the trial versions.

A final consequence of the narrow financial resources available, was the limited turnout of organizations willing to evaluate the platform. The turnout might have been greater, if a financial motive (i.e. some kind of reward/prize) was offered to organizations, for their participation. In any case, it must be noted that Cyprus is a small market, nevertheless, and the absolute number of Cypriot organizations utilizing an incident reporting solution, is not expected to be high. However, this low number of participants might have implications related to the generalizability of this project's findings. According to Hackshaw (2008, p.1143), although there is "nothing wrong with conducting well-designed small studies", the results of such studies need to be carefully interpreted, since they may yield unreliable or imprecise estimates, or they may over-estimate the magnitude of an association. The author also suggests that data from such studies "should be used towards designing larger confirmatory studies" (Hackshaw, 2008, p.1143). Section "8.5 – Future work" of this chapter, states such a future intention for conducting a larger study, in order to confirm the validity of the evaluation activity of the developed platform.

Time limitations were also an important element. The project had to be completed within a predefined period of time – according to the requirements of the University of East London. The researcher had to balance research, work and family commitments and produce the best possible outcome. Inevitably, some features were not implemented and were thus left for a future iteration of this project. These are documented in the next section of this chapter.

A further limitation of this project was the realization of pseudoanonymity rather than the true anonymity of the participants. This was a strategic decision which was taken after mature consideration and careful reflection. A truly anonymous (public) blockchain, without a central authority, would be difficult to manage - and probably ineffective. There would be no way to handle misbehaving participants, as well as the submission of spam/untrue/misleading incidents, factors which could eventually deter honest users from participating in the reporting process and undermine the platform's value. True anonymity had to be sacrificed in order to create an effective solution.

Since the developed artefact is a working prototype (a proof-of-concept solution), it did not undergo extensive testing and this poses as a further limitation. Before the artefact is ready for the production environment, it should undergo extensive grey box testing, including stress testing.

Another limitation was that incident reporting was eventually made available (through the platform) only for the “Detection” phase of the incident management lifecycle, as described by the ISO 27035 standard, and was not made available for the “Lessons learned” phase. This means that users who submit an incident, do not have the option of updating their entry with further information, after their initial submission (although they are being warned to evaluate their entry before submission). This is again a by-product of thoughtful consideration, since the researcher opted to give participants an enhanced feeling of trust towards the immutability and transparency of the solution, rather than provide them with the ability to edit incidents.

A final limitation has to do with the current state of the blockchain ecosystem, including general adoption, development and standardization. Blockchain is a new technology and although many governments, organizations and academics express an interest in its exploration, it is still in its infancy. There are no standardized procedures for development, there are no standardized features and components and the available resources and support options are very limited. Most DApps are still in experimental stage and the various blockchain components rarely glue well together and operate as planned. The development community is very small in size, while so many different blockchain implementations exist (with different capabilities, structure, programming languages, consensus algorithms and more). These factors made the task of getting help from the community, when needed, an exceptionally challenging task and made the overall development process distressing. Blockchain still has a long road ahead for mass adoption.

8.4. Discussion

To sum up, this research project had three primary goals:

- to identify and evaluate existing information security incident reporting schemes and solutions,
- to evaluate the use of blockchain technology as a resolution towards the inherent problems of existing reporting solutions, and,
- to design, develop and evaluate an incident reporting solution, which provides added value to users, and increases their level of motivation towards the reporting of information security incidents.

All goals of this project have been achieved.

The existing incident reporting platforms were identified and evaluated, and the results of this evaluation were utilized towards the design of the proposed solution. It is important to note, that although every effort was taken to identify all, available, incident reporting platforms (through a rigorous search process and by utilizing carefully crafted criteria), a possibility exists that some platforms/solutions may have not been identified. Therefore, some unique features (if any), of these hypothetically unidentified platforms, may have not been included in the overall evaluation process.

Blockchain technology has also been thoroughly examined, as part of this research project. Blockchain appeared to be a suitable candidate for accommodating the required solution. The technology appears able to confront a number of organizational concerns, such as negative publicity, through its inherent anonymity features, as well as decrease the various high costs associated with reporting and its processes. It also offers additional benefits, such as increased availability, immutability and transparency levels. Could all of these features, including anonymity, have been achieved through utilizing conventional technologies, instead of blockchain? Probably, yes. However, the cost of combining a multitude of conventional technologies towards achieving the same objectives, would probably be significantly higher. Furthermore, this is the first attempt in examining whether blockchain, specifically, with its various inherent attributes, can provide the added value that conventional technologies, cannot (and hence the problem of incident under-reporting).

Therefore, a blockchain-based incident reporting solution has been created. The solution did not attempt to provide a resolution towards all the reporting demotivators, but rather to successfully confront some of them. Confronting these individual demotivators, however, does not necessarily mean an increase in the overall user motivation for reporting. User "motivation" is a complex and multi-dimensional concept and future work (section "8.5") will attempt to further demystify it. However, through the evaluation activity, this research project suggests that by confronting a number of demotivators, the motivational reporting level of users can be improved. The literature identified a need for user anonymity in the reporting process, and the evaluation activity indicated such a rise in the anonymity level of users. The evaluation activity also indicated a decrease in cost by utilizing IRDA, another demotivator identified through literature. There were, however, various limitations, which are discussed in section "8.3", with the limited number of participants in the evaluation activity being a major limitation.

There are also other topics worthy of discussion. Although the pilot study conducted before the initiation of this project indicated that organizations could potentially be interested in utilizing such a solution, interest in IRDA cannot be taken for granted. Anonymity and low cost may not be sufficient for organizations to integrate IRDA into their reporting processes. Regarding this project's methodology, the selection of Design Science research can be thought of as being effective, as the meticulous execution of the framework by Peffers et al (2007) led to a successful end result. This, however, does not imply that selecting a different methodology (such as Action research) would not provide a similarly good output, although authors such as Iivari and Venable (2009) consider that a client-researcher relationship is required for conducting Action research.

During IRDA’s design and development, a number of identified positive features of the other platforms were successfully integrated into the proposed solution. Easy-to-use and navigate interfaces were created, familiar and standardized incident reporting forms and incident taxonomies were used, as well as other useful features, such as multi-factor authentication, encrypted communication channels, social features and more. However, even more features could have been implemented, and these are presented in section “8.5.” of this report. Also, during the demonstration activities, the administrator test cases were performed solely by the researcher, for reasons stated in section “6.2.3”. However, these test cases should have ideally been executed by a potential customer of the solution, such as a CSIRT/CERT or similar authority. Such an execution environment could potentially add up to the overall credibility level of the solution. Lastly, the demonstration activity indicated that the produced solution operates as planned, and that the design output matches the design input. However, as section “8.3.” also points out, more extensive testing could have been conducted, since only black-box testing was utilized. Due to time (and cost) considerations, such an action was unfortunately not possible to conduct during this, proof-of-concept, stage of IRDA.

8.5. Future work

This final section of this chapter describes a number of possible future enhancements to this work.

The platform is currently partly decentralized. Incident data is saved and retrieved from the chain, however the application cannot be considered as fully decentralized, since the front-end components are hosted on a traditional server environment, with a centralized structure. The solution would involve hosting the currently centralized components to a decentralized storage system. Common decentralized storage systems used in blockchain implementations, include, “IPFS”, “Storj”, “Dat”, “Swarm” and “Sia” (Nizamuddin et al, 2019; Heinisuo et al, 2019). The InterPlanetary File System (IPFS) is “a distributed file system, which integrates successful ideas from previous peer-to-peer systems, including DHTs, BitTorrent, Git, and SFS” (Benet, 2014) and appears to be the most popular choice amongst these systems (Nizamuddin et al, 2019; Heinisuo et al, 2019). It is open source, content addressable, and can be used for storing and sharing a large volume of files with high efficiency (Nizamuddin et al, 2019). Due to its design properties, it also has no single point of failure, while its ultimate goal is to build a new decentralized Internet architecture, by replacing the Hypertext Transfer Protocol (HTTP) (Heinisuo et al, 2019). In addition to the standard IPFS environment, the Azure BaaS platform also offers decentralized storage for private blockchain implementations, with a beta version of the IPFS module available on their marketplace. Although, during the latter stages of implementation actions, an effort was initiated to include this module in the platform's ecosystem, it was eventually abandoned, due to time limitations.

A final step towards the complete decentralization of the incident reporting DApp, would be the decentralization of its domain name. On the Internet, the Domain Name System (DNS) is used to translate human-readable domain names into IP addresses, which can then be loaded by internet browsers. On the Ethereum blockchain, the same task can be accomplished through the Ethereum Name Service (ENS), which, unlike the traditional, centralized, DNS, operates in a decentralized way (Antonopoulos & Wood, 2018). ENS, which is actually a DApp itself, is supported by a number of other DApps, for the registration, auction, and management of registered names (Antonopoulos & Wood, 2018). ENS could be utilized to obtain an ". eth" top-level domain for the incident reporting DApp, which would be accessible through ENS-compatible browsers.

Future work could also include actions aiming towards the overall improvement of the aesthetics, functionality and features of the platform. The current GUI might be simple and effective but lacks those design elements which could make it more aesthetically pleasing to users. Also, users of the platform are not currently informed about new incident submissions, and therefore must manually check the platform, for new entries. A mechanism could be employed (in the form of an automated e-mail message or a message received through a chatbot, in a private instant messaging application) to inform users as soon as a new incident is submitted. Furthermore, spam/test incidents can currently be seen on the platform. Even if the administrator removes the user from having access to the DApp, any spam/test/maliciously-intended incidents he/she has previously submitted will always remain on the blockchain. This was, indeed, a strategic decision, in order to increase the immutability and transparency of the platform. However, a filtering mechanism could be employed in the future, to at least enable the administrator to hide those incidents from appearing through the platform's web interface. A further enhancement could include creating an anonymous forum (to complement the chat functionality), for the users to be able to discuss important matters and essentially store that content for future reference. Currently, the anonymous chat does not save any interaction content. Another useful feature would be creating a page section in the platform for receiving live feeds (e.g. through RSS updates) from various CSIRTs or other incident-focused organizations. Also, the platform has not been optimized for viewing through mobile devices; this could be implemented at some point in the future. Nevertheless, it is important to note, that at the time of writing, there were no mobile-browsers available, with Web3.0 support for connecting to custom networks. A beta version of Metamask (mobile version) was available in both Android and Apple stores, however it only supported connections to the Ethereum main and test networks (although support for custom networks in the near future appears imminent).

Another future task could be testing the platform's durability (and user acceptance and response) in a public environment, by deploying the application into Ethereum's main network. Although some changes would have to be implemented beforehand (such as switching to a different consensus algorithm, since PoA is not currently supported in Ethereum main net) it would be very interesting to see how users would behave in such a scenario, and whether Adebayo et al's (2019) theoretical framework could indeed work in practice.

Future work, however, does not only include tasks of a technical nature. Beginning with the examination of the users' motivation for reporting incidents, this project successfully confronted some of the demotivators, which according to the identified literature, cause the issue of incident under-reporting. Through the evaluation activity, some individual elements (e.g. anonymity, cost) which directly affect user reporting motivation were examined and evaluated, and an improvement in these elements (through the new platform) was signified. Nevertheless, whether this improvement in individual elements can improve the overall user motivation for reporting was not explicitly examined. As seen from various studies (e.g. Johnson, 2002; Metzger et al, 2011; Ahmad et al, 2012; Etzioni, 2014; Ruefle et al, 2014; Humphrey, 2017; Housen-Couriel, 2018) there is a variety of reasons which can have an effect on the user/organizational motivation for reporting. Motivation can be considered as a key factor towards the increase of incident reporting. No matter the technology, uniqueness, ease of use and overall attractiveness of a solution, if the overall reporting motivation is low, then under-reporting will constantly remain an issue. Nevertheless, motivation appears to be a complex notion and a lot of questions remain unanswered: is creating an organizational culture, which encourages reporting, adequate, for an organization to increase its reporting statistics? If employees are financially, or otherwise, rewarded for reporting incidents, can that increase their reporting rate? On the other hand, could punishment be a more effective solution than reward? Are there any employee behavioural traits, which determine whether an employee essentially reports incidents, or rather prefers to ignore them? Is there a fine line between motivation and over-motivation? Could over-motivation actually increase the rate of false-positive incident reports and thus create other issues? All of these questions could yield interesting results. A future task could thus involve the examination of organizational/user reporting motivation, through a more holistic approach.

Furthermore, and as already mentioned in the limitations section, the low number of participants during the evaluation activity limit the generalizability of this project’s findings. Therefore, a future, confirmatory, evaluation activity, utilizing a significantly greater number of participants (probably not just from Cyprus, but from other parts of the world) would certainly be of high value. In addition, this work relied solely on secondary data from existing literature, in order to identify the organizational demotivators for the problem of incident under-reporting. A future task could include designing a study towards the collection of primary data, in order to confirm the validity of existing demotivators or to identify any new ones, and to examine whether these align well with existing literature.

Regarding incident reporting, in general, this project also identified a lack of standard taxonomies for security incidents, in line with previous recent findings (e.g. studies by Humphrey, 2017; ENISA, 2018). It would be interesting to examine why this complexity exists and whether it would be possible to design a universally accepted taxonomy. This would probably make the sharing of incidents between organizations a lot more efficient and could also aid in the overall harmonization of statistics.

A final future task could include the possible exploitation of intelligence systems in the reporting process, in line with the emergence of a newer stage of blockchain technology, “Blockchain 4.0”. As already stated in the literature review chapter, this stage involves the inclusion of artificial intelligence in the blockchain environment: since AI allows computers to learn from data, while blockchain provides data accuracy, which is useful for feeding data into the AI system and for recording its outputs, the benefits of both worlds can be combined (Angelis & Da Silva, 2019). A future effort could include the introduction of mechanisms in the blockchain reporting app, which could predict future attacks and targets, based on historical data and current trends, as well as the introduction of a safety net, for identifying possible false-positive incident submissions, based again on existing data and examination of tendencies.

LIST OF REFERENCES

Abrams, M., Jajodia, S., and Podell H., 1995, Information Security: An Integrated Collection of Essays, IEEE Computer Society Press.

Acquisti A, Friedman A, Telang R, 2006, Is there a cost to privacy breaches? An event study. The Fifth Workshop on the Econom. Inform. Security (WEIS), Robinson College, University of Cambridge, London.

Adebayo, A., Rawat, D.B., Njilla, L. and Kamhoua, C.A., 2019, Blockchain-enabled Information Sharing Framework for Cybersecurity. Blockchain for Distributed Systems Security, p.143.

Ahmad, A., Hadgkiss, J., and Ruighaver, A.B, 2012, Incident Response Teams–Challenges in Supporting the Organisational Security Function, Computers & Security (31:5), pp. 643-652.

Ahmad, A., Maynard, S.B. and Shanks, G., 2015. A case analysis of information systems and security incident responses. International Journal of Information Management, 35(6), pp.717-723.

Albakri, A., Boiten, E. and De Lemos, R., 2018, Risks of sharing cyber incident information. In Proceedings of the 13th International Conference on Availability, Reliability and Security (p. 58). ACM.

Alharby Maher & Aad van Moorsel, 2017, Blockchain-based Smart Contracts: A Systematic Mapping Study, Fourth International Conference on Computer Science and Information Technology (CSIT-2017)

Amazon, 2019, Blockchain on AWS - Easily build scalable blockchain and ledger solutions [online] Available at: https://aws.amazon.com/blockchain/

Ambre, A. Shekokar, N., 2015, Insider threat detection using log analysis and event correlation. Procedia Computer Science, 45, pp.436-445.

Ammous, S., 2016. Blockchain Technology: What is it good for?, Available at SSRN 2832751.

Amoroso E., 1994, Fundamentals of Computer Security Technology, Prentice-Hall PTR, Upper Saddle River, NJ.

Androulaki Elli, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolić, Sharon Weed Cocco, and Jason Yellick, 2018, Hyperledger fabric: a distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference (EuroSys '18). ACM, New York, NY, USA, Article 30, DOI: https://doi.org/10.1145/3190508.3190538

Angelis, J. and da Silva, E.R., 2019. Blockchain adoption: A value driver perspective. Business Horizons, 62(3), pp.307-314.

Antonopoulos, A.M. and Wood, G., 2018. Mastering ethereum: building smart contracts and dapps. O'Reilly Media.

Archer, L.B, 1984, Systematic method for designers. In, Cross, N., (ed.), Developments in design methodology, London: John Wiley, 57-82.

Armerding, T., 2018, The 18 biggest data breaches of the 21st century. [online] CSO Online. Available at: https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html

Asada, Y., Kanno, T. and Furuta, K., 2006, Application of Semantic Web to Incident Reporting. In 2006 SICE-ICASE International Joint Conference (pp. 955-958). IEEE.

Aviram, A. and Tor, A., 2003, Overcoming impediments to information sharing. Ala. L. Rev., 55, p.231.

Axon, L., 2015, Privacy-awareness in Blockchain-based PKI, CDT Technical paper series 21/15

Ayres, L.T., Curtin, C.M. and Ng, T.A., 2010, Standardizing breach incident reporting: Introduction of a key for hierarchical classification. In 2010 Fifth IEEE International Workshop on Systematic Approaches to Digital Forensic Engineering (pp. 79-83). IEEE.

Babüroglu, O.N. and Ravn, I., 1992, Normative action research. Organization Studies, 13(1), pp.019-34.

Bach, L.M., Mihaljevic, B. and Zagar, M., 2018, Comparative analysis of blockchain consensus algorithms, In 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (pp. 1545-1550), IEEE.

Baliga, A., 2017, Understanding blockchain consensus models, In Persistent.

Baliga, A., Subhod, I., Kamat, P. and Chatterjee, S., 2018, Performance evaluation of the quorum blockchain platform, arXiv preprint arXiv:1809.03421.

Baskerville, R., 1999, Investigating information systems with action research. Communications of the association for information systems, 2(1), p.19.

Baskerville R., 2008, What design science is not, European Journal of Information Systems, 17:5, 441-443, DOI: 10.1057/ejis.2008.45

Baskerville, R., Baiyere, A., Gregor, S., Hevner, A. and Rossi, M., 2018, Design science research contributions: finding a balance between artifact and theory. Journal of the Association for Information Systems, 19(5), p.3.

Baskerville, R., Spagnoletti, P. and Kim, J., 2014, Incident-centered information security: Managing a strategic balance between prevention and response. Information & management, 51(1), pp.138-151.

Baskerville, R. and Myers, M.D., 2004, Special issue on action research in information systems: Making IS research relevant to practice: Foreword. MIS quarterly, pp.329-335.

Baskerville, R. and Wood-Harper, A.T., 1998, Diversity in information systems action research methods. European Journal of information systems, 7(2), pp.90-107.

Belsis, M.A., Simitsis, A. and Gritzalis, S., 2005, Workflow based security incident management. In Panhellenic Conference on Informatics (pp. 684-694). Springer, Berlin, Heidelberg.

Benbasat, I., and Zmud, R. W, 1999, Empirical Research in Information Systems: The Practice of Relevance, MIS Quarterly (23:1), pp. 3-16.

Ben-Menahem A., 2018, 3 reasons why Azure's infrastructure is secure [online] Available at: https://azure.microsoft.com/en-us/blog/3-reasons-why-azure-s-infrastructure-is-secure/

Benet, J., 2014. Ipfs-content addressed, versioned, p2p file system. arXiv preprint arXiv:1407.3561.

Berenson, M., Levine, D. & Kreihbel, T., 2004, Basic business statistics: concepts and application, (9th ed.), Englewood Cliffs, NJ:Prentice-Hall Publishing Co.

Bernsmed, K. and Tondel, I.A., 2013, Forewarned is forearmed: indicators for evaluating information security incident management, In 2013 Seventh International Conference on IT Security Incident Management and IT Forensics (pp. 3-14), IEEE.

Blaikie, N., 2003, Analyzing quantitative data: From description to explanation, Sage.

Blair, R.C. and Higgins, J.J., 1985, A comparison of the power of the paired samples rank transform statistic to that of Wilcoxon's signed ranks statistic, Journal of Educational Statistics, 10(4), pp.368-383.

Bodó, R. and Kouril, D., 2014, Efficient Management of System Logs using a Cloud, In The International Symposium on Grids and Clouds (ISGC) 2013 (Vol. 179, p. 009), SISSA Medialab.

Bogdan, R.C., Biklin S.K, 1998, Qualitative research for education: An introduction to theory and methods, 3rd ed., Boston: Allyn and Bacon.

Boland, R.J. and Lyytinen, K., 2004, Information systems research as design: Identity, process, and narrative. In Information Systems Research (pp. 53-68), Springer, Boston, MA.

Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J.A. and Felten, E.W., 2015, Sok: Research perspectives and challenges for bitcoin and cryptocurrencies, In 2015 IEEE Symposium on Security and Privacy (pp. 104-121), IEEE.

Bonneau, J., Narayanan, A., Miller, A., Clark, J., Kroll, J.A. and Felten, E.W., 2014, Mixcoin: Anonymity for Bitcoin with accountable mixes, In International Conference on Financial Cryptography and Data Security (pp. 486-504), Springer, Berlin, Heidelberg.

Bowerman, B. & O'Connell, R., 2007, Business statistics in practice, New York: McGraw-Hill Irwin Publishing Co.

Bragagnolo, S., Rocha, H., Denker, M. and Ducasse, S., 2018, SmartInspect: solidity smart contract inspector, In 2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE) (pp. 9-18), IEEE.

Brékine, A., Papathanasiou, A., Kavallieros, D., Ziegler, S., Hemmens, C., Rodriguez, A.Q., Germanos, G., Kokkinis, G., Leventakis, G., Armin, J. and Bothos, J., 2019, Network Threat Analysis. In Internet of Things Security and Data Protection (pp. 81-92), Springer, Cham

Briggs, P., Jeske, D. and Coventry, L., 2017, The design of messages to improve cybersecurity incident reporting. In International Conference on Human Aspects of Information Security, Privacy, and Trust (pp. 3-13), Springer, Cham

Brown, R. G., Carlyle, J., Grigg, I., and Hearn, M, 2016, Corda: an introduction. R3 CEV, August.

Buterin, V., 2014, Ethereum white paper: a next generation smart contract & decentralized application platform, First version.

Buterin, V., 2018. What is Ethereum?, Ethereum Official webpage, [online] Available: http://www. ethdocs. org/en/latest/introduction/what-is-ethereum.html

Buterin, V. and Griffith, V., 2017, Casper the friendly finality gadget. arXiv preprint arXiv:1710.09437.

Cachin, C. and Vukolić, M., 2017, Blockchain consensus protocols in the wild. arXiv preprint arXiv:1707.01873.

Cai, W., Wang, Z., Ernst, J.B., Hong, Z., Feng, C. and Leung, V.C., 2018, Decentralized applications: The blockchain-empowered software system. IEEE Access, 6, pp.53019-53033.

Campbell, K., Gordon, L.A., Loeb, M.P. and Zhou, L., 2003, The economic cost of publicly announced information security breaches: empirical evidence from the stock market, Journal of Computer Security, 11(3), pp.431-448.

Carlozo, L., 2017, What is blockchain?, Journal of Accountancy, 224(1), p.29.

Casey, M.J. and Vigna, P., 2018, In blockchain we trust. Technol. Rev, 121, pp.10-16.

Casino Fran, Thomas K. Dasaklis, Constantinos Patsakis, 2019, A systematic literature review of blockchain-based applications: Current status, classification and open issues, Telematics and Informatics, Volume 36, Pages 55-81.

Castro, M. and Liskov, B., 2002, Practical Byzantine fault tolerance and proactive recovery. ACM Transactions on Computer Systems (TOCS), 20(4), pp.398-461.

Cavusoglu H, Mishra B, Raghunathan S, 2004, The effect of Internet security breach announcements on market value of breached firms and Internet security developers, Internat. J. Electronic Commerce 9(1):69–105.

Cebe, M., Erdin, E., Akkaya, K., Aksu, H., & Uluagac, A.S., 2018, Block4Forensic: An Integrated Lightweight Blockchain Framework for Forensics Applications of Connected Vehicles, CoRR, abs/1802.00561.

Chabinsky, S., 2014, The business necessity of cybersecurity: It's not an IT issue.Security: Solutions for Enterprise Security Leaders, 51(3), 56.

Chase, B. and MacBrough, E., 2018, Analysis of the XRP Ledger consensus protocol, arXiv preprint arXiv:1802.07242.

Chatzigeorgiou, C., Toumanidis, L., Kogias, D., Patrikakis, C. and Jacksch, E., 2017, A communication gateway architecture for ensuring privacy and confidentiality in incident reporting, In 2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA) (pp. 407-411), IEEE.

Chen, G., Xu, B., Lu, M. and Chen, N.S., 2018, Exploring blockchain technology and its potential applications for education, Smart Learning Environments, 5(1), p.1.

Chen, T., Li, X., Luo, X. and Zhang, X., 2017, Under-optimized smart contracts devour your money, In 2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER) (pp. 442-446), IEEE.

Cheng, L., Liu, F. and Yao, D., 2017, Enterprise data breach: causes, challenges, prevention, and future directions. WIREs Data Mining Knowl Discov, 7: e1211. doi:10.1002/widm.1211

Cheswick W and Bellovin S, 1994, Firewalls and Internet Security: Repelling the Wily Hacker, Addison-Wesley Publishing Company, Reading, MA.

Choo R, 2011, The cyber threat landscape: Challenges and future research directions. Computers and Security 30 (2011) 719-731

Chou, Y. L., 1989, Statistical analysis for business and economics, New York: Elsevier Science Publishing Co.

Christidis, K. & Devetsikiotis, M., 2016, Blockchains and smart contracts for the internet of things, IEEE Access 4, 2292–2303.

ChronoBank, 2018, Solidity storage sample contract, Github, [online] Available at: https://github.com/ChronoBank/solidity-storage-lib/blob/master/contracts/Storage.sol

Cohen F, 1995, Protection and Security on the Information Superhighway, John Wiley & Sons, New York, 1995.

Cohen F, 1997, Information System Attacks: A Preliminary Classification Scheme, Computers and Security, Vol. 16, No. 1, pp. 29-46.

Cohen, L. and Manion, L., 1994, Research methods in education, 4th ed., London: Routledge.

Cole, R., Purao, S., Rossi, M. and Sein, M., 2005, Being proactive: where action research meets design research. ICIS 2005 Proceedings, p.27.

Collis, J. and Hussey, R., 2013, Business research: A practical guide for undergraduate and postgraduate students. Macmillan International Higher Education.

Concordat UK to Support Research Integrity, 2012, Universities UK [online] Available at: https://www.universitiesuk.ac.uk/policy-and-analysis/reports/Documents/2012/the-concordat-to-support-research-integrity.pdf

Condon, E., He, A. and Cukier, M., 2008, Analysis of computer security incident data using time series models. In 2008 19th International Symposium on Software Reliability Engineering (ISSRE) (pp. 77-86). IEEE.

Conover, W.J., 1973, On methods of handling ties in the Wilcoxon signed-rank test, Journal of the American Statistical Association, 68(344), pp.985-988.

Conover, W. J., 1999, Practical Nonparametric Statistics. Wiley, Hoboken, NJ. 3rd edition.

Cook, T. and Campbell, D., 1979, Quasi-experimentation: design and analysis issues for field settings, Houghton Mifflin: Boston.

Coombs WT and Holladay SJ, 2012, The Handbook of Crisis Communication: John Wiley & Sons.

Couch, S., Kazan, Z., Shi, K., Bray, A. and Groce, A., 2018, A Differentially Private Wilcoxon Signed-Rank Test, arXiv preprint arXiv:1809.01635.

Courtois, N.T. and Mercer, R., 2017, Stealth Address and Key Management Techniques in Blockchain Systems, In ICISSP (pp. 559-566).

Creswell J W., 2014, Research Design: Qualitative, Quantitative, and mixed methods approaches, Sage publishing, 4th edition

Cusick J, Ma G., 2010, Creating an ITIL inspired incident management approach: roots, response, and results, In: Network Operations and Management Symposium Workshops (NOMS Wksps), 2010 IEEE/IFIP; pp. 142e8.

CYCSO – Cyprus Cyber Security Organization, 2018, About CYCSO [online] Available at: http://www.cycso.org/

Danyliw, R., 2016. The Incident Object Description Exchange Format Version 2 (No. RFC 7970) [online] Available at: https://tools.ietf.org/html/rfc7970

Davis, A.J., Kamal, M., Schoonover, T.V., Nabukenya, J., Pietron, L.R. and Vreede, G.D., 2006, Incident response planning using collaboration engineering process development and validation, In Inaugural Workshop on Information Security and Assurance (WISA).

Davison, R. M., Martinsons, M. G., and Kock, N., 2004, Principles of Canonical Action Research, Information Systems Journal (14:1), pp. 65-86.

De Angelis, S., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A. and Sassone, V., 2018, Pbft vs proof-of-authority: applying the cap theorem to permissioned blockchain.

Demeyer, S., 2011, Research methods in computer science, In 27th IEEE International Conference on Software Maintenance (ICSM).

Demšar, J., 2006, Statistical comparisons of classifiers over multiple data sets, Journal of Machine learning research, pp.1-30.

Denicolo, P. and L. Becker, 2012, Developing research proposals. Sage.

Denning, D., 1999, Information Warfare and Security, Upper Saddle River, NJ, Pearson.

Dennis A., 2001, Relevance in Information Systems Research, Communications of the AIS (6).

DePoy, E. and L. N. Gitlin, 2015, Introduction to research: Understanding and applying multiple strategies. Elsevier Health Sciences.

Derrick, B. and White, P., 2017, Comparing two samples from an individual Likert question, International Journal of Mathematics and Statistics, 18(3).

Di Pierro, M., 2017, What is the blockchain?. Computing in Science & Engineering, 19(5), pp.92-95.

Dilenschneider, R.L. and Hyde, R.C., 1985, Crisis communications: Planning for the unplanned. Business Horizons, 28(1), pp.35-38.

Dinh, T.T.A., Wang, J., Chen, G., Liu, R., Ooi, B.C. and Tan, K.L., 2017, Blockbench: A framework for analyzing private blockchains, In Proceedings of the 2017 ACM International Conference on Management of Data (pp. 1085-1100), ACM.

Do, H.G. and Ng, W.K., 2017, Blockchain-based system for secure data storage with private keyword search, In 2017 IEEE World Congress on Services (SERVICES) (pp. 90-93), IEEE.

Doane, D. & Seward, L., 2007, Applied statistics in business and economics, New York:McGraw-Hill Irwin Publishing Co.

EBA - European Banking Authority, 2017, Guidelines on major incident reporting under Directive (EU) 2015/2366 (PSD2) [online] Available at: https://eba.europa.eu/regulation-and-policy/payment-services-and-electronic-money/guidelines-on-major-incidents-reporting-under-psd2

Eekels, J., and Roozenburg, N.F.M., 1991, A methodological comparison of the structures of scientific research and engineering design: their similarities and differences, Design Studies, 12, 4, 197-203.

ENISA, 2010, Good Practice Guide for Incident Management [online] Available at: https://www.enisa.europa.eu/publications/good-practice-guide-for-incident-management

ENISA, 2012, Cyber Incident Reporting in the EU: An overview of security articles in EU legislation [online] Available at: https://www.enisa.europa.eu/publications/cyber-incident-reporting-in-the-eu/at_download/fullReport

ENISA, 2013, Incident Reporting for Cloud Computing [online] Available at: https://www.enisa.europa.eu/publications/incident-reporting-for-cloud-computing

ENISA, 2018, Reference Incident Classification Taxonomy: Task Force Status and Way Forward [online] Available at: https://www.enisa.europa.eu/publications/reference-incident-classification-taxonomy

Ermilov, D., Panov, M. and Yanovich, Y., 2017, Automatic Bitcoin address clustering, In 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 461-466), IEEE.

Estdale, J. and Georgiadou, E., 2018, Applying the ISO/IEC 25010 Quality Models to Software Product. In European Conference on Software Process Improvement (pp. 492-503), Springer, Cham.

Ettredge ML, Richardson VJ, 2003, Information transfer among Internet firms: The case of hacker attacks, J. Inform. Systems 17(2):71–82.

Farell, R., 2015, An analysis of the cryptocurrency industry, Wharton Research Scholars, 130.

Etzioni, A., 2014, The private sector: A reluctant partner in cybersecurity, Geo. J. Int'l Aff., 15, p.69.

Fan, K., Ren, Y., Wang, Y., Li, H. and Yang, Y., 2017, Blockchain-based efficient privacy preserving and data sharing scheme of content-centric network in 5G, IET Communications, 12(5), pp.527-532.

Finne, T., 2000, Information systems risk management: Key concepts and business processes, Computers& Security, 19, 3, 234-242.

Fitzpatrick, K.R. and Rubin, M.S., 1995, Public relations vs. legal strategies in organizational crisis decisions, Public Relations Review, 21(1), pp.21-33.

Friedlmaier, M., Tumasjan, A. and Welpe, I.M., 2018, Disrupting industries with blockchain: The industry, venture capital funding, and regional distribution of blockchain ventures. In Venture Capital Funding, and Regional Distribution of Blockchain Ventures, Proceedings of the 51st Annual Hawaii International Conference on System Sciences (HICSS).

García-Barriocanal, E., Sánchez-Alonso, S. and Sicilia, M.A., 2017, Deploying metadata on blockchain technologies, In Research Conference on Metadata and Semantics Research (pp. 38-49), Springer, Cham.

Garg A, Curtis J, Halper H, 2003, Quantifying the financial impact of IT security breaches, Inform. Management & Computer Security 11(2):74–83.

Gatteschi V., F. Lamberti, C. Demartini, C. Pranteda and V. Santamaría, 2018, To Blockchain or Not to Blockchain: That Is the Question, in IT Professional, vol. 20, no. 2, pp. 62-74, Mar./Apr. 2018. doi: 10.1109/MITP.2018.021921652

Geisler, J., 2010, Software for Medical Systems, In Mission-Critical and Safety-Critical Systems Handbook (pp. 147-268), Newnes.

Gemalto's Breach Level Index Report, 2018, Breach Level Index: Data Privacy & New Regulations take place [online], Available at: https://breachlevelindex.com/request-report

Glisson, W.B., McDonald, A., and Welland, R., 2006, Web Engineering Security: A Practitioner's Perspective, in: 6th international conference on Web engineering, Palo Alto, USA.

Goharshady A, Behrouz A, Chatterjee K, 2018, Secure Credit Reporting on the Blockchain [online] Available at: https://arxiv.org/abs/1805.09104

Gong, S. and Lee, C., 2020. BLOCIS: Blockchain-Based Cyber Threat Intelligence Sharing Framework for Sybil-Resistance. Electronics, 9(3), p.521.

Gonzalez J.J., 2005, Towards a Cyber Security Reporting System – A Quality Improvement Process. In: Winther R., Gran B.A., Dahll G. (eds) Computer Safety, Reliability, and Security. SAFECOMP 2005, Lecture Notes in Computer Science, vol 3688. Springer, Berlin, Heidelberg

Gordon L, Martin P. Loeb, William Lucyshyn, 2003, Sharing information on computer systems security: An economic analysis,Journal of Accounting and Public Policy, Volume 22, Issue 6, 2003, Pages 461-485

Gordon, L.A., Loeb, M.P. and Sohail, T., 2010, Market value of voluntary disclosures concerning information security, MIS quarterly, pp.567-594.

Gordon L., Martin P. Loeb, William Lucyshyn, Lei Zhou, 2015, The impact of information sharing on cybersecurity underinvestment: A real options perspective, Journal of Accounting and Public Policy, Volume 34, Issue 5, Pages 509-519

Graf R and King R, 2018, Neural network and blockchain based technique for cyber threat intelligence and situational awareness, 10th International Conference on Cyber Conflict (CyCon), Tallinn, pp. 409-426. doi: 10.23919/CYCON.2018.8405028

Greenspan G., 2015a, Ending the Bitcoin vs Blockchain Debate, MultiChain, blog, [online]. Available: http://www.multichain.com/blog/2015/07/bitcoin-vsblockchain-debate/

Greenspan, G., 2015b, Avoiding the pointless blockchain project. MultiChain, blog. [online] Available at: https://www.multichain.com/blog/2015/11/avoiding-pointless-blockchain-project/

Greenspan, G., 2015c, Multichain private blockchain-white paper. [online] Available at: http://www.multichain.com/download/MultiChain-White-Paper.pdf.

Gregor, S. and Hevner, A.R., 2013, Positioning and presenting design science research for maximum impact, MIS quarterly, pp.337-355.

Grimaila, M.R., Mills, R.F., and Fortson, L.W., 2008, An Automated Information Asset Tracking Methodology to Enable Timely Cyber Incident Mission Impact Assessment, Proceedings of the

2008 International Command and Control Research and Technology Symposium (ICCRTS 2008), Bellevue, WA.

Grimaila, M.R., Schechtman, G., Mills, R.F. and Fortson, L.W., 2009, Improving cyber incident notification in military operations, In IIE Annual Conference. Proceedings (p. 2357), Institute of Industrial and Systems Engineers (IISE).

Grispos, G., Glisson, W.B. and Storer, T., 2014, Rethinking security incident response: The integration of agile principles, arXiv preprint arXiv:1408.2431.

Grispos, G., Glisson, W., and Storer, T., 2015, Security Incident Response Criteria: A Practitioner's Perspective, in The 21st Americas Conference on Information Systems, Puerto Rico, USA.

Grispos, G., Glisson, W.B., Bourrie, D., Storer, T. and Miller, S., 2017, Security incident recognition and reporting (SIRR): an industrial perspective, arXiv preprint arXiv:1706.06818.

Hackshaw A., 2008, European Respiratory Journal Nov 2008, 32 (5) 1141-1143; DOI: 10.1183/09031936.00136408

Haferkorn, M. and Diaz, J.M.Q., 2014, Seasonality and interconnectivity within cryptocurrencies-an analysis on the basis of Bitcoin, Litecoin and Namecoin, In International Workshop on Enterprise Applications and Services in the Finance Industry (pp. 106-120), Springer, Cham.

Hansman, S., & Hunt, R., 2005, A taxonomy of network and computer attacks, Computers & Security, 24, 31–43.

Harrison, K. and White, G., 2012, Information sharing requirements and framework needed for community cyber incident detection and response, In 2012 IEEE Conference on Technologies for Homeland Security (HST) (pp. 463-469), IEEE.

Hasan, H., 2003, Information systems development as a research method, Australasian Journal of Information Systems, 11 (1), 4-13.

Hassani, H., 2017, Research methods in computer science: The challenges and issues, arXiv preprint arXiv:1703.04080.

Hausken, K., 2007, Information sharing among firms and cyber-attacks, Journal of Accounting and Public Policy, 26(6), pp.639-688.

He, Y., Johnson, C.W., 2012, Generic security cases for information system security in healthcare systems.

He, Y., Johnson, C., Renaud, K., Lu, Y. and Jebriel, S., 2014, An empirical study on the use of the generic security template for structuring the lessons from information security incidents, In 2014 6th International Conference on Computer Science and Information Technology (CSIT) (pp. 178-188), IEEE.

He, Y, Johnson, C., 2017, Challenges of information security incident learning: An industrial case study in a Chinese healthcare organization; Informatics for Health and Social Care, Vol.42(4), p.393-408

Heinisuo, O.P., Lenarduzzi, V. and Taibi, D., 2019, Asterism: Decentralized File Sharing Application for Mobile Devices, In 2019 7th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud) (pp. 38-47), IEEE.

Hennin, S., 2008, Control system cyber incident reporting protocol, In 2008 IEEE Conference on Technologies for Homeland Security (pp. 463-468), IEEE.

Hevner, A., & Chatterjee, S., 2010, Design Research in Information Systems: Theory and Practice, New York: Springer.

Hevner, A., March, S.T., and Park, 2004, J. Design Research in Information Systems Research. MIS Quarterly, 28, 1, pp. 75-105.

Hoffman, J.I., 2015, Comparison of two groups in Biostatistics for medical and biomedical practitioners, Academic Press.

Hothorn T., 2019, Coin - Conditional Inference Procedures in a Permutation Test Framework [online] Available at: https://www.rdocumentation.org/packages/coin/versions/1.3-1

Housen-Couriel D, 2018, Information Sharing for the Mitigation of Hostile Activity in Cyberspace: Comparing Two Nascent Models, European Cybersecurity Journal, 4(3), pp.44-50.

Hovav A, D'Arcy J, 2003, The impact of denial-of-service attack announcements on the market value of firms, Risk Management and Insurance Rev. 6(2):97–121.

Hove C, Tårnes M., 2013, Information security incident management: an empirical study of current practice, Norwegian University of Science and Technology

Hove, C., Tårnes, M., Line, M.B. and Bernsmed, K., 2014, Information security incident management: identified practice in large organizations, In 2014 Eighth international conference on IT security incident management & IT forensics (pp. 27-46), IEEE.

Howard, J.D., Longstaff, T.A., 1998, A common language for computer security incidents (No. SAND98-8667), Sandia National Labs., Albuquerque, NM (US); Sandia National Labs., Livermore, CA (US).

Hsu, C., Wang, T., Lu, A., 2016, The Impact of ISO 27001 certification on firm performance, In 2016 49th Hawaii International Conference on System Sciences (HICSS) (pp. 4842-4848), IEEE.

Hughes, A., Park, A., Kietzmann, J. and Archer-Brown, C., 2019, Beyond Bitcoin: What blockchain and distributed ledger technologies mean for firms, Business Horizons.

Humphrey, M., 2017. Identifying the critical success factors to improve information security incident reporting (Doctoral dissertation).

Hung, C.C., Chen, K. and Liao, C.F., 2019, Modularizing Cross-Cutting Concerns with Aspect-Oriented Extensions for Solidity, In 2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON) (pp. 176-181), IEEE.

Husák, M. and Cegan, J., 2014, PhiGARo: Automatic phishing detection and incident response framework, In 2014 Ninth International Conference on Availability, Reliability and Security (pp. 295-302), IEEE.

Iansiti, M. and Lakhani, K.R., 2017, The truth about blockchain. FderHarvard Business Review, 95(1), pp.118-127.

IBM, 2017, Three ways blockchain Explorers chart a new direction [online] Available at: https://www.ibm.com/services/insights/c-suite-study/blockchain

IBM, 2019, IBM Blockchain - Now delivering value around the world [online] Available at: https://www.ibm.com/blockchain

Icove D, Seger K and VonStorch W, 1995, Computer Crime: A Crimefighter's Handbook, O'Reilly & Associates, Inc., Sebastopol, CA

Iivari J., 2007, A paradigmatic analysis of information systems as a design science, Scandinavian Journal of Information Systems, 19 (2): 39.

Iivari, J. and Venable, J.R., 2009, Action research and design science research-seemingly similar but decisively dissimilar.

IOD & Barclays Policy report, 2016, Cyber Security: Underpinning the digital economy [online] Available at: https://www.iod.com/Portals/0/Badges/PDF's/News%20and%20Campaigns/Infrastructure/Cyber%20security%20underpinning%20the%20digital%20economy.pdf?ver=2016-04-14-101230-913

Ipsos MORI Social Research Institute and the University of Portsmouth, 2017, Cyber security breaches survey 2017, version 4.5 [online] Available at: https://www.ipsos.com/sites/default/files/2017-04/sri-cybersecurity-breaches-survey-2017.pdf

ISO/IEC 15288: 2015, 2015, Systems and Software Engineering: System Life Cycle Processes [online] Available at: https://www.iso.org/standard/63711.html

ISO/IEC 25010:2011, 2011, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models [online] Available at: https://www.iso.org/standard/35733.html

ISO/IEC 27001:2013, 2013, Information technology -- Security techniques -- Information security management systems -- Requirements [online] Available at: https://www.iso.org/standard/54534.html

ISO/IEC 27002:2013, 2013, Information technology -- Security techniques -- Code of practice for information security controls [online] Available at: https://www.iso.org/standard/54533.html

ISO/IEC 27035:2011, 2011, Information technology - Security techniques - Information security incident management, 2011

ISO/IEC 27035:2016, 2016, Information technology -- Security techniques -- Information security incident management -- Part 1: Principles of incident management [online] Available at: https://www.iso.org/standard/60803.html

Jaatun MG, Albrechtsen E, Line MB, Tøndel IA, Longva OH., 2009, A framework for incident response management in the petroleum industry, Int J Crit Infrastruct Prot, 2:26e37.

Jaikumar, V, 2002, "Organisations should build an incident response team", Computer World Canada, vol.9, no.16.

Jamieson, S., 2004, Likert scales: how to (ab) use them, Medical education, 38(12), pp.1217-1218.

Jarvinen, P., 2007, Action research is similar to design science, Quality & Quantity, 41(1), pp.37-54

Jiang, P., Guo, F., Liang, K., Lai, J. and Wen, Q., 2017, Searchain: Blockchain-based private keyword search in decentralized storage, Future Generation Computer Systems.

Johnson, C. 2002, Reasons for the Failure of Incident Reporting in the Healthcare and Rail Industries, in Components of System Safety, Springer, pp. 31-57

Johnson, C.W., 2015, Contrasting approaches to incident reporting in the development of safety and security–critical software.

Johnson, C., Badger, M., Waltermire, D., Snyder, J. and Skorupka, C., 2016, Guide to cyber threat information sharing (No. NIST Special Publication (SP) 800-150 (Draft)), National Institute of Standards and Technology.

Johnson, M.W., Christensen, C.M. and Kagermann, H., 2008, Reinventing your business model, Harvard business review, 86(12), pp.57-68.

Johnson, R.B., Onwuegbuzie, A.J. and Turner, L.A., 2007, Toward a definition of mixed methods research, Journal of mixed methods research, 1(2), pp.112-133.

Joyce, A.L., Evans, N., Tanzman, E.A. and Israeli, D., 2016, International cyber incident repository system: Information sharing on a global scale, In 2016 International Conference on Cyber Conflict (CyCon US) (pp. 1-6), IEEE.

Kacha, P., 2010, OTRS: CSIRT WorkFlow Improvements. CESNET, Tech. Rep., 10.

Kacha, P., 2014, Idea: security event taxonomy mapping, In 18th International Conference on Circuits, Systems, Communications and Computers.

Kamath, R., 2018, Food traceability on blockchain: Walmart's pork and mango pilots with IBM. The JBBA, 1(1), p.3712.

Kampanakis, P., 2014, Security automation and threat information-sharing options, IEEE Security & Privacy, 12(5), pp.42-51.

Kane, E., 2017, Is Blockchain a General-Purpose Technology? [online] Available at SSRN: https://ssrn.com/abstract=2932585

Kannan K, Rees J, Sridhar S, 2007, Market reactions to information security breach announcements: An empirical study, Internat. J. Electronic Commerce 12(1):69–91

Kaspersky, 2017, The Human Factor in IT Security: How Employees are Making Businesses Vulnerable from Within [online] Available at: https://www.kaspersky.com/blog/the-human-factor-in-it-security

Kaufmann JB and Kesner IF, 1994, The myth of full disclosure: A look at organizational communications during crises, Business Horizons, vol. 37, p. 29

Kavassalis P, Harald Stieber, Wolfgang Breymann, Keith Saxton, Francis Joseph Gross, 2018, An innovative RegTech approach to financial risk monitoring and supervisory reporting, The Journal of Risk Finance, Vol. 19 Issue: 1, pp.39-55.

Keller, G., 2005, Statistics for management and economic (7th ed.), Belmont, CA: Thomson Publishing Co.

Khan, M.E. and Khan, F., 2012, A comparative study of white box, black box and grey box testing techniques, Int. J. Adv. Comput. Sci. Appl, 3(6).

Khurana, H., Basney, J., Bakht, M., Freemon, M., Welch, V. and Butler, R., 2009, Palantir: a framework for collaborative incident response and investigation, In Proceedings of the 8th Symposium on Identity and Trust on the Internet (pp. 38-51), ACM.

Kijewski, P. and Pawliński, P., 2014, Proactive detection and automated exchange of network security incidents, Abgerufen am, 20.

Killam, L., 2013, Research terminology simplified: Paradigms, axiology, ontology, epistemology and methodology, Sudbury, Laura Killam.

Killcrece, G., Kossakowski, K.P., Ruefle, R. and Zajicek, M., 2003, State of the practice of computer security incident response teams (CSIRTs) (No. CMU/SEI-2003-TR-001). CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.

Kiltz, S., Lang, A. and Dittmann, J., 2007, Taxonomy for computer security incidents, In Cyber Warfare and Cyber Terrorism (pp. 421-418), IGI Global.

Kitcher P., 2001, Science, truth, and democracy, Oxford Studies in the Philosophy of Science). Oxford, UK: Oxford University Press.

Knapp, T.R., 1990, Treating ordinal scales as interval scales: an attempt to resolve the controversy, Nursing research, 39(2), pp.121-123.

Knight, R.F. and Pretty, D.J., 1997, The impact of catastrophes on shareholder value, Templeton College.

Kock, N., Gray, P., Hoving, R., Klein, H., Myers, M., and Rockart, J., 2002, IS Research Relevance Revisited: Subtle Accomplishment, Unfulfilled Promise, or Serial Hypocrisy, Communications of the AIS (8), Article 23.

Koens, T. and Poll, E., 2018, What Blockchain Alternative Do You Need?,. Data Privacy Management, Cryptocurrencies and Blockchain Technology (pp. 113-129), Springer, Cham.

Koivunen E., 2010, Why wasn't I notified: information security incident reporting demystified. In: 15th Nordic Conference in Secure IT Systems (NordSec 2010)

Kopp, E., Kaffenberger, L. and Jenkinson, N., 2017, Cyber Risk, Market Failures, and Financial Stability. International Monetary Fund.

Kostina, A., Miloslavskaya, N. and Tolstoy, A., 2009, Information security incident management process, In Proceedings of the 2nd international conference on Security of information and networks (pp. 93-97), ACM.

Kotulic AG, Clark JG., 2004, Why there aren't more information security research studies. Inf Manag; 41(5):597e607.

Kuhn, T.S., 2012, The structure of scientific revolutions, University of Chicago press.

Kulikova, O., Heil, R., van den Berg, J. and Pieters, W., 2012, Cyber Crisis Management: A decision-support framework for disclosing security incident information, In 2012 International conference on cyber security (pp. 103-112), IEEE.

Kurowski S, Frings S., 2011, Computational documentation of IT incidents as support for forensic operations, In: IT Security Incident Management and IT Forensics (IMF), 2011 Sixth International Conference on; pp. 37e47.

Lakatos, I., 1978, The Methodology of Scientific Research Programmes, J. Worral and G. Currie (Eds.), Cambridge, UK: Cambridge University Press.

Landwehr C, Bull A, McDermott J & Choi W, 1994, A Taxonomy of Computer Security Flaws, ACM Computing Surveys, Vol. 26, No. 3, September 1994, pp. 211-254.

Lau, F., 1999, Toward a framework for action research in information systems studies, Information Technology & People, 12(2), pp.148-176.

Lee C, 2017, A Study on Introducing Cyber Security Incident Reporting Regulations for Nuclear Facilities, CYBER 2017: The Second International Conference on Cyber-Technologies and Cyber-Systems

Lemieux V.L, 2016, Trusting records: is Blockchain technology the answer?, Records Management Journal, Vol. 26 Issue: 2, pp.110-139

Leszczyna, R. and Wrobel, M.R., 2014, Security information sharing for smart grids: Developing the right data model, In The 9th International Conference for Internet Technology and Secured Transactions (ICITST-2014) (pp. 163-169), IEEE.

Liang, G., Weller, S.R., Luo, F., Zhao, J. and Dong, Z.Y., 2018, Distributed blockchain-based data protection framework for modern power systems against cyber attacks, IEEE Transactions on Smart Grid.

Lin, I.C. and Liao, T.C., 2017, A Survey of Blockchain Security Issues and Challenges. IJ Network Security, 19(5), pp.653-659.

Lindqvist U and Jonsson E, 1997, How to Systematically Classify Computer Security Intrusions, Proceedings of the 1997 IEEE Symposium on Security and Privacy, IEEE Computer Society Press, Los Alamitos, CA, May, pp. 154-163.

Line, M.B., 2013, A case study: preparing for the smart grids-identifying current practice for information security incident management in the power industry, In 2013 Seventh international conference on IT security incident management and IT forensics (pp. 26-32), IEEE.

Line, M.B. and Albrechtsen, E., 2016, Examining the suitability of industrial safety management approaches for information security incident management, Information & Computer Security, 24(1), pp.20-37.

Lipton A., 2018, Blockchains and distributed ledgers in retrospective and perspective, The Journal of Risk Finance, Vol. 19 Issue: 1, pp.4-25.

Liu, H. and Tan, H.B.K., 2009, Covering code behavior on input validation in functional testing. Information and Software Technology, 51(2), pp.546-553.

Lu, Y., 2019, The Blockchain: State-of-the-Art and Research Challenges, Journal of Industrial Information Integration.

Luu L., D.-H. Chu, H. Olickel, P. Saxena, A. Hobor, 2016, Making smart contracts smarter, Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security, pp. 254–269.

Ma, Q., Schmidt, M. B., & Pearson, J. M., 2009, An integrated framework for information security management, Review of Business, 30(1), 58–69.

Ma, S., Hao, W., Dai, H.N., Cheng, S., Yi, R. and Wang, T., 2018, A Blockchain-based Risk and Information System Control Framework. In 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (pp. 106-113), IEEE.

Mackenzie, N. and Knipe, S., 2006, Research dilemmas: Paradigms, methods and methodology. Issues in educational research, 16(2), pp.193-205.

Majchrzak M & A. Gasser L, 2002, A Design Theory for Systems that Support Emergent Knowledge Processes, MIS Quartery 26/3 179-212.

Makedon, F., Ye, S., Steinberg, T., Zhao, Y., Ford, J., Xiao, Z. and Sudborough, B., 2003, A Security Incident Sharing and Classification System for Building Trust in Cross Media Enterprises, In Cross-Media Service Delivery (pp. 157-168), Springer, Boston, MA.

Makori, A.C. and Oenga, L., 2010, A Survey of Information Security Incident Reporting for Enhanced Digital Forensic Investigations, IJCIR, pp.19-31.

Mattila J, 2016, The Blockchain Phenomenon – The Disruptive Potential of Distributed Consensus Architectures, ETLA Working Papers 38, The Research Institute of the Finnish Economy

Mattioli R, Leguesse Y., 2018, Reference Incident Classification Taxonomy Task Force Update 53rd TF-CSIRT meeting – Hamburg, Germany, ENISA [online] Available at: https://www.first.org/resources/papers/hamburg2018/ReferenceTaxonomy-FIRST.pdf

March, S.T. and Smith, G.F., 1995, Design and natural science research on information technology, Decision support systems, 15(4), pp.251-266.

Markus, M.L., Majchrzak, A. and Gasser, L., 2002, A design theory for systems that support emergent knowledge processes, MIS quarterly, pp.179-212.

Marshall, J., 2009, The cyber scenario modeling and reporting tool (cybersmart), Cybersecurity Applications & Technology Conference for Homeland Security (pp. 305-309), IEEE.

Meek, G.E., Ozgur, C. and Dunning, K., 2007, Comparison of the t vs. Wilcoxon signed-rank test for Likert scale data and small samples, Journal of modern applied statistical methods, 6(1), p.10.

Meek, G.K., Roberts, C.B. and Gray, S.J., 1995, Factors influencing voluntary annual report disclosures by US, UK and continental European multinational corporations, Journal of international business studies, 26(3), pp.555-572.

Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G.M. and Savage, S., 2013, A fistful of bitcoins: characterizing payments among men with no names, In Proceedings of the 2013 conference on Internet measurement conference (pp. 127-140).

Menges, F. and Pernul, G., 2018, A comparative analysis of incident reporting formats, Computers & Security, 73, pp.87-101.

Mertens, D.M., 2005, Research methods in education and psychology: Integrating diversity with quantitative and qualitative approaches, 2nd ed., Thousand Oaks: Sage.

Mesa, A.P., Ardila, F. and Mármol, F.G., 2019, sSIEM-IoT: A blockchain-based and distributed SIEM for the Internet of Things.

Metzger, S., Hommel, W., and Reiser, H. 2011, Integrated Security Incident Management--Concepts and Real-World Experiences, IT Security Incident Management and IT Forensics (IMF), 2011 Sixth International Conference on: IEEE, pp. 107-121.

Microsoft, 2019, Azure Blockchain - Develop, test, and deploy secure blockchain apps [online] Available at: https://azure.microsoft.com/en-us/solutions/blockchain/

Miers, I., Garman, C., Green, M. and Rubin, A.D., 2013, Zerocoin: Anonymous distributed e-cash from bitcoin, In 2013 IEEE Symposium on Security and Privacy (pp. 397-411), IEEE.

Miloslavskaya, Natalia, and Alexander Tolstoy, 2019, New SIEM System for the Internet of Things, In World Conference on Information Systems and Technologies, pp. 317-327, Springer, Cham

Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W. and Qijun, C., 2017, A review on consensus algorithm of blockchain, In 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 2567-2572), IEEE.

Mitropoulos, S., Patsos, D. and Douligeris, C., 2006, On Incident Handling and Response: A state-of-the-art approach, Computers & Security, 25(5), pp.351-370.

Moreno, J., Serrano, M.A., Fernandez, E.B. and Fernández-Medina, E., 2020. Improving Incident Response in Big Data Ecosystems by Using Blockchain Technologies. Applied Sciences, 10(2), p.724.

Morkunas, V.J., Paschen, J. and Boon, E., 2019, How blockchain technologies impact your business model, Business Horizons.

Morris, N., 2018, Trade finance blockchain race is about to start, Ledger Insights [online] Available at: https://www.ledgerinsights.com/wetrade-trade-finance-blockchain-race/.

Moser, M., 2013, Anonymity of bitcoin transactions, In: Münster bitcoin conference.

Mottur, P.A. and Whittaker, N.R., 2018, Vizsafe: The Decentralized Crowdsourcing Safety Network, In 2018 IEEE International Smart Cities Conference (ISC2) (pp. 1-6), IEEE.

Mougayar W, 2016, The business blockchain: promise, practice, and application of the next Internet technology, Wiley.

Mtsweni, J., Shozi, N.A., Matenche, K., Mutemwa, M., Mkhonto, N. and Jansen van Vuuren, J., 2016, Development of a semantic-enabled cybersecurity threat intelligence sharing model.

Nabors E., 2017, Introduction to Mailgun email automation [online] Available at: https://support.rackspace.com/how-to/introduction-to-mailgun-email-automation/

Nakamoto S., 2008, Bitcoin: A peer-to-peer electronic cash system.

Narayanan, A., Bonneau, J., Felten, E., Miller, A. and Goldfeder, S., 2016, Bitcoin and cryptocurrency technologies.

Nesmith B., 2018, Forbes: CEOs: The Data Breach Is Your Fault [online] Available at: https://www.forbes.com/sites/forbestechcouncil/2018/06/26/ceos-the-data-breach-is-your-fault/#761a441758b0

Neumann P and Parker D, 1989, A Summary of Computer Misuse Techniques, Proceedings of the 12th National Computer Security Conference.

Newman C.A., 2018, The New York Times: When to Report a Cyberattack? For Companies, That's Still a Dilemma [online] Available at: https://www.nytimes.com/2018/03/05/business/dealbook/sec-cybersecurity-guidance.html

Nguyen, A., Gardner, L. and Sheridan, D., 2019, Towards Ontology-Based Design Science Research for Knowledge Accumulation and Evolution, In Proceedings of the 52nd Hawaii International Conference on System Sciences.

Nieswiadomy, R. M., 2011, Foundations of Nursing Research, 6th ed., Pearson

NIST Recommendations of the National Institute of Standards and Technology, 2012, Computer Security Incident Handling Guide, Special Publication 800-61, Revision 2

Nizamuddin, N., Salah, K., Azad, M.A., Arshad, J. and Rehman, M.H., 2019, Decentralized document version control using ethereum blockchain and IPFS, Computers & Electrical Engineering, 76, pp.183-197.

Noether, S. and Mackenzie, A., 2016, Ring confidential transactions, Ledger 1, pp.1-18.

Norman, G. 2010, Likert scales, levels of measurement and the “laws” of statistics, Advances in Health Sciences Education, 15(5), 625-632.

Nowruzi, M., Jazi, H.H., Dehghan, M., Shahmoradi, M., Hashemi, S.H. and Babaeizadeh, M., 2012, A comprehensive classification of incident handling information, In 6th International symposium on telecommunications (IST) (pp. 1071-1075), IEEE.

Noyes, C., 2016, Bitav: Fast anti-malware by distributed blockchain consensus and feedforward scanning, arXiv preprint arXiv:1601.01405.

Nunamaker J.F., Chen M., Purdin T., 1991, Systems Development in Information Systems Research, Journal of Management Information Systems 7/3 89-106.

OECD, 2015, Frascati Manual 2015: Guidelines for collecting and reporting data on research and experimental development, OECD Publishing.

Oracle, 2019, Oracle Blockchain Platform - Ready to Build Blockchain Platform [online] Available at: https://www.oracle.com/cloud/blockchain/

Osborne, T. T. 2001, Building an Incident Response Program To Suit Your Business, The SANS Institute [online] Available at: https://www.sans.org/reading-room/whitepapers/incident/paper/627

Owen, C., 1997, Design Research: Building the Knowledge Base. Journal of the Japanese Society for the Science of Design, 5 (2): 9-20.

O'Leary, Z., 2004, The essential guide to doing research, London: Sage.

PAC & Telefonica, 2015, Incident Response Management: How European Enterprises are Planning to Prepare for a Cyber Security Breach report. [online] Available at: https://www.elevenpaths.com/incident-response-management-2/index.html

Papas, N., R.M. O'Keefe, and P. Seltsikas, 2012, The action research vs design science debate: reflections from an intervention in eGovernment, European Journal of Information Systems 21(2), pp. 147–159

Patel, V.M., Chellappa, R., Chandra, D. and Barbello, B., 2016, Continuous user authentication on mobile devices: Recent progress and remaining challenges, IEEE Signal Processing Magazine, 33(4), pp.49-61.

Patraşcu, A. and Patriciu, V.V., 2013, Beyond digital forensics. A cloud computing perspective over incident response and reporting, In 2013 IEEE 8th International Symposium on Applied Computational Intelligence and Informatics (SACI) (pp. 455-460), IEEE.

Peck, M.E., 2017, Blockchain world-Do you need a blockchain? This chart will tell you if the technology can solve your problem, IEEE Spectrum, 54(10), pp.38-60.

Peffers, Tuure Tuunanen, Marcus Rothenberger, and Samir Chatterjee., 2007, A Design Science Research Methodology for Information Systems Research, J. Manage. Inf. Syst. 24, 3, 45-77, DOI=http://dx.doi.org/10.2753/MIS0742-1222240302

Peirce, C.S., 1931, Collected papers of charles sanders peirce, Harvard University Press.

Pilkington, M., 2016, 11 Blockchain technology: principles and applications, Research handbook on digital transformations, 225.

Pipkin, D.L., 2000, Information Security Protecting the Global Enterprise, Hewlett-Packard Company.

Pirounias, S., Mermigas, D. and Patsakis, C., 2014, The relation between information security events and firm market value, empirical evidence on recent disclosures: An extension of the GLZ study, Journal of Information Security and Applications, 19(4-5), pp.257-271.

Pontiveros, B.B.F., Norvill, R. and State, R., 2018, Recycling smart contracts: Compression of the ethereum blockchain, In 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS) (pp. 1-5), IEEE.

Prat, N., Comyn-Wattiau, I. and Akoka, J., 2014, Artifact Evaluation in Information Systems Design-Science Research-a Holistic View, In PACIS (p. 23).

Pratt, John W., 1959, Remarks on Zeros and Ties in the Wilcoxon Signed Rank Procedures, Journal of the American Statistical Association, 54, 655-67

Pries-Heje, J., Baskerville, R. and Venable, J.R., 2008, Strategies for Design Science Research Evaluation, In ECIS (pp. 255-266).

Prpić, J., 2017, Unpacking Blockchains, arXiv preprint arXiv:1703.06117.

Purao, S., 2013, Truth or dare: The ontology question in design science research, Journal of Database Management (JDM), 24(3), pp.51-66.

Putter, J, 1955, The Treatment of Ties in Some Non-parametric Tests, The Annals of Mathematical Statistics, 26, 368-86.

Putz, B., Menges, F. and Pernul, G., 2019. A secure and auditable logging infrastructure based on a permissioned blockchain. Computers & Security, 87, p.101602.

Ramesh, V., Glass, R.L. and Vessey, I., 2004, Research in computer science: an empirical study, Journal of systems and software, 70(1-2), pp.165-176.

Rapoport, R.N., 1970, Three dilemmas in action research, Human Relations, 23, 499-513.

Raval, S., 2016, Decentralized applications: harnessing Bitcoin's blockchain technology, O'Reilly Media, Inc.

Rebollo, O., Mellado, D., Fernández-Medina, E. and Mouratidis, H., 2015. Empirical evaluation of a cloud computing information security governance framework. Information and Software Technology, 58, pp.44-57.

Reed-Mohn, A., 2007, Incident Reporting Systems, Faculty of Computer Science and Media Technology, Gjøvik University College

Reid, F. and Harrigan, M., 2013, An analysis of anonymity in the bitcoin system, In Security and privacy in social networks (pp. 197-223), Springer, New York, NY.

Reijers, W., O'Brolcháin, F., & Haynes, P., 2016, Governance in Blockchain Technologies & Social Contract Theories, Ledger 1, 134-151.

Reynolds B and Seeger MW, 2005, Crisis and Emergency Risk Communication as an Integrative Model, Journal of Health Communication, vol. 10, pp. 43-55.

Riesco, R., Larriva-Novo, X. and Villagra, V.A., 2020. Cybersecurity threat intelligence knowledge exchange based on blockchain. Telecommunication Systems, 73(2), pp.259-288.

Rodrigues, B., Bocek, T., Lareida, A., Hausheer, D., Rafati, S. and Stiller, B., 2017, A blockchain-based architecture for collaborative DDoS mitigation with smart contracts, In IFIP International Conference on Autonomous Infrastructure, Management and Security (pp. 16-29), Springer, Cham.

Roll-Hansen, N., 2009, Why the distinction between basic (theoretical) and applied (practical) research is important in the politics of science, London School of Economics and Political Science, Contingency and Dissent in Science Project.

Ron, D. and Shamir, A., 2013, Quantitative analysis of the full bitcoin transaction graph, In International Conference on Financial Cryptography and Data Security (pp. 6-24), Springer, Berlin, Heidelberg.

Rosati, P., Cummins, M., Deeney, P., Gogolin, F., van der Werff, L. and Lynn, T., 2017, The effect of data breach announcements beyond the stock price: Empirical evidence on market activity, International Review of Financial Analysis, 49, pp.146-154.

Rosati, P., Deeney, P., Cummins, M., Van der Werff, L. and Lynn, T., 2019, Social media and stock price reaction to data breach announcements: Evidence from US listed companies, Research in International Business and Finance, 47, pp.458-469.

Rossi, M., and Sein, M.K., 2003, Design research workshop: a proactive research approach, 26th Information Systems Research Seminar in Scandinavia, Haikko Finland: The IRIS Association

Ruefle, R., Dorofee, A., Mundie, D., Householder, A.D., Murray, M. and Perl, S.J., 2014, Computer security incident response team development and evolution, IEEE Security & Privacy, 12(5), pp.16-26.

Ryba, M., Poniewierski, A., Sulwiński, J. and Górnisiewicz, M., 2009, The methodology for managing the abuse of IT systems, EDPACS The EDP Audit, Control, and Security Newsletter, 40(5), pp.1-13.

Sajana, P., Sindhu, M. and Sethumadhavan, M., 2018, On Blockchain Application: Hyperledger Fabric and Ethereum, International Journal of Pure and Applied Mathematics, 118(18), pp.2965-2970.

Samaniego, M. and Deters, R., 2016, Blockchain as a Service for IoT, In 2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (pp. 433-436), IEEE.

Sankar, L.S., Sindhu, M. and Sethumadhavan, M., 2017, Survey of consensus protocols on blockchain applications, In 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS) (pp. 1-5), IEEE.

SANS Institute, 2011, Incident Handler's Handbook [online] Available at: https://www.sans.org/reading-room/whitepapers/incident/incident-handlers-handbook-33901

Sasson, E.B., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E. and Virza, M., 2014, Zerocash: Decentralized anonymous payments from bitcoin, In 2014 IEEE Symposium on Security and Privacy (pp. 459-474), IEEE.

Saunders, M., Lewis, P., & Thornhill, A., 2007, Research Methods for Business Students (5th ed.), Pearson Education.

Schneier, B., 2011, Secrets and Lies: Digital Security in a Networked World, John Wiley & Sons.

Schultz, E.E., 2007, Computer forensics challenges in responding to incidents in real-life settings, Computer Fraud & Security, 2007(12), pp.12-16.

Schuster, B., 2018, What is the third generation of blockchain technology? HackerNoon. [online] Available at https://hackernoon.com/what-is-the-third-generation-ofblockchain-technology-36a46af5ccbc

Schwartz D, N. Youngs, A. Britto, 2014, The ripple protocol consensus algorithm, Ripple Labs Inc White Paper, vol. 5.

Schwartz, P.M. and Janger, E.J., 2006, Notification of data security breaches, Mich. L. Rev., 105, p.913.

Seibold, S. and Samman, G., 2016. Consensus: Immutable agreement for the Internet of value. KPMG [online] Available at: https://assets.kpmg.com/content/dam/kpmg/pdf/2016/06/kpmgblockchain-consensus-mechanism.pdf.

SentinelOne, 2016, Ransomware Research Data Report [online] Available at: https://go.sentinelone.com/rs/327-MNM-087/images/Data%20Summary%20-%20English.pdf

Settanni, G., Skopik, F., Shovgenya, Y., Fiedler, R., Carolan, M., Conroy, D., Boettinger, K., Gall, M., Brost, G., Ponchel, C. and Haustein, M., 2017, A collaborative cyber incident management system for European interconnected critical infrastructures, Journal of Information Security and Applications, 34, pp.166-182.

Shackelford S & Myers S, 2016, Block-by-Block: Leveraging the Power of Blockchain Technology to Build Trust and Promote Cyber Peace, Yale Journal of Law and Technology, Kelley School of Business Research Pa per No. 16-85.

Shedden, P., Ahmad, A. and Ruighaver, A.B., 2010, Organisational learning and incident response: promoting effective learning through the incident response process.

ShenTu, Q. and Yu, J., 2015, Transaction remote release (TRR): A new anonymization technology for bitcoin, arXiv preprint arXiv:1509.06160.

Shermin, V., 2017, Disrupting governance with blockchains and smart contracts, Strategic Change, 26(5), pp.499-509.

Shetty, S., Kamhoua, C.A. and Njilla, L.., 2019, Blockchain for Distributed Systems Security, Wiley-IEEE Computer Society Press.

Simon, H., 1996, The Sciences of Artificial, 3rd Edition, MIT Press, Cambridge, MA

Somekh, B. and Lewin, C., 2005, Research methods in social sciences, London: Sage.

Sonnenberg, C. and Vom Brocke, J., 2012, Evaluations in the science of the artificial–reconsidering the build-evaluate pattern in design science research, In International Conference on Design Science Research in Information Systems (pp. 381-397), Springer, Berlin, Heidelberg.

Sorrels, D., Grimaila, M.R., Fortson, L.W., and Mills, R.F., 2008, An Architecture for Cyber Incident Mission Impact Assessment (CIMIA), Proceedings of the 2008 International Conference on Information Warfare and Security (ICIW 2008), Peter Kiewit Institute, University of Nebraska Omaha.

Spanos, G. and Angelis, L., 2016, The impact of information security events to the stock market: A systematic literature review, Computers & Security, 58, pp.216-229.

Spruit, M. E. M. and Gerhardt, W., 1997, Information Security and The Significance of Organization, First ACM Workshop on Education in InfoSec, Monterey

Spruit, M. E. M., 1998, Competing Against Human Failing, The IFIP TC11 14th International Conference on Information Security (IFIP/SEC'98), Vienna/Budapest

Status Network, 2017, The Status Network whitepaper: A strategy towards mass adoption of Ethereum [online] Available at: https://status.im/whitepaper.pdf

Stephenson P., 2004, Managing digital incidents – a background, Computer Fraud & Security 2004 (12), 2004, pp. 17–19.

Stikvoort D., 2015, Incident Classification/Incident Taxonomy according to eCSIRT.net – International version, S-CURE bv, PRESECURE GmbH and SURFnet [online] Available at: https://www.trusted-introducer.org/Incident-Classification-Taxonomy.pdf

Sukhwani, H., Martínez, J.M., Chang, X., Trivedi, K.S. and Rindos, A., 2017, Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric), In 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS) (pp. 253-255), IEEE.

Sultan Karim, Umar Ruhi, Rubina Lakhani, 2018, Conceptualizing Blockchains: Characteristics & Applications, 11th IADIS International Conference Information Systems 2018.

Susman, G. and R. Evered, 1978, An Assessment of The Scientific Merits of Action Research, Administrative Science Quarterly, (23) 4, pp. 582-603.

Sveen F, Jose M. Sarriegi, Eliot Rich, Jose J. Gonzalez, 2007, Toward viable information security reporting systems, Information Management & Computer Security, Vol. 15 Issue: 5, pp.408-419

Sveen F, Sarriegi, J.M. and Gonzalez, J.J., 2009, The role of incident reporting in reducing information security risk, In Twenty Seventh International Conference of the System Dynamics Society, The System Dynamics Society.

Swan, M., 2015, Blockchain: Blueprint for a new economy, O'Reilly Media, Inc.

Symantec, 2016, Internet Security Threat Report, Volume 21 [online] Available at: https://www.symantec.com/content/dam/symantec/docs/reports/istr-21-2016-en.pdf

Takeda, H., Veerkamp, P., Tomiyama, T., and Yoshikawam, H., 1990, Modelling Design Processes, AI Magazine, 37-48.

Tapscott, D. and Tapscott, A., 2017, How blockchain will change organizations, MIT Sloan Management Review, 58(2), p.10.

Tarala J, Tarala K., 2015, Open Threat Taxonomy version 1.1, Enclave Security, Florida [online] Available at: https://www.auditscripts.com/resources/open_threat_taxonomy_v1.1a.pdf

Tasca, P. and Tessone, C.J., 2017, Taxonomy of blockchain technologies, Principles of identification and classification, arXiv preprint arXiv:1708.04872.

Taylor, P.J., Dargahi, T., Dehghantanha, A., Parizi, R.M. and Choo, K.K.R., 2019, A systematic literature review of blockchain cyber security, Digital Communications and Networks.

Threatvine, 2018, Using Threatvine to meet your NISD obligations [online] Available at: https://www.surevine.com/wp-content/uploads/2018/03/threatvine_nisd_final_2018_03_26-1.pdf

Tondel, I.A., Line, M.B. and Jaatun, M.G., 2014, Information security incident management: current practice as reported in the literature, Computers & Security, Vol. 45, pp. 42-57.

Umeh, J., 2016, Blockchain double bubble or double trouble?, Itnow, 58(1), pp.58-61.

Underwood, S., 2016, Blockchain beyond bitcoin. Communications of the ACM, 59(11), pp.15-17.

Usher, R., 2002, A diversity of doctorates: fitness for the knowledge economy?.,Higher Education Research & Development, 21(2), pp.143-153.

Usmani, K., Mohapatra, A.K. and Prakash, N., 2013, An improved framework for incident handling. Information Security Journal: A Global Perspective, 22(1), pp.1-9.

Vaishnavi, V., Kuechler, W., and Petter, S., 2004/19, Design Science Research in Information Systems" January 20, 2004 (created in 2004 and updated until 2015 by Vaishnavi, V. and Kuechler, W.); last updated (by Vaishnavi, V. and Petter, S.): 2019. URL: Http://www.desrist.org/design-research-in-information-systems/.

Vakilinia, I., Tosh, D.K. and Sengupta, S., 2017, Attribute based sharing in cybersecurity information exchange framework, In 2017 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS) (pp. 1-6), IEEE.

Valenta, M. and Sandner, P., 2017, Comparison of ethereum, hyperledger fabric and corda, Frankfurt School, Blockchain Center.

Vázquez, D.F., Acosta, O.P., Spirito, C., Brown, S. and Reid, E., 2012, Conceptual framework for cyber defense information sharing within trust relationships, In 2012 4th International Conference on Cyber Conflict (CYCON 2012) (pp. 1-17), IEEE.

Venable, J.R., 2009. Identifying and addressing stakeholder interests in design science research: An analysis using critical systems heuristics, In Information Systems–Creativity and Innovation in Small and Medium-Sized Enterprises (pp. 93-112), Springer, Berlin, Heidelberg.

Venable, J., Pries-Heje, J. and Baskerville, R., 2012, A comprehensive framework for evaluation in design science research, In International Conference on Design Science Research in Information Systems (pp. 423-438), Springer, Berlin, Heidelberg.

Von Solms, B., & von Solms, R., 2005, From information security to business security?, Computers & Security, 24(4), 271–273.

Vranken H, 2017, Sustainability of bitcoin and blockchains, Current Opinion in Environmental Sustainability Volume 28, Pages 1-9.

Wack, J. P., 1991, Establishing a Computer Security Incident Response Capability, US National Institute of Standards and Technology, Gaithersburg, Md, NIST Special Publication 800-3.

Wagner C., A. Dulaunoy G., Wagener and Iklody A., 2016, MISP: The Design and Implementation of a Collaborative Threat Intelligence Sharing Platform, Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security (WISCS'16), pp 49–56.

Walls, J., Widmeyer, G., and El Sawy, O., 1992, Building an Information System Design Theory for Vigilant EIS. Information Systems Research, 3, 1, 36-59.

Walls, J., Widmeyer, G., and El Sawy, O., 2004, Assessing Information System Design Theory in Perspective: How Useful was our 1992 Initial Rendition?, Journal of Information Technology Theory & Application (JITTA), 6, 2, 43-58.

Wang, H., Chen, K. and Xu, D., 2016, A maturity model for blockchain adoption, Financial Innovation, 2(1), p.12.

Warren, W. and Bandeali, A., 2017, 0x: An open protocol for decentralized exchange on the Ethereum blockchain, URl: https://github. com/0xProject/whitepaper.

Werlinger R, Hawkey K, Muldner K, Jaferian P, Beznosov K., 2008, The challenges of using an intrusion detection system: is it worth the effort?, In: Proceedings of the 4th symposium on Usable privacy and security (SOUPS '08), New York, NY, USA: ACM;. pp. 107-118.

Werlinger, R., Muldner, K., Hawkey, K., and Beznosov, K. 2010, Preparation, Detection, and Analysis: The Diagnostic Work of It Security Incident Response, Info. Management & Comp. Security (18:1), pp. 26-42.

Whitman, M. E. & H. J. Mattord, 2005, Principles of Information Security, Thomson Course Technology.

Wiik, J., Gonzales, J.J., & Kossakowski, K-P., 2005, Limits to Effectiveness in Computer Security Incident Response Teams, Twenty-Third International Conference of the System Dynamics Society, The System Dynamics Society, Boston, MA.

Wöhrer, M. and Zdun, U., 2018, Design patterns for smart contracts in the ethereum ecosystem, In 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (pp. 1513-1520), IEEE.

Wolff, J., 2014, Models for cybersecurity incident information sharing and reporting policies, TPRC.

Wood, G., 2014, Ethereum: A secure decentralised generalised transaction ledger, Ethereum project yellow paper, 151(2014), pp.1-32.

Wood, G., 2015, Whisper communications protocol, URI: https://github.com/ethereum/wiki/wiki/Whisper

Wood-Harper, T., 1985, Research Methods in Information Systems: Using Action Research, in E. Mumford et al., (eds.) Research Methods in Information Systems, Amsterdam: North-Holland, pp. 169-191

Wüst, K. and Gervais, A., 2018, Do you need a Blockchain?, In 2018 Crypto Valley Conference on Blockchain Technology (CVCBT) (pp. 45-54), IEEE.

Xu X, C. Pautasso, L. Zhu, V. Gramoli, A. Ponomarev, A. B. Tran, and S. Chen, 2016, The blockchain as a software connector, 13th Working IEEE/IFIP Conference on Software Architecture (WICSA), pp. 182-191, IEEE.

Yaga, D., Mell, P., Roby, N. and Scarfone, K., 2018, Blockchain technology overview (No. NIST Internal or Interagency Report (NISTIR) 8202 (Draft)), National Institute of Standards and Technology.

Yang, C., Chen, X. and Xiang, Y., 2018, Blockchain-based publicly verifiable data deletion scheme for cloud storage, Journal of Network and Computer Applications, 103, pp.185-193.

Yayla, A.A. and Hu, Q., 2011, The impact of information security events on the stock value of firms: The effect of contingency factors, Journal of Information Technology, 26(1), pp.60-77.

Yli-Huumo J, Ko D, Choi S, Park S, Smolander K, 2016, Where Is Current Research on Blockchain Technology? —A Systematic Review, PLoS ONE 11(10): e0163477, doi:10.1371/journal. pone.0163477

Zafar, H., Ko, M., and Osei-Bryson, K.-M. 2012, Financial Impact of Information Security Breaches on Breached Firms and Their Non-Breached Competitors, Information Resources Management Journal (25:1), pp. 21-37.

Zhao, J.L., Fan, S., Yan, J., 2016, Overview of business innovations and research opportunities in blockchain and introduction to the special issue, Financial Innovation 2 (1), 28.

Zheng, Z., Xie, S., Dai, H.N. and Wang, H., 2016, Blockchain challenges and opportunities: A survey, Work Pap.

Zheng, Z., Xie, S., Dai, H., Chen, X. and Wang, H., 2017, An overview of blockchain technology: Architecture, consensus, and future trends, In 2017 IEEE International Congress on Big Data (BigData Congress) (pp. 557-564), IEEE.

Zhu, B., Joseph, A. and Sastry, S., 2011, A taxonomy of cyber-attacks on SCADA systems, International conference on internet of things and 4th international conference on cyber, physical and social computing (pp. 380-388), IEEE.

Zyskind, G. and Nathan, O., 2015, Decentralizing privacy: Using blockchain to protect personal data, In 2015 IEEE Security and Privacy Workshops (pp. 180-184), IEEE.

APPENDICES

APPENDIX A – RISK ASSESSMENT

The following table describes the risk assessment conducted as part of this research project:

Table values		
a) Likelihood of Risk	**b) Impact**	**c) Risk Rating (c = a x b)**
1 = Low (Unlikely)	1 = Minor	1-2 = Low
2 = Moderate (Likely)	2 = Considerable	3-4 = Medium
3 = High (Very likely)	3 = Major	6-9 = High

Risk assessment						
Risk element	**Description**	**Likeli-hood (a)**	**Impact (b)**	**Risk rating (c = a x b)**	**Mitigating actions**	**Final risk rating**
Health & safety concerns	Electric devices malfunction leading to accidents (e.g. fire).	1	2	2	Researcher's equipment and work environment follow all standard safety precautions.	**1**
UEL ethical guidelines violation	Research project breaches UEL ethical research guidelines.	1	3	3	Ethical approval received from relevant committee prior to commencing the demonstration & evaluation activities. Ethical guidelines were followed throughout the project's course.	**1**
Loss of data	Loss of research data due to equipment failure or accident.	1	3	3	Research data was regularly backed-up in cloud provider.	**1**
Project failure due to limited funds	The project could fail if funds were not available for necessary project purchases (e.g. Microsoft Azure, Office 365).	1	3	3	Funding was secured before the research project's initiation	**1**

Project failure due to hardware /software failures	Hardware and software necessary for conducting this. project could unexpectedly fail.	1	2	2	Provisions were taken in order to have readily available backup equipment in case of hardware/software failure.	**1**
Project failure due to lack of supervisi on	Project could fail due to unproper supervision.	1	3	3	Supervision meetings were carried out regularly and draft work was submitted for review. Annual monitoring review ensured adequate work progression.	**1**
Project failure due to time mismana gement	Researcher is a full-time working professional. If work/research/per sonal life balance is unproperly managed, implications could be devastating.	2	3	6	Adequate pre-planning and proper time management for all researcher's activities. Strict deadlines were set and followed.	**2**

APPENDIX B – DAPP SMART CONTRACT & TEST CODE

The code of the main smart contract used for IRDA is presented below:

```
pragma solidity ^0.5.0;
/**
 * @title ReportsStorage
 * @dev Core smart contract of IRDA platform, which stores all reports
 * also returns report data by ID
 */
contract ReportsStorage {
   // Current count of submitted reports
   uint256 private _index;
   // Struct data of report
   struct Report {
      string ttl;
      uint256 dtsubmit;
      string reportJSON;
   }
   // Mapping of reports by id
   mapping(uint256 => Report) private _reports;
   // Event should be triggered when a new report will be submitted
   event EvtReport(uint256 _reportID);
   // Empty fallback method
   function () external payable {}
   /**
    * @dev Submitting report with data will store it on contract storage
    * @param _ttl Title of report, should be text-string
    * @param _reportJSON Data of report, should be JSON-string
    */
   function submitReport(string calldata _ttl, string calldata _reportJSON) external {
      _index++;
      _reports[_index] = Report(_ttl, block.timestamp, _reportJSON);
      emit EvtReport(_index);
   }
   /**
    * @dev Returns the report data by id
```

```
     * @return data of report (title, creation timestamp and JSON)
     */
    function getReport(uint256 reportID) external view returns (
        string memory ttl,
        uint256 dtsubmit,
        string memory reportJSON
    ) {
        Report memory report = _reports[reportID];
        return (
            report.ttl,
            report.dtsubmit,
            report.reportJSON
        );
    }
    /**
     * @dev Returns the count of submitted reports
     * @return the current count of submitted reports
     */
    function getReportsCount() external view returns (uint256 ret) {
        return _index;
    }
}
```

The JavaScript code used for executing the smart contract tests is presented below:

```
const assert = require('assert')
const fetch = require('node-fetch')
const { time } = require('@openzeppelin/test-helpers')
const ReportsStorage = artifacts.require('./ReportsStorage.sol')

contract('ReportsStorage', (accounts) => {
    let contractInstance
    const user1 = accounts[0]
    const user2 = accounts[1]

 const generateRandomData = async () => (await fetch('https://randomuser.me/api', { method: 
'GET' })).text()
```

```
const demoTitle = "Report title 1"
const demoTitle2 = "Report title 2"

let demoJson
let demoJson2

before(async () => {
  contractInstance = await ReportsStorage.new()

  demoJson = await generateRandomData()
  demoJson2 = await generateRandomData()
})
context('◯ Empty state checks', async () => {
  it("initial reports count should be zero", async () => {
    const count = await contractInstance.getReportsCount()
    assert(count, '0', "report count issue")
  })
  it("get empty report data from contract", async () => {
    const data = await contractInstance.getReport('1')

    const title = data['0']
    const timestamp = data['1']
    const json = data['2']

    assert(typeof title === 'string' && title.length === 0, true, "report title issue")
    assert(timestamp, '0', "report timestamp issue")
    assert(typeof json === 'string' && json.length === 0, true, "report json issue")
  })
})
context('◯ Submit report', async () => {
  it("should submit report", async () => {
    await contractInstance.submitReport(demoTitle, demoJson, { from: user1 })
  })
  it("reports count should be increased after submit", async () => {
    const count = await contractInstance.getReportsCount()
    assert(count, '1', "reporst count issue")
  })
```

```
    it("check report data from contract", async () => {
      const data = await contractInstance.getReport('1')
      const title = data['0']
      const timestamp = data['1']
      const json = data['2']
      const now = await time.latest()

      assert(title, demoTitle, "report title issue")
      assert(timestamp, now, "report timestamp issue")
      assert(json, demoJson, "report json issue")
    })
  })
  context('◯ Submit 2nd record', async () => {
    it("should submit report", async () => {
      await contractInstance.submitReport(demoTitle2, demoJson2, { from: user2 })
    })
    it("reports count should be increased after second report", async () => {
      const count = await contractInstance.getReportsCount()
      assert(count, '2', "reporst count issue")
    })
    it("check 2nd report data from contract", async () => {
      const data = await contractInstance.getReport('1')
      const title = data['ttl']
      const timestamp = data['dtsubmit']
      const json = data['reportJSON']
      const now = await time.latest()

      assert(title, demoTitle2, "report title issue")
      assert(timestamp, now, "report timestamp issue")
      assert(json, demoJson2, "report json issue")
    })
  })
})
```

APPENDIX C – PARTICIPANTS RECRUITMENT E-MAIL

Dear Sir/Madam,

As part of my doctoral research project, I have very recently developed an innovative platform for the reporting of information security incidents, amongst organizations. If your organization is currently utilizing (or has utilized, at some point, in the past) an information security incident reporting platform, then it would be eligible to participate in the functionality testing and evaluation of this new software. More specifically, you would be requested to complete a first questionnaire, evaluating your current (or previously used) incident reporting platform and then execute five (5) simple test cases on the developed system. You would then have to complete a second questionnaire, evaluating your experience with the newly developed system.

Participating organizations can be of any size and operate in any business sector. As mentioned above, the only major eligibility prerequisite is that participating organizations should currently be using (or have used, at some point, in the past) an existing incident reporting platform, either commercially available or open source/free.

Please declare your interest by replying to u1445442@uel.ac.uk stating the name and business sector of your organization, as well as the name of the incident reporting platform you are currently using (or have used).

Declaration of interest deadline: 22/11/2019, 20.00 hrs

Best regards,

Alexis Michail

Doctoral researcher

University of East London

APPENDIX D – ETHICAL APPROVAL

University of East London

Dear Alexis

Application ID: ETH1920-0040

Project title: Tackling the challenges of Information Security Incident Reporting: A decentralized approach

Lead researcher: Mr Alexis Michail

Your application to Research, Research Degrees and Ethics Sub-Committee meeting was considered on the 15th of November 2019.

The decision is: **Approved**

The Committee's response is based on the protocol described in the application form and supporting documentation.

Your project has received ethical approval for 2 years from the approval date.

If you have any questions regarding this application please contact your supervisor or the secretary for the Research, Research Degrees and Ethics Sub-Committee meeting.

Approval has been given for the submitted application only and the research must be conducted accordingly.

Should you wish to make any changes in connection with this research project you must complete 'An application for approval of an amendment to an existing application'.

The approval of the proposed research applies to the following research site.

Research site: Nicosia, Cyprus

Principal Investigator / Local Collaborator: Mr Alexis Michail

Approval is given on the understanding that the UEL Code of Practice for Research and the Code of Practice for Research Ethics is adhered to.□□

Any adverse events or reactions that occur in connection with this research project should be reported using the University's form for Reporting an Adverse/Serious Adverse Event/Reaction.

The University will periodically audit a random sample of approved applications for ethical approval, to ensure that the research projects are conducted in compliance with the consent given by the Research Ethics Committee and to the highest standards of rigour and integrity.

Please note, it is your responsibility to retain this letter for your records.

With the Committee's best wishes for the success of the project

Yours sincerely

Fernanda Silva

Research, Research Degrees and Ethics Sub-Committee

APPENDIX E – INSTRUCTIONS TO PARTICIPANTS FOR DEMONSTRATION ACTIVITIES

Demonstration activities - Test instructions

Please remember to complete Questionnaire A – "Questionnaire for evaluating features of current (or previously utilized) incident reporting platform"* ***before* *initiating the following activities and questionnaire B – "Questionnaire for evaluating features of newly developed incident reporting platform"* ***after*** *you have concluded the activities.*

Thank you once more for accepting to test and evaluate the functionality of IRDA, the Incident Reporting DApp!

For performing your assigned test cases, please follow the steps below:

1. Download and install a Web3 enabled browser or install the Metamask extension (recommended) to your current Chrome/Firefox/Opera browser. The following steps apply to users who have installed the Metamask extension.

2. Create an account on Metamask – remember to securely save your password and Seed Phrase.

3. Provide a copy of your public key to the researcher and wait for the researcher's confirmation in order to proceed with registration.

4. After you have received the researcher's confirmation, change your Metamask's network connection to "Custom RPC" and input the following address as RPC URL: https://master1.blockchain.azure.com:3200/-iTfL2lI0UAMam3QhMILMsG5

5. Save the new network.

6. Navigate to https://alexis-michael.eu/reporting/

7. A Metamask pop-up window will now appear in your browser - Login to your Metamask account (if you are not already logged-in) and allow IRDA to connect to your account.

8. Click on the "Sign up" tab and complete the required information (E-mail, phone number and password). Please select a phone number you currently have access to, since the one-time-password will be sent to this number.

9. Click on "Get started" button – check your mobile phone for an SMS message containing a six-digit number. Input this number to the designated field and continue.

10. IRDA's homepage should now be visible on your screen. Please log-out of your account, and follow the below instructions for executing the test cases:

The following test cases should be executed in sequential order:

Test case: UTC01 – User Login

1. Login to your Metamask account (connected to the IRDA network).

2. Navigate to https://alexis-michael.eu/reporting/

3. Allow IRDA to connect to your account

3. Fill-in your e-mail account and password and click on “login”.

4. A pop-up window will appear stating that the verification SMS has been sent. Click the “OK” button to close this window.

5. Check your registered mobile phone for an SMS message containing a six-digit number. Input this number to the designated field and click “login”.

6. IRDA’s homepage should now be visible on your screen.

Test case: UTC02 – User submit incident

1. While on IRDA’s homepage, click on the “Submit incident” button.

2. Fill in all available fields (at least the minimum required fields – indicated with a red asterisk) of the form with details of a mock incident.

3. Click “Preview & submit” button.

4. Review content of form and if satisfied with content click “submit” button, otherwise click “edit form” button.

5. A Metamask pop-up window will now appear asking you to sign your transaction. Proceed to signing your transaction.

6. After a moment or so, the transaction ID should appear on your screen.

Test case: UTC03 – User view incident

1. While on IRDA’s homepage, click on the “View incidents” button.

2. The incident submitted through the previous test case (UTC02) should be evident in the relevant table and located at the top of the list.

3. Click on any field of the specific row of the incident.

4. Confirm the data displayed is identical to the data submitted as part of UTC02 execution.

Test case: UTC04 – User ask for help

1. While on IRDA's homepage, click on the "Ask for help" button.

2. Fill-in your e-mail address and a message containing the text "Please don't leave me!" – you can also include your name in the relevant field (optional).

3. Click the "Send" button.

4. Wait for admin confirmation regarding message receival.

Test case: UTC05 – User chat

1. While on platform's homepage, click on the "Live chat" button.

2. Notify researcher to join the chat session.

3. After the researcher has joined the chat and sent an acknowledgment message, type a message with content: "Hello fellow anonymous!"

4. Click on the "submit" button.

5. The researcher should reply with a second acknowledgment message.

6. Messages sent and received should be visible on your screen.

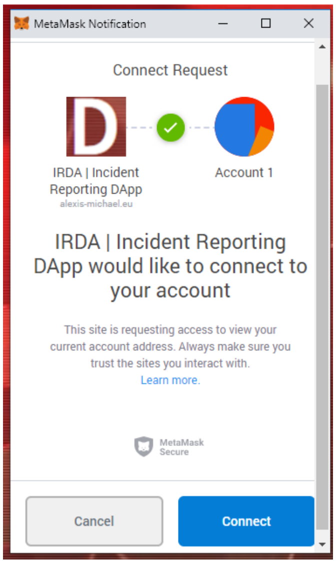
Please report the results (success/failure) of the above test cases to the researcher, at your earliest convenience, and in any case on or before 05/12/2019. In case of a failed test case please contact the researcher immediately and document your exact actions leading to this outcome, as well as any error messages (if any) displayed to you.

Thank you for your time and effort!

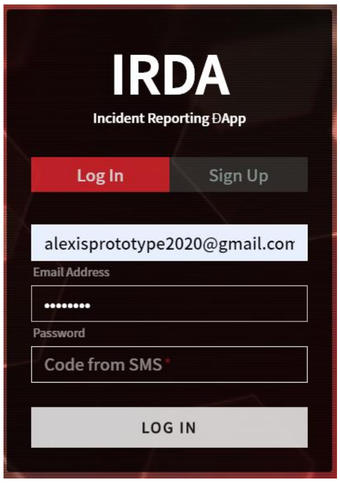
APPENDIX F – ILLUSTRATIVE EXAMPLE OF USER PERFORMING TEST CASES

The following example illustrates the flow of actions for logging-in to the platform, submitting an incident, viewing that incident in a list and tracing that incident through Epirus explorer:

1. Login to your whitelisted Metamask account. Navigate to https://alexis-michael.eu/reporting.



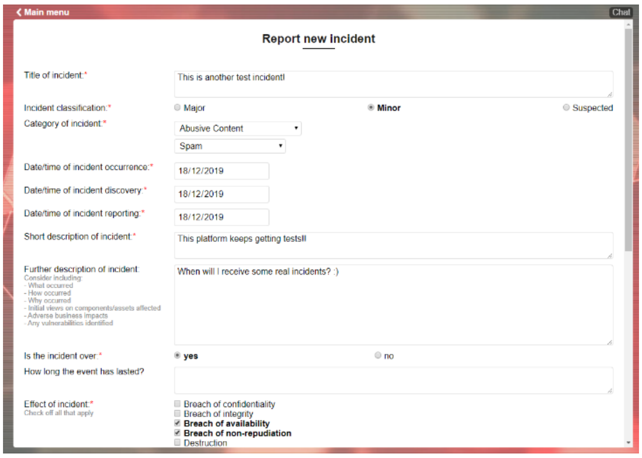
2. When prompted, allow IRDA to connect to your Metamask account.



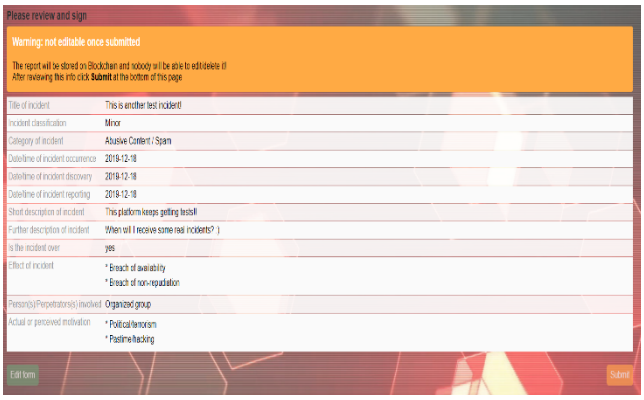
3. Type your e-mail address and password and click login. When the SMS containing the OTP arrives, enter the code in the relevant field and click login once more

4. While on the DApp's homepage, select the "Submit incident button"

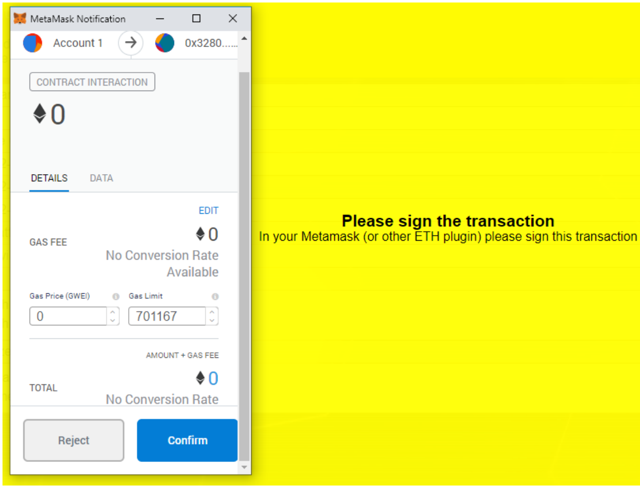
5. Complete the report form with details of a mock incident.



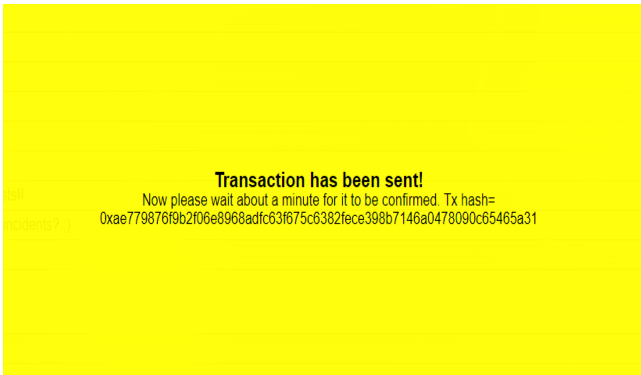
6. Confirm the incident's details before final submission.



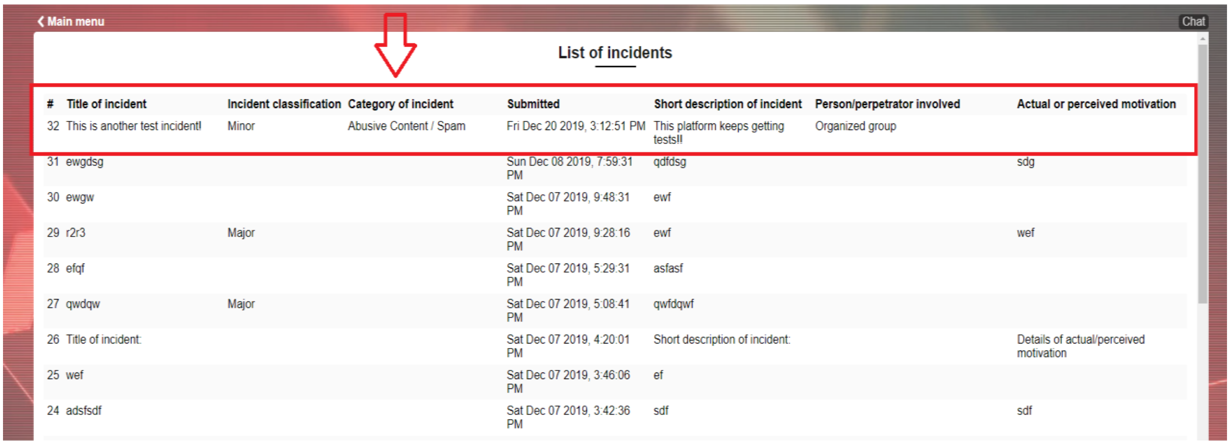
7. Sign the transaction with Metamask



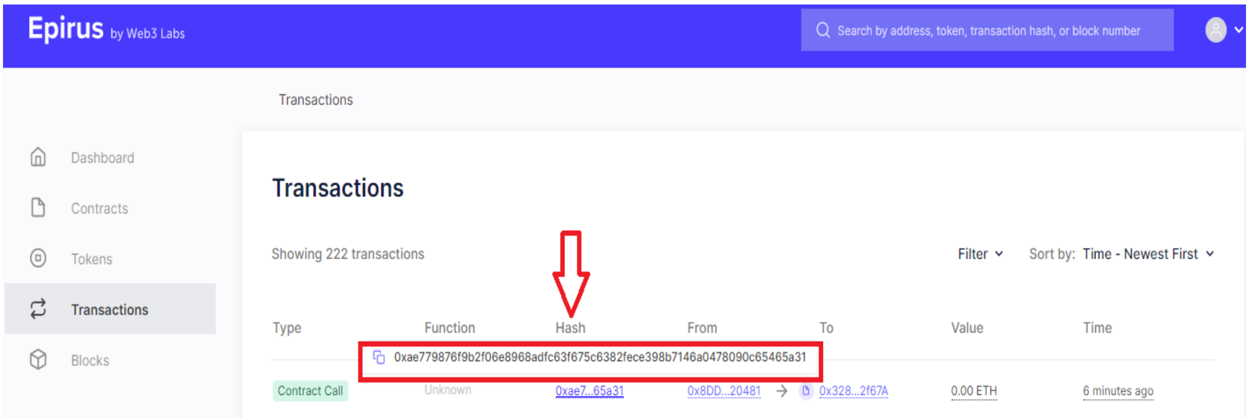
8. Transaction has been submitted!



9. Transaction has been registered in incidents list



10. Navigate to: http://epirus-8ea3d7.westeurope.cloudapp.azure.com/transactions and confirm that transaction hashes match



The incident has been successfully submitted on the blockchain!

APPENDIX G – VENABLE ET AL's (2012) FOUR-STEP METHOD FOR EVALUATING DSR PROJECTS

Venable et al's (2012) four-step evaluation framework for DSR projects is presented below:

1. Analyze the context of the evaluation – the evaluation requirements. As a first step, we need to identify, analyse, and priorities all of the requirements or goals for the evaluation portion of the DSR project.

a. Determine what the evaluands are/will be. Will they be concepts, models, methods, instantiations, and/or design theories?

b. Determine the nature of the artefact(s)/evaluand(s). Is (are) the artefact(s) to be produced a product, process, or both? Is (are) the artefact(s) to be produced purely technical or socio-technical? Will it (they) be safety critical or not?

c. Determine what properties you will/need to evaluate. Which of these (and/or other aspects) will you evaluate? Do you need to evaluate utility/effectiveness, efficiency, efficacy, ethicality, or some other quality aspect (and which aspects)?

d. Determine the goal/purpose of the evaluation. Will you evaluate single/main artefact against goals? Do you need to compare the developed artefact against with other, extant artefacts? Do you need to evaluate the developed artefact(s) for side effects or undesired consequences (especially if safety critical)?

e. Identify and analyse the constraints in the research environment. What resources are available – time, people, budget, research site, etc.? What resources are in short supply and must be used sparingly?

f. Consider the required rigor of the evaluation. How rigorous must the evaluation be? Can it be just a preliminary evaluation or is detailed and rigorous evaluation required? Can some parts of the evaluation be done following the conclusion of the project?

g. Prioritize the above contextual factors to determine which aspects are essential, more important, less important, nice to have, and irrelevant. This will help in addressing conflicts between different evaluation design goals.

2. Match the needed contextual factors (goals, artefact properties, etc.) of the evaluation (from step 1) to the criteria in figure 2 ("DSR Evaluation Strategy Selection Framework") , looking at the criteria in both white portions relating to a single dimension and the blue areas relating to a single quadrant. The criteria statement that match the contextual features of your DSR project will determine which quadrant(s) applies(y) most or are most needed. It may well be that more than one quadrant applies, indicating the need for a hybrid methods evaluation design.

3. Select appropriate evaluation method(s) from those listed in the selected, corresponding quadrant(s) in figure 3 ("DSR Evaluation Method Selection Framework"). If more than one box is indicated, selecting a method present in more than one box may be helpful. The resulting selection of evaluation methods and together with the strategy(ies) (quadrant(s)) constitutes a high-level design for the evaluation research.

4. Design the DSR evaluation in detail. Ex ante evaluation will precede ex post evaluation, but more than one evaluation may be performed, and more than one method used, in which case the order of their use and how the different evaluations will fit together must be decided. Also, the specific detailed evaluations must be designed, e.g. design of surveys or experiments. This generally will follow the extant research methods literature.

APPENDIX H – EVALUATION QUESTIONNAIRES

A. Questionnaire for evaluating features of current (or previously utilized) incident reporting platform

*Please complete this questionnaire providing answers regarding your current (or previously utilized) incident reporting platform. This questionnaire should be completed **before** testing the new platform. Please mark your selections with a "√", "x" or "+" symbol, in black or blue ink.*

On a scale from 1 to 10, with 1 being the lowest (poor) and 10 being the highest (excellent) score, please rate the following features of your current (or previously utilized) incident reporting platform:

1. How would you rate the level of user anonymity the platform provides?

No or negligent anonymity ⟷ Excellent level of anonymity

1	2	3	4	5	6	7	8	9	10

2. How would you rate the overall cost of purchasing, operating, managing, and maintaining the platform (including any staff training costs)?

Very expensive ⟷ Free or very low-cost

1	2	3	4	5	6	7	8	9	10

3. How would you rate the ease of understanding the platform's features and overall functionality?

Very hard to understand ⟷ Very easy to understand

1	2	3	4	5	6	7	8	9	10

4. How would you rate the overall ease of using the platform (including GUI design and simplicity in the reporting processes)?									
Very hard to use				↔					Very easy to use
1	2	3	4	5	6	7	8	9	10

5. How would you rate the level of customer support offered by the platform's provider?									
Very bad support				↔					Excellent support
1	2	3	4	5	6	7	8	9	10

6. How would you rate the overall level of performance and efficiency of the platform?									
Very bad				↔					Excellent
1	2	3	4	5	6	7	8	9	10

7. How would you rate the overall level of security of the platform?									
Very bad				↔					Excellent
1	2	3	4	5	6	7	8	9	10

8. How would you rate the overall level of accessibility of the platform?									
Very bad				↔					Excellent
1	2	3	4	5	6	7	8	9	10

9. How would you rate the social features (e.g. chat, forum etc.) offered by the platform (if, any)?									
Very bad									Excellent
1	2	3	4	5	6	7	8	9	10

10. How would you rate the platform's availability (i.e. uptime) level?									
Very bad									Excellent
1	2	3	4	5	6	7	8	9	10

11. How would you rate the overall platform's transparency features including the presence of any auditability mechanisms?									
Very bad									Excellent
1	2	3	4	5	6	7	8	9	10

B. Questionnaire for evaluating features of newly developed incident reporting platform

*Please complete this questionnaire providing answers regarding the newly developed incident reporting platform. This questionnaire should be completed **after** testing the new platform. Please mark your selections with a "√", "x" or "+" symbol, in black or blue ink.*

On a scale from 1 to 10, with 1 being the lowest (poor) and 10 being the highest (excellent) score, please rate the following features of your current (or previously utilized) incident reporting platform:

1. How would you rate the level of user anonymity the platform provides?

No or negligent anonymity ⟷ Excellent level of anonymity

1	2	3	4	5	6	7	8	9	10

2. How would you rate the overall cost of purchasing, operating, managing, and maintaining the platform (including any staff training costs)?

Very expensive ⟷ Free or very low-cost

1	2	3	4	5	6	7	8	9	10

3. How would you rate the ease of understanding the platform's features and overall functionality?

Very hard to understand ⟷ Very easy to understand

1	2	3	4	5	6	7	8	9	10

4. How would you rate the overall ease of using the platform (including GUI design and simplicity in the reporting processes)?									
Very hard to use									Very easy to use
1	2	3	4	5	6	7	8	9	10

5. How would you rate the level of customer support offered by the platform's provider?									
Very bad support									Excellent support
1	2	3	4	5	6	7	8	9	10

6. How would you rate the overall level of performance and efficiency of the platform?									
Very bad									Excellent
1	2	3	4	5	6	7	8	9	10

7. How would you rate the overall level of security of the platform?									
Very bad									Excellent
1	2	3	4	5	6	7	8	9	10

8. How would you rate the overall level of accessibility of the platform?									
Very bad									Excellent
1	2	3	4	5	6	7	8	9	10

9. How would you rate the social features (e.g. chat, forum etc.) offered by the platform (if, any)?									
Very bad			⟷						Excellent
1	2	3	4	5	6	7	8	9	10

10. How would you rate the platform's availability (i.e. uptime) level?									
Very bad			⟷						Excellent
1	2	3	4	5	6	7	8	9	10

11. How would you rate the overall platform's transparency features including the presence of any auditability mechanisms?									
Very bad			⟷						Excellent
1	2	3	4	5	6	7	8	9	10

APPENDIX I – "R" SCRIPT USED FOR SIGNIFICANCE TESTING

The following "R" code was used for executing the eleven significance tests of this project:

```
before <- c(x1, x2, x3, x4, x5, x6)
after <- c(y1, y2, y3, y4, y5, y6)
my_data <- data.frame(
        group = rep(c("before", "after"), each = 6),
        weight = c(before,  after)
        )
print(my_data)
library("dplyr")
group_by(my_data, group) %>%
  summarise(
    count = n(),
    median = median(weight, na.rm = TRUE),
    IQR = IQR(weight, na.rm = TRUE)
  )
before <- subset(my_data,  group == "before", weight,
         drop = TRUE)
after <- subset(my_data,  group == "after", weight,
         drop = TRUE)
library(PairedData)
pd <- paired(before, after)
plot(pd, type = "profile") + theme_bw()
library(coin)
wilcoxsign_test (before ~ after, distribution = "exact")
```

APPENDIX J – RESEARCH METHODOLOGY DETAILS

1. Research philosophy

Saunders et al (2007) explain, that some vital assumptions about the researcher's view and "understanding of the world", can be indicated by the research philosophy adopted by the researcher. These, world-related, assumptions, naturally underpin the research process (Saunders et al, 2007), and are important to review, since people may conduct research for an entire career, without considering the philosophical implications of their passively received areas of interest and research methods (Kuhn, 2012). Vaishnavi et al (2004/19) argue, that in multi-paradigmatic or pre-paradigmatic communities, such as information systems, researchers should certainly consider the "fundamental bases of the socially constructed realities in which they operate in". There are three major ways of thinking about research: Ontology, Epistemology and Axiology (Saunders et al, 2007; Collis & Hussey, 2013; Vaishnavi et al, 2004/19)

a) Ontology

According to Vaishnavi et al (2004/19), Ontology is the study that describes the nature of reality: what is real and what is not, what is derivative and what is fundamental; and whether the researcher is committed to objectivism or subjectivism in his view of reality (Saunders et al, 2007). Ontological questions include "What exists", "What is true" and "How can we sort existing things?" (Killam, 2013).

b) Epistemology

Vaishnavi et al (2004/19) describe Epistemology as the study that explores the nature of knowledge and refers to questions such as how knowledge is acquired and "how we come to know what we know" (Killam, 2013). It is a philosophical assumption concerned with items of knowledge acceptable as valid knowledge (Collis & Hussey, 2013). Epistemological questions include "On what does knowledge depend upon?” and “How can we be certain of what we know?” (Vaishnavi et al, 2004/19).

c) Axiology

According to Saunders et al (2007), Axiology is a branch of philosophy that studies “judgments about value”. It refers to what the researcher believes is valuable and ethical, and these basic beliefs guide the researcher’s decision making (Killam, 2013). Axiological questions include “What values does an individual or group hold and why?” (Vaishnavi et al, 2004/19).

2. Research paradigms

Mertens (2005) argues, that the researcher’s theoretical framework is what influences the exact nature of the definition of research. According to the same author, this framework, as distinct from a theory, is referred to as the “paradigm” and effects the way knowledge is interpreted and studied (Mertens, 2005). It is this choice of paradigm that outlines the intent, motivation and expectations of the research (Mackenzie & Knipe, 2006). According to Mackenzie & Knipe (2006), there is a number of different paradigms discussed in literature, although different sources may sometimes use different terms, which may ultimately lead to confusion. Some of the most common paradigms referred to in research, are the following (Mackenzie & Knipe, 2006):

a) Positivist and post-positivist paradigm

Positivism, which is also referred to as “science research” or “scientific method”, is based on the empiricist, rationalistic, philosophy that originated with Aristotle (Mertens, 2005), and signifies a deterministic philosophy in which outcomes or effects are determined by causes (Creswell, 2014). In order to control and/or predict forces that surround us, positivists utilize observation and measurement, in order to describe an experience, or test a theory (O’Leary, 2004). Post-positivism replaced positivism after the second World War (Mertens, 2005), and is driven by the assumption that “any piece of research is influenced by a number of well-developed theories, apart from, and as well as, the one which is being tested" (Cook & Campbell, 1979, p.24). This paradigm is most commonly associated with quantitative methods of data collection and analysis (Mackenzie & Knipe, 2006).

b) Interpretivist/constructivist paradigm

Understanding the "world of human experience", through the underlying idea that "reality is socially constructed", is the approach to research that interpretivists/constructivists take (Mertens, 2005). Therefore, researchers incline to rely upon the "participants' views of the situation being researched" (Creswell, 2014). This paradigm is most commonly associated with qualitative methods of data collection and analysis, or a combination of both, where quantitative data may be used in a way that expands or supports qualitative data (Mackenzie & Knipe, 2006).

c) Transformative paradigm

The transformative paradigm appeared during the 1980s and 1990s, partially because of the dissatisfaction related to the existing paradigms, but also due to a realization that a lot of the psychological and sociological theory related to the existing (and dominant) paradigms, "had been developed from the white, able-bodied, male, perspective, and was based on the study of male subjects" (Mertens, 2005). According to Creswell (2014, p.9), transformative researchers trust that "inquiry needs to be intertwined with politics and a political agenda", and their action agenda includes reforms able to "change the lives of the participants, the institutions in which individuals work or live, and the researcher's life" (Creswell, 2014). This paradigm is most commonly associated with a mixed methods approach (although researchers can opt for utilizing merely quantitative or qualitative approaches), for data collection and analysis (Mackenzie & Knipe, 2006), since such an approach allows the development of "more complete portraits of our social world, through the use of multiple perspectives and lenses (Somekh & Lewin, 2005, p.275).

d) Pragmatic paradigm

According to Mackenzie and Knipe (2006), Pragmatism is not loyal to any one system of philosophy or reality, but pragmatists rather focus on the “what” and “how” of the research problem (Creswell, 2014). The research problem is placed as central, and all approaches are applied to understanding the problem (Creswell, 2014); data collection and analysis methods most likely to provide insights into the problem are chosen, with no commitment, whatsoever, to any alternative paradigm (Mackenzie & Knipe, 2006). This paradigm, according to Creswell (2014), provides an opportunity for multiple methods, different assumptions and worldviews.

The following table by Mackenzie and Knipe (2006) is very informative, as it presents the language most commonly associated with the major research paradigms:

Positivist/ Postpositivist	Interpretivist/ Constructivist	Transformative	Pragmatic
Experimental Quasi-experimental Correlational Reductionism Theory verification Causal comparative Determination Normative	Naturalistic Phenomenological Hermeneutic Interpretivist Ethnographic Multiple participant meanings Social and historical construction Theory generation Symbolic interaction	Critical theory Neo-marxist Feminist Critical Race Theory Freirean Participatory Emancipatory Advocacy Grand Narrative Empowerment is sue oriented Change-oriented Interventionist Queer theory Race specific Political	Consequences of actions Problem-centred Pluralistic Real-world practice oriented Mixed models

Table J1. Language commonly associated with major paradigms (Mackenzie & Knipe, 2006)

Mackenzie and Knipe (2006) support, that while data collection methods can be combined, a researcher, does, usually, philosophically, align, with one of the research paradigms; although Hassani (2017) argues, that in many situations, a combination of approaches that these paradigms suggest, would better serve the research design - rather than a single paradigm - and a mix/multi-methods paradigm has received attention in studies (Johnson et al, 2007). Furthermore, Ramesh et al (2004) pointed out, that research in computing has been conducted according to a broad range of approaches and paradigms. In any way, the philosophical alignment of the researcher effects every decision made in the research process, including the choice of methodology (Mackenzie & Knipe, 2006).

3. Research approaches

The following section describes “Development”, “Design” and “Action” research approaches. It also details the similarities between the Design Science research and the Action research approaches.

a) Development research

Development research could be described as a disciplined investigation, for the purpose of improving either the developer, or the artefact being developed, in the general context of the development of a product, program, or software (Hasan, 2003). It can also be thought of as "proof by demonstration" (Nunamaker et al, 1991). Nunamaker et al (1991) argue that the advancement of Information Systems and practice often comes from new system concepts. However, these concepts, on their own, do not necessarily ensure a system’s subsistence, and therefore artefacts must be developed, in order to test the underlying concepts.

According to Hasan (2003), there must exist a research agenda in a system development project - since it is also research - although the progress of the project is usually determined by the system requirements. A researcher must therefore state the research problem, the questions to be asked and the consequent objectives, and must also be able to interpret the research findings in terms of research contributions to knowledge (Hasan, 2003). These contributions may be in the innovative nature of the artefact, its ability to improve workplace performance, or in the depiction of a new method of product development and they must be verifiable, usually through the success of the artefact as a proof of concept (Hasan, 2003).

Based on Nunamaker et al (1991) previous work, Hasan (2003) proposed an explicit framework for development research, which includes five stages of systems development:

i) Concept design: At this stage, the researcher ought to find, synthesize, use and apply existing knowledge, to identify gaps and develop a meaningful research objective.

ii) Constructing the system's architecture: At this stage, the researcher designs the architecture of the system, defines components, models, algorithms and data structures.

iii) Prototyping: At this stage, the researcher develops the proof-of-concept, which could be presented as a single working prototype, or could involve the iterative analysis, design and implementation of an evolving prototype.

iv) Product development: At this stage, the prototype's specifications are formalized, in order to build, test and evaluate a robust system.

v) Technology transfer: If the artefact is successful, it may seem appealing to a greater audience, and therefore, at this stage, it may be possible to evaluate the use of the artefact, with case studies or experiments, which may even trigger a new research cycle.

As Nunamaker et al (1991) point out, Development research can be described as a multimethodological approach to Information Systems research, but it is just one of the many available methodologies.

b) Design Science research

According to Vaishnavi et al (2004/19), research can be broadly defined as an activity which contributes to the understanding of a phenomenon; in the case of Design Science research, however, all or part of the phenomenon might not naturally occur, but rather be created (Lakatos, 1978). March & Smith (1995) state, that whereas natural science tries to understand reality, an “artificial” science attempts to create things that serve human purposes.” In the same context, Design Science, according to Hevner et al (2004, p.77), “creates and evaluates IT artefacts intended to solve identified organizational problems”. According to the same authors, Design Science involves a “rigorous process to design artefacts to solve observed problems, to make research contributions, to evaluate the designs, and to communicate the results to appropriate audiences”, whereas such artefacts may include “constructs, models, methods, and instantiations”. They go on by stating that research should represent a verifiable contribution, and rigor must be applied both in the development of the artefact, as well as in its evaluation. They also add that the development of the artefact should be a search process that draws from existing theories and knowledge, to come up with a solution to a defined problem, which is interesting to the research community (Gregor and Hevner, 2013) and that research must be effectively communicated to appropriate audiences. It is therefore evident that Design Science research aligns with pragmatism and it must pass both the tests of science and practice (Markus et al, 2002).

In order to understand the Design discipline and its research process, a general model for generating and accumulating knowledge has been proposed by Owen (1997):

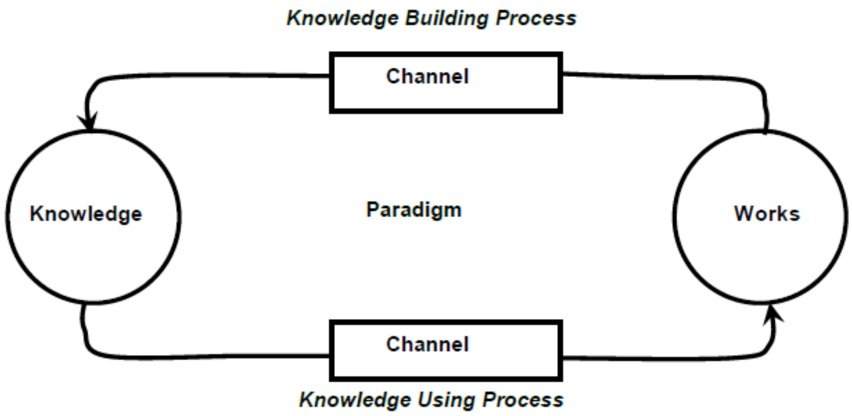


Figure J1. Model for generating and accumulating knowledge by Owen (1997)

Owen (1997, p.11) explains that “knowledge is generated and accumulated through action”, and that “doing something and judging the results is the general model”, where knowledge is creatively used to create works, which are consequently evaluated to build knowledge.

Vaishnavi et al (2004/19) argue that “learning through building”, the underlying philosophy of the Design Science research, is not exclusively used in the Information Systems domain, as the domains of engineering, education and health care, also utilize such an approach: programs of treatment are designed and empirically evaluated in health care, while the same analogy applies to new learning programs and curricula, in education. According to the same authors, Design Science research is a set of synthetic and analytical techniques and perspectives for performing research in the Information Systems domain, which typically involves the creation of an artefact and/or design theory, in order to improve the current state of practice and the existing research knowledge (Baskerville et al, 2018). According to Vaishnavi et al (2004/19), design means "to invent and bring into being", and therefore design science deals with creating a new artefact that does not exist. In order to provide a better understanding of the different forms of knowledge contribution of design science research, Vaishnavi et al (2004/19) provide the following table:

	Output	Description
1	Constructs	The conceptual vocabulary of a domain
2	Models	Sets of propositions or statements expressing relationships between constructs
3	Frameworks	Real or conceptual guides to serve as support or guide
4	Architectures	High level structures of systems
5	Design Principles	Core principles and concepts to guide design
6	Methods	Sets of steps used to perform tasks—how-to knowledge
7	Instantiations	Situated Implementations in certain environments that do or do not operationalize constructs, models, methods, and other abstract artifacts; in the latter case such knowledge remains tacit.
8	Design Theories	A prescriptive set of statements on how to do something to achieve a certain objective. A theory usually includes other abstract artifacts such as constructs, models, frameworks, architectures, design principles, and methods.

Table J2. Outputs of Design Science research by Vaishnavi et al (2004/19)

Vaishnavi et al (2014/19, p.7) argue that Design Science research differentiates itself from routine design by the production of “new, true and interesting” knowledge, and that it is mostly desirable to produce an artefact using “state-of practice application, with state-of-practice techniques, and readily available components”. According to Peffers et al (2007, p.2), although Design Science research “has been slow to diffuse into the mainstream of Information Systems research”, several researchers have been successful in bringing Design Science into the Information Systems research community, successfully making the case for its value and validity, and integrating “design” as a major component of research.

Although a generally accepted process for conducting Design Science research does not exist (Peffers et al, 2007), there is a number of available process models for conducting Design Science research, such as those by March and Smith (1995), Hevner et al (2004), Peffers et al (2007) and Purao (2013). Although the overall research process slightly differs, from model to model, they all share some basic attributes, such as identifying the problem, defining the objectives, designing, implementing and evaluating the artefact, and communicating the message to a greater audience.

c) Action research

Action research traces its origins to the social sciences – due to the massive social changes that the second World War brought (Baskerville & Myers, 2004) – and was explicitly introduced as a research methodology to the information systems community by Wood-Harper (1985), who incorporated action research concepts into an action-based systems development methodology called "Multiview" (Baskerville, 1999). Action research aims "to contribute both to the practical concerns of people in an immediate problematic situation, and to the goals of social science, by joint collaboration within a mutually acceptable ethical framework" (Rapoport, 1970, p.499). It has increased in importance for information systems towards the end of the 1990s, as the results it produces are considered to be highly relevant, since they are grounded in practical action, in order to solve a problem, while carefully informing theory (Baskerville, 1999). It therefore has a dual goal, by contributing both to research and practice (Iivari & Venable, 2009), and refers to a class of approaches, rather than a single, monolithic, research approach (Baskerville, 1999). According to Baskerville (1999), a wide-spread agreement is found in literature, on four common characteristics of Action research, which include "an action and change orientation", "a problem focus", "an organic process involving systematic and sometimes iterative stages", and "collaboration among participants".

Action research aims to create organizational change, while simultaneously studying the process (Baburoglu and Ravn 1992), and it is usually an iterative research approach, strongly oriented towards collaboration and change, involving both researchers and subjects (Baskerville & Myers, 2004). The action researcher is change-oriented; he/she believes that complex processes can be studied best by introducing changes into these processes, and consequently, observing their effects (Baskerville, 1999). Action research can be thought to belong to the post-positivist paradigm (Baskerville, 1999), although more recent literature situates Action research in pragmatism (Baskerville & Myers, 2004).

Baskerville and Myers (2004) describe a variety of forms in Action research, including "canonical action research", "collaborative practice research", “participatory action research” and “dialogical action research”. Regarding information systems, according to Baskerville (1999), these forms were inventoried and analysed from different perspectives: for example, one perspective (Baskerville & Wood-Harper, 1998) identified ten distinct forms of action research in information systems, while another perspective (Lau, 1999) outlined specific characteristics to identify research projects as members of a class of action research approaches.

According to Baskerville (1999), the most dominant action research description is the one provided by Susman and Evered (1978), although this particular approach has more recently been extended into a form known as “participatory action research” (Baskerville, 1999). After establishing a client-system research environment, five, identifiable, phases are iterated: “diagnosing”, “action planning”, “action taking”, “evaluating” and, finally, “specifying learning”. The following figure illustrates this action research structural cycle:

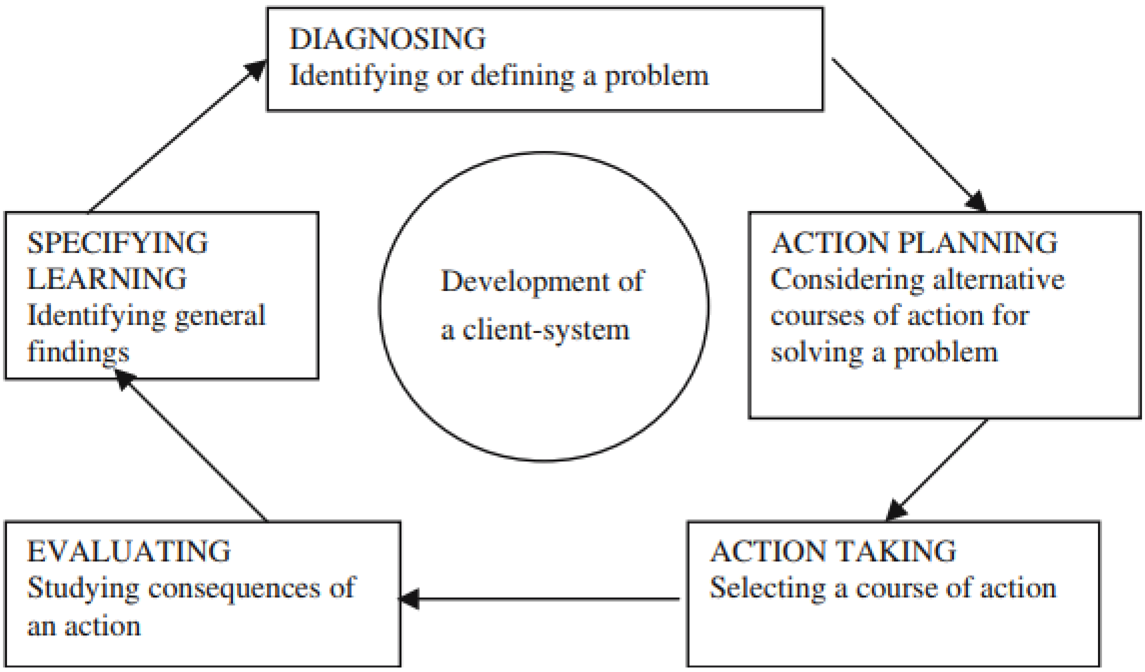


Figure J2.. Action research structural cycle by Susman and Evered (1978)

An Action research framework has also been proposed by Lau (1999), which includes four dimensions:

i) The **"conceptual foundation dimension"**, which includes the research aim, the theoretical assumptions and the perspective.

ii) The **"study design dimension"**, which includes the background of the research, the envisioned change, the participants, the sources and the duration; the methodological details of the study, in general.

iii) The **"research process dimension"**, which describes the sequence of steps by which action research is conducted, and should include one or more iterations of problem diagnosis, action interventions, reflective learning and extraction of general lessons, and,

iv) The **"role expectations dimension"**, which describes the capacity and expectations of both the researcher and the study participants, and also includes specifying competencies and evaluating ethical issues.

d) Similarities between Design Science research and Action research

According to Nguyen et al (2019), several studies comparing Design Science research and Action research have concluded that both methodologies share many common characteristics. The similarity between Design Science research and Action research has been identified by a number of studies, including those by Jarvinen (2007), Cole et al (2005) and Papas et al (2012).

Jarvinen (2007) presented a side-by-side comparison of the two approaches, comparing the cyclical process of action research and the general methodology of design science research, and concluded that there are many similarities; although the approaches might utilize different names for the various steps of each, the actual content is very similar. The following table illustrates the most important characteristics of both approaches:

Action research	Design science
AR-1: Action research emphasizes the utility aspect of the future system from the people's point of view.	DS-4: Design science's products are assessed against criteria of value or utility.
AR-2: Action research produces knowledge to guide practice in modification.	DS-2: Design science produces design knowledge (concepts, constructs, models and methods).
AR-3: Action research means both action taking and evaluating.	DS-3: Building and evaluation are the two main activities of design science.
AR-4: Action research is carried out in collaboration between action researcher and the client system.	DS-5: Design science research is initiated by the researcher(s) interested in developing technological rules for a certain type of issue. Each individual case is primarily oriented at solving the local problem in close collaboration with the local people.
AR-5: Action research modifies a given reality or develops a new system.	DS-1: Design science solves construction problems (producing new innovations) and improvement problems (improving the performance of existing entities).
AR-6: The researcher intervenes in the problem setting.	DS-5: Design science research is initiated by the researcher(s) interested in developing technological rules for a certain type of issue. Each individual case is primarily oriented at solving the local problem in close collaboration with the local people.
AR-7: Knowledge is generated, used, tested and modified in the course of the action research project.	DS-6: Knowledge is generated, used and evaluated through the building action.

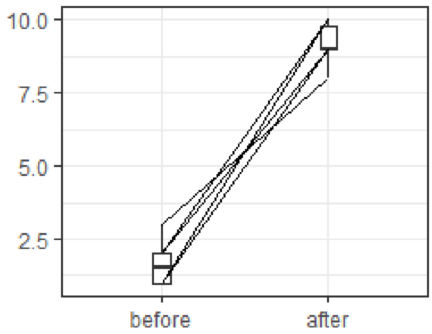
Table J3. Comparison of Action research and Design science by Jarvinen (2007)

Jarvinen (2007), therefore, concluded that there is a very high fit between the two approaches, and thus, Design Science research and Action research should be considered as similar approaches. Furthermore, Cole et al (2005) stated that both approaches share common assumptions regarding ontology, epistemology, and axiology.

The research community, however, does not reach a consensus regarding the similar nature of the two approaches. Authors such as Iivari and Venable (2009), argue that the two approaches are “decisively dissimilar”, as some activities of Design Science are always mutually exclusive from Action research, and that it is often the case that Action research does not share the paradigmatic assumptions and research interests of Design Science research. According to Peffers et al (2007, p.33), perhaps the clearest distinction between the two approaches is found in their conceptual origins, as Action research comes from the concept of the researcher being an “active participant”, in solving practical problems, in organizational contexts, whereas Design science originates “from a history of design as a component of engineering and computer science research”. Furthermore, Design Science research assumes no, specific, client-researcher relation and/or collaboration, contrary to Action research, which usually requires the existence of a particular client-researcher relationship (Iivari & Venable, 2009). According to Venable (2009, p.105), “clients” of a Design Science research project, would be “the set of all members of the generalised class of all people or organizations, who could potentially be motivated to solve instances of the generalised class of problem(s) addressed by the project’s outcome/artefact”.

APPENDIX K – SIGNIFICANCE TESTS

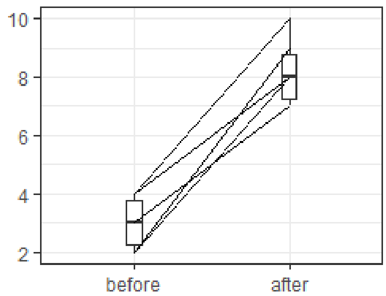
<table>
<tr><th colspan="14">Significance test 1: Anonymity</th></tr>
<tr><td>Question</td><td colspan="13">How would you rate the level of user anonymity the platform provides?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>1</td><td>2</td><td>2</td><td>1</td><td>3</td><td>1</td><td>9</td><td>9</td><td>10</td><td>9</td><td>8</td><td>10</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">1.5</td><td colspan="3">1</td><td colspan="3">9</td><td colspan="3">0.75</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="13">
</td></tr>
<tr><td>Connected objectives</td><td colspan="13">O_1, O_3</td></tr>
<tr><td>Type of test</td><td colspan="13">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="13">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td>H_0</td><td colspan="12">Level of user anonymity is not significantly different with IRDA</td></tr>
<tr><td>H_1</td><td colspan="12">Level of user anonymity is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="13"><pre>
        Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
         stratified by block
Z = -2.226, p-value = 0.03125
</pre></td></tr>
<tr><td>Verdict</td><td colspan="13">A Wilcoxon-Pratt Signed-Ranked test indicated that there is evidence to suggest that the level of user anonymity after using IRDA is significantly higher than before using IRDA (Z = -2.23, p = 0.03, α = 0.05), a partial requirement of objectives O_1, O_3</td></tr>
</table>

Table K1. Significance test for anonymity

<table>
<tr><th colspan="14">Significance test 2: Cost</th></tr>
<tr><td>Question</td><td colspan="13">How would you rate the overall cost of purchasing, operating, managing, and maintaining the platform (including any staff training costs)?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>4</td><td>3</td><td>2</td><td>2</td><td>3</td><td>4</td><td>8</td><td>7</td><td>9</td><td>8</td><td>7</td><td>10</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">3</td><td colspan="3">1.5</td><td colspan="3">8</td><td colspan="3">1.5</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="13">
</td></tr>
<tr><td>Connected objectives</td><td colspan="13">O_1</td></tr>
<tr><td>Type of test</td><td colspan="13">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="13">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td>H_0</td><td colspan="12">Level of cost is not significantly different with IRDA</td></tr>
<tr><td>H_1</td><td colspan="12">Level of cost is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="13">

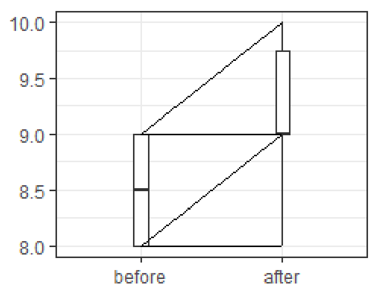
```
        Exact Wilcoxon-Pratt Signed-Rank Test

data:  y by x (pos, neg)
         stratified by block
Z = -2.2323, p-value = 0.03125
```

</td></tr>
<tr><td>Verdict</td><td colspan="13">A Wilcoxon-Pratt Signed-Ranked test indicated that there is evidence to suggest that the overall cost of IRDA is significantly lower than that of other reporting platforms ($Z = -2.23$, $p = 0.03$, $\alpha = 0.05$), a partial requirement of objective O_1</td></tr>
</table>

Table K2. Significance test for cost

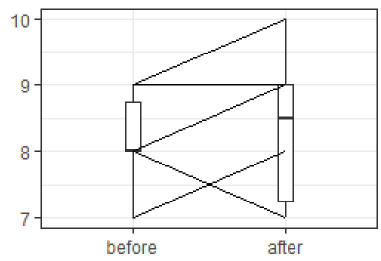
<table>
<tr><th colspan="13">Significance test 3: Ease of understanding</th></tr>
<tr><td>Question</td><td colspan="12">How would you rate the ease of understanding the platform’s features and overall functionality?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>8</td><td>8</td><td>8</td><td>9</td><td>9</td><td>9</td><td>9</td><td>9</td><td>8</td><td>9</td><td>10</td><td>10</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">8.5</td><td colspan="3">1</td><td colspan="3">9</td><td colspan="3">0.75</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="12">
</td></tr>
<tr><td>Connected objectives</td><td colspan="12">O_2, ITa_4</td></tr>
<tr><td>Type of test</td><td colspan="12">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="12">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td>H_0</td><td colspan="11">Level of ease of understanding is not significantly different with IRDA</td></tr>
<tr><td>H_1</td><td colspan="11">Level of ease of understanding is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="12">Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
stratified by block
Z = -2, p-value = 0.125</td></tr>
<tr><td>Verdict</td><td colspan="12">A Wilcoxon-Pratt Signed-Ranked test indicated that there is no evidence to suggest that IRDA’s ease of understanding is significantly different than that of other platforms (Z = -2, p = 0.13, α = 0.05), a partial requirement of objectives O_2, ITa_4</td></tr>
</table>

Table K3. Significance test for ease of understanding

<table>
<tr><th colspan="14">Significance test 4: Ease of use</th></tr>
<tr><td>Question</td><td colspan="13">How would you rate the overall ease of using the platform (including GUI design and simplicity in the reporting processes)?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>7</td><td>8</td><td>9</td><td>9</td><td>8</td><td>8</td><td>8</td><td>9</td><td>10</td><td>9</td><td>7</td><td>7</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">8</td><td colspan="3">0.75</td><td colspan="3">8.5</td><td colspan="3">1.75</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="13">
</td></tr>
<tr><td>Connected objectives</td><td colspan="13">O_2</td></tr>
<tr><td>Type of test</td><td colspan="13">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="13">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td>H_0</td><td colspan="12">Level of ease of use is not significantly different with IRDA</td></tr>
<tr><td>H_1</td><td colspan="12">Level of ease of use is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="13">Exact Wilcoxon-Pratt Signed-Rank Test

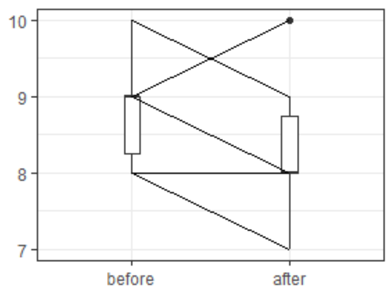
data: y by x (pos, neg)
stratified by block
Z = -0.44721, p-value = 1</td></tr>
<tr><td>Verdict</td><td colspan="13">A Wilcoxon-Pratt Signed-Ranked test indicated that there is no evidence to suggest that the level of IRDA’s ease of use is significantly different than that of other platforms ($Z = -0.45$, $p = 1$, $\alpha = 0.05$), a partial requirement of objective O_2</td></tr>
</table>

Table K4. Significance test for ease of use

Significance test 5: Customer support

Question	How would you rate the level of customer support offered by the platform's provider?

Results	Before						IRDA	After					
	A	B	C	D	E	F		A	B	C	D	E	F
	8	8	9	10	9	9	n = 6	7	8	8	9	10	8
Medians (M) & Inter-quartile range (IQR)	M			IQR				M			IQR		
	9			0.75				8			0.75		

Boxplot of paired results	
Connected objectives	O_2
Type of test	Two-tailed
Confidence interval	95%
Hypotheses	H_0: Level of ease of customer support is not significantly different with IRDA H_1: Level of ease of customer support is significantly different with IRDA

Results

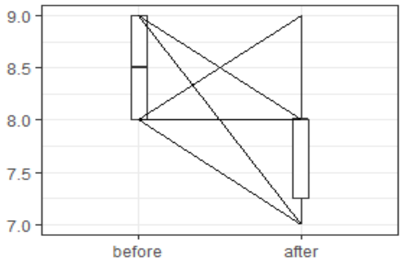
```
        Exact Wilcoxon-Pratt Signed-Rank Test

data:  y by x (pos, neg)
         stratified by block
Z = 1.3416, p-value = 0.375
```

Verdict	A Wilcoxon-Pratt Signed-Ranked test indicated that there is no evidence to suggest that the level of IRDA's customer support is significantly different than that of other platforms ($Z = 1.34$, $p = 0.38$, $\alpha = 0.05$), a partial requirement of objective O_2

Table K5. Significance test for level of customer support

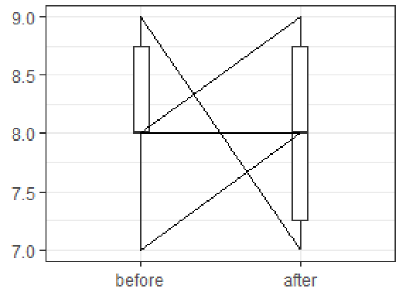
<table>
<tr><th colspan="15">Significance test 6: Performance & Efficiency</th></tr>
<tr><td>Question</td><td colspan="14">How would you rate the overall level of performance and efficiency of the platform?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>9</td><td>9</td><td>8</td><td>9</td><td>8</td><td>8</td><td>7</td><td>8</td><td>9</td><td>8</td><td>8</td><td>7</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">8.5</td><td colspan="3">1</td><td colspan="3">8</td><td colspan="3">0.75</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="13"></td></tr>
<tr><td>Connected objectives</td><td colspan="13">O_2</td></tr>
<tr><td>Type of test</td><td colspan="13">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="13">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td>H_0</td><td colspan="12">Level of ease of performance & efficiency is not significantly different with IRDA</td></tr>
<tr><td>H_1</td><td colspan="12">Level of ease of performance & efficiency is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="13">Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
stratified by block
Z = 1.41, p-value = 0.3125</td></tr>
<tr><td>Verdict</td><td colspan="13">A Wilcoxon-Pratt Signed-Ranked test indicated that there is no evidence to suggest that the level of IRDA's performance and efficiency is significantly different than that of other platforms ($Z = 1.41$, $p = 0.31$, $\alpha = 0.05$), a partial requirement of objective O_2</td></tr>
</table>

Table K6. Significance test for performance and efficiency

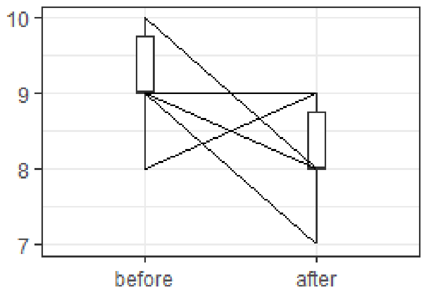
<table>
<tr><th colspan="15">Significance test 7: Security</th></tr>
<tr><td>Question</td><td colspan="14">How would you rate the overall level of security of the platform?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>7</td><td>8</td><td>8</td><td>9</td><td>8</td><td>9</td><td>8</td><td>9</td><td>8</td><td>7</td><td>9</td><td>7</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">8</td><td colspan="3">0.75</td><td colspan="3">8</td><td colspan="3">1.5</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="14"></td></tr>
<tr><td>Connected objectives</td><td colspan="14">O_2</td></tr>
<tr><td>Type of test</td><td colspan="14">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="14">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td>H_0</td><td colspan="13">Level of security is not significantly different with IRDA</td></tr>
<tr><td>H_1</td><td colspan="13">Level of security is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="14"><pre>
        Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
         stratified by block
Z = 0.21381, p-value = 1
</pre></td></tr>
<tr><td>Verdict</td><td colspan="14">A Wilcoxon-Pratt Signed-Ranked test indicated that there is no evidence to suggest that the level of IRDA's security is significantly different than that of other platforms ($Z = 0.21$, $p = 1$, $\alpha = 0.05$), a partial requirement of objective O_2</td></tr>
</table>

Table K7. Significance test for security

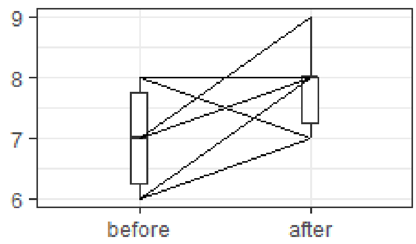
<table>
<tr><th colspan="15">Significance test 8: Accessibility</th></tr>
<tr><td>Question</td><td colspan="14">How would you rate the overall level of accessibility of the platform?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>9</td><td>10</td><td>9</td><td>10</td><td>8</td><td>9</td><td>7</td><td>8</td><td>9</td><td>8</td><td>9</td><td>8</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">9</td><td colspan="3">0.75</td><td colspan="3">8</td><td colspan="3">0.75</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="14">
</td></tr>
<tr><td>Connected objectives</td><td colspan="14">O_2</td></tr>
<tr><td>Type of test</td><td colspan="14">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="14">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td colspan="2">H_0</td><td colspan="12">Level of accessibility is not significantly different with IRDA</td></tr>
<tr><td colspan="2">H_1</td><td colspan="12">Level of accessibility is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="14"><pre>   Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
         stratified by block
Z = 1.6036, p-value = 0.1875</pre></td></tr>
<tr><td>Verdict</td><td colspan="14">A Wilcoxon-Pratt Signed-Ranked test indicated that there is no evidence to suggest that the level of IRDA's accessibility is significantly different than that of other platforms (Z = 1.60, p = 0.19, α = 0.05), a partial requirement of objective O_2</td></tr>
</table>

Table K8. Significance test for accessibility

<table>
<tr><th colspan="14">Significance test 9: Social features</th></tr>
<tr><td>Question</td><td colspan="13">How would you rate the social features (e.g. chat, forum etc.) offered by the platform (if, any)?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>6</td><td>8</td><td>7</td><td>8</td><td>7</td><td>6</td><td>8</td><td>8</td><td>9</td><td>7</td><td>8</td><td>7</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">7</td><td colspan="3">1.5</td><td colspan="3">8</td><td colspan="3">0.75</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="13">
</td></tr>
<tr><td>Connected objectives</td><td colspan="13">O_2</td></tr>
<tr><td>Type of test</td><td colspan="13">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="13">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td colspan="2">H_0</td><td colspan="11">Level of social features is not significantly different with IRDA</td></tr>
<tr><td colspan="2">H_1</td><td colspan="11">Level of social features is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="13">Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
stratified by block
Z = -1.4967, p-value = 0.25</td></tr>
<tr><td>Verdict</td><td colspan="13">A Wilcoxon-Pratt Signed-Ranked test indicated that there is no evidence to suggest that the level of IRDA’s social features is significantly different than that of other platforms ($Z = -1.50$, $p = 0.25$, $\alpha = 0.05$), a partial requirement of objective O_2</td></tr>
</table>

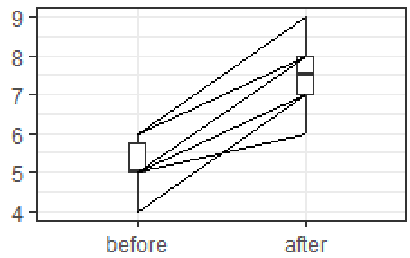
Table K9. Significance test for social features

<table>
<tr><th colspan="14">Significance test 10: Availability</th></tr>
<tr><td>Question</td><td colspan="13">How would you rate the platform’s availability (i.e. uptime) level?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA
n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>9</td><td>8</td><td>9</td><td>9</td><td>8</td><td>8</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">8.5</td><td colspan="3">1</td><td colspan="3">10</td><td colspan="3">0</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="13">
</td></tr>
<tr><td>Connected objectives</td><td colspan="13">O_3</td></tr>
<tr><td>Type of test</td><td colspan="13">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="13">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td colspan="2">H_0</td><td colspan="11">Level of platform availability is not significantly different with IRDA</td></tr>
<tr><td colspan="2">H_1</td><td colspan="11">Level of platform availability is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="13">Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
stratified by block
Z = -2.2514, p-value = 0.03125</td></tr>
<tr><td>Verdict</td><td colspan="13">A Wilcoxon-Pratt Signed-Ranked test indicated that there is evidence to suggest that the level of platform availability of the IRDA is significantly higher than that of other platforms ($Z = -2.25$, $p = 0.03$, $\alpha = 0.05$), a partial requirement of objective O_3</td></tr>
</table>

Table K10. Significance test for availability

<table>
<tr><th colspan="14">Significance test 11: Transparency</th></tr>
<tr><td>Question</td><td colspan="13">How would you rate the overall platform’s transparency features including the presence of any auditability mechanisms?</td></tr>
<tr><td rowspan="3">Results</td><td colspan="6">Before</td><td rowspan="5">IRDA

n = 6</td><td colspan="6">After</td></tr>
<tr><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td>6</td><td>6</td><td>5</td><td>5</td><td>4</td><td>5</td><td>8</td><td>9</td><td>6</td><td>8</td><td>7</td><td>7</td></tr>
<tr><td rowspan="2">Medians (M) & Inter-quartile range (IQR)</td><td colspan="3">M</td><td colspan="3">IQR</td><td colspan="3">M</td><td colspan="3">IQR</td></tr>
<tr><td colspan="3">5</td><td colspan="3">0.75</td><td colspan="3">7.5</td><td colspan="3">1</td></tr>
<tr><td>Boxplot of paired results</td><td colspan="13">
</td></tr>
<tr><td>Connected objectives</td><td colspan="13">O_3</td></tr>
<tr><td>Type of test</td><td colspan="13">Two-tailed</td></tr>
<tr><td>Confidence interval</td><td colspan="13">95%</td></tr>
<tr><td rowspan="2">Hypotheses</td><td colspan="2">H_0</td><td colspan="11">Level of platform transparency is not significantly different with IRDA</td></tr>
<tr><td colspan="2">H_1</td><td colspan="11">Level of platform transparency is significantly different with IRDA</td></tr>
<tr><td>Results</td><td colspan="13">Exact Wilcoxon-Pratt Signed-Rank Test

data: y by x (pos, neg)
stratified by block
Z = -2.2323, p-value = 0.03125</td></tr>
<tr><td>Verdict</td><td colspan="13">A Wilcoxon-Pratt Signed-Ranked test indicated that there is evidence to suggest that the level of platform transparency of the IRDA is significantly higher than that of other platforms ($Z = -2.23$, $p = 0.03$, $\alpha = 0.05$), a partial requirement of objective O_3</td></tr>
</table>

Table K11. Significance test for transparency

APPENDIX L – COMPLIMENTARY EVALUATION METHOD

According to the ISO/IEC 25010 (2011), the quality of a software/system refers to “the degree to which the software/system satisfies the stated and implied needs of its various stakeholders, and thus provides value”. Estdale and Georgiadou (2018) argue, that this value (to organizations and users), arises from the software’s actual behaviour in use, and that ISO/IEC 25010 aids in identifying such value, by dividing software characteristics in two quality models (“quality in use” and “product quality”), and thus enables direct assessment of the developed software. The “quality in use” model assesses the outcome of interaction, when a software is used in a particular context, and is composed of five characteristics, which are subdivided in further sub-characteristics (ISO/IEC 25010, 2011). The “product quality” model, assesses the static properties of the software and the dynamic properties of the system, and is composed of eight characteristics, which are also subdivided in further sub-characteristics (ISO/IEC 25010, 2011). According to the standard, both models provide a set of characteristics, against which the stated quality requirements of the software/system can be compared for completeness, and are applicable to both software products and systems. The models can be utilized by those responsible for evaluating the software/system’s quality, such as developers, quality assurance, control staff, and independent evaluators (ISO/IEC 25010, 2011).

a) Designing the evaluation in more detail:

Complimentary evaluation method	
Purpose	The purpose of this evaluation method, is to complement the main evaluation method, by performing a high-level assessment of the developed software against the requirements posed by the international standard “ISO/IEC 25010:2011 - Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models”. This evaluation method aims to assess the quality of the developed software (artefact), by utilizing an internationally renowned software quality standard.

Evaluation method	Participant observation (researcher).
Evaluation method details	The developed software's features/characteristics will be compared by the researcher against the ISO/IEC 25010 Software Quality Model requirements. According to the standard, software quality is assessed over two broad dimensions, "product quality" and "quality in use". "Product quality" relates to the static and dynamic properties of the software, and includes assessment of eight, distinct, characteristics, whereas "quality in use" evaluates the outcome of human interaction with the software, and includes the assessment of five, distinct, characteristics.
Participants	Researcher.
Roles and responsibilities	Researcher will compare the features/characteristics of the developed software against the requirements of the standard and document the results.
Timeframe	09/12/2019 – 13/12/2019
Prerequisites	Access to the text and provisions of the ISO/IEC 25010:2011 standard.
Assumptions	The various characteristics will be assessed on a high level of detail (overview, low degree of granularity).
Expected outcome	A table describing how the developed software satisfies (or not) the various requirements/provisions of the standard.
Data analysis method of results	One-to-one comparison of characteristics and qualitative interpretation of results.

Table L1. Second evaluation method details

b) Results and analysis

As previously indicated, this is a complementary, qualitative, evaluation method, and therefore the various software characteristics were evaluated, by the researcher, on a high, non-exhaustive, overview, level. The following table summarizes the evaluation results:

Quality in use		
Characteristics	**Sub-characteristics**	**IRDA software**
Effectiveness		The software developed met its specified goal, which was to create a manual, private, incident reporting platform, with specific characteristics, which can encourage incident reporting by users and organizations. Users can view and submit incidents and communicate in an anonymous fashion, without losing any of the benefits offered by other reporting solutions.
Efficiency		Although a decentralized solution is undoubtfully slower than a centralized one, the number of both users and expected incidents for every platform instance are considerably low, to expect any major efficiency/performance issues. The choice of a private blockchain implementation along with a suitable consensus algorithm (PoA/IBFT) also ensured that transactions are processed in a light-weight manner. The first evaluation method also indicated that there is no evidence to suggest that the level of IRDA's performance and efficiency is significantly different than that of other platforms.
Satisfaction	Usefulness	Participants who tried IRDA expressed their satisfaction with the software, successfully completed all test cases, and positively evaluated the various software features.
	Trust	Based on the success of the test cases performed by all participants (including the researcher), it is expected that the software will behave as intended.

	Pleasure	Participants did not express any dissatisfaction while utilizing the various features of the software.
	Comfort	Participants did not express any physical discomfort while using the software.
Freedom from risk	Economic risk	IRDA is considerably less expensive to operate than any other commercial alternatives. Furthermore, due to its anonymity features, the economic risk associated with potential concerns (i.e. economic risk associated with organizational reputation concerns) is mitigated.
	Health & safety risk	No health and safety risks applicable to users were identified.
	Environmental risk	No environmental risk was identified. The choice of a private blockchain implementation, along with a suitable algorithm (PoA/IBFT), also ensured that transactions are more environmentally friendly than open blockchain implementations utilizing 'traditional' algorithms (such as PoW)
Context coverage	Context completeness	IRDA can be easily used by non-expert or non-technical users, using widely available equipment (personal computers, laptops, mobile devices), with a limited set of prerequisites (i.e. having an Ethereum wallet and a Web3.0 capable browser). IRDA can operate in low network bandwidth conditions, although it cannot operate offline (i.e. a complete lack of internet connectivity would not allow users to view/submit incidents, chat etc).
	Flexibility	Although the software currently operates in a particular BaaS environment (Microsoft Azure), it could easily be transferred to any other cloud provider (such as AWS for example), or even operate in a local environment. If, at some point in the future, further platform functionality is required, new smart contracts could be deployed (with associated front-end modifications).

Product quality		
Characteristics	**Sub-characteristics**	**IRDA software**
Functional suitability	Completeness	The software satisfies all stated and implied needs and objectives, as those were set in Chapters four (Objectives) and five (Design & Implementation) of this report.
	Correctness	The various test cases performed, indicated that the software provides the correct/expected results, with the needed degree of precision.
	Appropriateness	The software's functions facilitate the accomplishment of specified tasks and objectives; the various functions are simple and easy to understand and use (by end users of the platform) and exclude any unnecessary steps.
Performance efficiency	Time-behaviour	Although no tests regarding the software's processing times and throughput rates were executed, the 100 transactions per second allowed by Quorum (and independently verified by other researchers, such as Baliga et al, 2018) are considered adequate for the platform's initial purposes.
	Resource utilization	No specific tests have been performed. However, according to Azure's platform statistics, with the selected implementation, the blockchain nodes run at a max CPU usage of 2.5%, with a memory usage of around 35%. If needed in the future (e.g. if significantly more users/incidents join/are recorded), additional resources can be purchased.
	Capacity	Again, no specific tests have been performed. However - and as also mentioned above – the numbers of virtual machines and blockchain nodes on Azure can be increased at any time, to

		accommodate demand in users/bandwidth/ storage.
Compatibility	Co-existence	The software is implemented on a BaaS platform and therefore its use should not have any detrimental impact, on any other service/product currently present in the local operational environment of the users.
	Interoperability	The software's front-end can smoothly invoke/receive/interpret communication with the blockchain, through designated smart contracts.
Usability	Appropriateness recognisability	Participants did not express any concerns related to the appropriateness of the software, for incident reporting purposes. They successfully performed all test cases and positively evaluated the software, during the first evaluation activity.
	Learnability	Participants did not express any concerns related to difficulties in learning to use the software and its features. They were provided with product/test instructions, they successfully performed all test cases, and then positively evaluated the software, during the first evaluation activity.
	Operability	Participants did not express any concerns related to operating the software and using its features.
	User error protection	The most serious error a user can make on the platform, is submitting an incident (through the relevant form) with inaccurate, wrong or missing content. Firstly, the software does not allow reports with empty content on the 'required' form fields. Secondly, when the user completes the form and clicks on the "Preview & submit" button, a warning message appears to the user, calling him/her to carefully review the incident before clicking the "submit" button, as the incident is not editable once it is submitted.

	User interface aesthetics	Participants did not express any concerns related to the aesthetics of the user interface. The GUI is simple and intuitive, and no excessive graphics were applied.
	Accessibility	The software can be accessed by any device with internet connectivity, an Ethereum wallet, and a Web3.0 capable browser. Where possible, users with disabilities can take advantage of their device/browser's disability features (such as zoom, contrast, text-to-speech etc) to browse content.
Reliability	Maturity	Although no specific tests (other than the test cases) have been performed, the software seems to be reliable under normal operation circumstances.
	Availability	100% blockchain availability is warranted; Quorum nodes ensure that actual content (incidents) is always available. However, since the front-end components were not stored in a decentralized fashion, their availability is limited to 99.9% (according to researcher's hosting provider).
	Fault tolerance	Hardware or software faults in a particular node do not hinder the platform's operational status, as remaining nodes can handle operations.
	Recoverability	In the event of an interruption or failure of a node, remaining nodes handle operations. When the failed node recovers/returns, the other blockchain nodes aid in updating its state accordingly.
Security	Confidentiality	Only pre-authorized users have access to the platform. Multi-factor authentication ensures that stolen platform credentials, on their own, do not grant access to the platform. Encryption ensures that data is protected in all of its states (at rest, in transit, in use).

	Integrity	Due to blockchain’s inherent characteristics, data (incidents) submitted on the platform cannot, in any way, be modified.
	Non-repudiation	Due to blockchain’s inherent characteristics, data (incidents) submitted on the platform cannot, in any way, be repudiated.
	Accountability	Although users of the platform can submit incidents and communicate in an anonymous fashion, between each other, the platform’s administrator can detect the originator of an incident submission. This allows the administrator to take appropriate action, should malicious behaviour be detected. However, the administrator cannot control/track content in the chat function, which is truly anonymous for every participant.
	Authenticity	The authenticity of a participant is initially ensured through the offline screening of a registration application, before being granted access to the platform. Approved users have their public key white-listed as a first authentication measure. Authenticity is also enhanced by utilizing two-factor authentication procedures (email/password combination, OTP via SMS).
Maintainability	Modularity	The platform is built of discrete components (both backend and frontend), which provide adequate modularity. The platform could change GUI, other frontend properties and complete operating environment, without requiring major modifications.
	Reusability	The principle of code reusability has been utilized throughout the development of the platform. Code snippets and smart contracts could be reused for any other similar project.

	Analyzability	Well-commented source code allows implementing potential changes with due care and due diligence.
	Modifiability	The software can be effectively and efficiently modified in the future, according to specific needs, without introducing defects or degrading its quality.
	Testability	Adequate test criteria can be defined for the platform. A number of functional test cases have already been executed by the participants (users & researcher), while non-functional tests can be scheduled and performed in the future.
Portability	Adaptability	The platform is highly adaptable. In case of increased usage, more nodes, storage and memory could easily be added. The platform could also be moved entirely to a new cloud provider or be locally installed. In addition, although the platform has not been optimized for mobile use, this can be done at some point in the future.
	Installability	As already mentioned, the platform can be moved/installed entirely to a new cloud provider or be locally installed. Regarding users, apart from having an Ethereum wallet and a Web 3.0-compatible browser, they do not need to execute any other local installations, since the platform supports web-based access.
	Replaceability	Users of the decentralized platform can, at any point, cease using the platform, and utilize any other platform or mechanism, without worrying about any "lock-in" risk. Incidents on the platform are submitted and presented in a standardized format (utilizing ISO/IEC 27035:2016 reporting template with eCSIRT.net mkVI" taxonomy), and users can import these in any other software.

Table L2. Second evaluation method results