UNIVERSITY OF SOUTHAMPTON

FACULTY OF PHYSICAL SCIENCES AND ENGINEERING

Electronic and Computer Science

A Framework to Secure a Document outside Its Organization

by

Zeyad Sabah Aaber

Thesis for the degree of Doctor of Philosophy in Computer Science

December 2016

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL SCIENCES AND ENGINEERING

Computer Science

Thesis for the degree of Doctor of Philosophy

A FRAMEWORK TO SECURE A DOCUMENT OUTSIDE ITS ORGANIZATION

Zeyad Sabah Aaber

Electronic information in any enterprise is an asset, and may be stored in a database or as electronic documents (word, PDF, and spreadsheet). This research focuses on the information leakage caused by documents going astray. Current document security statistics suggest that 63% of information was leaked as documents in 2012. Half of this was due to employee unawareness, and it has cost billions in terms of Intellectual Property IP, effort and money. The problem is that individual documents are almost defenceless outside the enterprise. Encryption and password protection are not impenetrable; it is only a question of time before the information is extracted using intense computing processing power.

This research contributes a new conceptual framework to secure a document, regardless of its place inside or outside the organisation. The framework combines the concepts of Active document, Digital Rights Management, Certificate Authorities, and content providers. The Tamper Proof Framework (TPF) will enable any document to be involved in decisions regarding the basic operations performed on it. Security experts and security-related IT staff confirmed these components.

The research produced a general conceptual framework that can be used in different domains. Formal methods were used to produce an event-based conceptual formal model. This model is an intermediate step for many future research directions.

Table of Contents

Chapter 1: Introduction 1

1.1 Research Question 2

1.2 Thesis structure 2

1.3 Contribution 3

Chapter 2: Literature Review 5

2.1 Security 5

2.1.1 Security goals 6

2.1.2 Data Security 6

2.1.3 Database Security 7

2.1.4 Information Security 7

2.1.5 Application/software security 8

2.1.6 Computer security 8

2.1.7 Network security 9

2.1.8 The Human factor in security 9

2.2 Information security 9

2.2.1 Confidentiality 10

2.2.2 Integrity 10

2.2.3 Availability 10

2.3 Document security 10

2.3.1 Access Control 11

2.3.2 Document Legalisation 12

2.4 Current Document Security Technologies 12

2.4.1 Electronic Document Management System 13

2.4.2 Electronic Records Management 13

2.4.3 Enterprise Content Management 13

2.4.4 Data Loss (leakage) Prevention 14

2.4.5 Digital/Information Rights Management 14

2.4.6 Issues with the current security technology 15

2.5 Public Key Infrastructure 15
2.6 Formal Methods 17
2.6.1 Formal Methods 17
2.6.2 Event-B 19
2.6.3 The Rodin Platform 21
2.6.4 Event-B Mathematical Language 22
2.7 Active Document 22
2.8 Summary 25
Chapter 3: Tamper-Proof Document proposal 27
3.1 Theory background 27
3.2 Initial framework functions 28
3.3 Initial framework components 29
3.4 Document access scenarios 30
Chapter 4: Research methodology 33
4.1 Research methods 33
4.1.1 Quantitative Methods 33
4.1.2 Qualitative Methods 34
4.1.3 Mixed Methods 36
4.2 Research methods applied 37
4.2.1 Triangulation 37
4.2.2 Methodology process steps 38
4.2.3 Expert Review 38
4.2.4 Expert review questions design 39
4.2.5 Survey 41
4.2.6 Survey questions design 42
4.2.7 Data analysis procedure 42
4.2.8 Modelling the framework using formal methods 43
4.3 Summary 44
Chapter 5: Findings and Discussion 45
5.1 Findings from Expert Review 45

5.1.1 General questions 45
5.1.2 Framework components 47
5.1.3 Summary 48
5.2 The modified framework 50
5.2.1 Framework Components 50
5.2.2 Framework Usage Scenarios 51
5.2.3 Overall system components 53
5.2.4 Overall framework workflow 55
5.3 Result from the survey 59
5.4 Discussion 60
5.5 Summary 62
Chapter 6: Modelling the framework using formal methods 65
6.1 Introduction to system modelling 65
6.2 TPF behavioural event-based modelling 66
6.3 Introduction to formal methods 69
6.4 Formal Conceptual Modelling of proposed framework: 70
6.4.1 Using Rodin to verify the model 74
6.5 Summary 78
Chapter 7: Conclusions and Future work 79
7.1 Contribution 79
7.2 Conclusions 79
7.3 Future work 80
Appendix A Survey Questions 81
A.1 General questions 81
A.2 Security related questions 82
Appendix B Expert Interview Questions 87
B.1 Demographic questions 87
B.2 General questions 87
B.3 Human negligence 88

B.4 Cross-domains 88
B.5 Legalisation 89
B.6 Information leakage 89

Appendix C Survey Data 91

C.1 Part one of the survey raw data 91
C.2 Part two of the survey raw data 98

References 103

List of Tables

Table 1 Issues with the current document security technologies 15

Table 2 Expert distribution 45

Table 3 Expert Overview 46

Table 4 Domain frequency for survey participants 59

Table 5 Survey response for the identified issue 59

Table 6 One-Sample Statistics for the components survey questions 60

Table 7 One-Sample Test with test value = 3 61

Table 8 the proposed framework states 67

Table 9 proposed framework event list 68

Table 10 Confirmed framework components mapping toward security triage CIA 71

Table 11 Framework machine States and their relative functions 73

Table 12 List of questions used in the survey about the framework components 85

Table 13 First part of the raw survey data 91

Table 14 Second part of the raw survey data 98

List of Figures

Figure 1 Document DRM architecture (after ho Eom 2012) 14

Figure 2 Anatomy of Event-B Models 21

Figure 3 Example of Event-B Outer and Inner Syntax 22

Figure 4 A detailed view of the Tamper-Proof framework structure..... 29

Figure 5 Document access scenarios when using the proposed framework..... 32

Figure 6 The methodology used in this research to answer the research questions 33

Figure 7 Mixed methodology using Triangulation to confirm the proposed framework 38

Figure 8 G*power Calculation for sample size..... 42

Figure 9 Cloud as customisation and delivery channel..... 50

Figure 10 TPF document access scenario..... 52

Figure 11 TPF overall structure 58

Figure 12 a sequence diagram for TPF when sharing a document with another organisation. 67

Figure 13 The behavioral event-based model of the proposed framework..... 69

Figure 14 the context of the framework as seen by Rodin tool 76

Figure 15 a formal model written in Event-B for any secure machine in an organisation part 1 77

Figure 16 a formal model written in Event-B for any secure machine in an organisation part 2 78

Figure 17 Cloud as customisation and delivery channel..... 84

Figure 18 The Proposed Framework structure 90

DECLARATION OF AUTHORSHIP

I, Zeyad Sabah Aaber declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

A FRAMEWORK TO SECURE DOCUMENTS OUTSIDE ORIGINATING ORGANIZATIONS

I confirm that:

This work was done wholly or mainly while in candidature for a research degree at this University;

Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;

Where I have consulted the published work of others, this is always clearly attributed;

Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;

I have acknowledged all main sources of help;

Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;

Parts of this work have been published as:

Aaber, Z. S., Crowder, R.M., Fadhel, N.F., & Wills, G.B., 2014. Preventing document leakage through active document. In 2014 World Congress on Internet Security, WorldCIS 2014. pp. 53–58.

Aaber, Z.S., Crowder, R.M., Chang, V.I., Fadhel, N.F., & Wills, G.B., 2015. Towards a framework for securing a document outside an organisational firewall. In Proceedings of the International Conference on Cloud Computing Technology and Science, CloudCom. pp. 1057–1062.

Aaber, Z.S., Wills, G.B. & Crowder, R.M., 2017. Protecting Document Outside Enterprise Network: A Confirmed Framework. In Enterprise Security. Springer, Cham, pp. 259–275.

Signed:

Date: 26/06/2017

Acknowledgements

I would like to thank all my family, especially my parents and siblings. To my Sarah, who always stood by my side no matter what happened, and to my daughters Fatima and Durah.

Thanks to my Supervisors Gary Wills and Richard Crowder the best ever one can get. Especially, Gary, you made my PhD journey approachable.

To all my friends and colleagues who supported me and helped survive through my PhD down times.

Definitions and Abbreviations

API	Application Programming Interface
B2B	Business-to-Business
CIA	Confidentiality, Integrity, Availability
CMS	Content Management System
DAC	Discretionary Access Control
DLP	Data Loss (leakage) Prevention
DMS	Document Management System
DoS	Denial of Service
ECM	Enterprise Content Management
ERM	Enterprise Rights Management
G2G	Government-to-Government
IG	Information Governance
IP	Intellectual Property
IRM	Information Rights Management
MAC	Mandatory Access Control
RBAC	Role-Based Access Control
XML	eXtensible Mark-up Language

Chapter 1: Introduction

Electronic documents are an essential part of the 21th century. An individual's files contain diverse and sometimes sensitive information varying from medical records to bank statements, for example. The same applies to corporate businesses and the government sector, but on a massive scale. They are sending documents that may contain more sensitive information than a single bank account. Their information may contain intellectual property, financial information, price-lists, and employees or patients' private data. Losing such information could have a devastating effect on the organisation and their data subjects, if it were to get into the wrong hands.

Document leakage is a continual threat that every organization is facing regardless of its domain or size. From time to time, the news tells of classified information leaked in its original document, an example of which is the PRISM scandal unveiled by Edward Snowden (Macaskill & Dance 2013). The US NSA had done its best to protect its data from being attacked from outside. Moreover, it had to enforce maximum-security measures and policies to protect its information assets from insider attacks. Yet, documents were still being leaked, mostly by employees, whether due to carelessness or on purpose.

Carnegie Mellon University conducted a survey in 2013 which found that 53% of the organisations participating agreed that "*damage caused by insider attacks is more damaging than outsider attacks*" (Software Engineering Institute 2013, p.5). It also revealed that 68% of insider attacks are either unintentional exposure of private data or Intellectual Property (IP) theft.

Document leakage can cause serious damage to any organisation in any domain. The National Health Service (NHS) suspended one of its staff who had sent an email to his home email address containing pay slip details for the entire staff at his hospital (Greatrex 2010). From the world of commerce, an engineering employee at General Motors stole hybrid car trade secrets and sold them to a rival manufacturer in China (Smallwood 2012).

So is it more important to focus on securing information from being attacked or on mitigating the damage from that attack? Since there is no "zero tolerant security system", this research will focus on mitigating the damage. The research question is "what conceptual framework is suitable for securing a document when it leaves the organisation's boundary?"

1.1 Research Question

This research explores document security in general and in the particular enterprises of government, education, healthcare, and business. At the end of this review the following question was the main theme which crossed all the literature:

What framework is suitable for securing documents when they go outside an organization firewall?

This question is divided into sub-questions:

SQ1. What security issues are there in documents used by cross-domain platforms?

SQ2. What technologies may be used to provide the framework with the ability to seamlessly integrate with existing presentation software?

SQ3. What security mechanisms facilitate secure document sharing in a collaborative environment between two organisations?

The first sub-question is addressed by the literature review in Chapter 2. Using the triangulation method, the second and third sub-questions are addressed. The third sub question is addressed in Chapter 6. The confirmed conceptual framework requires a long time to implement, so this work uses formal methods to build a model of the framework as an intermediate step. The software implementation may be part of the future work.

1.2 Thesis structure

Chapter 2 is the literature review and covers related aspects of document security such as the legalisation of electronic documents. In addition, there is a review of the previous approaches used to tackle the problem.

Chapter 3 proposes a solution to fill the gap, identified from the literature review, and gives the theoretical background on which this proposal is based. A framework is proposed to answer the research question. All changes to the framework are explained, and the final version of the framework presented.

Chapter 4 describes the methodology and principles of the mixed method that is used in this research.

Chapter 5 presents the findings of the methodology used to confirm the framework components. These are discussed and used to modify the initial framework.

Chapter 6 models the proposed framework using formal methods. A behavioural event-driven model is used to give abstract view of the states in the framework.

The work concludes with conclusions and possible future work.

1.3 Contribution

The main contribution of this work is to introduce a general conceptual framework to secure documents when they leave the originating organisation. Document security is a critical issue for organisations in all the domains surveyed. However, some domains are not serious enough in their endeavours to ensure their documents are safe. Others do pay attention, but cannot share their documents with others due to incompatibility or uncertainty. Unsecure document sharing is a euphemism for information leakage. These challenges are identified in the literature. The triangulation method is used to confirm these findings and to explore any overlooked issues.

Chapter 2: Literature Review

This chapter looks at the history of document security and the concept of "Active Document". The word "Active" in this context can be defined as the ability of a document to participate in making a decision regarding the operations (view, edit, amend and delete) that are performed on it. However, there are other interpretations for this word when it used in web technology.

2.1 Security

To understand what is meant by security in documents or document security, here is a brief review of security literature.

Security is defined as the "condition of being protected against danger or loss. In the general, it is a concept similar to safety" with "an added emphasis on being protected from dangers that originate from outside", that "something not only is secure but that it has been secured" (Pfleeger et al. 2006). The Federal Standard 1037C (Littlefield & Rowman 1997) defines security as:

"1. A condition that results from the establishment and maintenance of protective measures that ensure a state of inviolability from hostile acts or influences.

2. With respect to classified matter, the condition that prevents unauthorized persons from having access to official information that is safeguarded in the interests of national security.

3. Measures taken by a military unit, an activity or installation to protect itself against all acts designed to, or which may, impair its effectiveness.

4. The combination of confidentiality, integrity and availability.

5. The protection of computer hardware, software, and data from accidental or malicious access, use, modification, destruction, or disclosure. Tools for the maintenance of security are focused on availability, confidentiality, and integrity."

In other words, security can represent physical security in the traditional way, Information Technology (IT) security, or a combination of both. This research concerns IT security.

2.1.1 Security goals

A secure computing system is one that achieves the right balance among the three security goals: confidentiality, integrity and availability (CIA). Confidentiality, sometimes referred as secrecy or privacy, ensures that "computer-related assets are accessed only by authorised parties" (Easttom 2012). Integrity means that "assets can be modified" only when they are under authorised control, while availability means that "assets can be accessed by those authorised at appropriate times" (Pfleeger et al. 2006).

The relationship between these three security goals can be summarised as:

- Increased protection of confidentiality can limit availability and affect integrity.
- Enhanced integrity will decrease confidentiality and limit availability.
- Wide availability increases the risk of compromising integrity and confidentiality.

In order to understand how security controls are applied within and outside the organisation, the hardware, systems, software, data, communications links, and role of personnel, will now be reviewed.

2.1.2 Data Security

Data is secured when it kept safe from corruption and access to it is appropriately controlled. Thus data security services ensure privacy. It also helps to protect users' personal data.

In the digital world, the way in which data is secures has changed. "In the old days [sic], data security and privacy were easily provided by storage in a locked box or file cabinet. Conversion of such records into digital data in databases on local and wide area networks markedly increases the provider's exposure to liabilities" as described by Albisser et al. (2003).

Data security also ensures that users are treated equally, and this is the responsibility of the legislation authority. For example, in the UK, the Data Protection Act 1998 "is used to ensure that personal data is accessible to those whom it concerns, and provides redress to individuals if there are inaccuracies. It states that only individuals and companies with legitimate and lawful reasons can process personal information and it cannot be shared" (UK Parliament 1998). In the global context, the International Standard ISO/IEC 27002:2005 covers data security and the Code of Practice for

information security management, and "is intended as a common basis and practical guideline for developing organizational security standards and effective security management practices. One of its cardinal principles is that all stored information/data, should be owned so that it is clear whose responsibility it is to protect and control access to that data".

Encryption is one of the well-known and effective ways to secure data. The most common cryptosystems are the Data Encryption Standard (DES), Advanced Encryption Standard (AES), and RSA algorithm. Cryptography has many techniques in the digital environment including hash functions, key exchange protocols, digital signatures, and certificates (Pfleeger et al. 2006).

2.1.3 Database Security

Protecting data is the core function of all secure systems. In many cases, this task is surrendered to the database management system (DBMS). The known requirements of database control include "physical database integrity, logical database integrity, element integrity, audit ability, access control, user authentication, and availability" (Pfleeger et al. 2006).

Databases may contain sensitive user data, and this sensitive data may also be subject to different levels of privilege which could conflict with access control.

Five main methodologies are employed to guarantee confidentiality in multilevel secure databases: trusted front end, integrity lock, distributed databases, commutative filters, and restricted views (Pfleeger et al. 2006).

2.1.4 Information Security

According to the National Institute of Standards and Technology (NIST), Information security is defined as "'protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction" (NIST 2014). Their core principles of information security are: confidentiality, integrity and availability. They categorize Information Systems into three sections: hardware, software, and communications, while their layers are physical, personal and organizational.

Many organizations from different domains, including governments, military, and businesses, collect confidential information about their employees, customers, and

products. The collected information is then “processed and stored on computers, and transmitted across networks” (Pfleeger et al. 2006). While the information may satisfy the organizations’ needs, the organizations are exposed to information leaks, unauthorised access and abuse of their data. According to the statutes and standards mentioned earlier (UK Parliament 1998; NIST 2014; ISO IEC 2005), protecting confidential information is now a legal requirement in many domains. In some cases, “information security has a significant effect on privacy, which is also viewed very differently in different cultures” (Pfleeger et al. 2006).

Basically, information security includes system authentication; logical and physical controls; non-repudiation; access control, information classification; cryptography and risk management.

2.1.5 Application/software security

Application security is the use of software, hardware, and procedural methods to protect applications from known external threats. Security controls built into applications minimize the likelihood that hackers will be able to influence the applications and access, steal, modify, or destroy sensitive data (Mao 2003).

The main goal of application security is to keep undesirable users from gaining access to application areas where they can access private and sensitive information or can corrupt data. Application security is not as simple as adding a username and password interface before using the application, but making applications secure involves cryptographic and encryption methods to protect sensitive data (Yoder & Barcalow 1998).

2.1.6 Computer security

Pfleeger (2006) has categorised the security flaws in computer development into two general classes: one is to compromise or modify data, and the other is to disturb computer services. They suggested countermeasures as: “development controls, operating system controls, and administrative controls”. The “development controls limit software development activities”; the operating system provides controls to limit “access to computing system objects”, while the “administrative controls limit the kinds of actions people can take”.

2.1.7 Network security

Network security consists of a primary computer network infrastructure, policies implemented by the network administrator to protect the network, and defending network-accessible resources from unauthorised access (Majchrzak & Usener 2011).

Network accessible resources include the network infrastructure, applications programs, and data. The strongest network security measures are firm authentication, access control, and encryption. Three controls are specific to network security: firewalls, an intrusion detection system, and secure e-mail (Jr & Coley 1999). A common method for assessing network security is a system for adaptive network security, which uses a network vulnerability-scanning assessment (Gleichauf & Randall 2001). Boyle has suggested a device and method that provides multi-level security for communications among computers and terminals on a network (Boyle et al. 1996), while Hershey proposed a system and method utilising a parallel finite state machine adaptive active monitor and response (Hershey et al. 1995).

2.1.8 The Human factor in security

Most computer-based security breaches are caused by either human or environmental factors (Liu et al. 2009). Zeadally et al. (2012) classifies them thus:

1. The administration of security (for example security planning and risk analysis).
2. The economics of cyber security (for example the cost/benefit analysis of investing in security).
3. The privacy policy and usage of the collected data, and the compliance with law and ethics that control malicious activities.

2.2 Information security

Information security is the most dynamic and challenging field in computer science. It is generally accepted that there is no such thing as a perfectly secure system to protect information, but there is a trade-off between risk and cost. Information security tries to ensure confidentiality, integrity and availability, and has been addressed for long time.

2.2.1 Confidentiality

This is the ability to restrict or control access to certain information to authorised users only. Some principle like “Least Privilege” is used to make sure that only users who have a true need for this data are able to access it. Losing confidentiality exposes the information to the risk of losing privacy, identity theft, or unauthorised access to that information. Access control, authentication and encryption, are some of the technologies used to enforce information confidentiality (Whitman & Mattord 2011).

2.2.2 Integrity

This is the act of assuring that the information is accurate and reliable and it has not been tampered with by authorised access or an unknown entity. It mainly covers authenticity, accountability and non-repudiation. Authenticity is the ability to prove that the information has not been changed in an unauthorised manner, while non-repudiation is the ability to record every action on the information, both sending and receiving, to prevent tampering and fraud. It does not guarantee the delivery of the information. Finally, accountability is the ability to link the user to every information action recorded as time, access level, and method used to perform the action (Easttom 2012).

2.2.3 Availability

This ensures that the information and other crucial assets are always available when needed by the user. The loss of availability occurs not only when a natural disaster happens, but may be as simple as delayed access to information or denial of access arising from a hacker attack. This may result in business disruption, revenue loss and loss of customer trust. Risks such as loss of privacy, fraud, information no longer being reliable, and loss of user confidence, are what security research is addressing (Stamp 2011).

2.3 Document security

In a digital world, documents could range from business-related to a private letter sent to a family member, although the focus here is on the Business-to-Business (B2B) and Government-to-Government (G2G) domains, where the scenario is multi-sites that need to exchange documents between them securely (Smallwood 2012). A wide range

of security aspects should be investigated to mitigate the chance of security breaches. Such aspects as policy, privacy, access control, trust management, and document legalisation, have been researched previously to introduce a model, framework or system which has the ability to cover as many of these aspects as possible. These aspects are highly interconnected and sometimes it hard to draw a line between them.

2.3.1 Access Control

Access control is another crucial aspect related to securing documents within any enterprise. The three current policies for implementing access control are: Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role-Based Access Control (RBAC). Despite a lot of research one these, none of them provides an invincible solution against improper usage of a document.

DAC, introduced by Lampson (1974), may be the most primitive of the access controls. It employs a usage matrix that aims to control who can and cannot perform actions, and what actions are allowed for each user. A lot of research has been done on the best mechanism to implement this policy, but the concept remains the same. However, DAC faces real problems in overcoming the Trojan horse threat (Samarati & Vimercati 2001).

MAC tries to solve the problem by using a Multilevel Security Policy. MAC distinguishes between the authenticated user and the processes which request access on their behalf, to provide more control on leakage and modification. The user at a given security level can only read a document from a higher security level without any right to modify it, but the same user can read and modify a document from a lower security level. Therefore, any process from that user that tries to modify a document at a higher level will be denied. While that solves the problem of unauthorised processing, it puts a major overhead on the system controlling the process. This overhead allows the system to be exposed to Denial of Service (DoS) attacks, when an attacker floods the system with lower priority processes and prevents the most important process from being granted access (Samarati & Vimercati 2001).

In the case of a large organizations, the access control policies mentioned above failed to provide proper solutions. The Role-Based Access control provides a flexible way to reflect the real-life context of the organisation. In most situations, the user's *role* in the organisation is more important than the user's *identity*. The RBAC groups and names the privilege and the users are then assigned to these groups in a way that fits naturally

with the organisation's real structure. The RBAC has been further developed since it was first introduced (Baldwin 1990). Some of the many advantages are: authorisation management, hierarchical roles, least privilege, and separation of roles. However, this mechanism faces serious threats from social engineering and has some limitations. For example, when the policy is role-oriented, it may prevent some users viewing data that is essential for their work, such as when a manager delegates their secretary to draft a report on their behalf.

However, another approach to force security policies is by adopting a complex mix of all the previous mechanisms, and this requires a unified control language suitable for reading by both the system and humans in different domains, such as eXtensible Mark-up Language (XML). Recently, XML has played a great role in access control mechanisms, and much research has taken place (Choi & Yeo 2012);(Abiteboul et al. 2008).

2.3.2 Document Legalisation

Electronic documents need to be legitimate (authenticated) so they can carry full legal status. Thus arises the need for a digital signature to replace a real signature. The digital signature requires an authority that guarantees the integrity of the signature and maintains its security. This authority may vary from one domain to another, as well as from country to country. This is one of the main reasons that the digital signature has such limited usage in international scale (Schmidt & Loebl 2005).

2.4 Current Document Security Technologies

The ultimate aim is to provide security for the document's entire lifecycle. This means that the document is secure from being created, through its use, until it has been marked as obsolete/archived. All the actions performed on that document throughout that period should be monitored, audited and controlled. This sounds similar to Information Governance (IG), but document security is only part of Information Governance (Smallwood 2012).

2.4.1 Electronic Document Management System

This is software defined as "the application of technology to save paper, speed up communications, and increase the productivity of business processes" (Sprague 1995), and is known as Document Management System (DMS). The DMS controls the lifecycle of documents in the organization—how they are created, reviewed, and published—and how they are ultimately disposed of or retained. The DMS has specific tasks in order to be accredited as effective DMS (Kofax 2014), which are:

1. Control document type; template and metadata to be created.
2. Where and how to store and access the document inside the organisation.
3. How to transfer the document between employees inside the organisation and what policy to use to control access to related documents.
4. How the document is converted from one stage to another inside the organisation. The legal requirement of how it is stored in the organisation record.

2.4.2 Electronic Records Management

ERM is software that manages all the enterprise's business documents and records, paying no attention to whether it is in digital or physical form. So electronic records like word, video, voice and e-forms are stored along with DVDs and CDs (Smallwood 2012). ERM provides a disposal policy, so it can set up a retention schedule to destroy selected data according to the organization's policy. During document retention, the ERM keeps the document from being altered in any way and even considers encryption of the document as a modification.

2.4.3 Enterprise Content Management

ECM is software that manages the enterprise's documents including web content, electronic documents and business records. The ECM is sometimes called a Content Management System CMS (Smallwood 2012). The ECM keeps only one copy of each document that is up-to-date, and can be accessed from the entire enterprise according to the access policy. Theoretically, ECM can manage any kind of content; however, it focuses on unstructured content (anything not database). Structured data is usually stored as reports or graphs as unstructured content. ECM allows the users to retrieve the latest version of any document very simply and provides excellent metadata control. Therefore, any user with an older version of the document can check against the latest version of that document from the ECM.

2.4.4 Data Loss (leakage) Prevention

DLP is a software and hardware combination that is designed to prevent any data leaving the organisation without inspection. If the data is sensitive or the user does not have the privilege of transmitting it elsewhere, the DLP system will terminate the transfer (Smallwood 2012). The DLP filters the document depending on its content, key words, and author. DLP uses cauterised management framework and deep content analysis to perform document security in use, transfer and storage.

2.4.5 Digital/Information Rights Management

IRM (or Enterprise Right Management, ERM) is the process of using a security wrapper to encapsulate any data in order to secure it when in use, transfer or storage (EMC Corporation 2008). This type of securing document is called persistent security. IRM provides granularity to file-level security, regardless of the file location inside or outside the organization (ho Eom 2012). Different vendors could provide IRM and it can be integrated with the existing authentication system, as shown in Figure 1.

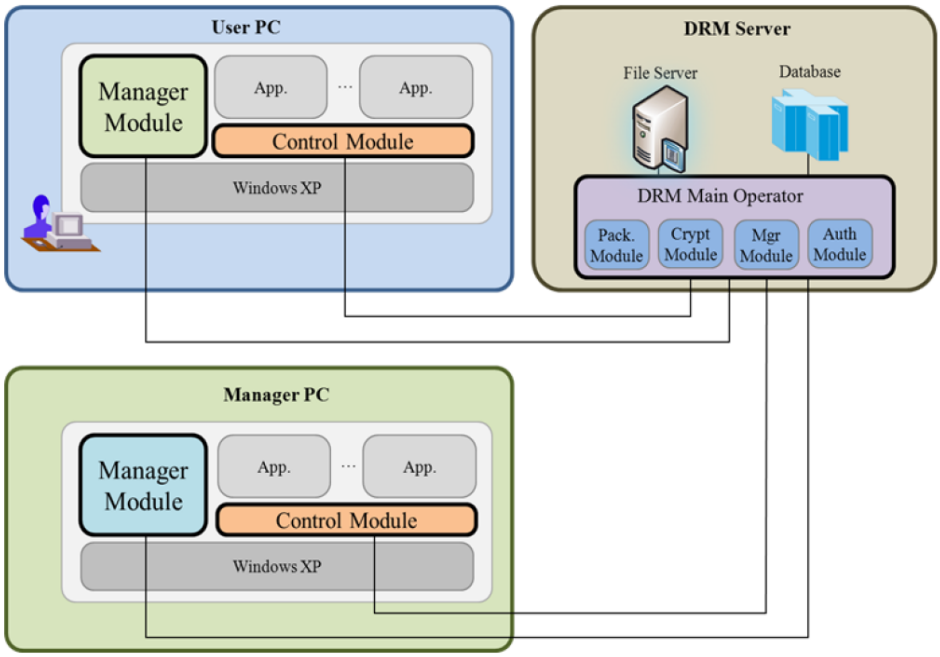

Figure 1 Document DRM architecture (after ho Eom 2012)

The IRM can utilise many technologies to secure the information, for example Content identification, Digital watermark, Digital fingerprint, and Encryption (Tassel 2006).

2.4.6 Issues with the current security technology

The document security technologies mentioned above have some limitations, listed below.

Table 1 Issues with the current document security technologies

Technology	Issues	Reference
DMS/EDMS	• Secures the document inside the organisation only. • It is not designed to manage cooperative work.	(Sprague 1995)
ERM	• Secures the document inside the organisation only. • The user can keep an offline copy so when the system performs its retention schedule on the original copy, the user's copy still exists. • Cannot perform encryption on stored records since the ERM considers it as a modification.	(Smallwood 2012)
ECM/CMS	• Secures the document inside the organisation only. • Once the user gets access to the file, there is no way to monitor or control what action he can perform on the file. • Any user can keep a copy of the document.	(Smallwood 2012)
IRM/ERM	• Compatibility issues, for instance MS-IRM is more compatible with MS Office files than other documents and files. • Usually depends on existing user authentication and management. • Does not have the same level of protection for all document types. Doing things right is not doing the right things.	(EMC Corporation 2008) (Manasdeep 2012)
DLP	• Secures the document inside the organisation only. • Cannot control encrypted files. • Cannot control privileged users.	(Smallwood 2012)

2.5 Public Key Infrastructure

Public-Key Infrastructure (PKI) is defined as a "set of hardware, software, people, policies, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates" (Toorani & Shirazi 2008). Another definition of PKI is a process created to enable users to implement public key cryptography, usually in a large setting as described by Pfleeger (2006). They noted that PKI would offer its users a set of identification and access control services, which would include creating certificates

correlating the user's identity with the public key. Certificate management is another service provided by PKI to perform signing certificates; confirm/deny if a certificate is still valid; revoking user certificates if withdrawn or if the user signing key has been compromised (Pfleeger et al. 2006).

PKI contains of the following components (Faraj Al-Janabi & Abd-Alrazzaq 2011; McKinley 2000; Vacca 2005; Toorani & Shirazi 2008):

- Registration Authority (RA): a body that verifies public keys and the identities of their holders before binding, ensuring keys meet the international standard.
- Certification Authority (CA): binds public key to the identities of their owners; responsible for issuing and revoking of Public key certificates.
- Validation Authority (VA): an agency that provides information on behalf of the CA.
- Public key certificate: a document that is signed by a CA certifying the accuracy of the binding of a public key and its owner's identity.
- Certificate Repository (CR): stores Public key certificates and Certification Revocation Lists (CRLs).
- Central Directory: a secure location for store and index keys.

The process of PKI has been outlined in the literature (Trcek 2006; Toorani & Shirazi 2008; Vacca 2005), and consists of the following steps:

1. A user applies to the RA for a certificate with his public key;
2. RA verifies and confirms the user's identity to the CA;
3. CA signs the public key certificate with CA's private key and issues the certificate to the user;
4. CA also sends information about issued certificates to VA;
5. The user can now sign electronic documents with his private key and attach the Public Key Certificate to the electronic Document;
6. The integrity of the electronic Document can then be verified on access and the user's identity can be checked by the VA on behalf of the CA.

PKI provides a hierarchical trust structure (Pfleeger et al. 2006): through PKI, a chain of CAs can be traced to find a trusted note from the signer's public key certificate, such that not only can the signer be tracked down, but also the CA, and the CA's CAs, all the way to the root CA. PKI are thus "trusted services that enables the secure transfer of information and supports a wide variety of E-Commerce applications" (Yeun & Farnham 2001). They also pointed out that a properly implemented PKI can provide

"Confidentiality: communications between two parties remain secret; Integrity: no unauthorized modification of information between two parties; Authentication: the process of reliably determining the identity of a communication party; and non-repudiation: impossible for communicating parties to falsely deny".

2.6 Formal Methods

An overview of formal methods is presented here, followed by a detailed explanation of Event-B (Russo 2011) and its toolset Rodin (Abrial et al. 2006). The main concepts of Event-B are described with the specific importance of proof obligations. The Rodin platform is introduced to provide a practical implementation of using Event-B.

2.6.1 Formal Methods

Mathematical techniques are important in all mature engineering disciplines. However, they have not been used heavily in software engineering (Woodcock et al. 2009). Discussion about their use and significance has attracted huge attention and is still doing so (Liu et al. 1997). Two schools of thought on this debate are:

- formal techniques deliver remedial and complete solutions to problems associated with system development.
- formal methods have little use or benefit to the development process.

Developing formal tools to reason about systems is certainly a challenging task. One view of the components which any formal method should contain is:

- A semantic model is defined as the mathematical structure where terms, formulae and the rules used, are given a precise meaning. The semantic model should reflect the underlying computational model of the proposed application.
- A specification language is the code with which systems and their activities are described. The specification language must have an appropriate semantics within the semantic model.
- Verification systems/refinement calculi are the mathematically sound rules that allow the verification of system properties and the stepping between specifications and implementations.
- Supporting tools, such as proof assistants and syntax and type checkers, are important for the formalism to be of any practical use.

According to Liu et al. (1997), a formal method should have clear development guidelines to facilitate its integration with development processes. The aim of this work is to enhance the existing Event-B verification system (by means of proof extensions) and specification language (by means of language extensions).

Challenges

Despite the availability of many formalisms and their supporting tools, there are many difficulties facing the integration of formal methods into the development process of computer systems. Real problems stem from the very nature of formal methods and computer engineering, some which are outlined below (Abrial 2007).

- Formal methods require computer engineers to think carefully about the system in question before proceeding to the coding stage. This is not helped by the fact that engineers "postpone any serious thinking" during the specification and design phases (Abrial & Hallerstede 2007), and accommodate a rather long and resource-hungry test phase.
- It is quite difficult to change the current development process. Within the industry, managers are reluctant to change the traditional way of approaching projects unless clear value will be gained.
- Modelling is not a simple activity as it is often accompanied by reasoning (Abrial 2007). A clear distinction between modelling and programming should be maintained, as the initial model of a program specifies the properties against which the final program will be evaluated.
- One of the main objectives of modelling is the ability to reason formally. Software engineers are not accustomed to this practice.
- One obstacle is the lack of attractive tool support, to make modelling and reasoning a seamless addition to the development process. This is one of the main selling points of Event-B and Rodin (Butler & Hallerstede 2007).

Classifications

Despite the difficulties and misconceptions that surround formal methods, important effort has been spent designing and implementing formal systems and tools that benefit from the rigour that mathematics offers. These formalisms can be organised into five categories (Liu et al. 1997):

- Model-based approach: A system is modelled using discrete mathematical structures to describe its properties. Operations describe the transitions between different states. This approach does not explicitly represent concurrency. Non-functional requirements (e.g. temporal requirements) can, in some cases, be expressed. Notable examples of this approach include Z (Woodcock & Davies 1996), the B Method (Abrial 1996), and VDM (Jones 1995).
- Logic-based approach: Logics are used to describe system properties including probabilistic and temporal behaviour. The axiomatic system can then be employed to validate system properties. In some cases, the logic can be extended with concrete programming constructs to provide an implementation-oriented language. Notable examples of this approach include Modal Logic (Goldblatt 2003) and Temporal Logic (Galton 1987).
- Algebraic approach: In this approach, an explicit definition of operations is given by axiomatically linking the behaviour of different operations without defining states. Algebraic formalisms, similar to model-based formalisms, do not provide an explicit representation of concurrency. A notable example of algebraic formalisms is OBJ (Goguen & Malcolm 2000).
- Process Algebra approach: CSP (Hoare 1978) and CCS (Vaglini 1991) are notable examples. The π-calculus (Strnadl 2006) is a formal approach to modelling mobility within concurrent systems. Concurrent processes are formally represented, and system behaviours are described as "constraints on all allowable observable communication between processes" (Strnadl 2006).
- Net-based approaches: Graphical notations with formal semantics are used to describe systems. Petri Nets (Reisig 2003) are a notable example.

2.6.2 Event-B

Here is a brief account of Event-B, describing what is meant by discrete systems, which are the subject matter of Event-B modelling.

Discrete Systems Modelling

Complex systems are made of many inter-related components that interact with an external environment. Although these systems often exhibit continuous behaviours, they appear as discrete traits most of the time. This essentially means that they can be abstracted using a discrete transition model. There could be many of these transitions,

but that does not change the nature of systems that are intrinsically discrete (Abrial & Hallerstede 2007).

A discrete model consists of a state which can be represented as variables. The choice of variables will depend on the level of abstraction of the model with regard to the real system. As in other applied sciences, there will be certain laws that govern the state of the model including its type. Such laws are referred to as *invariants*.

A discrete model can be subject to a number of transitions, or *events*. Each of these events has a *guard*, which is the condition under which the event is allowed to take place. Furthermore, each event has an action associated with it. The action describes the effect that the occurrence of the event has on the state of the model.

In the discrete modelling of complex systems, it is assumed that the execution of events takes no time (Abrial & Hallerstede 2007). When no event is allowed to occur (guards of all events are false), the execution of the model stops and is said to have deadlocked (Abrial & Hallerstede 2007). If many guards are true, only one event is allowed to occur. The choice of the event to occur in the latter case is non-deterministic.

Event-B Modelling

Event-B is a formalism for discrete system modelling based on the B method (Abrial 1996). Event-B modelling is carried out using *first-order predicate logic* with *equality* and *set theory*. The approach provides facilities to reason about models using *proof obligations*. These in turn implicitly represent the *semantics* of Event-B models (Hallerstede 2011). A brief descriptive account of Event-B modelling is given below, but a formal description is available (Abrial & Hallerstede 2007).

An Event-B model consists of *contexts* and *machines*. Contexts represent the static aspects of the model whereas machines describe its dynamic aspects. Figure 2 summarises the anatomy of Event-B models.

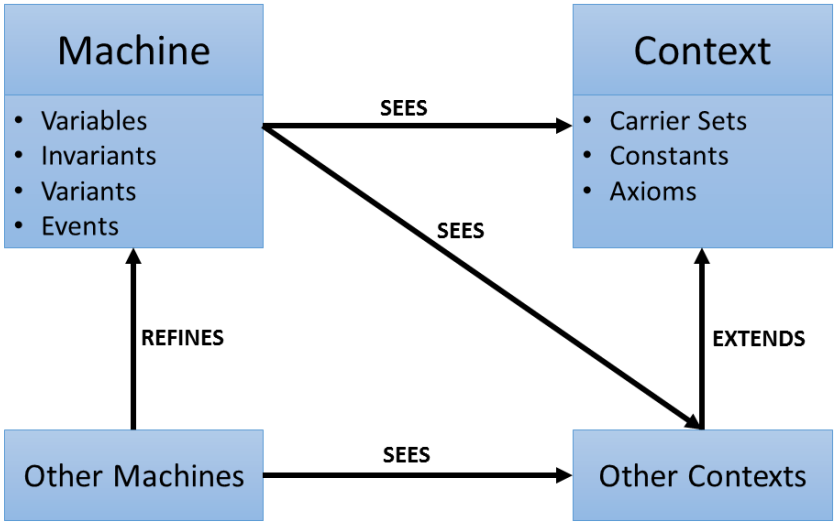

Figure 2 Anatomy of Event-B Models

2.6.3 The Rodin Platform

The Rodin platform (Abrial et al. 2006) is an integrated modelling environment for Event-B. It provides facilities and tools to develop and reason about models in a reactive manner, inspired by modern integrated development environments (IDEs) such as Eclipse (Platform 2013). When developing Java programs using Eclipse, the user is not required to initiate the compilation process. Rather, the IDE reacts to changes in code in a seamless manner, which provides effective feedback to the developer. Analogously, in Rodin, while developing a model of a complex system, static checking, proof obligation generation, and management, are carried out seamlessly to provide immediate feedback to the modeller. The combination of static checking and proof obligation generation in Rodin can be thought of as an extended static checker (Detlefs 1995) for Event-B. More precisely, the Rodin platform provides the capability to:

- develop models in Event-B by specifying contexts and machines,
- analyse models by means of static checking including syntax and type checking,
- semantically analyse models by means of proof obligations generated as appropriate,
- carry out a mathematical proof in order to verify model consistency.

In order to strike a good balance between usability and effectiveness, Rodin is designed to satisfy the following requirements (Butler & Hallerstede 2007):

- Design-Time Feedback: the tool responds quickly to changes and provides feedback that can be easily related to models;
- Distinct Proof Obligation Generation and Verification phases: the tool decouples modelling and proving while maintaining the link between the two activities (i.e. traceability) in case the automatic proofs fail.

2.6.4 Event-B Mathematical Language

Figure 3 shows an example of a simple context. Context C0 defines a constant *minimum*. The first axiom asserts that minimum is a partial function from the set of sets of naturals to the set of naturals. The second axiom ensures that minimum associates non-empty sets of natural numbers with their least element using the usual ordering, <, on natural numbers. The syntax used to write Event-B models can be divided into two levels:

1. Outer Syntax: the level of syntax that corresponds to the unboxed parts of the context definition in Figure 3. This syntax is used to identify the components of specific contexts and machines.
2. Inner Syntax: this level of syntax relates to the boxed parts in Figure 3. This syntax is used to identify the mathematical formulae for invariants, axioms, guards and actions.

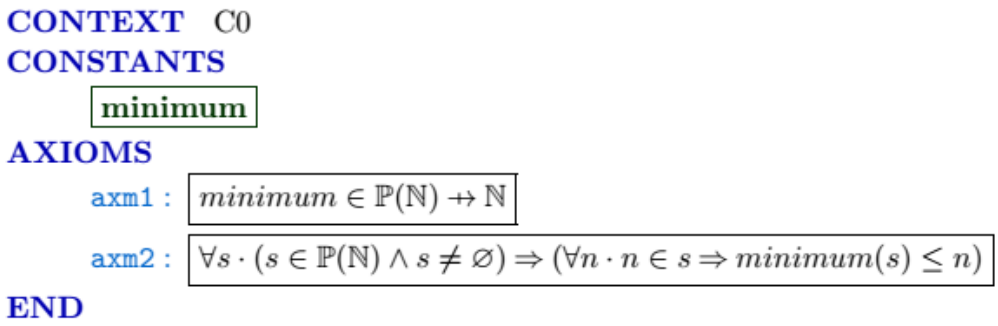

Figure 3 Example of Event-B Outer and Inner Syntax

2.7 Active Document

The concept of Active Document was first introduced by Quint & Vatton (1994). They suggested that a document could be *active* when the document has a set of features in addition to its basic logical structure, while a *document manipulation system* has the ability to read those features using some mechanisms. They claimed the benefit of the active document concept was in cooperative editing and authoring of documents, such

as user interface and on document indexing. Most of the following research was focused on the same aspect (collaborative document authoring and editing) without addressing any security features.

LaMarca et al. (1999) were working for Xerox Palo Alto on document management systems, when they introduced document-centred collaboration. Earlier studies had failed to depend on extended document properties (Giampaolo 1999;Richter & Cabrera 1998). Their approach was based on separating the coordination information stored in the document, from the actual document data. As a result, the document carries its semantics within it, which can be read by middleware designed for that purpose. This middleware will read this semantic information and convert it to actions on the fly (without the need to open the file).

Their prototype project *Placeless Document System* explored the new features they proposed (Dourish et al. 2000). At that point, the phrase *Active Properties* was used to represent the semantic information stored inside the document by their middleware. These active properties extended the uniform document properties and metadata to represent not only structure but behaviour as well. This was followed by a paper which took Placeless system case studies as an example (Dourish 2003).

Nam & Bae (2002) introduced a framework for processing active documents in the business domain. In their terminology, an Active document is one that contains both business rules and data. Their framework focused on combining business rules and data for a web form on the client side, which then validates the business rules before triggering any event on the database server side. However, their proposed framework works on web forms with a DBMS back end, in particular Oracle. Moreover, they did not mention any security aspects.

Abiteboul et al. (2009) used the words “active document” again but to capture user interaction with Web 2.0 applications. They used the active document to add some semantic to the information available on the web.

Neumann & Lenz (2010) introduced a content-oriented workflow system for the medical care domain. They used the term “active document” to describe a software agent that reflects the user role in displaying information proper to their level of access. They adopted the same principle introduced by LaMarca et al. (1999), which mainly focused on the difference between the content and coordination information to facilitate easy use of their proposed workflow system *The alpha-Flow*. Todorova &

Neumann (2011) introduced a project that made use of the active properties in the previous *alpha-Flow* system, to build an auxiliary system that added more features to the main system in a distributed environment. Although they introduced a new point of view for the patient record in the medical domain, they failed to address the benefits of their system for security. Aspects of confidentiality, privacy, access control and integrity, are not covered in their system. Moreover, they tailored-built their system to health domain requirements and it is not guaranteed to work in other domains.

The most recent and security-related concept of active document was introduced by Munier et al. (2012). They proposed a new Enterprise Digital Right Management (E-DRM) architecture to secure files being shared among various parties over the cloud (main organisation and subcontractors or outsourcing entity). Data encapsulation was used to store security-related information (access control, audit, and metadata) as well as the *Security Kernel*, which is basically a piece of code to perform the security checks and decisions (Munier et al. 2012). To read the document, the user needs to have a *licence* explicitly describing their right to access the document and a *Trusted Viewer*. The trusted viewer could be a lightweight viewer embedded inside the document itself, or a heavy trusted viewer that uses the Application programming interface (API). Another way to view the document content is to export the data in eXtensible Mark-up Language (XML) to be displayed by any regular XML viewer.

Whichever way the document is viewed, the security kernel will accept or reject the user access, collecting and attaching metadata to the action, and finally calculating new data during the action. The document then stores the new data and waits to connect to a server, specialised for synchronisation purposes, to synchronise the new data (Munier et al. 2012).

Finally, the architecture tries to bind the management and security to one solution. While this binding works in the example provided, it may not work as a general solution. The collaboration part of the binding needs a server to synchronise the data between different versions of the document. This adds more vulnerability to the proposed system by bringing in cloud computing issues and communication threats.

Munier et al. (2014) used the same idea but this time emphasised the role of metadata. Each document contains metadata about the author, date, reviews, notes, and other editorial information. They make use of these metadata to automate the security and

collaborative editing in their model. Nevertheless, some of the information they extract from the metadata (for example geo-tagging information) could violate user privacy.

2.8 Summary

The literature review can be summarised into these points.

1. There is ongoing interest in making the document intelligent and as a result more interactive. But there is no agreed body of work in this regard, in any field (security, management and cooperative work, etc.).
2. Until 2012, the focus was to make use of active documents to facilitate easier workflow systems, without paying attention to security.
3. The only research that studied active document from a security prospective is Munier et al. (2012), but their approach is arguable in many respects.
4. There is always a worry about the user's and the organization's privacy. It is hard to draw a line between the information that the system needs to reveal and what is considered private data (Choi et al. 2013, p.128).

Chapter 3: Tamper-Proof Document proposal

The previous chapter concluded that the active document concept has gained attention from both practical and theoretical researchers, but that security has hardly been covered at all and certainly inadequately.

The most recent security-related research was conducted by Munier et al. (2012), whose architecture is arguable in several points. Starting with the document viewing mechanism, the architecture proposes a lightweight embedded trusted viewer, which seems to be an agent more than a viewer, and this agent may face some security threats. In its turn, the heavyweight viewer uses the API to communicate with the security kernel, which may contain security errors. The XML export option is not viable since the XML does not provide enough security measures to protect the confidentiality or the integrity of the data (Sun, Ramachandran, et al. 2017).

Another point is the licence file security, and the mechanism by which it is exchanged. The licence file was encrypted with PKI and could be broken if the outsourced entity did not have enough security awareness. The proposed architecture suggests that all the security-related information (access control information) is stored in the license file, which is basically an encrypted XML file. This makes the licence file the main target for an attacker since it may be the weakest link and is vulnerable (Wang et al. 2016).

A Tamper-Proof Document framework is proposed, based on the concepts mentioned in the literature. The framework novelty comes from the unique way of combining these concepts and utilising them in the security field. It derives from another conceptual frameworks like Chang et al. (2016).

Tampering with a document happens when an unauthorised access or modification is performed on the document. To prevent any tampering, the framework collaborates with the document itself. Using the active document concept facilitates collaboration. To defend the document against that tampering, the DRM/IRM concept is used.

3.1 Background theory

The Tamper-Proof framework comprises the concepts Active Document and Information Rights Management. The Active Document concept was first mentioned by Quint & Vatton (1994), who defined it as the “result of a combination of some

specific features in documents and some mechanisms in a document manipulation system".

By opening new possibilities, this concept makes the metadata of the document more useful, and hence it can change the behaviour of the presentation software. These new possibilities were used in the "Placeless Document" project, for collaborative document editing management (Dourish et al. 2000). The framework proposed here injects some pieces of information and programming code into the document file. The injected document will have the ability to provide more information without being opened, as well the capacity to perform simple tasks.

Digital/Information Rights Management provides continuous security for the document inside its wrapper. The wrapper usually replaces the actual document presentation software in order to control the operations performed on the document (copy, past, cut, delete and print). Moreover, it used to provide a secure channel for authenticating its users and enforcing the organisation's security policy, even outside the organization's firewall. The framework aims to convert each machine on which it is installed as a wrapper for the document, whether inside or outside the organisation (Chang et al. 2013). This facilitates the identification of each machine and user inside the organisation, and the privileges of that user or machine. The framework aims to integrate seamlessly with the existing presentation software, which interface the user is familiar with, to monitor the operations being performed by the user on that document. By doing so, the framework will enforce the security policy of the organisation with less effort and better granularity (Chang & Ramachandran 2016).

3.2 Initial framework functions

The proposed framework has three main components in order to perform correctly: presentation software integration module, operating system integration module, and document black box module. These components are expected to execute a group of functions that in total represent the main features of the framework. These functions are divided into document-side functions and system-side functions.

Document-side functions (Active Properties) focus on the following aspects: maintain extended metadata, environment detection, and document destruction. These functions are performed by the document black box module. System-side functions are mainly security and validation functions. Security functions are represented by

producing the document identifier number, the machine identification number, injecting active properties into the document, encryption and decryption document content, and control presentation software operations (Anwar, Inayat, et al. 2017). These functions are performed by both the presentation integration module and the operating system integration module. Validation functions are those that are responsible for enforcing the security policy on the document use, and are: testing the integrity of the active properties within the document, maintaining information about collaboration requirements, and monitoring the operating system file operations (delete, copy and cut). These functions are performed completely by the operating system integration module. Figure 4 shows a detailed view of the framework.

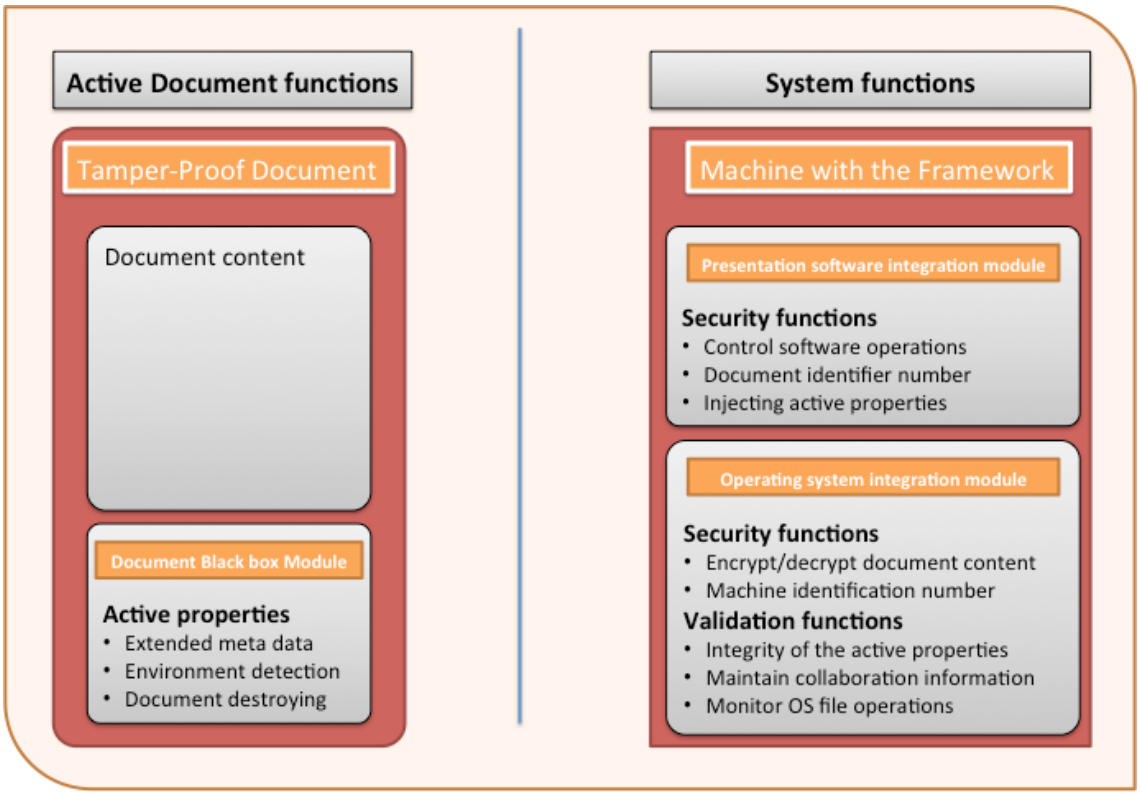

Figure 4 Detailed view of the Tamper-Proof framework

3.3 Initial framework components

Another concept adopted by the framework is the D/IRM. It provides continued security for the document inside its wrapper. The wrapper is used to replace the actual document presentation software, in order to control the operations performed on the document (copy, past, cut, delete and print). Moreover, it provides secure channels to authenticate the users and enforce the organisation's security policy, even outside the organization firewall. The framework aims to use each machine on which it is installed,

whether inside or outside the organisation, as a wrapper for the document. This facilitates the identification of each machine and user inside the organisation, and the privileges of that user or machine. The framework aims to integrate seamlessly with existing presentation software, whose interface the user is familiar with, to monitor the operation to be performed by the user on that document. By doing this, the framework will enforce the security policy of the organisation with less effort and better granularity. The proposed framework has three main components in order to perform correctly: presentation software integration module, operating system integration module, and document black box module. These components are expected to execute a group of functions that in total represent the main features of the framework. These are mainly divided into document-side functions and system-side functions.

1. The **Document black-box module** is responsible for providing Document-side functions (Active Properties). These functions mainly focus on the following aspects: maintaining extended metadata, environment detection, and document destruction.
2. The **Presentation software integration module** performs system-side functions that are mainly security and validation functions. These functions are: producing document identifier numbers, injecting active properties inside the document, and controlling presentation software operations (Anwar, Mohamad Zain, et al. 2017).
3. The **Operating system integration module** performs security and validation functions. Validation functions are those responsible for enforcing security policies on document usage (Sun, Liao, et al. 2017). These functions are: testing the integrity of the active properties inside the document, maintaining information about collaboration requirements, and monitoring the operating system file operations (delete, copy, and cut). The security functions are mainly machine identification numbers and encryption and decryption of document content.

3.4 Document access scenarios

The proposed framework uses these concepts to enable the document itself to have the functional ability to perform simple detection and irreversible tasks. The detection task is to examine the surrounding environment to check whether it has the framework installed or not. These capabilities along, with other information, are embedded in the document when it is created, or edited inside an environment that has the framework.

There are several use cases where there is a need for this framework, as explained below.

First case: standalone or self-standing documents. Large organizations and governments invest in the securing of documents within their networks, and transfer them between remote sites using PKI. In some instances, the documents are sent to insecure computers, basically for a user who is not allowed to view the document. The objective is to facilitate the securing of a document when it is sent from a secure network to another site regardless of the security level of the recipient.

Second case: domain independent file authorities. Most organizations use the common office document (Word document, Excel spreadsheet, PowerPoint presentation and PDF). These documents are presented in the same way whatever the organization domain, using one of the well-known presentation programs (Microsoft Office, Open Office, iWork, or LibreOffice), while the security and management of the document is actually different. The objective is to find a solution that integrates with existing presentation software and is domain independent.

Third case: the trade-off between the security and availability of the document. The cost of leaked information is increasing dramatically. Even with the latest expensive technology implemented inside the organisation, information leakage still exists when the organisation wants to share its information with other organisations. What if the second organisation depends on the first to provide it with security mechanisms? This may limit the availability of the shared document but does mitigate information leakage.

When a document is transferred to another environment there are two main scenarios. In the first scenario, if the environment does not have the framework then the file performs the simple irreversible task (alteration of document characters or displacing document layers). In the second scenario, if the environment has the framework, the latter will use some information implanted in the file to automatically enforce the organisation's security policy, while the document need do nothing. Figure 5 shows a flow of the expected scenarios.

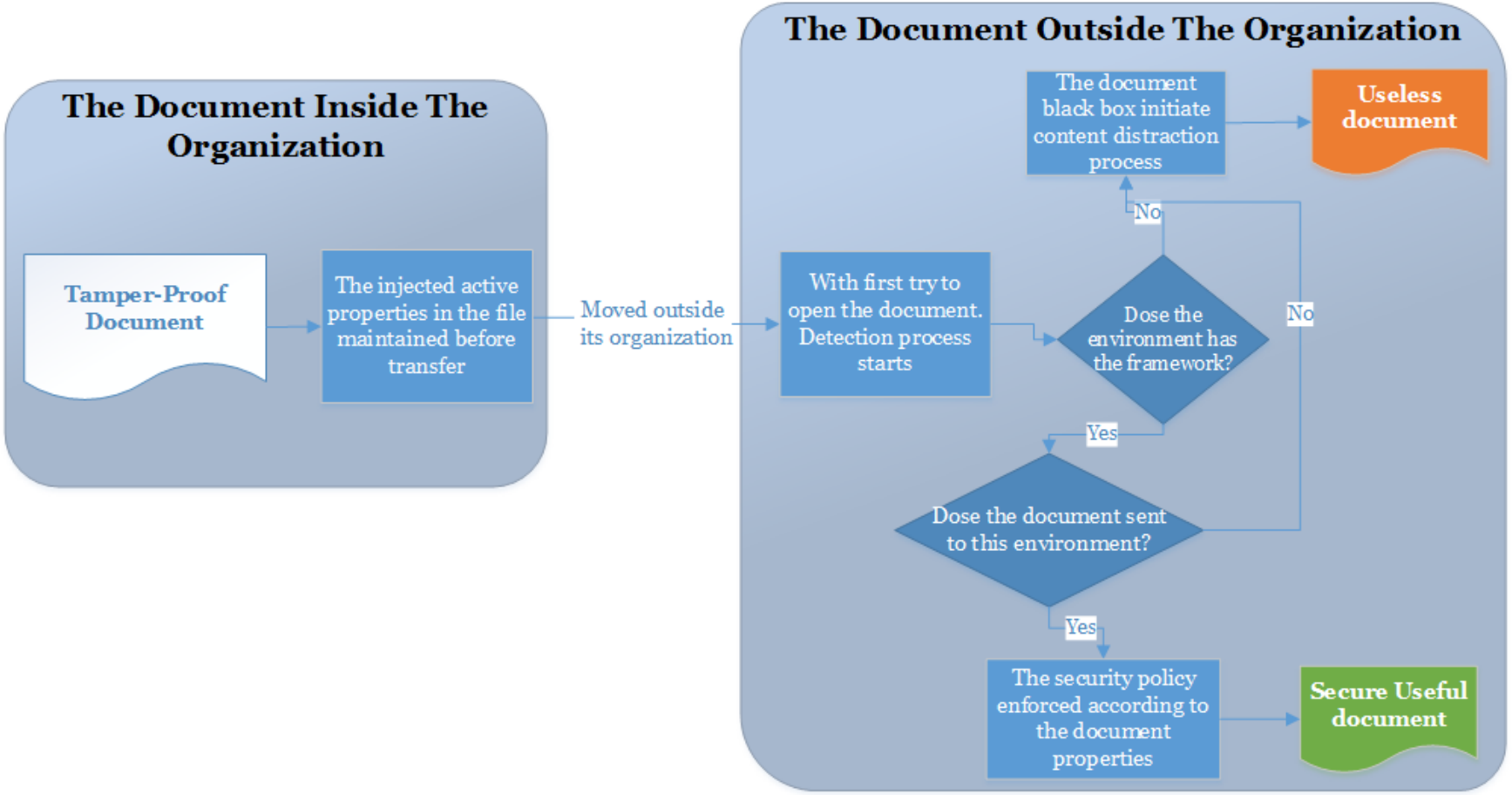

Figure 5 Document access scenarios when using the proposed framewor

Chapter 4: Research methodology

This chapter explores the main methods used to conduct research on security system verification. These methods are the ways that are used to collect and analyse data. There are three main methods used in this field; qualitative, quantitative and mixed methods. These methods are explained, and the mixed method was chosen for the reasons provided in the context. The complete methodology to tackle issues in securing documents in shared environments is shown in Figure 6.

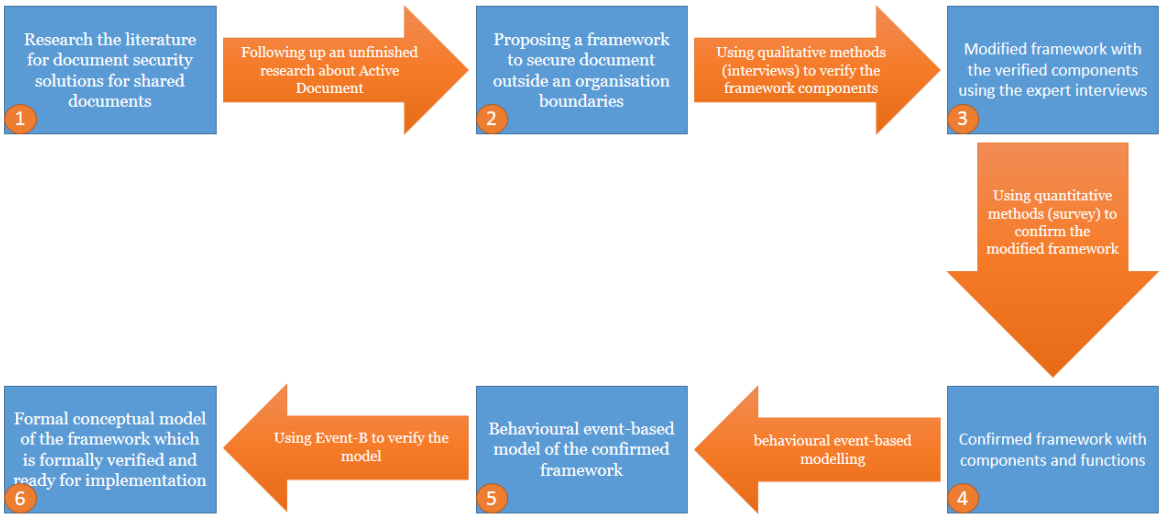

Figure 6: The methodology used in this research to answer the research questions

4.1 Research methods

There are many ways to conduct research verification. This section shows a list of the most used methods:

4.1.1 Quantitative Methods

Quantitative research is based on collecting and analysing data that can be represented in numbers. This research method is used when there is a developed theory that needs to be confirmed (Recker 2013).

Quantitative methods use questionnaires as a main approach to gather the required data. A questionnaire consists of a set of questions for gathering participants' responses in a pre-designed and standardised manner. Demographic data and users' opinions are the main data that is gathered using questionnaires. Their main advantage is that they can effortlessly be circulated to a large number of respondents (Preece et al. 2002).

Questionnaire responses can be structured or unstructured. Structured responses are easier to capture and analyse. There are five formats for structured responses (Bhattacherjee 2012):

1. Dichotomous response: choosing from two possible responses
2. Nominal response: choosing from more than two unordered responses
3. Ordinal response: choosing from more than two ordered responses
4. Interval-level response: choosing from a 5-point or 7-point scale
5. Continuous response: usually a blank space for the respondent to fill

Two different techniques are used for analysing quantitative data (Bhattacherjee 2012): descriptive analysis, where statistics are used to describe, combine and present the concepts of interest or show the relationships between these concepts, and inferential analysis, where statistics are used to test a hypothesis. Software tools such as SPSS can aid in this analysis.

In quantitative studies it is important to recruit a sample that statistically represents the population in order to generalise the findings (King & Horrocks 2010). This type of sampling is called random sampling, where participants are chosen randomly from a wider population (Recker 2013).

4.1.2 Qualitative Methods

Qualitative research methods involve collecting, analysing and interpreting data that cannot usually be shown in the form of numbers. They provide in depth understanding of a problem or situation. Hence, they are useful for exploratory research where a phenomenon is not well researched or is still developing (Recker 2013).

There are four main types of qualitative methods: observation, interviews, documents and audio-visual materials (Creswell 2007). The most commonly used

method is interviews. Interviews are described as “a conversation with a purpose” (Preece et al. 2002).

Interviews are categorised as: open-ended/unstructured, structured and semi-structured, depending on the amount of control the interviewer holds over the interview (Preece et al. 2002). The interviewer imposes control by determining a fixed set of questions prior to the interview. Another categorisation for interviews is based on the number of participants. They can be one-to-one or group interviews. Each of these categories has its benefits (Preece et al. 2002):

- Unstructured interviews: usually produce rich data since the interviewees are given the opportunity to mention things that the interviewer may not have considered.
- Structured interviews: are easier to analyse because the study is standardized. The same questions are given to each participant with a specific set of answers.
- Semi-structured interviews: use both closed and open questions and share features with structured and unstructured interviews.
- Focus groups or group interviews: allow diverse or delicate issues to be raised and usually involve between three and ten people.

In qualitative research, a large amount of data is produced and it is not always clear what parts of the data are relevant to the study. The most popular technique for analysing qualitative data is coding (Recker 2013; Creswell 2007). Coding is a way of categorising chunks of data by assigning labels or meaning to them. Data is usually organised around the core ideas or themes found in the study. These codes may be determined prior to data collection, or they may develop as the researcher is exposed to the data and broadens his perspective (Preece et al. 2002). Tools such as NVivo may be used to help researchers analyse and keep track of the data.

Due to the detailed and intense work required in qualitative research, it is necessary to limit sample size (Anderson 2010); sample size is not decided based on mathematical calculations. The most important factor for sampling in quantitative studies is to recruit a diverse sample that is able to enlighten the research topic (King & Horrocks 2010). This is called purposive sampling, where participants are chosen because they possess certain properties or expertise (Recker 2013).

4.1.3 Mixed Methods

As a response to the criticisms faced by qualitative or quantitative methods, a growing number of researchers are conducting mixed methods studies which explicitly combine both approaches (Recker 2013). Moreover, since qualitative methods are hard to generalise to a larger population (Recker 2013), quantitative methods can be used to confirm the findings and generalise them. Also, collecting different types of data from different sources by different methods helps develop a clearer picture of the problem being studied (Kaplan & Duchon 1988).

There are five major justifications for using mixed methods (Johnson & Onwuegbuzie 2004):

- Triangulation: where the findings of the study will be confirmed by using different methods to study the same problem
- Complementary: where the findings from one method will be used to elaborate and clarify the findings from the other method
- Initiation: where different methods are used to attempt to discover contradictions that will lead to reshaping the research questions
- Development: where the findings from one method will be used to inform the other methods
- Expansion: where different methods will be used to study different problems to expand the scope of the research

Triangulation refers to using two or more methods to investigate a problem. It may be used for three different purposes: to validate the findings of a study, to generalise the findings, and to get a better understanding of an issue (Barron 2006). Jick (1979) suggests that the use of multiple methods has the potential to reveal “unique variance” which may have been overlooked when applying a single method. Triangulation has four main forms (Barron 2006):

- Data triangulation: involves collecting data from different sources or people at different times.
- Investigator triangulation: involves the data being collected and analysed by different investigators or researchers to mitigate the subjective impacts of individual investigators.
- Theoretical triangulation: involves approaching data from different theoretical perspectives.

- Methodological triangulation: involves using different methods to collect and analyse the same data to compare the findings.

4.2 Research Methods Applied in This Research

This preliminary study applies a mixed methods approach to explore the effect of the proposed components for the framework. The mixed methods approach was chosen to strengthen the results of the study by validating the findings through triangulation (Kaplan & Duchon 1988). In the next section, a description is given of how the triangulation was performed and of the individual methods applied.

4.2.1 Triangulation

In order to refine and confirm the factors influencing the document protecting outside its organisation firewall, a methodological triangulation was performed. It involved combining and comparing data discovered from a detailed literature review, an expert review and a questionnaire survey (See Figure 7). The triangulation was performed in three stages, since each method should be applied independently (Jupp 2006). The results from each stage were then compared.

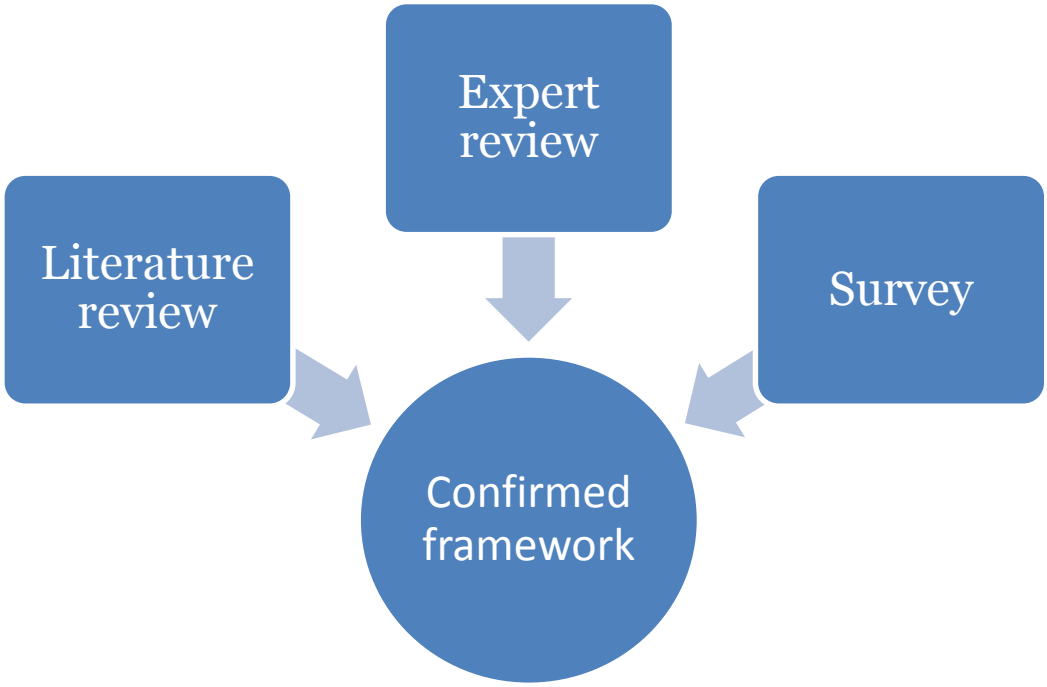

Figure 7: Mixed methodology using Triangulation to review and confirm the proposed framework

4.2.2 Methodology process steps

For the first stage, a thorough research review on related literature was conducted in order to initially propose the framework components. After that, interviews were conducted with experts to review the framework components, in order to improve the framework by adding, deleting or modifying the components. These improvements are described in detail later in the next chapters. Finally, an online survey was conducted to confirm the modified framework.

4.2.3 Expert Review

Interviews were used to conduct an exploratory and confirmation study, since there are some solutions that solve a part of the problem, and to discuss the proposed framework. The interview research method was chosen because it enables in-depth discussions and explorations.

The initial framework proposed from the literature review was reviewed by interviewing experts working in the cyber security field in the four suggested domains (Government, Healthcare, Academia, and Business). Experts were chosen for interview at this exploratory stage since the findings from a sample of experts have more credibility than findings from a sample that includes non-experts (Bhattacherjee 2012).

Qualitative studies usually depend on non-probability sampling where participants are chosen based on non-random criteria (Bhattacherjee 2012). In expert sampling, participants are chosen based on their knowledge in the area being studied (Bhattacherjee 2012). In this type of sampling, size depends on saturation (Guest et al. 2006). Saturation is reached when no new knowledge can be gleaned. Guest et al. (2006) suggest that saturation is usually reached by twelve interviews.

For the purpose of this review, sixteen cyber security experts were interviewed. A person is considered an expert if they have at least five years' experience of working on cyber security field within a particular domain.

The expert review was based on conducting semi-structured interviews with sixteen cyber security experts from four domains in the UK and Iraq. These experts were from government, healthcare, education and business domains. The interviews were conducted face-to-face, over the phone and online, based on the availability and location of each expert.

In the online interviews, the experts were asked to answer the questions and approached again for clarification when necessary. The two main objectives of these interviews were:

- To review the factors identified from the literature review conducted previously in order to improve the framework (by adding, deleting and modifying components).
- To identify additional factors that are unique to the domain or culture that may have been overlooked previously in the literature.

The semi-structured interviews included both closed and open questions. The closed questions were concerned with getting the experts' opinions on the factors in the proposed framework. The experts were also allowed to comment on these proposed factors. The open questions had the objective of identifying further factors from the experts that had not been identified in the desk-based study.

The interview questions were pre-tested on two fellow researchers at the University of Southampton. Based on this pre-test, it was decided that rather than showing respondents a diagram of the framework and asking their opinion, the respondents would be asked their opinion on each individual framework component and allowed to make further comments.

Prior to conducting the interviews, ethical approval was sought and obtained from the University of Southampton's ethics committee. The reference for the ethics approval is ERGO/FoPSE/13224.

4.2.4 Expert review questions design

The interview questions were designed to confirm the importance of each component of the framework. The questions were divided into three main parts; first: demographic and experience, second: general questions about security features and technologies used, third: confirming and exploring framework components. Answering these questions facilitate capturing the experts' knowledge about securing documents in different domains. This knowledge would help to confirm and modify the proposed framework components, ultimately, answering the research question:

What is an applicable framework for securing documents when they go beyond an organisation firewall?

These interviews help to answer two of the sub-questions of the main research question above, which are:

1. What are the security issues in documents used across different domains?
2. What are the security mechanisms that facilitate secure document sharing in a collaborative environment between two organisations?

The actual questions used in the interview are as follows:

First: Demographic questions:

1. What is your organisation domain?
2. Which of these roles fits your fits your job description?
3. How long have you been working in Cyber Security?

Second: General questions about security features and technologies used:

1. From your experience working on cyber security, what are the current security mechanisms that you are aware of? Mechanisms that are used to secure a document in an organisation?
2. From your experience working on cyber security, can you talk about the vulnerable and most frequent issues facing document security in an organisation that you have worked in or conjunction with to solve these issues?
3. From your experience working on cyber security, could you describe in detail the mechanisms or solutions that organisations use to secure documents outside their network firewall?

Third: confirming and exploring framework components:

The research so far proposes a framework with set of components, can you please confirm and elaborate on each of these components:

1. In your opinion, what is considered as human negligence in document leakage?
2. From your experience, are you aware of human negligence that commonly occurs in your domain?
3. From the previous question on human negligence, in your domain, what do writes in its IT security policy to prevent or mitigate human

negligence? In other words, if you were asked to write a policy/ recommendation, what would it be?

4. In your opinion, does your domain normally share documents with third party services?
    a. What is the security level of these shared documents?
5. In your opinion, what is the dependency level of your domain on third party services?
6. In your opinion, what are the common mechanisms that used in your domain to ensure safe use of documents at the third parties' infrastructure that reflects these documents' security level?
7. Can you tell me about current legalization policy that you aware of regarding authorizing and verifying document integrity?
8. Can you tell me about what are the most vulnerable and frequent challenges facing this policy implementation?
9. Now, (showing a diagram of the proposed framework and a list of components with brief descriptions about each one), can you please state your confirmation of these components functionality as described.

4.2.5 Survey

Questionnaires were chosen to confirm the updated framework resulting from the expert reviews. This approach was chosen for its ability to confirm and quantify the findings from quantitative research (Recker 2013). This approach is favourable because it is an established method for capturing unobservable data such as participants' opinions, can be used to capture data about a large population that cannot be observed directly, and allows respondents to respond at their own convenience (Bhattacherjee 2012).

In qualitative research, random sampling is employed which allows the findings of the study to be generalized to the population (Bhattacherjee 2012). Calculating random sample sizes is usually estimated mathematically based on preselected parameters (Guest et al. 2006).

In this study, G* Power software (Faul et al. 2009) was used to calculate the minimum sample size. The calculation was performed for a t-test to find the difference between a mean and constant. See Figure 8, from this calculation it was determined that the minimum sample size was 15.

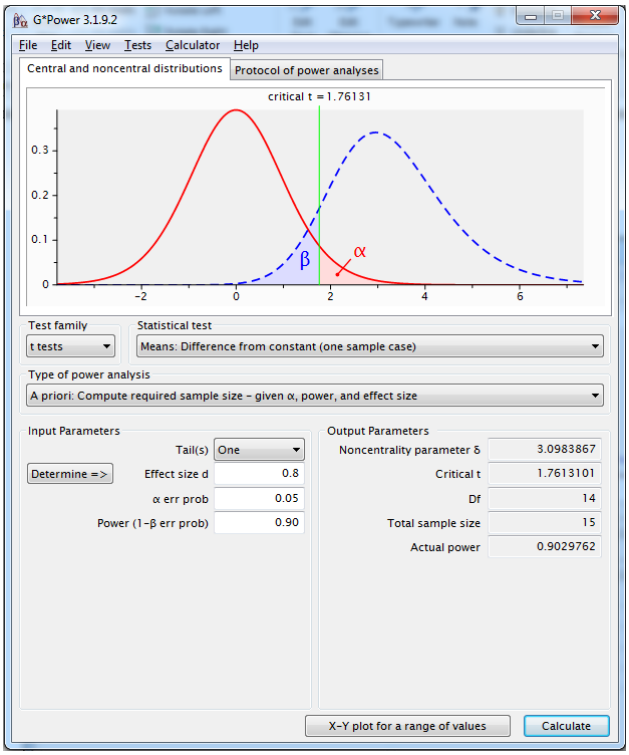

Figure 8: G*power Calculation for sample size

4.2.6 Survey questions design

The questionnaire is divided into two main parts. The first part asks a couple of nominal questions about the respondents' organisation type, demographic data, and experience to confirm their eligibility for this study. The second part was constructed using a five point Likert-type scale (Bhattacherjee 2012) with the following ratings: strongly agree = 5; agree = 4; neutral = 3; disagree = 2 and strongly disagree = 1. The purpose of the questions in this part is to confirm the proposed framework components to secure documents outside an organisation firewall. The University of Southampton's iSurvey application was used to generate the online survey. Prior to administering the online questionnaire, it was pre-tested by two computer science researchers at the University of Southampton.

This survey was performed by administering an online questionnaire to confirm the factors in the updated framework resulting from the expert review. Ninety participants complete the survey. These participants were reached via security groups in academia, business and healthcare. It was decided to administer the questionnaire online as this method is convenient for respondents. Respondents

were approached by email or via social media (LinkedIn) and asked to complete the online questionnaire.

Ethical approval was obtained from the University of Southampton's ethics committee. The reference for the ethics approval is ERGO/FoPSE/13224.

4.2.7 Data analysis procedure

There were two main analyses in this research for the collected data. The first was the theme-based analysis (thematic analysis) for the experts interview, which is usually used with qualitative methods (Boyatzis 1998). The second analysis method was statistical methods to confirm the modified framework components through sampling the population.

Thematic analysis was used to identify, analyse and report the themes within the collected raw data. The themes reflect outlines that exist within the collected data, and the outlines describing the occurrence. Therefore, thematic analysis is a method of organising and describing raw data in a manner that helps researchers identify important things to define in detail about their research questions (Braun & Clarke 2006).

For the qualitative data analysis, the raw data needs to be split into themes, and NVivo software was used for that purpose. Each participant was given a node, and each node had its own characteristics that reflect the demographic information. After that, data from the transcripts were assigned to related codes for each node.

For the survey analysis, a statistical error calculation method was used. Two types of errors are considered when calculating the minimum acceptable sample size (Bhattacherjee 2012); Type 1 (α) errors occur when rejecting a true null hypothesis and Type 2 (β) errors occur when a false null hypothesis is not rejected. The likelihood of these errors occurring can be reduced by increasing the sample size (Bhattacherjee 2012). By convention, α is set to 0.05 for a 95% confidence interval and $(1-\beta)$ is set to 0.9 for 10% of missing an association (Bhattacherjee 2012). Another parameter considered is effect size, which refers to the magnitude of the association between the predictor and outcome variables. Cohen (1988) defines three different effect sizes: small (d=0.2), medium (d=0.5) and large (d=0.8). In exploratory studies, the effect size is usually set at large (Cohen 1988).

4.2.8 Modelling the framework using formal methods

The triangulation used in this methodology resulted in confirmed components and functionality. The final part of the methodology was to step forward in implementing the framework. The implementation could be just for a proof of concept; however, any implementation is out of this research's scope. To specify the confirmed functionality in more detail, formal method modelling will be used. Formal methods are branch of mathematics that emerged to help design, analyse and validate software to be error-free and robust (Butler et al. 2004). Formal methods are based on fundamental mathematic concepts like sets and functions. This enables them to analyse each property of the software model.

The behavioural perspective modelling was used to model the dynamic behaviour response of the framework. Behavioural modelling has two main approaches; data-driven modelling and events-driven modelling (Kellner 1988). Events-driven modelling is commonly used to model events stimulated (triggered) that required the system to respond. In some cases, events have data associated with it. This modelling considers any events, regardless of whether they were internal or external events (Said et al. 2009). However, it assumes that the system has number of defined stages (states) and the events transfer the system from one state to another (Harel 1988). This type of behavioural modelling is the most suitable for our framework.

Formal Conceptual Modelling is a general approach to model real world systems beyond software functions. This approach uses human knowledge to make abstractions and refinements of a system (Vliet 2007). This approach is used in this research, and the Event-B formal method was chosen to model the proposed framework.

4.3 Summary

This chapter highlighted the research methods used in this study and the rationale behind these choices. A mixed method approach was used to review and confirm the proposed framework for securing documents outside the organisation firewall. This approach utilized a methodological triangulation of expert reviews and a questionnaire survey. The expert review was based on conducting semi-structured

interviews with sixteen cyber security experts from the four investigated domains (government, healthcare, education, and business). The findings from these reviews were used to develop the initial framework. An online survey was created; 113 cyber security professionals responded to the online survey but only 90 cases were valid. The results were used to confirm the reviewed framework.

Chapter 5: Findings and Discussion

There are previous studies and some solutions to secure a document when it leaves the originating organisation firewall. However, many issues and challenges are presented through the literature facing current solutions, as mentioned in section 2.4.6. This study consisted of an expert review to evaluate and identify framework components and a survey to confirm the identified components. In this chapter, the results from both the expert review and the survey are presented and discussed.

5.1 Findings from Expert Review

The expert review was conducted with sixteen experts in total. These experts had at least five years' experience in working on IT projects within one or more of the suggested domains. They were approached with an email containing an overview about the framework, the aim of the interviews and a consent form. The purpose of the expert review was to review the factors and components identified from literature and to identify further factors if there were any. A saturation point was reached from the tenth interview; after that, no new information were added, and the remaining interviews were only to confirm the findings. The reviews were constructed in the form of semi-structured interviews that the researcher obtained permission to record. There were twenty questions classified in four main categories (general questions, factors affecting document security, and framework components). The gathered answers were grouped and analysed as described in the next sections.

5.1.1 General questions

They are five general questions which work as references and an introduction. These questions aim to provide validation that the experts were selected randomly and match the experience criteria. These sixteen experts were divided as showed in Table 2. The interviews were anonymised and kept in a secure computer at the university. The experts' identity was kept anonymous by using codes as an alternative to names, as shown in Table 3.

Table 2: Expert distribution

Domain	UK	Iraq	Total
Local Government	2	2	4
Healthcare	2	2	4
Higher Education	2	2	4
Business	2	2	4
Totals	8	8	16

Table 3: Expert Overview

Expert code	domain	country	Job title
A	Healthcare	UK	Project Manager
B	Healthcare	UK	IT Manager
C	Healthcare	IQ	IT Department Manager
D	Healthcare	IQ	System Administrator
E	Education	UK	IT Manager
F	Education	UK	IT Manager
G	Education	IQ	System Administrator
H	Education	IQ	Computer Centre Manager
I	Business	UK	IT Manager
J	Business	UK	IT Manager
K	Business	IQ	System Administrator
L	Business	IQ	System Administrator

M	Government	UK	ICT Team Manager
N	Government	UK	IT Manager
O	Government	IQ	System Administrator
P	Government	IQ	System Administrator

5.1.2 Framework components

In this section, all the interviews were interesting and positively arguable. All the experts confirmed that the proposed framework has the right components, but there were some characteristics (methods) which were the centre of the debate. These characteristics are listed below:

1. What type of encryption will be used?
    a. Answer: this is not determined yet. It will be determined in the first step of the future work.
2. The ability of the file to auto run once it is copied or downloaded at the receiver machine.
    a. Answer: the file will check the environment once the user tries to open it, not the moment it lands at the receiver computer.
3. What if the body of the document that contains the actual information is copied using bit-by-bit operations from DOS or Linux and the attacker has many tries to crack it?
    a. Answer: the document body is useless by its own, because it is part of the original body and the remaining parts are provided by the framework.
4. What is the security measure for communication channels when sharing a document via email or online.
    a. Answer: channel security is outside the scope of my research. However, the document itself is not useful without the framework.

On the other hand, the experts suggested some new ideas and proposed replacements for some components. They were really interested by the concept of keeping the same software, as there is no need to train the end users. Ten of the experts, including seven from the UK, were focusing on training cost and time. Expert I said:

> "In real life, no business can effort throw a massive amount of money and hundreds of training hours away just because another software provides one more security feature".

They suggested using the cloud and the Security Assertion Mark-up Language (SAML) as delivery mechanisms to share and force security policies. SAML is an Extensible Mark-up Language (XML) based data format used to exchange authorisation and authentication. Moreover, the XML signature may be used to verify the SAML source and the data integrity. Some of them suggest using Attribute Based Access Control System (XACML) to provide more granularity to user access rights.

The result of the feedback from the expert interviews were reflected on the initial framework design and described in section 3.2.4 the modified framework. Most of the discussion with the experts was about technologies and mechanisms that may be used in the framework and which are better. There were good remarks and points made in the discussion, but they were more useful at the implementation level rather than the level this research is at.

5.1.3 Summary of the finding from the expert interviews

From the literature in Chapter 2, three main factors were identified. These factors were human negligence, legalisation, and cross-domain sharing. In the second section of the interview, the experts were asked to state their opinion about these factors. The first question was:

"In your experience, what do you think about these factors identified by the literature?

They all confirmed these factors as main and have "precise" ones. While this is the case, all of them were hesitant (cautious and thoughtful) about the legalisation factor. The UK experts were cautious about different legislations for different domains and countries. They referred to it as "private policy", with Expert A saying:

> "Policy makers and administration interference prevent applying a unified legal framework".

Meanwhile, Iraqi experts were hesitant to consider it as a factor for document security, but more as an implementation hindrance. Expert O said:

> "Our legal system is not ready to handle such a new conflict"

Moreover, Expert P said:

> "The parliament had passed very broad guidelines three years ago; they were so broad no directorate could decide how to apply regulations that make them feel safe when using digital documents".

Legalisation is more about the will to secure electronic document and the ability to validate this security across different domains, and this was not always hindered by technology. Thus, the legalisation factor was removed from the proposed framework focus.

The next question was:

"Depending on your experience, what do you think was overlooked by the literature review?"

Expert B said:

> "Cost is an essential factor to consider when you're proposing a new solution. When the cost is high, we start looking for big players in this industry like MS or Adobe".

Meanwhile, Expert F focused on upgrading and changing the ecosystem used by the user, commenting:

> "Upgrading a system is not an easy and cheap process."

The findings from the expert reviews were very encouraging in term of agreeing with the identified issues with secure document sharing. However, it was difficult to pass by the technical details for each component. Each expert speaks for its domain requirements and challenges; they wanted to know which technology used where. This led us to our next point, which is the technology combination to provide more security. Expert M said:

> "When the DES threatened, they used triple DES or 3DES to encrypt their hard disks".

Expert K said,

> "From my experience, using a trusted computing environment with DRM reduces document leakage risk almost to none".

However, most of the experts usually end their sentences with how costly it is to do that, and to what extent the other party will be convinced to spend for that matter.

Another essential point made by Expert H was:

> "Integrating your solution with the existing office suite will directly reduce capital invested in staff training significantly".

This referred to how much it costs an organisation to train its staff to use new software. The remaining questions were about the framework components, as discussed in the next section.

5.2 The modified framework

The initial framework concept and components were discussed with fourteen security experts. These discussions were carried out as a part of the interviews to confirm the research findings. As a result of these discussions, a refined version of the framework emerged. This refined version is still called the Tamper-Proof Framework (TPF). The following paragraphs describe the framework structure and main components modifications.

The main concepts behind TPF remain the same, which are:

- Providing a means to extend Organisation A's security policy to be applicable on Organisation B.
- The document produced in this framework is useless outside the framework environment.

The TPF is composed from two main parts: Active Document and System Functions. Active Document is a document that has active security properties created in a machine that has TPF installed. System Functions are secure background services in the operating system which monitor some system calls and translate the active properties into security rules. The TPF will depend on the cloud to provide secure customisation and delivery channels, as shown in Figure 9.

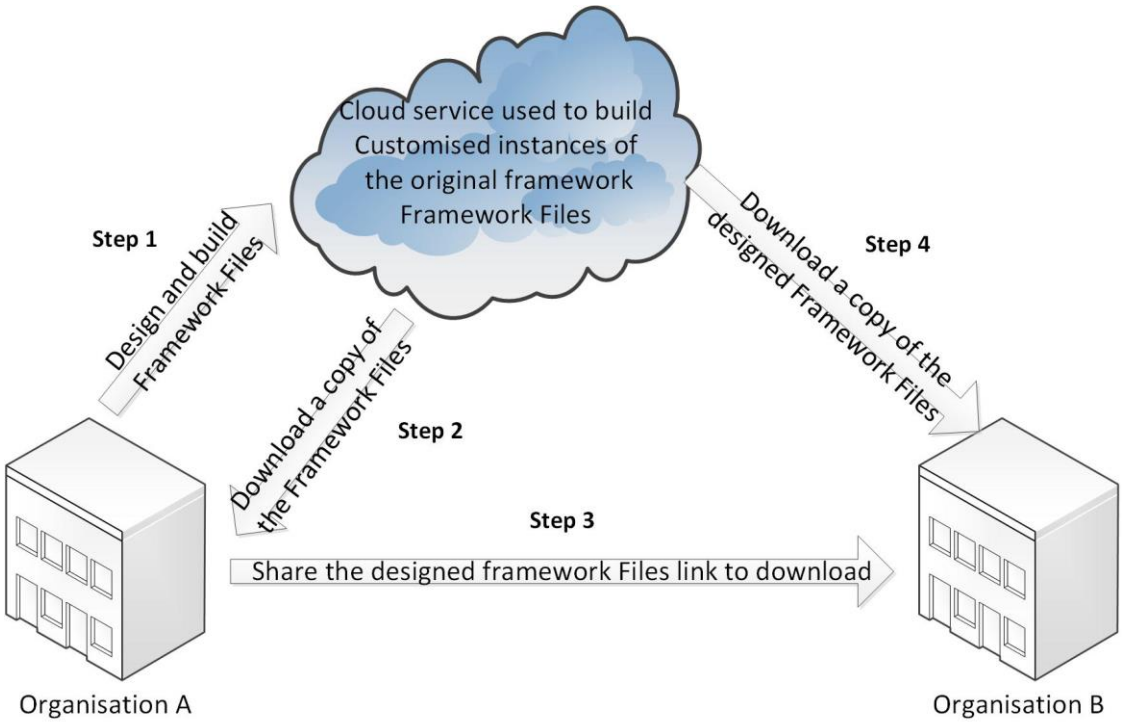

Figure 9: The Cloud as customisation and delivery channels

5.2.1 Framework Components

New terms were used in the modified TPF. All the terms used in the TPF context are explained in this paragraph:

- Active Properties: features inserted inside a document that can be read by another piece of software designed for that.
- Active Document: a document that has active properties. These active properties are implanted by the Framework files. The Active Document is composed from two main parts: Public Information and Secret Information.

• Public Information (Pre-Processing information): plain text Default Security Policy and Active Properties that help the System Functions to detect and authenticate the document's origin. This security policy is used by the System Functions to check for the initial security requirements.

• Secret Information (Post-Processing information): encrypted data containing security parameters. These parameters are:

- Access control policy: the required user authentication

- Context parameters: when, where, what and who is authorised to open this document.
- Verification code: for the active information in the Public Information. It is a hash value for the Public Information.
- Security Mechanism Parameters: contains technical information about the encryption technique, content verification, and retention policy. This technical information is for the Actual Encrypted Data processing.
- Actual Encrypted Data: part of or all of the actual data of the document. This data is encrypted for the second time and can be only accessed if all the validations are passed.

• Framework Files: customised version of the TPF distributed in an executable format and works as a background service. It integrates with the operating system and the presentation software (Microsoft Office, PDF Reader) to monitor and control their activities. In addition, it has special functions called System Functions that perform security operations (encrypting, decrypting, and hashing).

• Presentation software: the default programs that are used to view the documents. This presentation software could be Microsoft Word, Adobe PDF Reader or Open Office.

5.2.2 Framework Usage Scenarios

The expected scenario is that Organisation A logs in to the cloud service to customise its own version of the TPF Framework Files to reflex its security policy. A trusted third party could provide the cloud service, or it can be provided by Organisation A itself. The interesting point here is the trust between the two organisations is out of this research's scope. Organisation A then shares the download link for the customised Framework Files with Organisation B. When both organisations install the Framework Files, they are ready to share documents and extend their security policies.

The Active Document has two main components: Pre-Processing information and Post-Processing information. Pre-Processing information is public information that can be read by any operating system, but is interpreted differently on OS's that have the Framework Files installed. This information is used to authenticate the document's source and to set the minimum-security policy parameters at the

destination machine. On the other hand, Post-Processing information is securely encrypted using asymmetric key encryption. The content of this information includes the following:

- Security parameters (access control, environment-monitoring level and context).
- Active properties to verify the document's content integrity.
- Part of or all the actual document content.

After the document is authenticated, the TPF will start to decrypt the Post-Processing information to retrieve the remaining security policy parameters. The System Functions will translate these parameters into security policy roles; in other words, deploying Organisation A's security policy. Finally, when all the settings go through correctly, the remaining document content is downloaded from the cloud. The cloud service could be provided by a third party or by Organisation A. These steps are shown in more detail in Figure 5. The document retention policy is also part of the Post-Processing information.

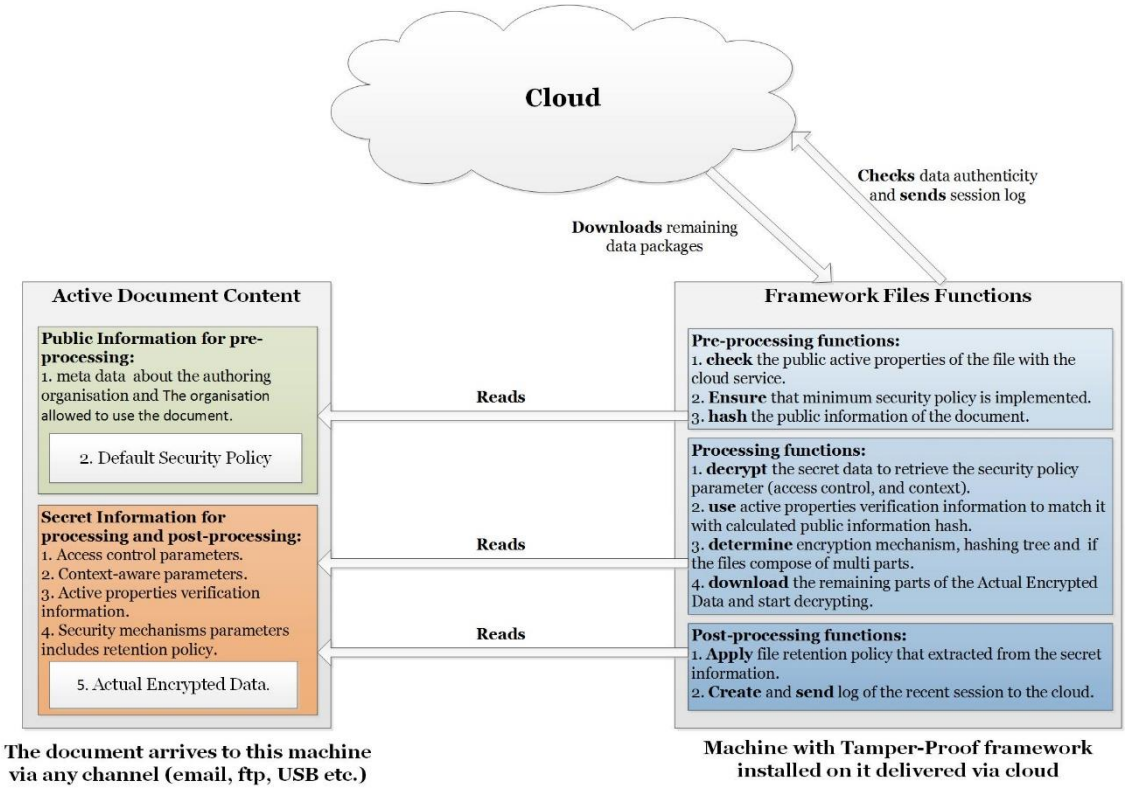

Figure 10: TPF document access scenario

The TPF Framework Files are composed from two main components: Active Document Readers and System Functions. Each component has two or more

modules inside to perform as designed. However, technical details of these modules are very specific, which will be useful in the implementation phase but not for the purposes of this research. These details include, but are not limited to, access control mechanisms and protocols, context information sensors, encryptions algorithms, key exchange mechanisms, reliability mechanisms, and hashing functions. The overall framework structure is shown in Figure 11.

5.2.3 Overall system components

This section summarises the overall framework components and functions, and the next section goes in detailed step of the framework workflow.

5.2.3.1 Active document

An active document is a document that has active security properties created in a machine uses the proposed framework. These Active Properties are divided into two main parts: Public Information and Secret Information. Public Information (Pre-Processing information) is the plain text Default Security Policy and Active Properties that help the System Functions to detect, authenticate, and check the integrity of the document. The Default Security Policy is used by the System Functions to check for the initial security requirements. Secret Information (Post-Processing information) is encrypted data that contains security parameters. These parameters are:

- Access control policy: the required user authentication information and mechanisms.
- Context parameters: when, where and what for each user authorised to open this document.
- Verification code: for the active information in the Public Information. It is a hash value for the Public Information in the document.
- Security Mechanism Parameters: contains technical information about the encryption technique, content verification, and retention policy. This technical information is for the System Functions in order to decrypt Actual Encrypted Data.
- Actual Encrypted Data: part of the actual data of the document. This data is encrypted for the second time and can be only accessed if all the validations from public information and verification code are passed.

5.2.3.2 System Functions

The Framework Files are the actual files that contain programming code to perform the System Functions. The authors assume that each enterprise will customise the Framework Files to fit their security policy. The customisation includes the access control policy, context parameters and default document retention settings. Framework Files are customised executable files distributed by the enterprise to work as a background services in a computer machine. They integrate with the operating system and the presentation software (the default program that is used to view the documents like Office suites, and PDF readers) to monitor and control their activities. In addition, they have special functions called System Functions that perform security operations (encrypting, decrypting, and hashing).

System Functions are automated background services in the operating system responsible for encryption/decryption, monitoring presentation software operations and translating the active properties into security rules. The system functions are divided into three main modules:

- Authentication Module: this module reads the public active properties of the active document.
- Encryption / Decryption Module: this module decrypts the actual data inside the active document.
- Logging/ tracking Module: this module logs all the operations done on the document and sends it back to the certification authority.

5.2.3.3 Certification authority

As described in Section 2.5, a certification authority provides continuously updated secure binding between the encryption keys and their authors (Zhou et al. 2002). The certification authority (CA) provides secure download services in addition to its default functions. This secure download service is the channel which the enterprise uses to share its security policy requirements. The CA could belong the enterprise itself or be provided by a third party. The trust issues of CA are out of the scope of this research. The workflow of CA is as follows:

1. The Enterprise A security officer submits their Framework Files to the CA and choses what this Framework Files aimed for (e.g. are they for everyone to download or for specific enterprises to share documents with?).
2. The CA assigns a pair of keys (public and private) to encrypt the enterprise's Framework Files.
3. The keys are exchanged with Enterprise A. The CA then sends the owner a sharable download link for the encrypted Framework Files.
4. Enterprise A sends that link with Enterprise B, which it aims to securely share documents with.
5. Enterprise B downloads and installs the Framework Files on their computers, and then they are ready to work on documents from Enterprise A.
6. Once an Active Document sent to Enterprise B from Enterprise A, the Framework Files will interpret its active properties and check for the integrity of the document with the CA.
7. Then the document will be opened or discarded depending on the integrity feedback from the CA. The integrity checking required to access to the Public Information is part of the Active Document.
8. If the document integrity check was clear, the Framework Files will decrypt the Secret Information part of the Active Document and then download the remaining document parts on the computer.
9. When the Active Document is in use, the Framework Files will monitor and control its usage (saving in another name, printing, or screen printing). When the user finishes using the document, the Framework Files will send session information to the CA to track this document's usage.

This workflow ensures the secure delivery of the Framework Files and the remaining data for the document, and provides the ability to track users' activities on the document.

5.2.4 Overall framework workflow

This section covers the workflow of the confirmed framework components' interactions, in steps. The section describes the process of sharing a document securely between two organisations, where both organisations use Framework Files within their computer systems. For this example, Organisation A will be considered as the authoring organisation. However, all the organisations using the same custom framework files will be able to author an active document that can be shared among them securely. The workflow starts from Organisation A, as shown in Figure 11. The numbered steps in Figure 11 are described in the list below, the list numbering reflects the actual step number in the figure.

1. Organisation A sets its sharing and usage policy that they want to apply on any document it shares with other organisations, including Organisation B. Setting the sharing and usage policy will produce custom Framework Files.
2. This step is mainly about sharing the custom Framework Files that are generated from the previous step. The proposed and confirmed sharing channel is a cloud service. This step includes two sub steps:
    a. Sharing the Framework Files to the cloud by Organisation A (the authoring organisation).
    b. Organisation B downloads the custom Framework Files from the cloud service. There is no limitation on how many organisations can download the Framework Files, unless the sharing and usage policy says otherwise.
3. Organisation A creates an active document using their computer systems that have the Framework Files. This active document will follow the security settings in the sharing and usage policy in the Framework Files. Organisation A can share this active document via any channel.
4. Once the active document at the receiving organisation it will be checked by the Framework Files. The check will be performed on two stages:
    a. Public information metadata checking: If the active file sharing policy destined to this organisation then proceed to the next check. Otherwise, the Framework Files use their own retention policy to destroy the active document. To perform this check, the Framework Files need to communicate with the certification authority.

   b. Secret information checking and processing: This information will be processed only when the first stage is passed successfully, meaning the active document is supposed to be opened in this organisation's computer systems. Then the Framework Files will use the Secret information to download the remaining parts of the document from the cloud using PKI.

5. This step focuses on verifying the Public and Secret information with the CA and cloud. This verification includes two sub steps:
   a. The first sub step is to verify the integrity of the Public information, and then use this Public information to retrieve the decryption keys for the Secret information in the next stage. If the Public information did not pass, the active document will be discarded.
   b. The second sub step is to download the remaining parts of the active document's secret data from the cloud using the keys obtained from the previous step. The downloaded parts will complete the actual document data. However, as the data will still be encrypted, the Framework Files need to contact the CA for the second time to get new keys for the full data decryptions. Part of the secret information authentication requirements. If the document required more authentication like online login or VPN connection in addition to the operating system user authentication.
6. Now the active document data is ready to decrypt and be processed according to the attached sharing and usage policy. This step is performed by a black box, which contains different algorithms for cyphering and deciphering the data as many times as required. Then it pipes the actual active document data to the policy integration module.
7. This step is all about monitoring and controlling the environment in which the active document is used. This step has three sub steps:
   a. Monitoring the piping of the data from the black box module in plain text format that is understandable by the default presentation software in the computer system used.
   b. Controlling the presentation software in which the active document will be presented to the end user. The presentation software could be the office suite, or a portable format viewer or editor. In this step, the Framework Files control what the usage rights are on that active

document for the specific user. All this is based on the sharing and usage policy that is integrated with the active document.

c. Controlling the operating system functions; for example, which programs have to be closed before opening the presentation software. Another example is to disable system functions, like printing the screen or using default software.

8. Each time the active document is decrypted and licenced correctly it is considered as a session. The final step is to collect and send all logs of the session to the cloud so that the authoring organisation can analyse and audit the document usage.

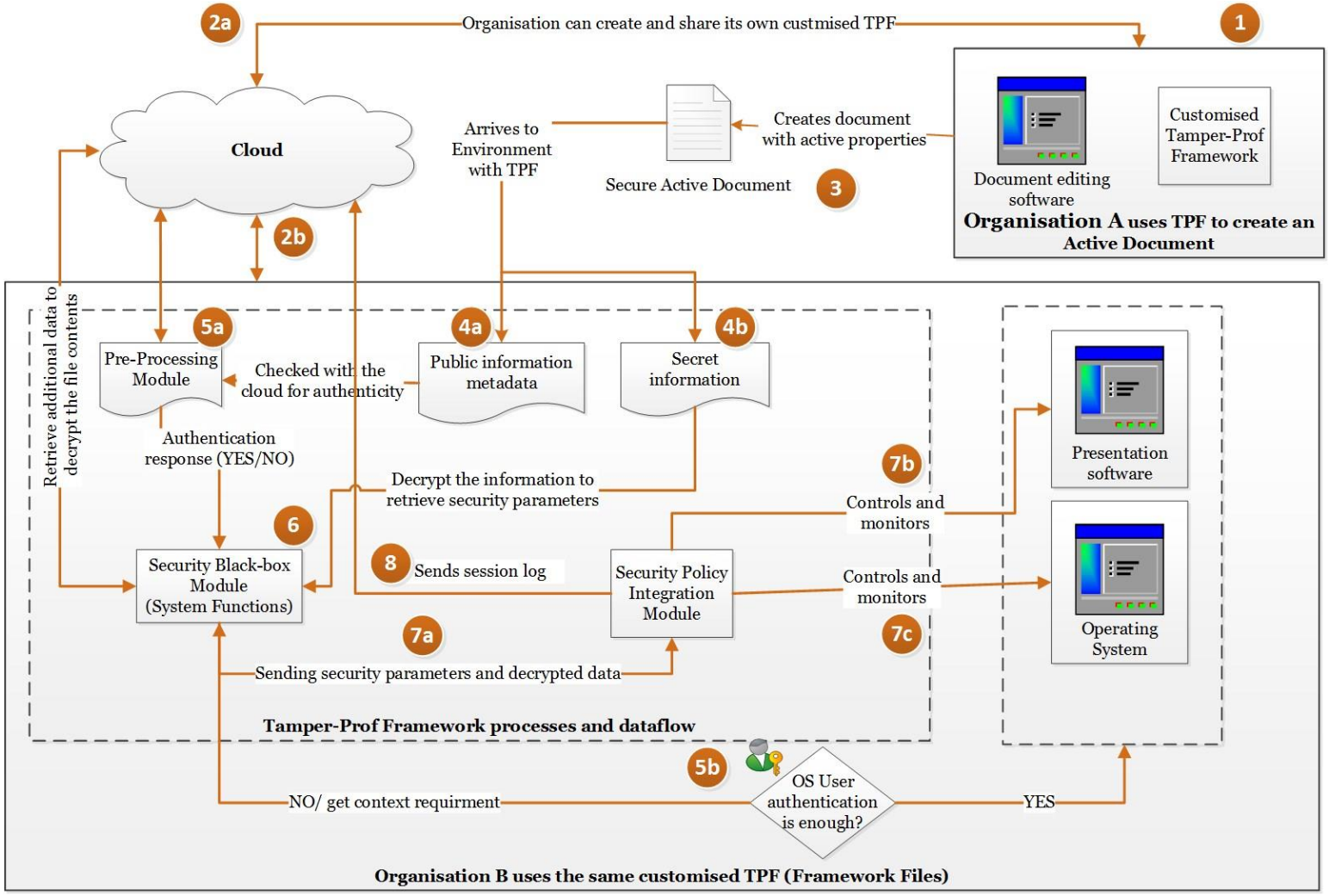

Figure 11: TPF overall structure

5.3 Result from the survey

The survey was designed to validate the modified framework components and revalidate the literature findings. The survey was filled by ninety IT professionals (which was validated by a couple of demographic questions). The survey questions were designed to measure to what extent the IT community agreed with the TPF components and findings. The results of the survey will be discussed in two main parts: firstly, the confirmation of findings, and secondly, the confirmation of framework components.

Table 4: Domain frequency for survey participants

Domain	Frequency	Percentage
Healthcare	15	16.7%
Education	34	37.8%
Business	14	15.6%
Government	18	20%
Other	9	10%

Table 5: Survey responses for the identified issues

Issue	Response	No. of cases	Percentage
Human negligence	Strongly Agree	25	27.8%
	Agree	38	42.2%
	Neutral	22	24.4%
	Disagree	3	3.3%
	Strongly Disagree	2	2.2%
Different domains Different security policy	Strongly Agree	37	41.1%
	Agree	20	22.2%
	Neutral	16	17.8%
	Disagree	17	18.9%
	Strongly Disagree	0	0%
Legislations	Strongly Agree	18	20%
	Agree	5	5.6%
	Neutral	25	27.8%
	Disagree	20	22.2%
	Strongly Disagree	22	24.4%

5.4 Discussion

The survey analysis gives a clear idea what the IT professionals were concerned about. The total number of participants was 113, but only 90 of them actually finished the survey, and therefore only their input is valid. The SPSS software was used to perform the statistical analysis of the participants' responses.

To verify the framework components, the participants were asked to answer direct questions on a scale of 1-5, where 1 meant "Strongly disagree" and 5 meant "Strongly agree". Table 6 below shows the mean for one sample statistics.

Table 6: One-Sample Statistics for the components' survey questions

Component	N	Mean	Std. Deviation	Std. Error Mean
Q1. Using Cloud or web application as distribution channel	90	3.82	1.128	.119
Q2. Using System Function as black box for security processes	90	3.82	1.128	.119
Q3. Splitting the Security policy into two parts	90	3.82	1.128	.119
Q4. Include context information as part of the security policy	90	3.79	1.117	.118
Q5. Store and upload session information for context-aware analysis	90	3.84	1.151	.121
Q6. Using some error correction codes as availability measure (like Erasure code)	90	3.89	1.011	.107
Q7. Using multi-level of encryptions	90	3.89	1.011	.107
Q8. Integrate the Framework Files with the existing legacy software	90	3.89	1.011	.107

From Table 6 above, every component has a mean higher than 3, which means that they are valid and statistically significant. It is worth mentioning that the mean values for some questions were the same. This is due to having more participants than the ideal number required.

If the 2-tailed value (significance value) in Table 7 is greater than 0.05, then the choice is not significant. From Table 7, we find that all the questions are significant.

Table 7 One-Sample Test with test value = 3

Questions	t	df	Sig. (2-tailed)	Mean Difference	95% Confidence Interval of the Difference	
					Lower	Upper
Q1	6.918	89	<.001	.822	.59	1.06
Q2	6.918	89	<.001	.822	.59	1.06
Q3	6.918	89	<.001	.822	.59	1.06
Q4	6.702	89	<.001	.789	.55	1.02
Q5	6.963	89	<.001	.844	.60	1.09
Q6	8.345	89	<.001	.889	.68	1.10
Q7	8.345	89	<.001	.889	.68	1.10
Q8	8.345	89	<.001	.889	.68	1.10

5.5 Summary

The triangulation method was used to explore and confirm the literature findings. The method was successful in exploring the aspects of the proposed framework. The experts, in general, were interested in a solution that is cheap, customisable and scalable. Furthermore, they provided constructive comments regarding the framework components. These comments were used to modify the initial framework components. The modified framework was now ready to be confirmed by the selected sample from the IT community.

The survey was used to confirm the components of the modified framework. The results were encouraging enough to proceed with the next step of this research. However, the survey results show common disagreement about the legal issues in relation with technology. This may be an indication that trust in the administration level is a key factor for the technology to work.

The TPF aims to secure electronic documents when they leave the boundaries (firewalls) of the authoring organisation. It proposes a new combination of technologies to perform this task. The initial framework structure was modified to reflect the discussion with the security experts interviewed, but the technology combinations remain the same. These technologies are:

- Active document: a document that has a set of properties more than the ordinary metadata. These properties are called “Active Properties” and can be read by other particular pieces of software (Quint & Vatton 1994).
- Cloud service: a very broad term referring to the Internet (Velte et al. 2010). It is used in this context as a delivery and distribution channel. The term in this context refers to a web site that is trusted by both parties to share the “System Functions files”.
- Persistent security: this means that the system is secure through all its states if certain characteristics are met (Bossi et al. 2004). There are different methods to control and monitor these characteristics of the system. One method is to install software that replaces all default programs that open certain types of files. This concept is used by Digital Rights Management (DRM) solutions to secure

digital media and documents. In this context, it is called “System Functions”.

These three concepts combined form the core of the TPF. Other techniques are proposed to provide standardised security policy communication, like XACML (Lorch et al. 2003). Another technique is the Erasure Code for reliable content delivery (Rizzo 1997).

Chapter 6: Modelling the framework using formal methods

This chapter will describe the modelling process for the proposed framework. Formal methods are defined as "mathematically rigorous" tools and technologies for design, specification, and verification of hardware and software systems. They will be used in in order to model the requirements of the proposed framework. A behavioural event-driven model used to give an abstract view of the states in the framework.

6.1 Introduction to system modelling

In general, system modelling uses graphical notations to represent each model. Currently, the most used notations are the ones based on the Unified Modelling Language (UML). The produced model is useful in the requirement engineering process of a system, the design process when the system engineer implements the system and after the system is implemented to check its functionality. However, it is good to remember that system modelling leaves out some details. That is mainly because the modelling is an abstraction of the system and it focuses on one perspective of the system (Vliet 2007). So, in order to get the full details of the system aspects, all the perspectives need to be modelled.

According to Kruntchen's 4 + 1 (Kruntchen 1995) system architecture views, the most matching perspective models are (Vliet 2007):

1. External perspective: modelling the environment of the system.
2. Interaction perspective: modelling the interaction between the system and its environment or the interaction between the system's internal components.
3. Structural perspective: modelling the structure of the organisation or data processing in the system.
4. Behavioural perspective: modelling the dynamic behaviour of the system and the way it responds to events.

The behavioural perspective modelling was used to model the dynamic behaviour response of the framework. The behavioural modelling has two main types of

behaviour triggers to model. The first is data-driven modelling, where the system has to make responses to process incoming data. This type of behavioural modelling requires a detailed data flow diagram. This modelling is most suitable for requirements gathering and the engineering process where it shows end-to-end data processing. The second is events-driven modelling, commonly used to model events that are stimulated (triggered), requiring the system to respond. In some cases, events have data associated with it. This modelling considers any event, regardless of whether they were internal or external events (Said et al. 2009). However, it assumes that the system has a number of defined stages (states) and the events transfer the system from one state to another (Harel 1988). This type of behavioural modelling is the most suitable for our framework.

The proposed framework assumes there are defined states for any document in the enterprise, and switching from one state to another needs not just the triggering event but also sufficient data. These states are listed in Table 8.

6.2 TPF behavioural event-based modelling:

The modelling process used information from the confirmed framework to define the states, events and data related to each state. The behavioural event-based modelling of the framework will show only the states and the events which make the transition from one state to another. The model will not show the flow of the data within the framework; however, it may include some information about the process used on the framework.

The TPF assumes that the sequence of actions on the document is:

1. Organisation A has the TPF and a user with right permission wanting to create or open a document.
2. The TPF checks the user's permissions and environment settings and behaves accordingly, either granting or denying the user request.
3. The user saves the document, and the TPF attaches active properties and a security policy to the document, as well as updating the CA.
4. The document is shared with another organisation (B), and a user on that organisation tries to open the shared document.

5. If Organisation B has the TPF, the TPF will recheck the document integrity with the CA, the authoring organisation security policy and the current user permission. If the document is sent to organisation B and the user has the right permissions, the document will open. Elsewise, the document will be encrypted to render it useless, as shown in figure below.
6. If the organisation B does not have the TPF, the document will not open and cannot be used. The user does not have enough data to render the document again.

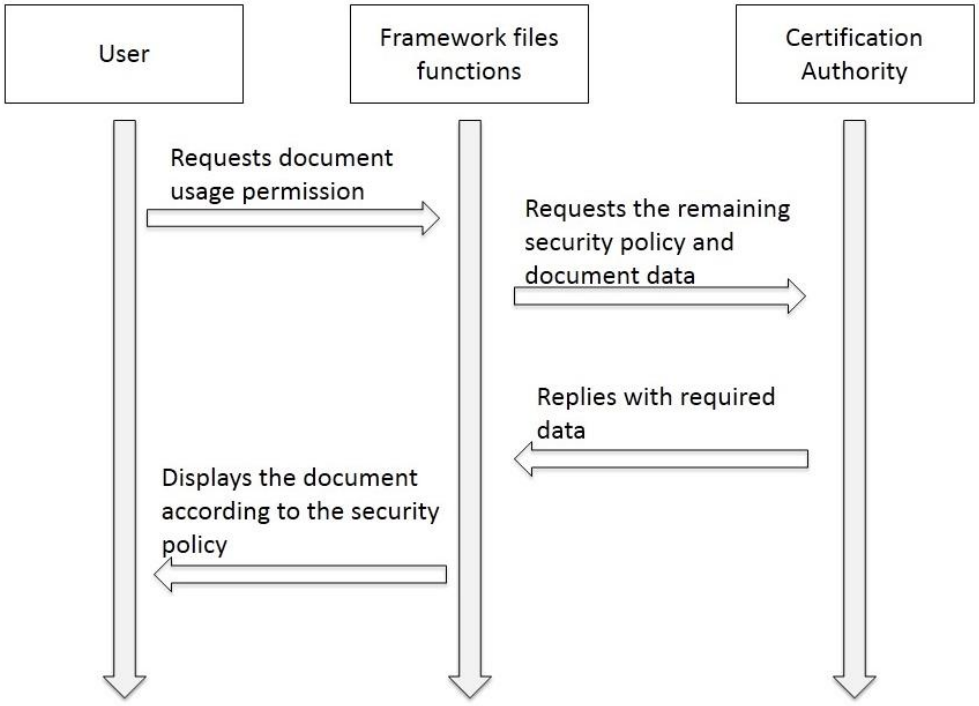

Figure 12: A sequence diagram for TPF when sharing a document with another organisation.

The behavioural modelling required states and events for all of the framework's activity sequence. In the tables below, the states and events are detailed. An important note is that behavioural event-based modelling has one issue; the states and events grow rapidly unless the level or granularity is determined. For the purposes for this research, two level states are enough, since the aim is to provide a common landscape of the proposed framework.

Table 8: The proposed framework states

Super state	State	Description
Protected document		Opening the environment to create or edit a document inside an environment that has the TPF. This super state has three normal states.
	New document	Creating a new document in the environment with the TPF
	Opened document	Opening the document with permissions matching the authoring organisation policy and enforced by the TPF
	Shared Document	A document in transit via portable media or through a network.
Useless document		A document opened in environment without the TPF.
Archived document		A document archived in an environment with the TPF in rest mode.

Table 9: Proposed framework event list

Events	Description
Check	Checking the user permissions and environment setup policy.
Setting-up	Setting up the environment security parameters according to the environment security policy.
Open	Opening the document according to the user permissions for editing or viewing.
Save	Saving the document and attaching the security policy of the authoring organisation.
Authorise	Checking the document integrity with the CA before starting any other event.
Re-check	Re-checking the document security parameters, user permissions, and environment settings. These setting will be enforced by the TPF to match the authoring organisation's security policy. This event happens outside the authoring organisation's firewall.
Confiscate	Encrypting the document with a random key so it will be unusable.
Update	Updating the log information with the CA for auditing purposes.
Archive	Storing the document in a protected environment according to the authoring organisation's security policy.

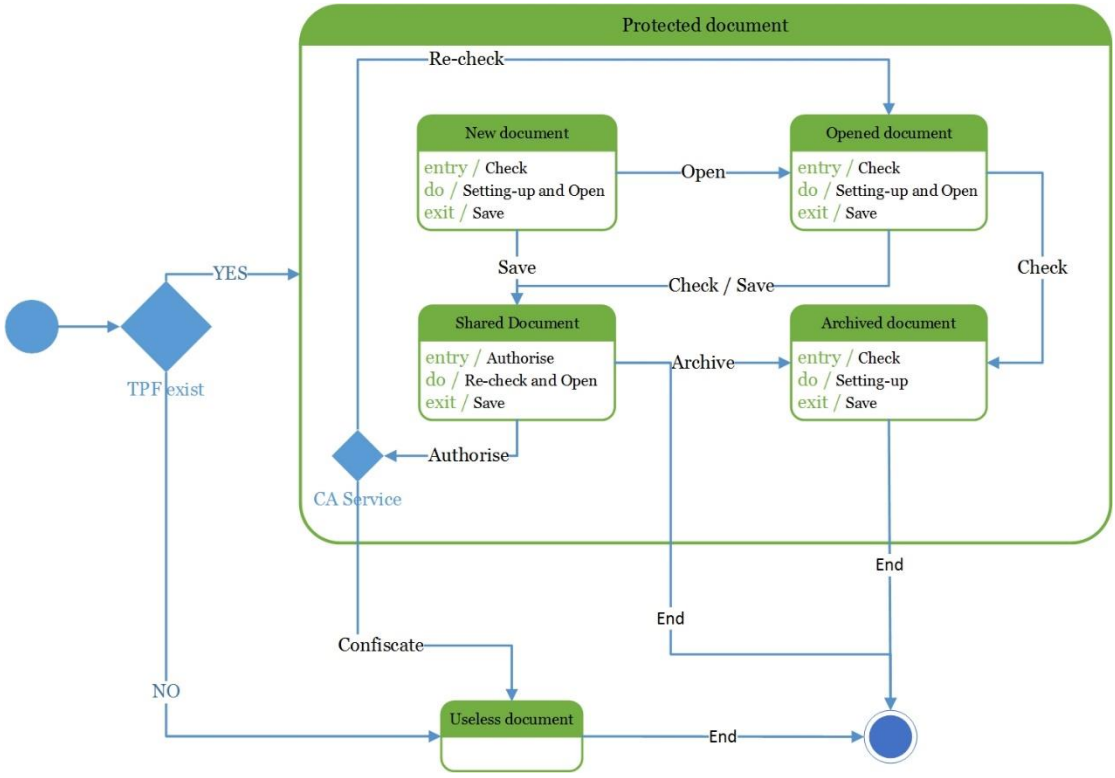

Figure 13: The behavioral event-based model of the proposed framework

6.3 Introduction to formal methods

Formal methods are mathematically rigorous tools and technologies to verify the requirements and design of software and hardware systems (Beth 1970). It utilises well-formed statements in mathematical logic that are used in formal verification, and rigorously subtracts any logical action that defies the mathematical statements (Davies 1988). Formal methods have massive application especially in safety and risk assessment in real time systems. Real systems (hardware and software) are known to have high complexity in term of states and events (Heitmeyer 2009). To overcome these challenges, formal methods use:

1. Abstraction: where too many details are not the main goal, such as in high level designs.
2. Prioritisation: applied only for the most critical parts of the system.
3. Least variable: analyses the models that have least states and events.
4. Divide and conquer: use iterations (refinements) in a hierarchical approach.
5. Automation: as much as possible using software tools.

There is no 'best' formal method in particular that works for all scenarios. Thus, the formal methods are used mainly in three different traditions; Formal Specification Languages (Larch, Z, and VDM), Reactive System Modelling (RSML and SCR), and Formal Conceptual Modelling (RML, Event-B and Telos) (Ii 2001). For the purposes of this research, the Formal Conceptual Modelling is what best fit to achieve overall verification of the framework states and functions.

Formal Conceptual Modelling is a general approach to modelling real world systems beyond software functions. This approach uses human knowledge to make abstractions and refinements of the system. This approach is used in this research and the Event-B formal method was chosen to model our proposed framework (Vliet 2007).

6.4 Formal Conceptual Modelling of proposed framework:

Event-B is a state-based formal method widely used for verification and specification purposes. It models a system using states in a hierarchal manner and gradually refines each state to add more events and specifications. In Event-B, a model has two main parts; the first is static (CONTEXT) and the second is dynamic (MACHINE) (Said et al. 2009). The CONTEXT defines the static parts of the model (SETS, CONSTANTS and AXIOMS) to introduce all constraints on the sets; meanwhile, the MACHINE has the dynamic components, such as VARIABLES, INVARIANTS, and EVENTS. Changing a VARIABLE in a machine changes its state and is controlled by the INVARIANTS. The verification process of a model involves testing the consistency and correctness of the constraints in all the refinements (Wright 2008). In this research, the Rodin platform was used to develop the formal model of the framework. Rodin is an Eclipse-based Integrated Development Environment for Event-B with an extensive library of plugins to extend its functionality.

Formal modelling using Event-B needs well-defined requirements for each state. These requirements will help to decide the CONTEXT and MACHINE parameters. The states of the framework are explained in Figure 13. The requirements for each state were defined from the confirmed framework components in Section 5.2. The Table 10 below shows these components and their mapping to the security triage CIA (Confidentiality, Integrity, and Availability).

Table 10: The confirmed framework components mapping to the security triage CIA (Before detailed mapping into state machine).

Component	Security triage mapping
	Confidentiality / Integrity / Availability
C1. Using a Cloud or web application as distribution channel	Integrity / Availability
C2. Using System Functions as a black box for security processes	Confidentiality / Integrity / Availability
C3. Splitting the Security policy into two parts	Confidentiality
C4. Including context information as part of the security policy	Confidentiality
C5. Storing and uploading session information for context-aware analysis	Confidentiality / Integrity
C6. Using some error correction codes as availability measures (like Erasure code)	Integrity
C7. Using multi-level encryptions	Confidentiality / Availability
C8. Integrating the Framework Files with the existing legacy software	Confidentiality / Availability

But these components did not provide enough details to be used in the model. Therefore, the components were divided into sub functions. The next step is to map these functions to each state, as shown in Figure 13. The detailed functions of the components were:

1. Using a Cloud or web application as distribution channel
    a. Checks data authenticity and sends a session log
    b. Downloads remaining data packages
2. Using System Functions as a black box for security processes
    a. Decrypts the secret data to retrieve the security policy parameters (access control and context).

   b. Uses active properties verification information to match it with the calculated Public information hash.
   c. Determines the encryption mechanism, hashing tree and if the file is composed of multiple parts.
   d. Downloads the remaining parts of the Actual Encrypted Data and starts decrypting.
3. Splitting the Security policy into two parts
   a. Applies the file retention policy extracted from the secret information.
   b. Meta data about the authoring organisation and the organisations that are allowed to use the document.
   c. Default Security Policy
4. Including context information as part of the security policy
   a. Access control parameters.
   b. Context-aware parameters.
   c. Active properties verification information.
   d. Security mechanisms parameters, including the retention policy.
5. Storing and uploading session information for context-aware analysis
   a. Checks data authenticity and sends a session log
6. Using some error correction codes as availability measures (like Erasure code)
   a. Downloads the remaining parts of the Actual Encrypted Data and start decrypting.
   b. Uses active properties verification information to match it with the calculated public information hash.
7. Using multi-level of encryptions
   a. Secret Information for processing and post-processing
   b. Actual Encrypted Data.
8. Integrating the Framework Files with the existing legacy software

The next step is to map these functions to their corresponding state in the framework. Table 11 below shows the mappings:

Table 11: The Framework machine States and their relative functions

States	Functions
New Document	Checks data authenticity and sends a session log
	Determines the encryption mechanism, hashing tree and if the file is composed of multiple parts
	Meta data about the authoring organisation and the organisations that are allowed to use the document
	Default Security Policy
	Access control parameters
	Context-aware parameters
	Active properties verification information
	Security mechanisms parameters, including the retention policy
	Secret Information for processing and post-processing
	Actual Encrypted Data.
Opened Document	Checks data authenticity and sends a session log
	Downloads remaining data packages
	Decrypts the secret data to retrieve the security policy parameters (access control and context).
	Uses active properties verification information to match it with the calculated public information hash
	Determines the encryption mechanism, hashing tree and if the file is composed of multiple parts

	Downloads the remaining parts of the Actual Encrypted Data and starts decrypting
	Meta data about the authoring organisation and the organisations that are allowed to use the document
Shared Document	Checks data authenticity and sends a session log
	Downloads remaining data packages
	Uses active properties verification information to match it with the calculated public information hash
	Determines the encryption mechanism, hashing tree and if the file is composed of multiple parts
	Meta data about the authoring organisation and the organisations that are allowed to use the document
Archived Document	Downloads remaining data packages
	Applies the file retention policy extracted from the secret information
Useless Document	

6.4.1 Using Rodin to verify the model

Building a model in Rodin requires details about CONTEXT, MACHINE, and the events that trigger a MACHINE to transfer from one state to another. Once the states and the functions were related to each state, the remaining step was to identify the SETS, CONSTANTS and AXIOMS for the CONTEXT and the VARIABLES, INVARIANTS, and EVENTS for the MACHINEs. However, the research identified a set of requirements that are generic, and therefore any refinements will not be applicable. The goal of the research was to identify a framework characterisation, not to implement one. There were two approaches to model the framework; the first was to define one CONTEXT and MACHINE and then refine it in a different iteration to represent each state, and the second was to define one CONTEXT but different MACHINEs, each representing one state of the document in the framework. The first approach would have provided more

granularity and control over the requirements, and thus was chosen for this research.

The first step is the abstraction of the framework's main functions and parameters. This process will set the groundwork for further refinement, depending on the final implementation of the framework. The main sets in these CONTEXTs are:

- DOCUMENTS: the set of all the documents that belongs to an organisation and created in environment uses the framework.
- USERS: the set of an organisation's users.
- ORGANISATIONS: the set of the organisations that use the framework and wish to share documents securely.
- CA_PROVIDERS: the set of Certificate Authorities and content providers.
- SECURITY_POLICY_PARAMETER: the set of security policy settings (for example, when, where, who, and any other access and context control parameters).
- SECURITY_TECHNOLOGY_PARAMETER: the set of encryption, hashing, and transmitting technologies, parameters and settings.

The framework abstraction will have events (functions) which change the variables in the framework to transform the states. These events are:

- CreateNewDocument: an event to create a new document in the framework
- OpenDocument: an event to open a document in the framework.
- SaveDocument: an event to save a document after editing or reading it in the framework.
- SettingupTheEnvironment: an event to enforce the parameters belonging to a security policy and technology sets.
- AuthoriseDocument: an event to contact the CA to check the information integrity in the document.
- AuthoriseUser: an event to check the user's permissions, carried out after the authorisation of the document has passed though positively.
- ArchiveDocument: an event to store a document in an archive, along with all related metadata.
- ConfiscateDocument: an event to encrypt unauthorised documents with random security parameters to render them unusable.

This information provided above was enough to start modelling the framework. The modelling started with a CONTEXT and one MACHINE. This setup would provide an overview of the required settings to build an actual software framework from this conceptual framework. The final Event-B file is shown below in Figure 14, Figure 15 and Figure 16.

```
CONTEXT
  FrameworkEnvironment
SETS
  DOCUMENTS      //   set of all the document belongs to an organisation and created in environment uses the framework
  USERS      //   the set of an organisation users
  ORGANISATIONS      //   set of the organisations that uses the framework and wising to share document securely
  CA_PROVIDERS      //   set of Certificate Authorities and content providers
  SECURITY_POLICY_PARAMETER      //   set of security policy setting
  SECURITY_TECHNOLOGY_PARAMETER      //   set of encryption, hashing, and transmitting technologies parameters and setting
CONSTANTS
  document
  user
  secured
  shared
AXIOMS
  secureddocument   :   document∈ DOCUMENTS⟶ORGANISATIONS
  securityFlag   :   secured∈SECURITY_POLICY_PARAMETER⟶ORGANISATIONS
  Anyuser   :   user∈USERS⟶ORGANISATIONS
  sharingFlag   :   shared∈DOCUMENTS⟶ORGANISATIONS
END
```

Figure 14: The context of the framework as seen by the Rodin tool

```
MACHINE
  Securedcomputer
SEES
  FrameworkEnvironment
VARIABLES
  pc0_NewDocument
  pc0_OpenDocument
  pc0_ArchivedDocument
  pc0_ShareDocument
  newdocument
  confiscated
  secure
INVARIANTS
  typeof_pc0_NewDocument   :   pc0_NewDocument ∈ BOOL
  typeof_pc0_OpenDocument   :   pc0_OpenDocument ∈ BOOL
  typeof_pc0_ArchivedDocument   :   pc0_ArchivedDocument ∈ BOOL
  typeof_pc0_ShareDocument   :   pc0_ShareDocument ∈ BOOL
  distinct_states_in_pc0   :   partition
                               ({TRUE}, {pc0_NewDocument} ∩ {TRUE}, {pc0_OpenDocument} ∩ {TRUE}, {pc0_ArchivedDocument} ∩ {TRU
  typeof_newdocument   :   newdocument∈DOCUMENTS
  typeof_confiscated   :   confiscated∈SECURITY_POLICY_PARAMETER
  typeof_secure   :   secure∈BOOL
EVENTS
  OpenDocument   ≙
  STATUS
   ordinary
  WHEN
   isin_pc0_NewDocument   :   pc0_NewDocument = TRUE
  THEN
   leave_pc0_NewDocument   :   pc0_NewDocument ≔ FALSE
   enter_pc0_OpenDocument   :   pc0_OpenDocument ≔ TRUE
  END

  initialise   ≙
  STATUS
   ordinary
  BEGIN
   enter_pc0_NewDocument   :   pc0_NewDocument ≔ TRUE
  END

  ArchiveDocument   ≙
  STATUS
   ordinary
  WHEN
   isin_pc0_OpenDocument   :   pc0_OpenDocument = TRUE
   isin_pc0_ShareDocument   :   pc0_ShareDocument = TRUE
  THEN
   leave_pc0_OpenDocument   :   pc0_OpenDocument ≔ FALSE
   leave_pc0_ShareDocument   :   pc0_ShareDocument ≔ FALSE
   enter_pc0_ArchivedDocument   :   pc0_ArchivedDocument ≔ TRUE
  END

  SaveDocument   ≙
  STATUS
   ordinary
  WHEN
   isin_pc0_OpenDocument   :   pc0_OpenDocument = TRUE
  THEN
   leave_pc0_OpenDocument   :   pc0_OpenDocument ≔ FALSE
   enter_pc0_ShareDocument   :   pc0_ShareDocument ≔ TRUE
  END

  SettingupTheEnvironment   ≙
  STATUS
   ordinary
  ANY
   isPolicy
   SecurityMetaData
   SecurityTechPara
  WHERE
   isPolicy_type   :   isPolicy ∈ SECURITY_POLICY_PARAMETER
   SecurityMetaData_type   :   SecurityMetaData ∈ SECURITY_POLICY_PARAMETER
   SecurityTechPara_type   :   SecurityTechPara ∈ SECURITY_TECHNOLOGY_PARAMETER
   isin_pc0_NewDocument   :   pc0_NewDocument = TRUE
  THEN
   pc0_actions1   :   secured=secure
  END

  ConfiscateDocument   ≙
```

Figure 15: A formal model written in Event-B for any secure machine in an organisation (part 1)

```
STATUS
 ordinary
ANY
 Sharable
 confiscated
WHERE
 Sharable_type   :   Sharable ∈ SECURITY_POLICY_PARAMETER
 confiscated_type   :   confiscated ∈ SECURITY_POLICY_PARAMETER
 isin_pc0_OpenDocument   :   pc0_OpenDocument = TRUE
THEN
 confiscate1   :   confiscated= secure
END

AuthoriseUser   ≙
STATUS
 ordinary
ANY
 authorised
WHERE
 authorised_type   :   authorised ∈ SECURITY_POLICY_PARAMETER
 isin_pc0_NewDocument   :   pc0_NewDocument = TRUE
THEN
 skip
END

AuthoriseDocument   ≙
STATUS
 ordinary
BEGIN
 skip
END

END
```

Figure 16: A formal model written in Event-B for any secure machine in an organisation (part 2)

6.5 Summary

Formal methods are the way to go to verify and precisely define system requirements. Formal methods have been used in the safety and security of critical infrastructures. The real popwer of it lies in its flexability. It can give an overall system abstraction, and it can provide in-depth software class requirments and variables. So it essentially depends on the goal of using these methods. In this research, it was used to verify the general concept of the components and provide a landscape to start up from for future research or implementations.

Chapter 7: Conclusions and Future work

This chapter will present the main contribution of the research and conclusions from the data collection stage to confirm the research proposal. Additionally, a plan is provided in this chapter to prove that the proposed framework is actually implementable and works as it should.

7.1 The Contribution

The main contribution of this thesis is to introduce a general conceptual framework to secure documents when they leave the authoring organisation's firewalls. Document security is a critical issue for organisations in all the domains being interviews and surveys. However, some domains do not pay the adequate level of attention to ensure their documents are safe. Others do pay attention, but they cannot share their documents with others due to incompatibility or uncertainty. Unsecure document sharing is another word for information leakage. These challenges are identified from the literature review. The triangulation method was used to confirm these findings and to explore any overlooked issues.

7.2 Conclusions

The triangulation method used was composed of three main components. These components were the literature review, expert interviews and finally a survey of security professionals. The literature review ended with a list of challenges and issues facing current document security solutions, as mentioned in Chapter 2:. Based on these results, a framework was proposed to mitigate these challenges. The next step was to discuss these findings and the proposed framework components with the cyber security experts. The experts review added valuable comments to the framework which resulted in a more detailed workflow and functions, such as using SAML and XACML. Some of the framework components were modified and the cloud-computing concept was added. The experts confirmed the identified issues; however, they stated that legalisation of a document depends more on the policy and politics than the technology itself. As a result, the legalisation of a document is still an issue but it is out of this research's scope.

Different domains understand document security differently, so there is no one solution that addresses all the security priorities of each domain. This thesis presents a general landscape of cross-domain document security. The final product of this thesis is a model of the conceptual framework. This model was verified using a formal model to verify the consistency of the conceptual framework's components and functions. However, the model is not aiming to produce directly implementable software framework model, as that would be beyond the research aims and exceed the allotted time span of the study.

7.3 Future work

There are many interesting directions for future work to continue this research. Here we explain some of these directions and the expected outcomes of each one.

Implementing a software implementation of the conceptual framework:

This will need more investigation about the best way to implement the framework. There are many technologies that currently provide different aspects of security (CIA), depending on the level of dependency on the user, and their average level of computer proficiency in that specific domain. Each domain has different requirements and concepts.

An expert system for security validation and assessment:

This would be based on the findings of this research's initial stages, and adding more data from case studies and further interviews. This information would provide guidelines and feedback about what the organisation should do, according to its domain.

Appendices

Appendix A Survey Questions

Appendix B Expert Interview Questions

Appendix C Survey Data

Appendix A Survey Questions

Ethics reference number: **ERGO**/FoPSE/**13224**	Version: 1	Date: 2014-11-28
Study Title: What is an appropriate framework for securing documents when they go outside an organization?		
Investigator: Zeyad S. Aaber Alkhafajy		

A.1 General Questions:

1. **What is your organisation domain?**

□ Healthcare

□ Educational

□ Business

□ Government

□ Other, please specify: .....

2. **Which of these roles fits your fits your job description?**

□ Security Expert

□ Security Policy maker

□ IT-related staff

□ other, please specify: .....

3. **How long have you been working in Cyber Security?**

□ 0-5 years □ 6-10 years □ More than 10 years

A.2 Security related questions:

4. **In your professional experience, could you list the ways (mechanisms, solutions) that organizations use to secure document outside their network firewall? For example DRM.**

    a. .....

    b. .....

    c. .....

    d. .....

5. **The literature identified the following issues as the most frequent document vulnerability, could you indicate the frequency or risk that each issues according to your work experience?**

Issues	Frequency
Human negligence	□ Low □ Medium □ High □ Highest
Cross-domains	□ Low □ Medium □ High □ Highest
Legalization	□ Low □ Medium □ High □ Highest

Issues	Risk meter
Human negligence	□ Low □ Medium □ High □ Highest
Cross-domains	□ Low □ Medium □ High □ Highest
Legalization	□ Low □ Medium □ High □ Highest

6. **In your professional opinion, what is classified as "Human Negligence" in document leakage? Tick ✓ where its applicable**

    a. ☐ losing portable storage includes organization documents.

b. ☐ leaving his workstation without logging out.

c. ☐ using weak password, or using his password with other personal services.

d. ☐ Using public workstation to do organization work.

e. ☐ other, please list……

f. ☐ other, please list……

g. ☐ other, please list……

7. From your professional experience, what are the common mechanisms to ensure safe and secure sharing of documents with third parties according to documents security level (as described in question 12)?

☐ Non

☐ Password protection

☐ Encryption

☐ Digital Right Management

☐ Mix

☐ Other: Please list them ……

8. According to your knowledge, what is current legalization policy in your domain regarding authorizing and verifying document integrity?

☐ None. Hard copy and live signature only.

☐ Digital signature.

☐ Smart card key exchange.

☐ Trusted computing.

☐ Network attached security hardware module.

- □ Others, please list:
    - .....
    - .....
    - .....
    - .....
    - .....

Now here is concept of proposed framework to secure document outside it organization, please read the paragraph and answer the questions after.

The Tamper-Proof Framework (TPF) is composed of two parts: Active Document and System Functions. Active document is a document that has security active properties; hence it was created in a machine that has TPF installed. System Functions are secure background services in the operating system that monitor some system calls and translate the active properties into security rules. The TPF will depend on the cloud to provide secure customisation and delivery channel see Figure 17. The cloud can be replaced with any website to facilitate data sharing between the two organisations. In addition, the customisation of the TPF could be offline on the organisation A premises.

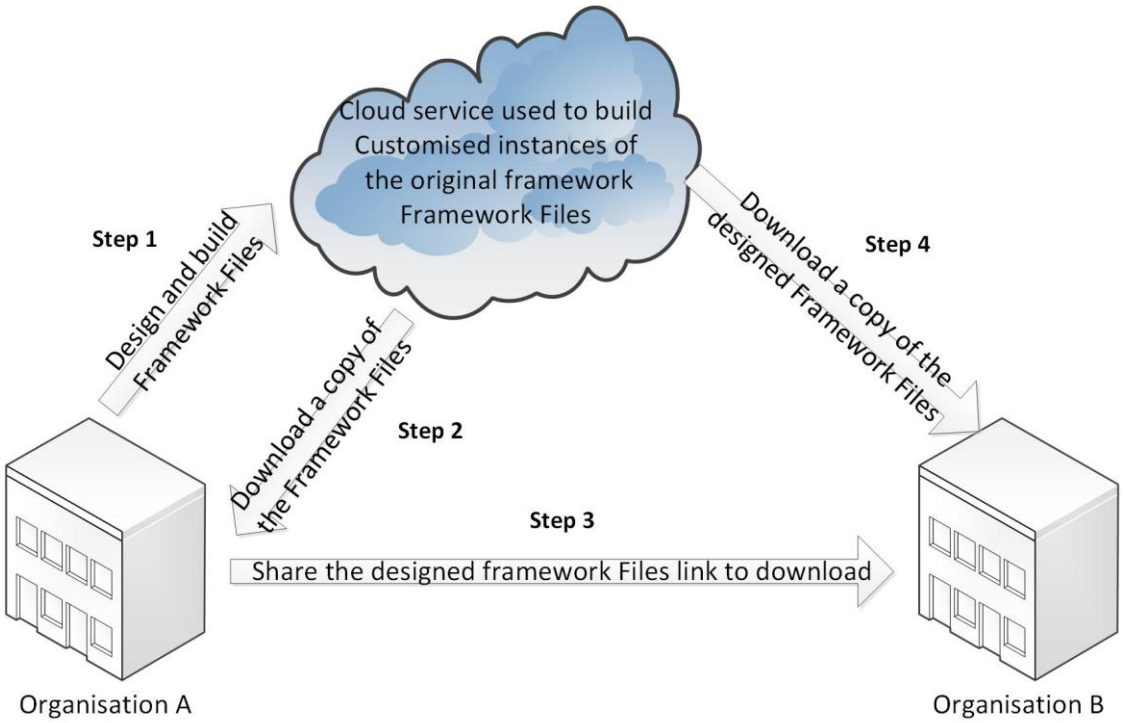

Figure 17 Cloud as customisation and delivery channel

New terms were used in the modified TPF. All the terms used in TPF context are explained here.

- **Active properties:** features inserted inside a document that can be read by software designed for that purpose.
- **Active Document:** a document that has active properties. These active properties were implanted by the Framework Files. The active document is composed of two parts: Public Information and Secret Information.
- **Public Information (Pre-Processing information):** plain text Default Security Policy and Active Properties that help the System Functions detect and authenticate the document's origin. This security policy is used by the System Functions to check for the initial security requirements.
- **Secret Information (Post-Processing information):** These encrypted data contain security parameters. These parameters are :
    - **Access control policy:** what is the required user authentication?
    - **Context parameter:** when, where, what and who is authorised to open this document.
    - **Verification code** for the active information in the Public Information. It is a hash value of the Public Information.

  - **Security Mechanism Parameter:** contains technical information about the encryption technique, content verification, and retention policy. This technical information is for the Actual Encrypted Data processing.
  - **Actual Encrypted Data:** part or all of the actual data of the document. This data is encrypted for a second time and can be accessed only if all the validations are passed.
- **Framework Files:** a customised version of the TPF distributed as executable code and working as a background service. It integrates with the operating system and the presentation software (Microsoft Office, PDF reader) to monitor and control their activities. In addition, it has System Functions that perform the security operations (encrypt, decrypt, and hashing).
- **Presentation software:** the default programs that are used to view the documents. This presentation software could be Microsoft Word, Adobe PDF Reader or Open Office.

Please answer these questions:

Table 12: List of questions used in the survey about the framework components

Component	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Q1. Using Cloud or web application as distribution channel					
Q2. Using System Function as black box for security processes					
Q3. Splitting the Security policy into two parts					
Q4. Including context information as part of the security policy					
Q5. Store and upload session information for context-aware analysis					

Component	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Q6. Using some error correction codes as availability measure (like Erasure code)					
Q7. Using multiple level of encryptions					
Q8. Integrate the Framework Files with the existing legacy software					

Appendix B Expert Interview Questions

Ethics reference number: **ERGO**/FoPSE/**13224**	Version: 1	Date: 2014-11-28
Study Title: What is an appropriate framework for securing documents when they go outside an organization?		
Investigator: Zeyad S. Aaber Alkhafajy		

B.1 Demographic questions

If you allow me, I would like to start with some background questions:

1. **What is your organisation domain?**

( ) Healthcare ( ) Educational ( ) Business ( ) Government

2. **Which of these roles fits your fits your job description?**

( ) Security Expert ( ) Security Policy maker ( ) IT-related staff

( ) Other, please specify::

_____

3. **How long have you been working in Cyber Security?**

( ) 0-5 years ( ) 6-10 years ( ) More than 10 years

B.2 General Questions:

2. From your experience working on cyber security, what are the current security mechanisms that you are aware of? Mechanisms that are used to secure a document in an organization?

3. From your experience working on cyber security, can you talk about the vulnerable and most frequent issues facing document security in an organization that you have worked in or conjunction with to solve these issues?

4. From your experience working on cyber security, could you describe in detail the mechanisms or solutions that organizations use to secure document outside their network firewall?

Thank you for your answers. Now, from the literature the following issues had been identified; Human negligence, Cross-domains security measures (compatibility) and Legalization. These defined as:

Human negligence: is a human act tensional or intentional that results in document leakage.

Cross-Domains: when a document is shared between two or more domains.

Legalization: the act of considering the document purport has the full legal value and can be treated as original or verified version of the original copy.

B.3 Human negligence:

2. In your opinion, what is considered as a Human negligence in document leakage?
3. From your experience, are you aware human negligence that commonly occurs in your domain?
4. From the previous question on human negligence, in your domain, what do writes in its IT security policy to prevent or mitigate Human negligence? In other words if you were asked to write a policy/ recommendation, what would it be?

B.4 Cross-domains:

1. In your opinion, does your domain normally share documents with third parties services?
    a. What is the security level of these shared documents?
2. In your opinion, what is the dependency level of your domain on third parties services?

3. In your opinion, what are the common mechanisms that used in your domain to ensure safe use of documents at the third parties infrastructure that reflects these documents security level?

B.5 Legalization:

1. Can you tell me about current legalization policy that you aware of regarding authorizing and verifying document integrity?

2. Can you tell me about what are the most vulnerable and frequent challenges facing this policy implementation?

B.6 Information leakage:

1. Keeping the points discussed before in mind (Human negligence; Cross-domains) in your opinion, what are the most frequent issues that causes document leakage?

2. From your experience, could you tell me a story that most recent or most devastating you aware of? Without mentioning names or entities.

3. From your experience, in your domain what is more important to secure the original document or the information in that document? In other word to what extend an organisation in your domain would compromise document availability to ensure its security.

Thank you for your time and answers. Now here is a structure diagram for the proposed solution to secure document outside its firewall boundaries. Next, describing concepts of the proposed solution. Finally asking the expert to comment on the components and concepts of the solution.

Appendix B

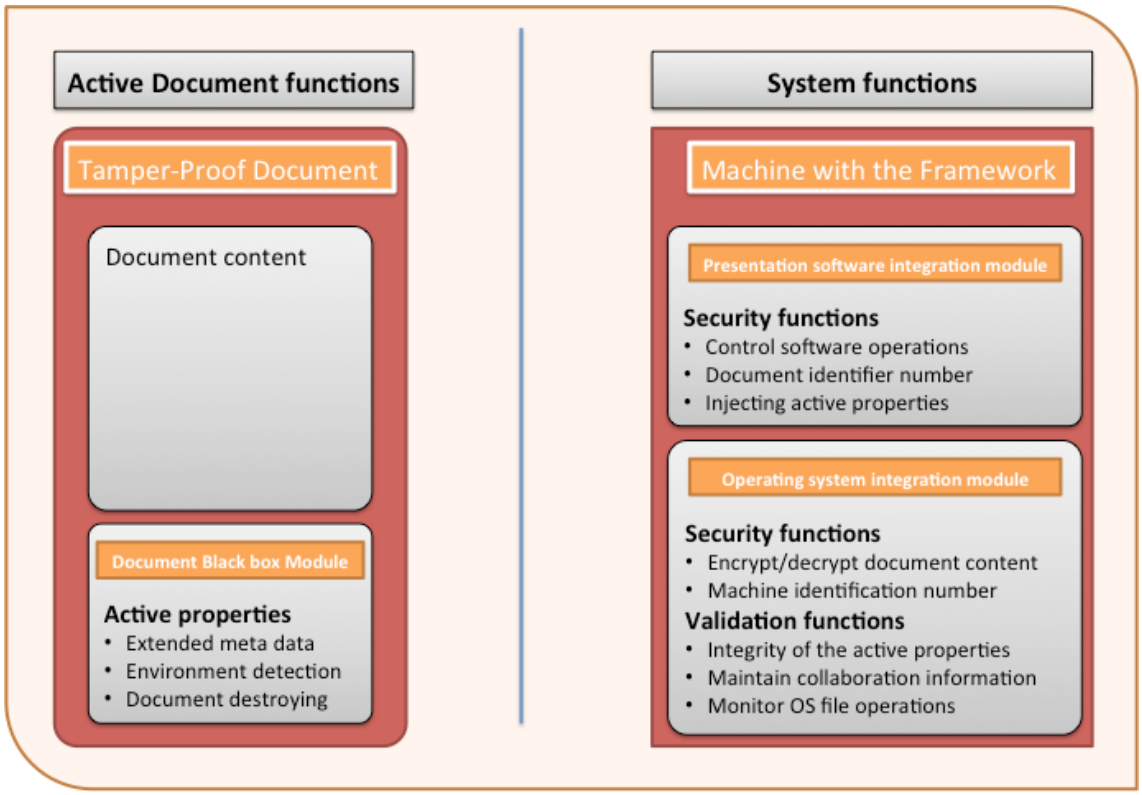

Figure 18: The Proposed Framework structure

Appendix C Survey Data

The survey is conducted using the University of Southampton iSurvey tool. This tool helped to create and collect the data from the participant. The data shown here is the raw data from that system. Each participant given and ID called “Case ID”. Due to the number of fields to be collected from each participant they cannot fit in one page in any given layout. So the data is splitting into two parts to fit the width of this thesis page.

C.1 Part one of the survey raw data

Table 13: First part of the raw survey data

case ID	domain	job description	e-document load	Human	Cross-domains	Legalization	other	Web-application-as-delivery-channel
1	healthcare	Security Expert	0% - 25%	5	5	5		5
2	education	Security Policy maker	26%-50%	4	4	4		4
3	business	IT-related staff	51%-75%	3	4	3		3
4	government	other	76%-100%	2	2	2	lost items	2
5	other	Security Expert	0% - 25%	1	5	1		1
6	healthcare	other	26%-50%	3	3	5		3
7	education	IT-related staff	51%-75%	3	5	4		3

8	business	other	76%-100%	3	4	3		3
9	government	Security Expert	0% - 25%	3	4	1		3
10	other	other	26%-50%	4	2	5		4
11	healthcare	IT-related staff	51%-75%	4	5	4		4
12	education	other	76%-100%	4	3	3		4
13	business	Security Expert	0% - 25%	4	5	1		4
14	government	other	26%-50%	4	4	5		4
15	other	IT-related staff	51%-75%	4	4	3		4
16	healthcare	other	76%-100%	4	2	1		4
17	education	Security Expert	0% - 25%	4	5	5		4
18	business	other	26%-50%	5	3	4		5
19	government	IT-related staff	51%-75%	5	5	3		5
20	other	other	76%-100%	5	2	1		5
21	other	Security Expert	0% - 25%	5	5	5		5

22	healthcare	Security Policy maker	26%-50%	5	3	3		5
23	education	IT-related staff	51%-75%	5	5	1		5
24	business	other	76%-100%	5	4	5		5
25	government	Security Expert	0% - 25%	5	4	4		4
26	other	other	26%-50%	4	2	3		3
27	other	IT-related staff	51%-75%	3	5	1		2
28	healthcare	other	76%-100%	2	3	5		1
29	education	Security Expert	0% - 25%	1	5	3		3
30	business	Security Policy maker	26%-50%	3	2	1		3
31	education	IT-related staff	0% - 25%	3	5	5		3
32	business	other	26%-50%	3	3	4		3
33	government	Security Expert	51%-75%	3	5	3		4
34	other	other	76%-100%	4	4	1	training	4
35	other	IT-related staff	0% - 25%	4	4	5	cost	4

36	healthcare	other	26%-50%	4	2	3		4
37	education	Security Expert	0% - 25%	4	5	1		4
38	business	other	26%-50%	4	3	5		4
39	education	IT-related staff	51%-75%	4	5	4		4
40	business	other	76%-100%	4	2	3		4
41	government	Security Expert	0% - 25%	4	5	1		5
42	other	Security Policy maker	26%-50%	5	2	5		5
43	other	IT-related staff	0% - 25%	5	5	3		5
44	healthcare	other	26%-50%	5	3	1		5
45	education	Security Expert	51%-75%	5	5	5		5
46	government	other	76%-100%	5	4	4		5
47	other	IT-related staff	0% - 25%	5	4	3		5
48	other	other	26%-50%	5	2	1		4
49	healthcare	Security Expert	51%-75%	5	5	5		3

50	education	Security Policy maker	76%-100%	4	3	3		2
51	government	IT-related staff	0% - 25%	3	5	5		1
52	other	other	26%-50%	2	2	4		3
53	other	Security Expert	0% - 25%	1	5	3		3
54	healthcare	Security Policy maker	26%-50%	3	3	2		4
55	education	IT-related staff	51%-75%	3	5	1		4
56	government	other	76%-100%	3	4	4		2
57	other	other	0% - 25%	3	4	3		5
58	other	IT-related staff	26%-50%	4	2	1		3
59	healthcare	other	0% - 25%	4	5	5		5
60	education	Security Expert	26%-50%	4	3	3		2
61	government	Security Policy maker	51%-75%	4	5	5		5
62	other	IT-related staff	76%-100%	4	2	4		4
63	other	other	0% - 25%	4	5	3		4

64	healthcare	Security Policy maker	26%-50%	4	3	2		4
65	education	Security Expert	0% - 25%	4	5	1		5
66	government	other	26%-50%	5	4	4		5
67	government	IT-related staff	51%-75%	5	5	3		5
68	other	other	76%-100%	5	4	1		5
69	other	Security Policy maker	0% - 25%	5	4	5		5
70	healthcare	Security Expert	76%-100%	5	2	3		5
71	education	Security Policy maker	0% - 25%	5	5	5		5
72	government	IT-related staff	26%-50%	5	3	4		4
73	government	other	0% - 25%	5	5	3		3
74	other	other	26%-50%	4	2	2		2
75	other	other	51%-75%	3	5	1		1
76	healthcare	Security Expert	76%-100%	2	3	4		3
77	education	Security Policy maker	0% - 25%	1	5	3		3

78	government	IT-related staff	26%-50%	3	4	1		4
79	government	other	0% - 25%	3	5	5		4
80	other	other	26%-50%	3	4	3		2
81	other	other	51%-75%	3	4	5	upgrading	5
82	education	Security Expert	76%-100%	4	2	4		3
83	government	Security Policy maker	0% - 25%	4	5	3		5
84	government	IT-related staff	26%-50%	4	3	2		2
85	other	other	0% - 25%	4	5	1		5
86	other	other	26%-50%	4	2	4		4
87	education	Security Policy maker	51%-75%	4	5	3		4
88	government	Security Expert	76%-100%	4	3	1		4
89	government	other	26%-50%	4	5	5		5
90	government	other	26%-50%	5	4	4		5

C.2 Part two of the survey raw data

Table 14: Second part of the raw survey data

case ID	system-functions	multipart-content-download	context-aware	session-log	availability-error-correction	multi Encryption	Legacy SW
1	4	5	5	5	4	4	4
2	4	4	4	4	3	3	3
3	4	4	3	4	2	2	2
4	2	2	2	2	1	1	1
5	2	5	1	5	3	3	3
6	3	3	3	3	3	3	3
7	3	5	3	5	3	3	3
8	3	4	3	4	3	3	3
9	3	4	3	4	4	4	4
10	3	2	4	2	4	4	4
11	3	5	4	5	4	4	4
12	3	3	4	3	4	4	4
13	4	5	4	5	4	4	4
14	4	4	4	4	4	4	4
15	4	4	4	2	4	4	4
16	2	2	4	5	4	4	4
17	4	5	4	3	5	5	5
18	4	3	5	5	5	5	5
19	3	5	5	2	5	5	5
20	3	2	5	5	5	5	5
21	3	5	5	3	2	2	2

22	3	3	5	5	5	5	5
23	4	5	5	4	3	3	3
24	4	4	5	5	5	5	5
25	4	4	5	4	4	4	4
26	3	2	4	4	4	4	4
27	3	5	3	2	2	2	2
28	3	3	2	5	5	5	5
29	3	5	1	3	3	3	3
30	3	2	3	5	5	5	5
31	3	5	3	2	4	4	4
32	3	3	5	5	4	4	4
33	3	5	4	4	2	2	2
34	4	4	3	2	5	5	5
35	4	4	2	4	3	3	3
36	4	2	1	2	5	5	5
37	4	5	3	5	2	2	2
38	4	3	3	3	5	5	5
39	4	5	3	5	3	3	3
40	3	2	3	2	5	5	5
41	4	5	4	5	4	4	4
42	4	2	4	3	4	4	4
43	4	5	4	5	2	2	2
44	4	3	4	4	5	5	5
45	4	5	4	5	3	3	3
46	4	4	4	4	5	5	5
47	3	4	4	4	2	2	2

48	4	2	4	2	3	3	3
49	4	5	5	5	4	4	4
50	4	3	5	3	4	4	4
51	4	5	5	5	4	4	4
52	4	2	5	2	4	4	4
53	3	5	5	5	4	4	4
54	2	3	5	4	4	4	4
55	3	5	5	4	4	4	4
56	3	4	5	4	4	4	4
57	3	4	4	2	5	5	5
58	2	2	3	5	5	5	5
59	2	5	2	3	5	5	5
60	3	3	1	5	5	5	5
61	3	5	3	2	2	2	2
62	4	2	3	5	5	5	5
63	4	5	5	3	3	3	3
64	4	3	4	5	5	5	5
65	2	5	3	4	4	4	4
66	2	4	2	5	4	4	4
67	2	5	1	4	2	2	2
68	2	4	3	4	5	5	5
69	4	4	3	2	3	3	3
70	4	2	3	5	5	5	5
71	2	5	3	3	4	4	4
72	4	3	4	5	4	4	4
73	4	5	4	2	3	3	3

74	3	2	4	5	4	4	4
75	3	5	4	4	4	4	4
76	3	3	4	4	4	4	4
77	3	5	4	2	4	4	4
78	3	4	4	5	4	4	4
79	3	5	4	3	4	4	4
80	4	4	5	5	4	4	4
81	4	4	5	2	4	4	4
82	4	2	5	5	5	5	5
83	4	5	5	3	5	5	5
84	4	3	5	5	5	5	5
85	4	5	5	4	5	5	5
86	4	2	5	5	2	2	2
87	4	5	5	4	5	5	5
88	4	3	4	4	3	3	3
89	4	5	3	2	5	5	5
90	2	4	2	5	4	4	4

References

Abiteboul, Serge, Omar Benjelloun, and Tova Milo. 2008. “The Active XML Project: An Overview.” *VLDB Journal* 17 (5): 1019–40. doi:10.1007/s00778-007-0049-y.

Abiteboul, Serge, Pierre Bourhis, and Bogdan Marinoiu. 2009. “Efficient Maintenance Techniques for Views over Active Documents.” In *Proceedings of the 12th International Conference on Extending Database Technology Advances in Database Technology - EDBT ’09*, 1076. New York, New York, USA: ACM Press. doi:10.1145/1516360.1516483.

Abrial, Jean-Raymond. 1996. *The B-Book: Assigning Programs to Meanings*. Cambridge University Press.

Abrial, Jean-Raymond. 2007. “Formal Methods: Theory Becoming Practice.” *Journal of Universal Computer Science* 13 (5): 619–28. doi:10.3217/jucs-013-05-0619.

Abrial, Jean-Raymond, Michael Butler, Stefan Hallerstede, and Laurent Voisin. 2006. “An Open Extensible Tool Environment for Event-B.” *Formal Methods and Software Engineering*. *Lecture Notes in Computer Science Volume 4260*, 588–605. doi:10.1007/11901433_32.

Abrial, Jean-Raymond, and Stefan Hallerstede. 2007. “Refinement, Decomposition, and Instantiation of Discrete Models: Application to Event-B.” *Fundamenta Informaticae* 77: 1–28. http://iospress.metapress.com/index/c74274t385t6r72r.pdf.

Albisser, A M, J B Albisser, and L Parker. 2003. “Patient Confidentiality, Data Security, and Provider Liabilities in Diabetes Management.” *Diabetes Technol.Ther.* 5 (1520–9156 (Print) LA–eng PT–Journal Article PT–Review SB–IM): 631–40. doi:10.1089/152091503322250659.

Anwar, Shahid, Zakira Inayat, Mohamad Fadli Zolkipli, Jasni Mohamad Zain, Abdullah Gani, Nor Badrul Anuar, Muhammad Khurram Khan, and Victor Chang. 2017. “Cross-VM Cache-Based Side Channel Attacks and Proposed Prevention Mechanisms: A Survey.” *Journal of Network and Computer Applications*. doi:10.1016/j.jnca.2017.06.001.

List of References

Anwar, Shahid, Jasni Mohamad Zain, Mohamad Fadli Zolkipli, Zakira Inayat, Suleman Khan, Bokolo Anthony, and Victor Chang. 2017. “From Intrusion Detection to an Intrusion Response System: Fundamentals, Requirements, and Future Directions.” *Algorithms* 10 (2). Multidisciplinary Digital Publishing Institute: 39.

Baldwin, R.W. 1990. “Naming and Grouping Privileges to Simplify Security Management in Large Databases.” In *Proceedings. 1990 IEEE Computer Society Symposium on Research in Security and Privacy*, 116–32. IEEE. doi:10.1109/RISP.1990.63844.

Beth, Evert W. 1970. *Formal Methods*. *Philosophia Mathematica*. Vol. s1-10. Dordrecht: Springer Netherlands. doi:10.1007/978-94-010-3269-8.

Bhattacherjee, Anol. 2012. *Social Science Research: Principles, Methods, and Practices*. *Health Research Policy and Systems BioMed Central*. Vol. 9. doi:10.1186/1478-4505-9-2.

Boyatzis, Richard E. 1998. “Transforming Qualitative Information: Thematic Analysis and Code Development.” *Transforming Qualitative Information Thematic Analysis and Code Development*. doi:10.1177/102831539700100211.

Boyle, John M, Eric S Maiwald, and David W Snow. 1996. “Apparatus and Method for Providing Multi-Level Security for Communication among Computers and Terminals on a Network.” Google Patents.

Braun, Virginia, and Victoria Clarke. 2006. “Braun, V ., Clarke, V .Using Thematic Analysis in Psychology., 3:2 (2006), 77-101.” *Qualitative Research in Psychology* 3: 77–101. doi:10.1191/1478088706qp063oa.

Butler, M J, M Leuschel, S Lo Presti, and P Turner. 2004. “The Use of Formal Methods in the Analysis of Trust (Position Paper).” *Trust Management, Lecture Notes in Computer Science* 2995: 333–39.

Butler, Michael, and Stefan Hallerstede. 2007. “The Rodin Formal Modelling Tool.” *BCSFACS Christmas 2007 Meeting Formal Methods In Industry London*, 1–5. http://eprints.ecs.soton.ac.uk/14949/.

Chang, Victor, Yen-Hung Kuo, and Muthu Ramachandran. 2016. “Cloud Computing Adoption Framework: A Security Framework for Business Clouds.” *Future Generation Computer Systems* 57 (April): 24–41. doi:10.1016/j.future.2015.09.031.

Chang, Victor, and Muthu Ramachandran. 2016. “Towards Achieving Data Security with the Cloud Computing Adoption Framework.” *IEEE Transactions on Services Computing* 9 (1): 138–51. doi:10.1109/TSC.2015.2491281.

Chang, Victor, Muthu Ramachandran, Robert J Walters, and Gary Wills. 2017. *Enterprise Security: Second International Workshop, ES 2015, Vancouver, BC, Canada, November 30–December 3, 2015, Revised Selected Papers*. Vol. 10131. Springer.

Chang, Victor, Robert John Walters, and Gary Wills. 2013. “The Development That Leads to the Cloud Computing Business Framework.” *International Journal of Information Management* 33 (3). Elsevier Ltd: 524–38. doi:10.1016/j.ijinfomgt.2013.01.005.

Choi, Jae-Myeong, and Sang-Soo Yeo. 2012. “Automated Security Level Conversion on Security Documents in Different Domains.” In *2012 15th International Conference on Network-Based Information Systems*, 681–84. IEEE. doi:10.1109/NBiS.2012.144.

Chul-Ki Nam, and J.-H.J. Bae. 2002. “A Framework for Processing Active Documents.” In *Proceedings 6th Russian-Korean International Symposium on Science and Technology. KORUS-2002 (Cat. No.02EX565)*, 122–25. IEEE. doi:10.1109/KORUS.2002.1027977.

Creswell, John W. 2007. *Qualitative Inquiry and Research Design: Choosing Among Five Approaches. Australasian Emergency Nursing Journal*. Vol. 11. doi:10.1016/j.aenj.2008.02.005.

Davies, Anthony C. 1988. “Introduction to Formal Methods of Software Design.” *Microprocessors and Microsystems* 12 (10): 547–53.

Detlefs, David L. 1995. “An Overview of the Extended Static Checking System.” In *In Proceedings of the First Workshop on Formal Methods in Software Practice*, 1–9. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.31.58.

Dourish, Paul. 2003. “The Appropriation of Interactive Technologies: Some Lessons from Placeless Documents.” *Computer Supported Cooperative Work (CSCW)* 12 (4): 465–90. doi:10.1023/A:1026149119426.

Dourish, Paul, W Keith Edwards, Anthony LaMarca, John Lamping, Karin Petersen, Michael Salisbury, Douglas B. Terry, and James Thornton. 2000. “Extending Document Management Systems with User-Specific Active Properties.” *ACM Transactions on Information Systems* 18 (2): 140–70. doi:10.1145/348751.348758.

Easttom, Chuck. 2012. *Computer Security Fundamentals*. Edited by Tonya Simpson, Betsy Brown, Sandra Schroeder, and Keith Cline. Second. Indianapolis, Indiana: Pearson.

EMC Corporation. 2008. “The Challenges of Deploying Information Rights Management Across the Enterprise.” Bedford, MA.

Eom, J ho. 2012. “Modeling of Document Security Checkpoint for Preventing Leakage of Military Information.” *International Journal of Security and Its Applications* 6 (4): 175–83.

Faraj Al-Janabi, Sufyan T., and Hussein Khalid Abd-Alrazzaq. 2011. “Combining Mediated and Identity-Based Cryptography for Securing E-Mail.” In *Communications in Computer and Information Science*, 194 CCIS:1–15. doi:10.1007/978-3-642-22603-8_1.

Faul, Franz, Edgar Erdfelder, Axel Buchner, and Albert-Georg Lang. 2009. “Statistical Power Analyses Using G*Power 3.1: Tests for Correlation and Regression Analyses.” *Behavior Research Methods* 41 (4): 1149–60. doi:10.3758/BRM.41.4.1149.

Galton, Antony. 1987. “Temporal Logic and Computer Science: An Overview.” In *Temporal Logics and Their Applications*, 1–52.

Giampaolo, D. 1999. *Practical File System Design with the Be File System*. 1st ed. San Mateo, CA: Morgan Kaufmann.

Gleichauf, RE, and WA Randall. 2001. Method and system for adaptive network security using network vulnerability assessment. *US Patent …*, issued 2001.

Goguen, Joseph A., and Grant Malcolm. 2000. *Software Engineering with OBJ: Algebraic Specification in Action*. *Software Engineering with OBJ: Algebraic Specification in Action*. Vol. 2. Springer Science & Business Media.

Goldblatt, Robert. 2003. "Mathematical Modal Logic: A View of Its Evolution." *Journal of Applied Logic*. doi:10.1016/S1570-8683(03)00008-9.

Greatrex, Jonny. 2010. "Bungling West Midlands Medics Lose 12,000 Private Patient Records - Birmingham Mail." *Sunday Mercury*.

Guest, G, A Bunce, and L Johnson. 2006. "How Many Interviews Are Enough?" *Field Methods* 18 (1): 59. doi:10.1177/1525822X05279903.

Hallerstede, Stefan. 2011. "On the Purpose of Event-B Proof Obligations." In *Formal Aspects of Computing*, 23:133–50. Springer. doi:10.1007/s00165-009-0138-3.

Harel, David. 1988. "On Visual Formalisms." *Commun. ACM* 31 (5): 514–530. doi:10.1145/42411.42414.

Heitmeyer, Constance L. 2009. "On the Role of Formal Methods in Software Certification: An Experience Report." *Electronic Notes in Theoretical Computer Science* 238 (4): 3–9. doi:10.1016/j.entcs.2009.09.001.

Hershey, Paul C, Donald B Johnson, An V Le, Stephen M Matyas, John G Waclawsky, and John D Wilkins. 1995. "Network Security System and Method Using a Parallel Finite State Machine Adaptive Active Monitor and Responder." Google Patents.

Hoare, C. A. R. 1978. "Communicating Sequential Processes." *Communications of the ACM* 21 (8): 666–77. doi:10.1145/359576.359585.

Ii, Albert. 2001. "Lecture 17 : Formal Modeling Methods What Are Formal Methods ? Example Formal Methods : Formal Methods in Software Engineering Validation : Predicting Behavior Three Traditions ... (1) Formal Specification Languages Three Basic Flavours : (2) Reactive S."

Iso Iec. 2005. "BS ISO/IEC 27002:2005 Information Technology — Security Techniques — Code of Practice for Information Security Management." *ISO*.

Jones, Cliff B. 1995. *Teaching Notes: Systematic Software Development Using VDM*. Vol. 2. Prentice-Hall Englewood Cliffs, NJ.

Jr, R E Wesinger, and C D Coley. 1999. “Firewall Providing Enhanced Network Security and User Transparency.” *US Patent 5,898,830*. Google Patents.

Jupp, Victor. 2006. *The SAGE Dictionary of Social Research Methods*. *The SAGE Dictionary of Social Research Methods*. 1 Oliver’s Yard, 55 City Road, London England EC1Y 1SP United Kingdom: SAGE Publications, Ltd. doi:10.4135/9780857020116.

Kaplan, B, and D Duchon. 1988. “Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study.” *MIS Quarterly* 12 (4): 571–86. doi:0166.

Kellner, Marc I. 1988. “Representation Formalisms for Software Process Modelling.” *Software Process Workshop on Representing and Enacting the Software Process*, 93–96. doi:10.1145/75110.75125.

Kofax. 2014. “Document Management.” *Kofax Inc*. http://www.kofax.com/electronic-signature.

Kruntchen, P. 1995. “Architectural Blueprints–the” 4+ 1” View Model of Software Architecture.” *IEEE Software* 12 (November): 42–50. doi:10.1145/216591.216611.

LaMarca, A, WK Edwards, and Paul Dourish. 1999. “Taking the Work out of Workflow: Mechanisms for Document-Centered Collaboration.” *ECSCW’99*, no. September: 12–16. doi:10.1007/978-94-011-4441-4_1.

Lampson, Butler W. 1974. “Protection.” *ACM SIGOPS Operating Systems Review* 8 (1). New York, NY, USA: ACM: 18–24. doi:10.1145/775265.775268.

Littlefield, and Rowman. 1997. *Telecommunications: Glossary of Telecommunication Terms*. Federal Standard. Division, National Communications System (U.S.). Technology & Standards Section, United States. General Services Administration. Information Technology.

Liu, Debin, XF Wang, and LJ Camp. 2009. “Mitigating Inadvertent Insider Threats with Incentives.” *Financial Cryptography and Data Security*, 1–16. doi:10.1007/978-3-642-03549-4_1.

Liu, X., Z. Chen, H. Yang, H. Zedan, and W.C. Chu. 1997. “A Design Framework for System Re-Engineering.” In *Proceedings of Joint 4th International Computer Science Conference and 4th Asia Pacific Software Engineering Conference*, 342–52. IEEE. doi:10.1109/APSEC.1997.640191.

Lorch, Markus, Seth Proctor, Rebekah Lepro, Dennis Kafura, and Sumit Shah. 2003. “First Experiences Using XACML for Access Control in Distributed Systems.” *Proceedings of the 2003 ACM Workshop on XML Security XMLSEC 03* 47 (C): 25. doi:10.1145/968559.968563.

Macaskill, Wen, and Gabriel Dance. 2013. “NSA Files Decoded: Edward Snowden’s Surveillance Revelations Explained | World News | Theguardian.com.” *The Guardian*.

Majchrzak, Tim A., and Claus A. Usener. 2011. “Evaluating E-Assessment for Exercises That Require Higher-Order Cognitive Skills.” In *Proceedings of the Annual Hawaii International Conference on System Sciences*, 48–57. doi:10.1109/HICSS.2012.252.

Manasdeep. 2012. “Information Rights Management Implementation and Challenges.” Mumbai.

Mao, Wenbo. 2003. *Modern Cryptography: Theory and Practice*. *Theory and Practice*. Vol. 170. doi:10.1093/aje/kwp410.

McKinley, Barton. 2000. “The ABCs of PKI.” *Network World*.

Miller, Lawrence C, and Peter H Gregory. 2016. *CISSP for Dummies*. John Wiley & Sons.

Munier, Manuel, Vincent Lalanne, Pierre-yves Ardoy, and Magali Ricarde. 2014. “Legal Issues About Metadata Data Privacy vs Information Security.” Edited by Joaquin Garcia-Alfaro, Georgios Lioudakis, Nora Cuppens-Boulahia, Simon Foley, and William M. Fitzgerald, Lecture Notes in Computer Science, 8247. Berlin, Heidelberg: Springer Berlin Heidelberg: 162–77. doi:10.1007/978-3-642-54568-9 11.

Munier, Manuel, Vincent Lalanne, and Magali Ricarde. 2012. “Self-Protecting Documents for Cloud Storage Security.” In *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, 1231–38. IEEE. doi:10.1109/TrustCom.2012.261.

Neumann, Christoph P., and Richard Lenz. 2010. “The Alpha-Flow Use-Case of Breast Cancer Treatment - Modeling Inter-Institutional Healthcare Workflows by Active Documents.” In *2010 19th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, 6:17–22. IEEE. doi:10.1109/WETICE.2010.8.

NIST. 2014. “Security and Privacy Controls for Federal Information Systems and Organizations Security and Privacy Controls for Federal Information Systems and Organizations.” *Sp-800-53Ar4*, 400+. doi:10.6028/NIST.SP.800-53Ar4.

Platform, Eclipse. 2013. “Eclipse Platform Compatibility.” http://www.eclipse.org/eclipse/.

Preece, Jenny, Yvonne Rogers, and Helen Sharp. 2002. *Interaction Design: Beyond Human-Computer Interaction*. *Design*. Vol. 18. doi:10.1016/S0010-4485(86)80021-5.

Quint, Vincent, and I Vatton. 1994. “Making Structured Documents Active.” *Electronic Publishing* 7 (November 1993): 55–74.

Recker, Jan. 2013. *Scientific Research in Information Systems: A Beginner’s Guide*. *Scientific Research in Information Systems*. doi:10.1007/978-3-642-30048-6.

Reisig, W. 2003. “An Introduction to Petri Nets.” *International Journal of General Systems* 32: 565–82. doi:10.1007/0-8176-4488-1_2.

Richter, Jeffrey, and Luis Felipe Cabrera. 1998. “A File System for the 21ST Century: Previewing the Windows NT 5.0 File System-Many Programming Tasks Will Be Simplified by Innovations in NTFS, the Windows NT 5.0 File System.” *Microsoft Systems Journal-US Edition*. [Redmond, Wash.]: Microsoft Corp., c1986-c1999., 19–36.

Rizzo, Luigi. 1997. “Effective Erasure Codes for Reliable Computer Communication Protocols.” *SIGCOMM Comput. Commun. Rev.* 27 (2). New York, NY, USA: ACM: 24–36. doi:10.1145/263876.263881.

Russo, Aryldo G. 2011. *Modeling in Event-B - System and Software Engineering by Jean-Raymond Abrial. Cs_Konzepte*. Vol. 36. doi:10.1145/1943371.1943378.

Said, Mar Yah, Michael Butler, and Colin Snook. 2009. "Class and State Machine Refnement in UML-B." *Integration of Model-Based Formal Methods and Tools (Workshop at iFM 2009)*.

Samarati, Pierangela, and SC de Vimercati. 2001. "Access Control: Policies, Models, and Mechanisms." *Foundations of Security Analysis and ...*. doi:10.1007/3-540-45608-2_3.

Schmidt, AU, and Z Loebl. 2005. "Legal Security for Transformations of Signed Documents: Fundamental Concepts." *Public Key Infrastructure*, 255–70. doi:10.1007/11533733_18.

Smallwood, RF. 2012. *Safeguarding Critical E-Documents: Implementing a Program for Securing Confidential Information Assets*. Hoboken, New Jersey: John Wiley & Sons, Inc.

Software Engineering Institute. 2013. "2013 US State of Cybercrime Survey."

Sprague, Ralph H. 1995. "Electronic Document Management: Challenges and Opportunities for Information Systems Managers." *MIS Quarterly* 19 (1): 29. doi:10.2307/249710.

Stamp, M. 2011. *Information Security: Principles and Practice*. *Linguistics*. Vol. 11. Hoboken, New Jersey: JohnWiley & Sons, Inc.

Strnadl, Christoph F. 2006. "Communicating and Mobile Systems. The P-Calculus." *Information Systems Management* 23 (4): 67–77. http://www.tandfonline.com/doi/abs/10.1201/1078.10580530/46352.23.4.20060901/95115.9.

Sun, Gang, Victor Chang, Muthu Ramachandran, Zhili Sun, Gangmin Li, Hongfang Yu, and Dan Liao. 2016. "Efficient Location Privacy Algorithm for Internet of Things (IoT) Services and Applications." *Journal of Network and Computer Applications*. doi:10.1016/j.jnca.2016.10.011.

Sun, Gang, Dan Liao, Hui Li, Hongfang Yu, and Victor Chang. 2017. “L2P2: A Location-Label Based Approach for Privacy Preserving in LBS.” *Future Generation Computer Systems* 74: 375–84. doi:10.1016/j.future.2016.08.023.

Todorova, Aneliya, and CP Neumann. 2011. “Alpha-Props: A Rule-Based Approach to ‘Active Properties’ for Document-Oriented Process Support in Inter-Institutional Environments.” *Lecture Notes in Informatics (LNI) ….*

Toorani, Mohsen, and Ali Asghar Beheshti Shirazi. 2008. “LPKI - A Lightweight Public Key Infrastructure for the Mobile Environments.” In *2008 11th IEEE Singapore International Conference on Communication Systems, ICCS 2008*, 162–66. doi:10.1109/ICCS.2008.4737164.

Trcek, Denis. 2006. *Managing Information Systems Security and Privacy*. Springer Science & Business Media.

Uk Parliament. 1998. “Data Protection Act 1998.” *Changes* 2002 (3): 1–128. doi:10.1080/713673366.

Vacca, John R. 2005. “Public Key Infrastructure: Building Trusted Applications and Web Services.” *Computing Reviews* 46 (2): 100.

Vaglini, G. 1991. *Communication and Concurrency*. *Information and Software Technology*. Vol. 33. Prentice hall New York etc. doi:10.1016/0950-5849(91)90083-N.

Vinet, Luc, and Alexei Zhedanov. 2011. “A ?missing? Family of Classical Orthogonal Polynomials.” *Journal of Physics A: Mathematical and Theoretical* 44 (8): 85201. doi:10.1088/1751-8113/44/8/085201.

Vliet, Hans Van. 2007. “Software Engineering: Principles and Practice.” *Wiley*, no. 3: 726. doi:10.1016/0950-5849(94)90015-9.

Wang, Zhiwei, Cheng Cao, Nianhua Yang, and Victor Chang. 2016. “ABE with Improved Auxiliary Input for Big Data Security.” *Journal of Computer and System Sciences*, December. doi:10.1016/j.jcss.2016.12.006.

Whitman, M, and H Mattord. 2011. *Principles of Information Security, 4th Edition*. 4th ed. Boston, MA, USA: Course Technology, Cengage Learning.

Woodcock, Jim, and Jim Davies. 1996. *Using Z: Specification, Refinement, and Proof*. Vol. 39. Prentice Hall Englewood Cliffs.

Woodcock, Jim, Peter Gorm Larsen, Juan Bicarregui, and John Fitzgerald. 2009. "Formal Methods: Practice and Experience." *ACM Computing Surveys* 41 (4): 1–36. doi:10.1145/1592434.1592436.

Wright, Stephen. 2008. "Using EventB to Create a Virtual Machine Instruction Set Architecture." *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5238 LNCS: 265–79. doi:10.1007/978-3-540-87603-8_21.

Yeun, Chan Yeob, and Tim Farnham. 2001. "Secure M-Commerce with Wpki." In *Proceedings of 1st International Workshop for Asian PKI*. Citeseer.

Yoder, Joseph W. J.W., and Jeffrey Barcalow. 1998. "Architectural Patterns for Enabling Application Security." *Proceedings of PLoP 1997* 51: 31.

Zeadally, Sherali, Byunggu Yu, Dong Hyun Jeong, and Lily Liang. 2012. "Detecting Insider Threats: Solutions and Trends." *Information Security Journal: A Global Perspective* 21 (4): 183–92. doi:10.1080/19393555.2011.654318.