Information Security Testing Plan Model for Secure Software Development

Bachelor's Thesis

Degree Programme in Business Information Technology

Autumn, 2022

Tommi Sipponen

HÄMEEN AMMATTIKORKEAKOULU
HÄME UNIVERSITY OF APPLIED SCIENCES

Tietojenkäsittelyn koulutus

Author Tommi Sipponen **Year** 2022

Subject Information Security Testing Plan Model for Secure Software Development

Supervisor Lasse Seppänen

TIIVISTELMÄ

Tämä opinnäytetyö tutkii tietoturvatestausmallin toteuttamista julkiselle toimijalle, Oikeusrekisterikeskukselle. Työn pääasiallinen tuotos on tukevan teoreettisen dokumentaation lisäksi testausdokumentti, josta asiakas saa pohjan käytettäväksi tuleville projekteille. Ensimmäisessä luvussa tarkastellaan projektissa käytettyjä työkaluja.

Toisessa luvussa tarkastellaan tietoturvan historiaa 1970-luvulta nykypäivään. Historian tutkiminen auttoi kokonaiskuvan muodostamisessa ja toi tarvittavaa kontekstia testaukseen, sillä testausta ei yleensä käsitellä tietoturvan näkökulmasta. Koska kyseessä on julkinen toimija, kolmas luku käsittelee valtionhallinnon tietoturvaprosesseja (VAHTI, KATAKRI) sekä tarkastelee olemassa olevia suojaustasoja, joita valtionhallinnossa käytetään sen varmistamiseksi, että dokumentteja lukevat vain valtuutetut henkilöt. Tämä oli tärkeää, sillä valtionhallinnon prosessit ja standardit ovat erilaisia kuin yksityisellä sektorilla, jolloin tarkastelu oli tarpeen.

Viidennessä luvussa tutustutaan prosessin määritelmään ja sen kehittämiseen tarvittavan kontekstin vuoksi. Kuudennessa tarkastellaan sovelluskehityksen historiaa tietoturvaan liittyvin osin. Tämä helpotti dokumentin muotoilua testausdokumentin

suuntaan. Lopuksi katsotaan olemassa olevaa prosessia ja tuloksena syntynyttä testausdokumenttia, joka toimitettiin asiakkaalle, sekä johtopäätöksiä.

Avainsanat Information Security, Testing, Models, Process Improvement, OWASP

Sivut 53 sivua, josta 15 sivua liitteitä

HÄMEEN AMMATTIKORKEAKOULU
HÄME UNIVERSITY OF APPLIED SCIENCES

Degree Programme in Business Information Technology

Author Tommi Sipponen **Year** 2022

Subject Information Security Testing Plan Model for Secure Software Development

Supervisor Lasse Seppänen

ABSTRACT

This thesis examines information security testing for the purpose of state client, Legal Register Centre. The main purpose of this thesis is to produce a document that the client can use for information security testing potentially utilised in relevant projects. First chapter will define tools used for this project.

The second chapter is about the history of information security from the 1970s to the present day. This was important since literature and processes governing testing tend to leave out information security aspect, so this was important for the author in order to establish context. Due to the nature of the governmental client, one chapter is dedicated in looking at relevant governmental security processes and frameworks such as VAHTI and KATAKRI as they relate to the subject. This was important due to difference in governmental and private sector standards, so a deeper look was necessary in order to follow the established frameworks of the client.

As security is paramount, there are multitude of factors to consider for public side projects in terms of security which was a surprise to the author. Once the context of information security has been established, a brief look is taken at the definition of

process and the process of improving them. Finally, a history of software testing in a broader sense to help establish context as it relates to information security. This made it easier to conceive the final testing document in a proper form. Final two chapters discuss the process of creation and the final document produced.

Keywords Information Security, Testing, Models, Process Improvement, OWASP

Pages 53 pages including appendices 15 pages

Abbreviations & Terminology

Abbr.	Term	Explanation
	Attack vector/Attack surface	An attack vector, or threat vector, is a way for attackers to enter a network or system.
BPMN	Business Process Modelling Notation	Visual modelling language for business analysis applications and specifying enterprise process workflows, which is an open standard notation for graphical flowcharts that is used to define business process workflows.
	Cyberspace	Imaginary, intangible, virtual-reality realm where (in general) computer-communications and simulations and (in particular) internet activity takes place.
IEEE		IEEE is the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity.
ICT	Information and Communications Technology	ICT refers to technologies that provide access to information through telecommunications. It is similar to Information Technology (IT) but focuses primarily on communication technologies. This includes the Internet, wireless networks, cell phones, and other communication mediums.
ISSEA	International Systems Security Engineering Association	The Information Systems Security Association (ISSA) is a not-for-profit, international organization of information security professionals and practitioners. It provides educational forums, publications, and peer interaction opportunities that enhance the knowledge, skill, and professional growth of its members.
ISTQB	International Software Testing Qualifications Board	The Board is the leading global certification scheme in the field of software testing providing certificate for software testing professionals.
IOT	Internet of Things	The Internet of Things (IoT) describes the network of physical objects "things" that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. These devices range from ordinary household objects to sophisticated industrial tools.

IT	Information Technology	The development, maintenance, and use of computer software, systems, and networks.
KATAKRI		Tool that government uses to audit their information security standards.
OS	Operating System	A software that communicates with the hardware and allows other programs to run.
OWASP	The Open Web Application Security Project	A non-profit foundation that works to improve the security of software. Through community-led open-source software projects, hundreds of local chapters worldwide, tens of thousands of members, and leading educational and training conferences, the OWASP Foundation is the source for developers and technologists to secure the web.
NIST	National Institute of Standards and Technology	The National Institute of Standards and Technology (NIST) was founded in 1901 and is now part of the U.S. Department of Commerce. NIST is one of the nation's oldest physical science laboratories
	Phishing	Phishing is a cybercrime in which a target or targets are contacted by email, telephone, or text message by someone posing as a legitimate institution to lure individuals into providing sensitive data such as personally identifiable information, banking and credit card details, and passwords. The information is then used to access important accounts and can result in identity theft and financial loss.
-	Redundant system	Redundancy involves the duplication of system functions to ensure that those functions will survive some level of damage to the system; in the context of communications systems, it is used to describe a system that can maintain communications with the surface when a single communication path is disrupted.
SAMM	Software Assurance Maturity Model	Tool for analysing and supporting complete software lifecycle in an effective and measurable way
	Scope creep	How a project’s requirements tend to increase over a project lifecycle, e.g. what once started out as a single deliverable becomes five. Or a product that began with three essential features, now must have ten. Or midway through a project, the needs of customers change, prompting a reassessment of the project

		requirements. Scope creep is typically caused by key project stakeholders changing requirements, or sometimes from internal miscommunication and disagreements.
SDLC	Software Development Cycle	Name for process aiming at developing fully functioning software, several models exist across time
VAHTI	The Government Information Security Management Board	Board is responsible for steering and reconciling the development of public sector, and particularly central government, information security in Finland.

Table of Contents

List of appendices

List of figures

List of tables

1 INTRODUCTION

Since the dawn of time, security of the information that we handle in our everyday lives, whether work or personal, has always been extremely important but now it has become more crucial than ever before. As technology continues to evolve with an accelerating pace, matters regarding information security become increasingly complicated. Certain procedures must be put in place to ensure secure handling of information. The possibility of a human error can never be fully removed but with proper training and instructions, its effect will be greatly diminished. Security has become an integral part of modern software development environment.

Once everyone in the organisation is aware of the importance of security, they will achieve a better understanding and handling of information security in software projects. Any reforms done to the instructions of information security must be clear and understandable by everyone in the organisation because not everyone in the organisation is prominent in IT. These people are often the reason for security breaches.

This can happen in many ways such as clicking a link in a received email from someone who claims to be from the IT Department of the organisation, for instance, which then causes malwares, leading to potential data theft and economic loss for the organisation as a result. Often these types of efforts to compromise information security are done in such a way that people do not even regard these attempts as such since they offer look and feel very legitimate.

The client of the thesis is the Legal Register Centre, a bureau under the Ministry of Justice of Finland. The purpose of this thesis is to help the client to improve their information security testing procedures in software development by looking at the current process and developing a testing plan for secure testing.

To learn best how to improve information security, four research questions will be presented and answered in this thesis: What is information security? How can it be

incorporated into testing? How does incorporating testing improve information security? What are the benefits of incorporating information security into testing?

The theoretical framework of this thesis covers research methods, research into information security, processes, and testing with emphasis on security. In practice, an overview of both the current and the improved information security process within the client organisation will be given. In addition, a security testing plan to be used in software projects within the client organisation will be provided to enable more secure software development. Finally, conclusion of the produced results as well as recommendations on how to client can proceed will be presented.

2 METHODS, TOOLS AND LIMITATIONS

A thesis can be viewed as a project. As such, it is appropriate to use relevant project management tools to manage it successfully. The author has chosen to use Trello as his management tool of choice to manage and maintain a clear view of the thesis. In addition, Mendeley will be utilised to support and ensure proper citation and management of references in accordance with the university's chosen style to ease the academic work.

Furthermore, this chapter will lay foundations to the techniques and research methods utilised by the author in their work, as well as the applied tools. In addition, the limitations and scope of the thesis are described in further detail.

2.1 Research Methods

For this thesis, literature, and research papers relevant to the topic will be gathered and analysed in a rigorous manner to ensure validity and reliability of the information using search engines and keywords as well as looking for literature in online libraries. It is worth noting that some papers and literature were not readily available, so they were not therefore utilised in creating the theoretical framework for this thesis. Qualitative analysis will be performed on existing reports and documentation within the organisation to gain a better understanding of the current process.

2.2 Kanban

In order to keep track of everything, it is imperative that a proper view is always maintained. Many methods and tools exist to keep track of a project. In this project, the author has chosen to utilise parts of Kanban.

Kanban is a Lean tool. Lean approach was first developed for car manufacturing industry in Japan during the 1950s. In Japanese it means 'signboard'. The main purpose of Lean is to eliminate all types of waste from the development process. This is achieved via a set of principles, which are: Building quality in, creating knowledge, deferring

commitment, fast delivery, respecting people and optimising the whole. This will lead to the maximal increase of value for the customer. In recent years, Kanban has become increasingly popular in software development. (Ahmad, Markkula, & Ovio, 2013, pp. 1–2.)

Kanban is a mechanism for Just-In-Time production, practiced by Toyota for optimising processes and improving efficiency, originating in 2004. The principles of Kanban include visualising the workflow, limiting the work in progress, constant measure and management of the flow, collaborative improvement and making process policies explicit. (Ahmad et al., 2013, p. 2.)

In this project, the author uses the Kanban board for visualisation and keeping track of the work in progress since the board clearly shows the assigned tasks and helps to prioritise. However, it should be noted that there are no set rules concerning the implementation of the Kanban board. (Ikonen, Pirinen, Fagerholm, Kettunen, & Abrahamsson, 2011, p. 2)

The author has chosen to use the online tool Trello to facilitate and enable the use of the Kanban board. Trello is a cloud-based solution that uses Kanban as the project management method. All tasks and activities within the project are shown in a single view that are easily visible. Trello enables visualisation of tasks into boards, dividing projects into groups and further sub-divide those groups into singular tasks. The interface of Trello is very user-friendly, making it ideal for managing a project of this size.

With Trello, the user can create an unlimited number of boards and designate them according to their wishes. Users can assign lists to each board and create cards to each of those lists, acting ask tasks. The user can also add individual to-do lists to each card, allowing division of any given task into several phases and dividing them among team members. The lists created are easily reordered. The tasks can be given color-coded labels that are fully customisable with words to designate statuses such as “doing” or “done”. Finally, to make sure that the project stays on schedule, all tasks can be given a due date. (Johnson, 2010, pp. 1–2.)

In this approach, the Kanban board consists of three columns. First column details the tasks that are to be done, the second one shows tasks in progress, and the third one presents tasks that have been completed.

2.3 **Mendeley**

Mendeley is a free reference management software that allows easy citation for academic papers. It is used by over 6 million researchers worldwide. (“Mendeley - Reference Management Software & Researcher Network,” n.d.)

With Mendeley, the user can add various articles or even webpages to a library, created within the software, with little effort via a browser plugin or manually. Mendeley automatically identifies all necessary information for referencing such as the authors, the year of publication and so forth. The user can easily annotate and highlight within the sources, search the necessary articles from the library while typing and most importantly, easily insert citations using the selected citation style and generate the bibliography with a single click. (Tugizimana, 2015)

2.4 **Limitations and Scope**

Part of the information processed in this thesis and possibly the parts of the results as well are classified. To conform with law, any such parts will not be included in the publication of the thesis, but generalised documents will be produced instead. In addition, the author will focus on information security testing and producing documentation regarding testing and the processes of the bureau and providing theory to support all of the above. Furthermore, as a list of predefined search terms was used, some relevant literature and useful research likely remained unspotted. Conducting risk analysis, writing test cases, describing testing tools, testing the model with a pilot system, inquest into possible automation, discussing the human and technological factors of information security, information security governance and the respective frameworks are considered out of scope and therefore are not included in the thesis.

3 INFORMATION SECURITY

Over 2,500 years ago Chinese general Sun Tzu Wu made remarks that are relevant even today: “If you know the enemy and know yourself, you need not fear the result of a hundred battles. If you know yourself but not the enemy, for every victory gained you will also suffer a defeat. If you know neither the enemy nor yourself, you will succumb in every battle”. (Tzu, n.d., p. 11)

Information security is not so different from war. Like war, it is a constant battle, between the technicians and managers, who defend information assets using multiple layers of security from threats and attacks from all sides, and hackers, people who would seek to use information for their own ends. These attacks are relentless and never-ending. To stay victorious, the defenders must know themselves and the enemy they face. People are already taking advantage of information security. Using a password or an anti-virus program is already practically applying information security, albeit on a superficial level.

This chapter will illustrate the basics of information security starting with the history and going forward with some concepts of the field that are required for understanding the context. The author will briefly mention some of the known cases of security breaches to help the reader understand why information security is critically important.

3.1 History

Information security has existed for a long time in one form or another. Some people liken information security to only computers, but it has been in existence for some time before that. This chapter will cover what are thought to be the most crucial time periods and events regarding information security.

3.1.1 1970–1990

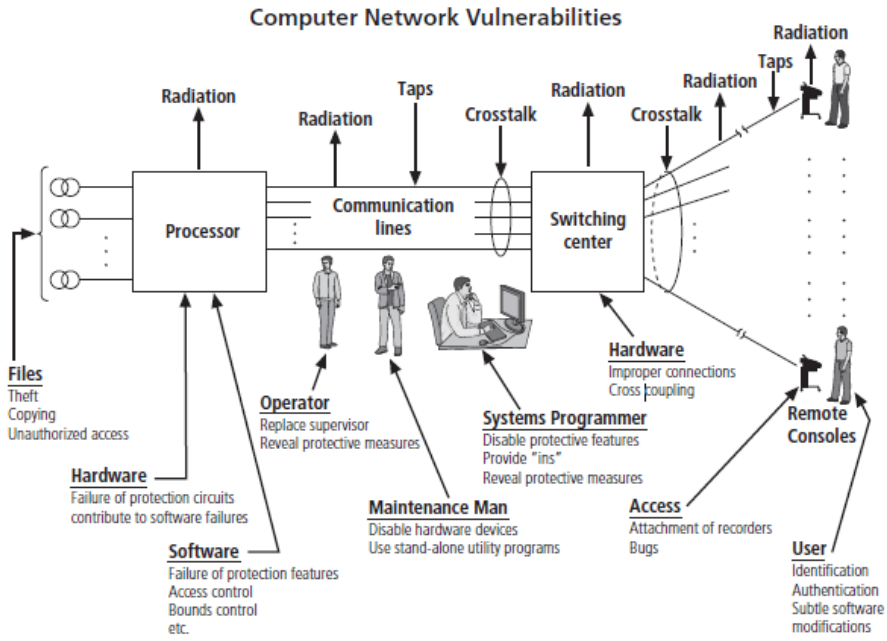
As Whitman & Mattord (2012, pp. 4–5) state, technology kept evolving and tasks required became more complex and sophisticated. During the Cold War, significant

advancements were made, resulting increase in the number of computers. To support this process, United States Department of Defence's Advanced Research Projects Agency, hereinafter referred to as DARPA, started researching the possibility of a redundant network communications system for military purposes. This project, known as ARPANET, later became what today is known as the Internet.

Later, in the 1970s, ARPANET became more widely used, therefore increasing the potential for its misuse. The lack of security within the system was brought to the attention of the researchers in 1967, after which securing the systems became one of the major concerns during development for military contractors. In 1973, Robert M. Metcalfe, pioneering engineer of the Internet and creator of Ethernet, identified problems with the security of ARPANET. It became clear that individual sites did not have sufficient safety controls to protect data from unauthorized access, passwords were vulnerable and user authentication did not exist. (Whitman & Mattord, 2012, pp. 5–6)

Afterwards, DARPA created a list of recommendations, known as Rand Report R-609, declassified and released to the public in 1979, seen in **Error! Reference source not found.**. It noted that security risks within military information systems could no longer be mitigated through the normal procedures. This report greatly expanded the definition of information security from physical safety to include securement of the data, limiting unauthorised access to said data and involving personnel on every level of the organisation information security. (Whitman & Mattord, 2012, pp. 6–7)

Figure 1 Vulnerabilities realised in the Rand Report (Whitman & Mattord, 2012, p. 8)



It was around this time that another type of security threat emerged; malware. According to Brown (2014), the first computer worm was created in 1979 at Palo Alto Research Center owned by Xerox. Initially, the program was meant to make computers more efficient but would eventually end up being warped later by hackers into a virus with a sole purpose of destruction or alteration of data.

ARPANET was first compromised in 1988, once a worm was uploaded there, resulting in debilitation of approximately 6,000 computers. The virus in question replicated itself, filling the memory banks of the computers. It was created by Robert Morris who was then given three years of probation and a penalty fine of 10,000 dollars. This was one of first instances of what could be described as “cyber-crime” even though the perpetrator was not looking for financial gain but instead created and unleashed the virus out of simple boredom. (Brown, 2014)

The first system to integrate security to its core functions was MULTICS. It was a time-sharing operating system. In the late 1970s, personal computers were created when microprocessor decentralised computing, moving it out of data centres, increasing networking dramatically. (Whitman & Mattord, 2012, pp. 7–8)

3.1.2 1990–present

As computers became more common towards the end of the 20th century and Internet gained ground, this also gave rise to many security issues. Despite the wide and ever-growing use of computers, security remained a low priority issue, which resulted in most of the problems faced. It was assumed that most users were trustworthy, so encryptions and authentications were deemed unnecessary. Early approaches to security had focused on physical security but since everything was networked, causing information to be more exposed security threats, other types of security had to be considered. (Whitman & Mattord, 2012, p. 9)

Today, the Internet connects a myriad of unsecured computer networks so that they engage in constant communication with one another. Security of each computer depends on the security of every other computer to which it is connected. Information security needs continuous improvement, especially in the light of constant threat of cyberattacks and information warfare between nations. (Whitman & Mattord, 2012, pp. 9–10)

Cyber-crime has become an increasingly troubling issue ever since the end of 20th century. First, the issue was not taken seriously, as can be deduced from an example case, where two teenage hackers took control of over 500 government, military, and private computer systems in the United States. This event is known as the “Solar Sunrise”. (Brown, 2014)

Since the turn of the century, the amount and variety of malware have increased exponentially with the growth of computing. This has caused some serious financial losses. For instance, the computer worm Code Red caused two billion dollars in damage by infecting Microsoft Windows NT and Windows 2000 servers, attempting to attack the White House website. (Brown, 2014.) In 2011, Norton estimated that in the previous year alone, there was over 32 billion dollars’ worth of criminal gain, the financial loss being inflicted to 74 million people solely in the United States. (Saini, Rao, & Panda, 2012)

By comparison, cyber incidents such as data breaches, security incidents, privacy violations and phishing can cause substantial losses for businesses. For instance, incidents mentioned above cost U.S companies approximately 8,5 billion U.S dollars a year over a period of ten years from 2004 to 2014, totalling 85 billion. The average cost of a data breach in the UK is approximately 1.4 million pounds and recovery back to normal conditions takes 9.3 months (Soomro, Shah, & Ahmed, 2016, p. 217). When the loss of information such as crucial company investment strategies or policies or personal data, potential loss of customers, decreased overall profits, possible downsizing and so forth is considered, it can potentially have a much larger impact than what the above numbers represent. It is good to keep in mind that the accuracy of these numbers is debatable since intellectual property often cannot be accurately measured in numbers. (Romanosky, 2016, p. 131)

In today's world, new methods of waging war exist predominantly online, to be used in several ways from hacking enterprise systems for financial gain to subverting democracy through influence on social media. This is a different type of war of the new era, cyber warfare. The first instance of organised cyber warfare was discovered in 2010, when a virus named Stuxnet, created for attacking the economic and industrial sectors, was discovered. Its sole purpose was to target systems that run nuclear power. This virus was used against Iran to halt its nuclear program. It has been debated that due to its complexity and sheer size, it was developed by either the United States or Israeli government for more than ten years. (Brown, 2014). Due to limitations of this thesis, the author will not go further into detail about cyber-crime, cyber warfare or threats to information, ICT, or cybersecurity.

To summarize, information security has existed long before modern computers and has become increasingly crucial as technology has developed further. Starting with military applications, this advancement led to more complex systems requiring more sophisticated solutions to secure them and especially the data they hold. Today data is everything because most systems themselves are now redundant. During the first few decades, security was not considered important, but when computers were

commercialised and became available to everyone, and with the increase of diverse types of malware and emergence of cyber warfare, the need for improving information security has ever since grown dramatically. Despite this, it is imperative to understand that information security is much more than protection against malware and viruses. It is a cohesive entirety, acting as a crucial component of our society.

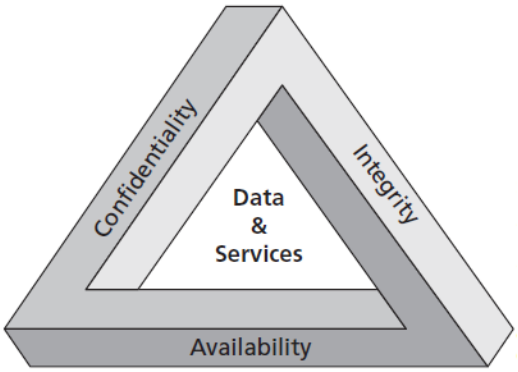
3.2 **Definitions**

This sub-chapter introduces some of the most important concepts of information security. It is also important to distinguish information, ICT, and cybersecurity. The three will be explained in detail with definitions by various authors of the field. At the end, comparison between them is presented to clarify the position of each, what they have in common and how are they different from one another as well as the conclusions of the author of the thesis when it comes to information security as a single concept.

3.2.1 Information security

Von Solms and Van Niekerk (2013, p. 98) define information security as "the preservation of the confidentiality, integrity and availability of information". The preserved information can be on many different forms, including print or handwritten, stored electronically, conveyed during conversation and so forth.

Figure 2 The C.I.A triad (Whitman & Mattord, 2012, p. 11)



Whitman and Mattord (2012, p. 10) define information security as "protection of information and its critical elements, including the systems and hardware that use, store and transmit the information". They argue that while ensuring the three principles within the C.I.A triad as depicted in **Error! Reference source not found.** has been the industry standard for a long time, it no longer addresses the constantly changing environment of the industry in a proper manner. Therefore, they have defined several additional critical characteristics of information, expanding the model seen in **Error! Reference source not found.**. These expanded characteristics include accuracy, authenticity, utility, and possession, bringing total number of these characteristics to seven. Von Solms & Van Niekerk extend this definition, bringing the number of this properties to a total of eight. They are further explained below in Table 1 along with the original elements from the C.I.A model. In addition, non-repudiation is included since it is considered by many authors to be an essential part of information security.

Table 1: The eight properties of information security (Whitman & Mattord, 2012, pp. 14-17, Von Solms & Van Niekerk, p.99)

Element	Description
Accuracy	When information has the required value that the user is expecting and is also free from any mistakes or errors, it can be considered accurate. This can be compromised if the information in question has been modified, whether intentionally or otherwise.
Authenticity	Information can be considered authentic as long as it is in the same state in which it was created, stored, transferred or placed rather than being a reproduction or fabrication.
Availability	Availability is essentially enabling authorisation of accessing information to people or computer systems with the required credentials or authorisation without delay, obstruction, or

	interference and to access that information in the required format.
Confidentiality	Information has confidentiality when it is protected from any exposure or being disclosed to unauthorised people or systems. This ensures that only those users with the proper privileges can access the information. There are several measurements to protect confidentiality, including classification of information, storing documents securely, applying security policies and educating the end-users.
Integrity	Information can be considered to have its integrity intact when it is complete, whole, and uncorrupted. This characteristic becomes under threat if the information is exposed to any form of disruption such as corruption, damage, or destruction, threatening its original state.
Non-repudiation	Information is non-repudiable when a proof of its integrity and origin of data is provided and can be verified by any third party at any time.
Possession	A possession of information is a state of control or ownership. Obtaining information means possessing it, even if it is not in a proper format or have other characteristics described here. Some characteristics above are intertwined but this is not always the case. For instance, a breach of possession does not necessarily lead to a breach of confidentiality but vice versa it is the exact opposite since a breach of

	confidentiality always leads to a breach of possession.
Utility	Information has utility when it has value, in other words when it can serve a purpose. If information is available but carries no meaning for the end-user, in that instance it is essentially worthless. (Whitman & Mattord, 2012, p. 17)

In summary, Von Solms (2013, p. 98) posits that information security used to be only a technical issue. As computers and networks evolved, this process had to be extended beyond the technical boundaries. Therefore, information security should be considered a process rather than a product or a technology. It is a vast entity consisting not only of technology and people, but also of legal, ethical, and professional aspects.

Some people think that information security equals cryptography and firewalls, which could not be further from the truth. As has been established, information security encompasses all aspects of protection and the applied standards are supposed to specify the countermeasures to protect a system from potential forms of exploitation. For this to be possible, vulnerabilities must be found and assessed. Simply having a firewall is not enough to prevent or detect attacks.

However, technical controls are still essential for maintaining secure environment for the information that requires protection. This can be achieved via many different methods such as access control, where systems specify who can use a specific resource and how they can use it or authentication whereby the user is required to confirm their identity to gain access. This can be done with ID cards, passwords, multi-factor authentication, retinal scans and so forth. System logs can be used to monitor activity of any user, so they can be held accountable. (Whitman & Mattord, p.298-306)

It is also worth noting that there is a notable difference between information security and information and communication technology security, as well as cybersecurity, which will be further described in detail below.

3.2.2 ICT security

Von Solms & Van Niekerk (2013, pp. 98–99) posit that ICT Security should be defined as "all aspects relating to defining, achieving and maintaining the confidentiality, integrity, availability, non-repudiation, accountability, authenticity and reliability of information resources. They also argue that ICT security can be considered a sub-category of information security due to information security including protection of underlying information resources.

Von Solms & Van Niekerk (2013, p. 98) assert that while ICT security is therefore very close to information security, there are still some differences that ought to be considered such as additional characteristics, namely non-repudiation, accountability and reliability.

Securing an ICT resource and an information resource are quite different. While an information resource can be any entity from where information or data is sent to or received from, ICT resource is an information resource that exists within an information technology (IT) system. The author makes another valid point via reminding that information can be considered secure only when everything else dealing with that information such as resources or processes are equally secure. (Von Solms & Van Niekerk, 2013, p. 99)

To fully understand information and ICT security, one must possess a complete view of the characteristics of security explained in section 3.2.1 since all of them have an essential role in information security and should be considered equally important. (Von Solms & Van Niekerk, 2013, p. 98)

In the case of ICT Security, it is the technological infrastructure that requires protection while in the case of information security ICT is the infrastructure keeping the information intact but the information itself is the asset that needs to be protected.

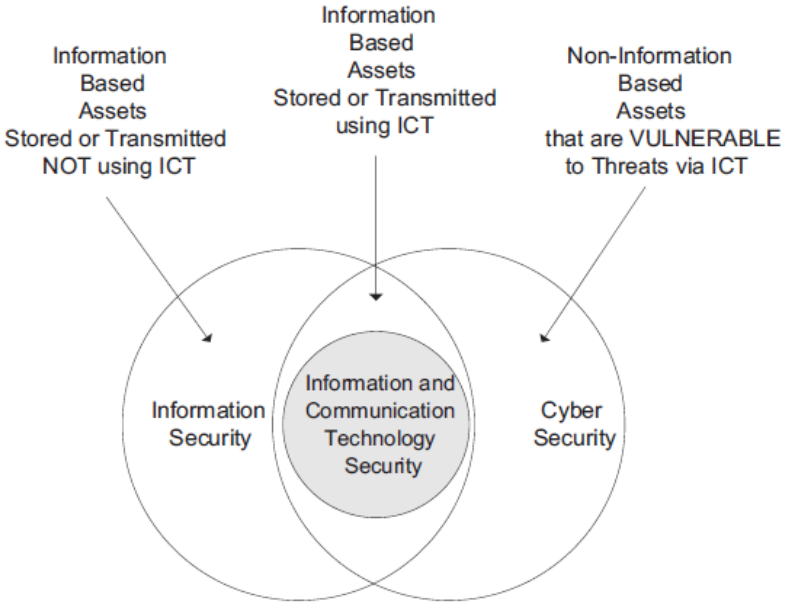
3.2.3 Cybersecurity

Cybersecurity is often mixed with information security, and even though they are not completely synonymous, they bear similarities. Von Solms and Van Niekerk (2013, p. 99) assert that while most cybersecurity incidents lead to a breach of one or more of the characteristics of information security established in chapter 3.2.1, it is not part of the scope of information security.

Within cybersecurity the assets requiring protection range from the person to household appliances like smart technology, all the way up to critical national infrastructure. In practice, everything that can be reached via cyberspace is vulnerable. In this instance, weak points in ICT and information are the underlying cause of the vulnerabilities that enable cyber-attacks to take place. (Von Solms & Van Niekerk, 2013, p. 100)

Unlike ICT or information security, cybersecurity has intangible dimensions requiring careful consideration and protection such as ethics, when it comes to cyber-bullying for instance. This goes beyond the boundaries of what is considered information. It could be therefore said that cybersecurity is an expansion of information security. It is more about protecting the people and other assets. **Error! Reference source not found.** helps to understand the relationship between the three layers of security.

Figure 3 The relationship between ICT, information and cybersecurity (Von Solms & Van Niekerk, 2013, p. 101)



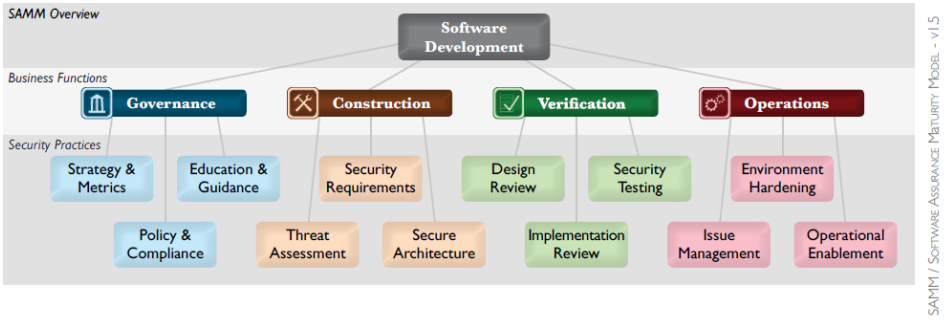
In summary, with ICT, the assets that require protection are parts of technology infrastructure. Information security extends this by including all aspects of information, going beyond the technology by taking into consideration the information that is not stored or communicated directly via ICT. Cybersecurity serves as an extension of information security and includes both tangible and intangible assets. In general, one can surmise that cybersecurity is about protecting everything that can be accessed via cyberspace, not just technology or information but people and other assets as well.

3.3 **OWASP**

The Open Web Application Project, also known as OWASP, is a non-profit organisation that focuses on improving software security, enabling the development, preparation, and support of trustworthy and reliable software, and bringing information security more accessible to general population. OWASP is not connected with any brand or technology but instead supports the use of commercial tools. OWASP aspires to produce impartial, practical, and cost-effective information about software security through transparency and with the focus on cooperative community. OWASP is a world-wide organisation, with most of its staff working as volunteers. (OWASP, Saarinen 2014, p.18)

3.3.1 SAMM v1.5

Figure 4 Overview of version 1.5 of SAMM (SAMM v1.5)



The Software Assurance Maturity Model, also known as SAMM, is a freely available for all to use and vendor neutral open framework designed to aid organisations in creating and implementing software security strategy that is tailored to the risks that the organisation faces. It is flexible and as such can be used by small, medium, and large organisations independent of the style of development. In addition, SAMM can be applied organisation-wide, single line of business or even an individual project. **Error! Reference source not found.** demonstrates how security practices are tied to each part of software development. (SAMM v1.5)

SAMM defines four business functions, each function consisting of category of activities related to software development. For each function, three security practices are defined, consisting of security-related activities that create assurance for the function. For each practice, SAMM defines three maturity levels, characterised by sophisticated objectives that are increasingly demanding than the previous level. (SAMM v1.5)

3.3.2 OWASP Top 10

Figure 5 OWASP Top 10 list, 2013-2017

OWASP Top 10 is a document that represents a broad consensus about the most critical security risks to web applications. It is an awareness document, designed to make organisations aware of the risks and help them to ensure that their web applications minimize these risks. **Error! Reference source not found.** lists the Top 10 from 2017 with comparison the previous edition that was released in 2013. (A1QA, 2017)

4 INFORMATION SECURITY IN THE GOVERNMENT

An abundance of jurisdiction exists regarding information security that must be followed when making decisions concerning acquirement of new devices or technologies for the Finnish Government or changes to existing, for instance, IT infrastructure. All actions of the Government and public authority are based on the law. In addition, they apply specific framework standards in their work. This chapter will compile the legislation, standards, entities, and classification of material within the Government regarding information security.

4.1 Legislation

The Government has issued several decrees and initiatives to improve information security. These announcements, like all the actions by public authority, are always based on existing legislation. This chapter summarises the laws and the initiatives that regulate information security or some parts of it within the Government.

4.1.1 European Union

After joining the European Union, Finland has been affected by directives, acts and statues regarding information security such as European Union Council Decision of 23rd September 2013 on the Security Rules for Protecting EU Classified Information (2013/488/EU). Most of these have been implemented in national law as is required from all members, the most recent being General Data Protection Regulation, also known as GDPR.

The latest development includes the EU Cybersecurity Act, agreed upon by member states on 11th of December 2018. This Act reinforces the already-existing mandate of ENISA (European Union Agency for Network and Information and Security). Its purpose is aiding nations to deal with cybersecurity threats. The Act also established a union-wide cybersecurity framework for certification and boosting the overall cybersecurity. Furthermore, ENISA will enable preparedness, promote the awareness, and assist the

member states in developing and implementing relevant policies in many different areas. (Commission, 2018)

4.1.2 National

The premise for information security in Finland starts from the Constitution. In Section 10, the following is stated:

"Everyone's private life, honour and the sanctity of home are guaranteed. More detailed provisions on the protection of personal data are laid down by an Act. The secrecy of correspondence, telephony and other confidential communications is inviolable. "(Constitution of Finland 731/1999 §10)

Other applied legislation include Publicity of Actions of Public Authority Act (621/1999), Data Protection Act (1050/2018), International Information Security Requirements Act (588/2004), Communications Market Act (393/2003), Protection of Privacy in Professional Life Act (759/2004), Data Protection Act of Electrical Communications (516/2004) and Electrical Communications Services Act (917/2014).

Furthermore, new legislation has been enacted and it will take effect on 1st of January 2020 (HE 284/2018). This new piece of legislation will unify some of the existing laws into a single Act, making information security legislation more comprehensive and clarifying how the public authority must handle information. It is meant to improve control of information within public authority. (HE 284/2018)

4.2 **VAHTI**

The Government Information Security Management Board (VAHTI) is an authority that was established by the Ministry of Finance for steering and development of public administration and central government information security. VAHTI is responsible for handling IS policies and matters regarding IS guidance. They promote implementation of Government Programmes and Government policy outlines. It acts in an advisory

capacity in decision-making, aiding the Government and the Ministry of Finance in preparation of decision relating to IS within central government.

VAHTI's main objective is developing information security. In practice, the areas that VAHTI focuses on are improvement of reliability, continuity, quality, risk management and contingency planning of the functions of the Government. This is achieved through its provision of general information security instructions for all public authorities. Their work has significantly improved information security within central government. The Government Resolution on Enhancing Information Security in Central Government, on November 26, 2009 established VAHTI's position as a key entity of information security within central government. (Ministry of Finance, 2010, p. 9)

4.3 **KATAKRI**

National Security Audit Criteria (KATAKRI) is an auditing tool for information security of the public authority established by the Ministry of Defence and controlled and maintained by National Security Authority (NSA). Its purpose is to evaluate and audit organisation's ability to protect classified information that is in the possession of public authority. Essentially it is a compilation of existing minimum requirements of information security, based on national legislation and Finland's international obligations.

It has three distinctive sections; security management, with the intent of making certain that the organisation has the required security management abilities, physical security, the purpose of which is describing the security requirements for the physical environment of classified information and information assurance, where the security requirements for the IT environment are given. (Ministry of Defense, 2015)

4.4 **JUDO**

Action Plan for Digital Security in Public Administration (JUDO) is an initiative by the Ministry of Finance, under the supervision of Population Register Centre. It is a program slated to cover the years from 2019 to-2022 with the purpose of developing digital

security within public authority. Within this initiative, digital security compasses of five parts: risk management, continuity of operations, reservation, informational security, cybersecurity and data protection. (Väestörekisterikeskus, 2017)

4.5 **Classification of information material**

Within the central government, four different security levels exist. These are collectively known as protection levels. Their main purpose is to ensure safety of all parties involved and to control the processing of classified documents by public authority. All of them will be described further within this sub-chapter. All internal documents requiring protection level within the Ministry of Justice have universally either Protection Level III or IV, therefore making those levels most relevant. However, in the interest of consistency, all four are represented here.

4.5.1 Protection level I

Protection level I, also known as TOP SECRET, is the highest available level. Documents can be given this level of protection if following condition holds true: “Unauthorised disclosure of secret information could cause particularly grave prejudice to a public interest referred to in a secrecy provision or to international relations, State security or national defence.” (Ministry of Finance, 2010, p. 63)

4.5.2 Protection level II

Protection level II, also known as SECRET, is the second highest available level. Documents can be given this level of protection if the following condition is true: “Unauthorised disclosure of secret information could cause significant prejudice to a public interest referred to in a secrecy provision or to international relations, State security or national defence.” (Ministry of Finance, 2010, p. 63)

4.5.3 Protection level III

Protection level III, also known as CONFIDENTIAL is the second lowest available level. Documents are given this level of protection if the following condition is true:

“Unauthorised disclose of secret information could cause prejudice to a public or private interest or right referred to in a secrecy provision or to international relations, State security or national defence”. (Ministry of Finance, 2010, p. 63)

4.5.4 Protection level IV

Protection level IV, also known as RESTRICTED, is the lowest available level. Documents are given this level of protection if the following condition is true: “unauthorised disclosure of secret information could be disadvantageous to a public or private interest referred to in a secrecy provision or, in the event of documents referred to in section 9(2) of the Decree on Information Security being involved, if unauthorised disclosure of information could be disadvantageous to a public or private interest or adversely affect the ability of a public authority to perform its functions.” (Ministry of Finance, 2010, p. 63)

For the processing rights, distribution, traceability, and IT processing of documents on each protection level, see Appendix A.

To summarize, the Government has significantly advanced their information security agenda, policies, and implementation in the recent years with help from VAHTI. Information security within the Government is regulated by law, both international and domestic and continues to do so through new legislation and initiatives like JUDO.

5 PROCESSES

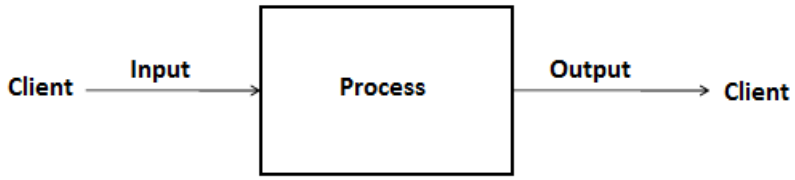
To create information security testing model, a process needs to be established. This chapter discusses processes in general, mainly how to define and analyse them through extant literature. Some analytical methods are introduced and described further in detail as well as how to model and develop a process. Finally, to be able to describe a process accurately, a Business Process Model Notation (BPMN) has been developed which will be further described at the end.

5.1 Definition

Process can mean many things. It is not straightforward concept and there is no ultimate right answer. Based on extant literature, a couple of illustrations will be given below to assist in establishing a frame for what constitutes a "process".

Martinsuo & Blomqvist, 2010 define processes as "event chains that create increased value for the customer for which the organisation uses its resources". **Error! Reference source not found.** shows one way of illustrating a process.

Figure 6 Simplified illustration of a process (Modified from Martinsuo & Blomqvist, 2010, p. 4)



Sharp & McDermott define processes as "a collection of interrelated activities, initiated in response to a triggering event, which achieves a specific, discrete result for the customer and other stakeholders of the process". (Sharp and McDermott, p.56)

(Luukkonen, Savolainen, & Tamminen, n.d., p. 8) assert that process is different depending on context. A process can be "a chain of events, a series of improvements or

series of handling a matter or chain of matters." It can be "a series of events, having or is depicted as having a certain direction, meaning, effect or result." A process can be "a series of interconnected actions and to implement them, resources are needed for transforming inputs into products."

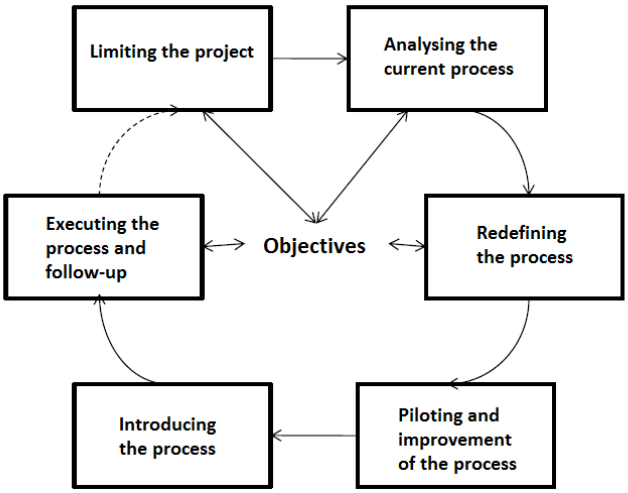
From these definitions the following conclusion can be derived; A process is a chain of interrelated activities that requires resources from the entity that implements it, whether it is an organisation, or an individual. In addition, process produces a result, such as for instance, a product or part of a product, perhaps serving as the required catalyst for the next step in a larger process for the customer or other stakeholders of said process. In the interest of consistency, hereafter the term "activity" is used to describe sub-processes, also known as tasks or events.

5.2 **Process development and modelling**

In previous sections, a process was defined and some of the methods for analysing it were also covered. This chapter will discuss how to model a process. The Business Process Modelling Notation 2.0 (BPMN 2.0) will be further introduced at the end of the chapter.

Modelling a process is an excellent way to demonstrate the current process or bring its potential flaws and development needs into light. Processes exist in every part of an organisation and its business. Within this chapter, distinctions as to the type and contents of a process are not made but the focus is in modelling a general process.

Figure 7 General stages of defining a process (Modified from Martinsuo & Blomqvist, 2010)



Error! Reference source not found. shows the steps that aid in developing a process. When beginning the effort to develop a process, it is imperative that the project is properly limited to prevent scope creep. Sometimes there is already knowledge in place about the existing processes, making the drawing the line easier. The business model of the organisation is in a central part of this.

Once the limitation of target development has been finalised, as much information as possible about the current process is required. If the process is yet to be created and is completely new, information about how its functions have been performed before is essential for mapping the process.

Information about the process can be collected in numerous ways; In addition to the methods mentioned in the previous chapter, interviews, group work, database analysis for existing performance and simulations are also some of the available options. They can be used to supplement any of the previously mentioned methods.

To properly visualise what is needed, the current process should always be compared to the objectives; how well does it answer to the current needs and what kind of flaws can be observed?

After the initial analysis has been performed, the areas of the process requiring refining must be identified. It is possible that the entire process needs to be redefined, depending on the needs of the customer like previously stated.

Whenever possible, the process should be piloted in a testing environment or under true conditions before extensive use. This enables observation of the process, allowing for alterations to the model if needed and noticing the potential errors, thus preventing problems later. In addition, this phase will give some insight into whether the renewed process is useful, and whether it solves the problems of the old one (Martinsuo & Blomqvist, 2010, pp. 9–10)

If the process works under testing conditions, the organisation can then continue with introducing and eventually implementing it. To make sure the process works as intended, a feedback loop should be created so the process can be developed further if necessary, establishing a policy of continuous improvement.

5.3 **BPMN**

Knowing how the business operates is the first and the most critical step of business process improvement. Business Process Model and Notation (BPMN), provides a graphical representation of business workflows that anyone, from business analyst to stakeholder, can easily understand; aiding in business process analysis and business process improvements. As stated by (Chinosi & Trombetta, 2012), BPMN is state of the art-model for providing clear picture of the existing process.

For the purposes of this thesis, client provided an existing BPMN model which can be seen at **Error! Reference source not found.** to help understand the existing Business Process that is related for information security during projects.

6 SOFTWARE TESTING

We use several types of software every day with an increasing pace. To avoid major disruptions, it is imperative that the said software works without errors. For this to be possible, extensive testing is required during the development. Often this is not the case and the resulting failures have led to major incidents. Testing is often one of the most undervalued and underfunded parts of the development cycle. This chapter will illustrate software testing and highlight its importance. Furthermore, its ties to information security will be shown. In addition, types of testing will be discussed more in detail.

6.1 Context

The importance of testing cannot be overstated and perhaps the best way to explain how ignoring thorough testing of the software can have disastrous consequences. This chapter will look at some infamous cases where testing was not done appropriately and what resulted from it.

Software issues can be something relatively mild such as the inability to use a credit card, leading to inconvenience and irritation to people, as happened in Germany on 1st of January 2010. People could not use their cards for a week. Some were affected more than others. The glitch was a result from wrong configuration that resulted in microchips of the cards not being able to process the change of year from 2009 to 2010. Fixing this error would end up costing an estimated 300 million euros. (Connolly, 2010)

One of the more major incidents included the flight of Ariane 5 on June 4th, 1996. A component of Ariane 4 was transferred to Ariane 5 without proper testing, resulting in the loss of the launcher and the satellites it carried, concurrently causing a loss of between a hundred million and half a billion U.S dollars and ten years of work (Dalal & Singh Chhillar, 2012; Jazequel & Meyer, 1997).The explosion was a result of a software error involving a failed conversion of an integer.

Software errors have also caused human fatalities in addition to economic loss. Between 1985 and 1987 a computer-controlled radiotherapy machine known as Therac-25 gave patients massive overdoses of radiation, resulting in six deaths because of programming errors. (Leveson & Turner, 1993)

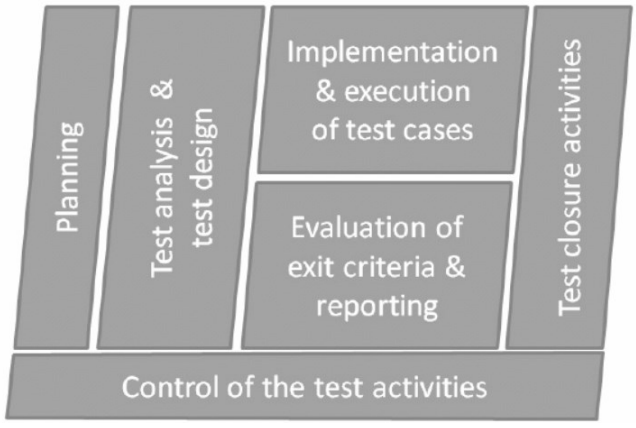
6.2 **Definitions**

This sub-chapter introduces some important concepts of software testing. Testing can be further divided into several sub-categories which will be covered in this chapter such as dynamic and static testing.

6.2.1 Testing

Testing is the attempted identification of defects and failures during software development so that they can be fixed, and quality of the product can thus be improved (Homès, 2012). It is important to emphasize that testing should take place from the beginning as soon as the team has something to test for and this should continue throughout the entire development cycle. This is because the later in the development process the defects are found, the more it will cost. **Error! Reference source not found.** shows fundamental testing processes.

Figure 8 Crucial processes of testing (Homès, 2012)



(Hass, 2008) defines testing through standards. IEEE 610 defines testing as “the process of operating a system or component under specified conditions, observing or recording the results and making an evaluation of some aspect of the system of component.” IEEE 829 states that testing is “The process of analysing a software item to detect the difference between existing and required conditions and to evaluate the features of the software items.” And finally, the ISTQB characterises testing as “The process consisting of all life cycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirement, to demonstrate that they are fit for purpose and to defect defects.”

Each of these three definitions have one thing in common; testing is considered a process. Processes were discussed before in Chapter 5. ISTQB mentions “static and dynamic” testing, which will be defined later more in detail. Testing is done on “system” or “component” or a “software item”. By looking at these statements, it is possible to determine that testing can be done to any product or part of a product. (Hass, 2008)

6.2.2 Static testing

According to ISTQB, static testing is “testing of a component or system at specification nor implementation level without execution of that software”. The purpose of static testing is to find defects in the written products like the code and the documents, results of human error. Thus, it can be considered a part of quality assurance. However, it is not synonymous to review. There are many types of static testing, such as audits, inspections and both technical and management reviews. Static testing can be used against plans, requirement specifications, strategies and so forth. These types are further illuminated in Appendix C. (Hass, 2008, pp. 276–278)

Benefits of static testing include earlier insights into product by management, higher efficiency, and productivity due to discovery of defects early. In addition, because fewer defects are presented, dynamic testing time is decreased. Doing static testing results in less defects sent to the customer, enabling higher level of trust between the two parties.

6.2.3 Dynamic testing

Dynamic testing is testing of the software itself, where the written code is being executed on a computer, thus requiring something executable and some form of a test environment. Dynamic testing is used to find the situations where the test object does not behave as it is supposed to. (Hass, 2008, p. 276)

6.2.4 Functional and non-functional testing

Functional testing is done to see what the software does in practice. This focuses on the output, checking whether it is as stated in the requirements or not. Non-functional testing measures how the software performs. This can refer to for instance performance, stress, and compatibility testing.

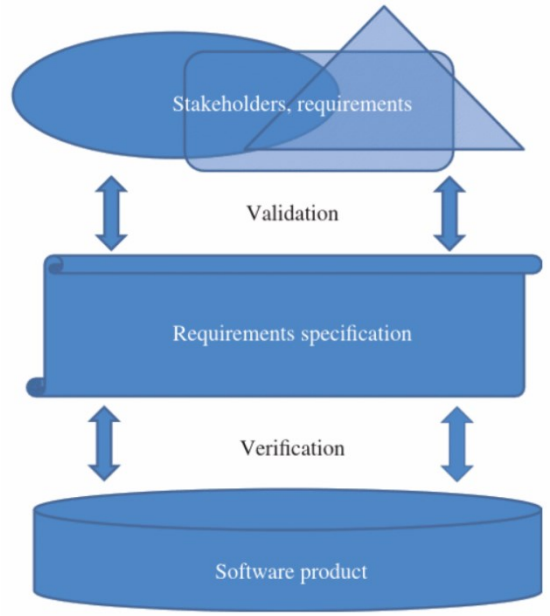
6.2.5 Verification and validation

Verification and validation are essential parts of the testing process, concluding each part of testing lifecycle like is shown in **Error! Reference source not found.**. In the ISO 9000 standard, verification is defined as “confirmation, through the provision of objective evidence, that specified requirements have been fulfilled.”

Verification asserts that the requirements of the product have been fulfilled and answers the questions; did the team produce the specified and is it being built correctly? Verification checks the relationship between documented specifications and the program itself. Due to this, it is repeatable and potentially automatable operation. Verification can be implemented with several techniques, the choice of which depends on the test object. Test object can be a document like plan, design or specifications or the code itself. Techniques include inspection, review and walk-through. After the code has been written, it can be analysed statically, to verify that it is written according to the standards and has no obvious faults. Usually this is followed by dynamic testing, executing the code.

Validation is defined as “confirmation, through the provision of objective evidence that the requirements for a specific intended use or application have been fulfilled.” Validation focuses on the use of validated component, whether the requirements for that specific use have been fulfilled, in relation to the needs of the users. Therefore, validation answers the question: has the correct product been built or is it being built? Validation asserts that specifications are valid and involves interaction with stakeholders and thus cannot be automated. **Error! Reference source not found.** illustrates this. (Hass, 2008, pp. 17–19; Homès, 2012, p. 7; Tchier & Mili, 2015, p. 25)

Figure 9 Relationship between validation and verification (Tchier & Mili, 2015, p. 27)



6.2.6 Costs, errors, impacts and principles

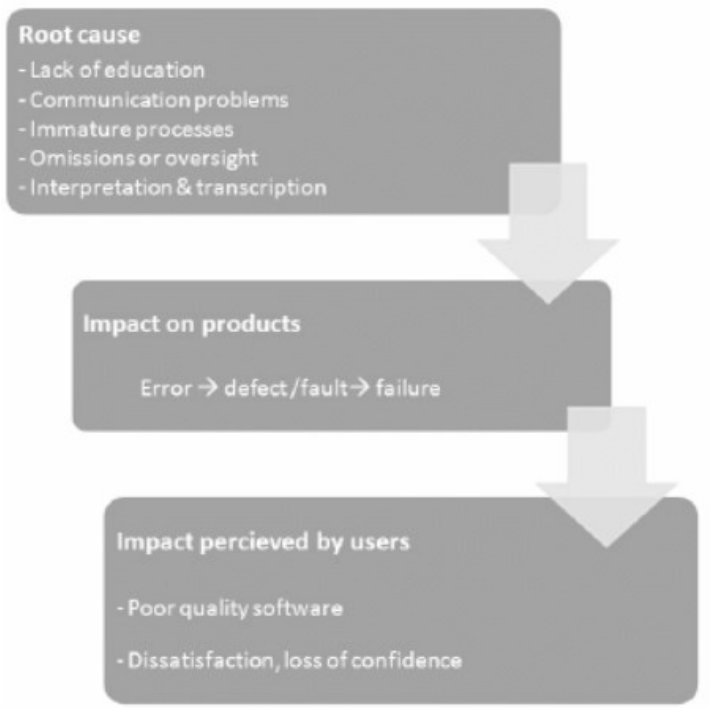
Some think testing stops at the release but in order to keep the software up-to-date, testing must, in fact, take place during the entire lifecycle of the software, up until it gets fully decommissioned, not only during the development. Even then, testing must be done to ensure smooth transition to the new software. Despite this, exhaustive testing is not a feasible approach. In other words, everything cannot be tested. Testing for every possible scenario can be considered impossible. Doing this would delay software releases indefinitely, costing insurmountable amounts of time, money, and resources.

According to (Bertolino, 2007, p. 1) it has been estimated that testing can consume fifty percent, or even more, of the whole software development budget so right allocation of resources is fundamental to keep the project from going overbudget. Flyvbjerg & Budzier (2011) state that projects have an average cost overrun of 27 percent. In addition, almost 70% of all projects fail (Billows, 2015).

(Hass, 2008) argues that testing is necessary because of human error. Like programmers who make mistakes in their code, so testers commit mistakes during the testing process, no matter how perfect the process might be. Yet these mistakes are not done on purpose. The important part is to learn from them and not to trip over the same wire twice.

(Homès, 2012, p. 8) and International Software Testing Qualifications Board (ISTQB) define error as “human action at the root of a defect” and defect as an “impact of human error on a product”, resulting in failure. In this instance, a failure is an unexpected mode of operation resulting from the execution of a defect that does not fulfil the expectations of the end user. It is often the case that the end user will notice the failure. This path gets further elucidated in **Error! Reference source not found.**. However, the end users should not be treated as testers because it is not their purpose. There is a myriad of probable causes for failures, ranging from communication problems and immature design to oversights. According to the Pareto principle, eighty percent of all software errors come from twenty percent of components.

Figure 10 Origin and consequences of failure (Homès, 2012, p. 8)



Two main approaches can be identified in software testing today; The traditional approach where tests are based on specification, systematic analysis, and requirements, leading to a myriad of documentation. Definitions and clear organisation of all activities such as test design and test cases are crucial. The second method is the agile approach where tests are based on the recommendations of Agile Manifesto. The main point is searching for risk and context associated defects. The difference is that agile is more suitable for smaller-scale projects and the traditional method is more applicable to large projects.(Homès, 2012, p. 6)

Some people mix testing with debugging. To avoid confusion, a clear distinction between the two will be made. As has been established before, testing is specifically about identifying defects and errors. Debugging in turn is fixing those defects, finding, and removing the exact causes of those failures. This is not part of the testing process and is not done by testers but rather the programmers because testers often do not have the knowledge or skills to perform this task. (Homès, 2012, p. 10)

Seven principles can be distinguished based on extant literature. They should be considered in every testing environment. Table 2 defines these principles and gives more insight into their purpose.

Table 2: The seven principles of testing (Homès, 2012, pp. 11–14)

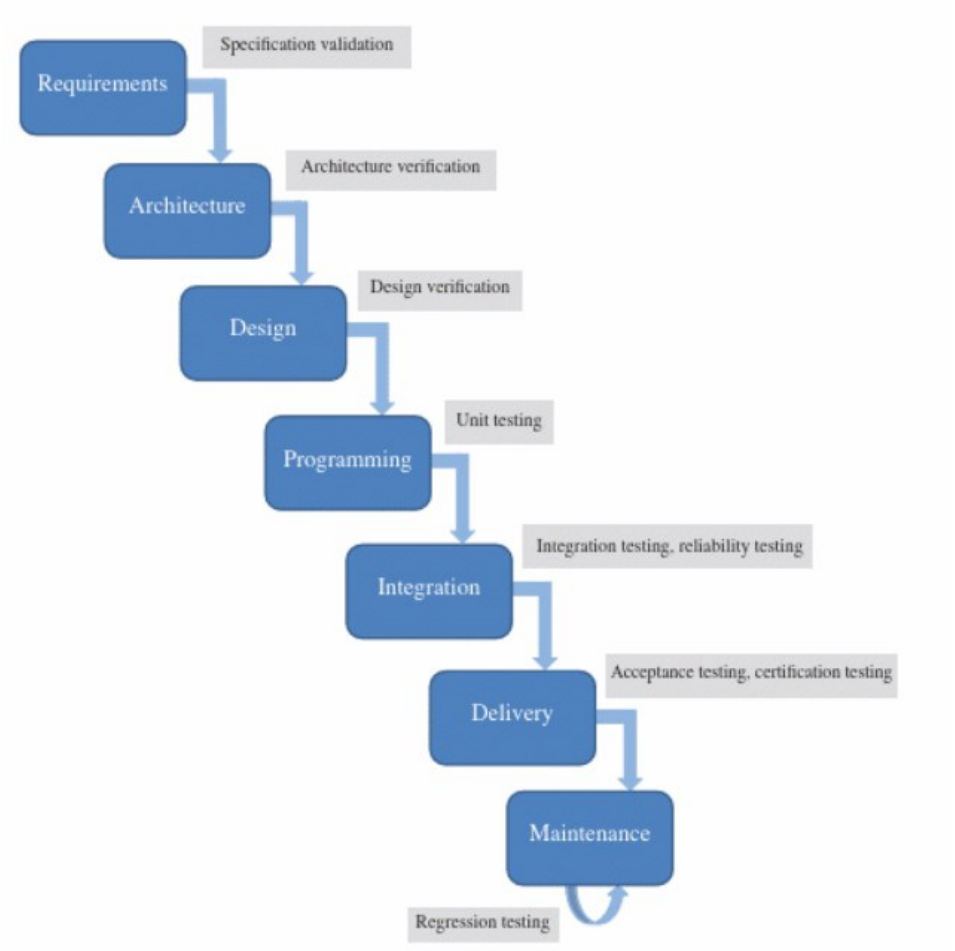
Principle	Explanation
Testing and the presence of defects	It is only possible to test presence of defects, not their absence. Testing does not guarantee identification of all defects within code.
Exhaustive testing is impossible	Not everything can be tested simply because the number of test cases would be infinite, which would be unrealistic. This forces a choice upon the tester of which tests to execute.
Test early in the development cycle	Testing costs increase later in the project and it is more efficient to start testing or find defects as early as possible. Smart design allows avoiding defects and makes identification and creating of test cases easier.
Defects aggregation	It can be established that defects concentrate around similar criteria. If you find a defect in a code, it might be worthwhile to look for other defects in that code. The Pareto principle applies here.
Pesticide paradox	Tests become less efficient when they are used over a prolonged period of time. This happens because same tests will not find any defects if executed repeatedly as software changes. To avoid this, tests must be modified accordingly during the testing process.

Testing depends on context	Testing will be done differently depending on the system. A system used by health care professionals will generally be tested more thoroughly than a system for a business for instance. This also applies to how the software evolves over time. As knowledge increases, tests will become system-specific.
Absence of errors fallacy	Even if the software has no defects, it still might not be suitable for the needs of the customer. If it is not, having no defects makes no difference to architecture, usability, or success of said software.

6.3 **Security testing within the software development life cycle**

Testing is not an independent process. It requires development. If nothing is being developed, there can be no testing. However, it is often the case that testing gets left out or is only conducted at the end of development. As has been established before, testing should be performed on each level of development as soon there is something that can be tested, from initial planning to maintenance after the software release. **Error! Reference source not found.** shows one example of a software development life cycle (SDLC).

Figure 11 A representation of software development life cycle with testing included, the waterfall model (Tchier & Mili, 2015, p. 26)



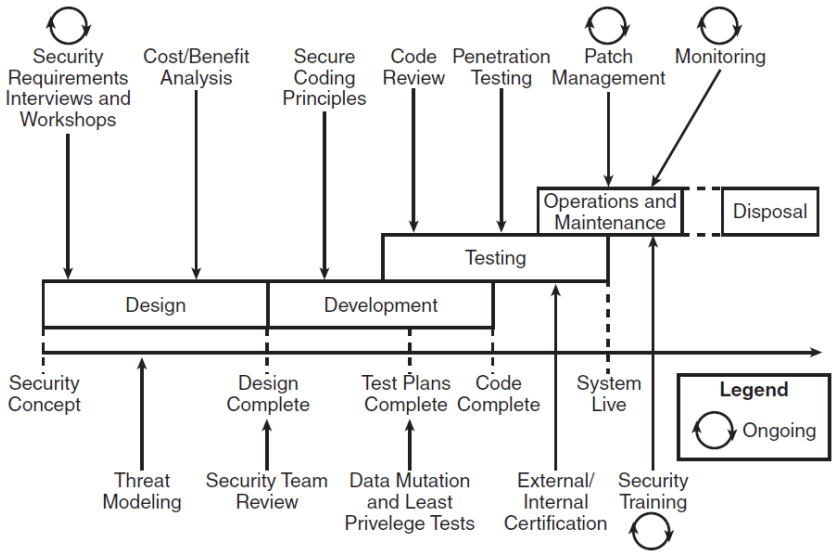
The extant literature agrees that four main types of testing exist. These are unit or component testing, integration testing, system testing and acceptance testing. For the purposes of this thesis, testing types relevant to the client, namely acceptance,

regression and security testing will be further inspected with emphasis on security testing.

Due to cybercrime and potential for disruption and damage, especially through Internet of Things (IoT) due to an increased attack surface, the author of this thesis would argue that security testing should be considered a mandatory part of the testing process. Security testing often composes mostly of non-functional testing but can also include functional test types such as how the software functions while facing an SQL-injection. For instance, escalated privileges for a user means larger attack surface or attack vector. The potential damage from security holes in a program and possible consequences necessitate security testing. The primary objective of security testing is to validate the requirements of the software related to security properties that were discussed further in chapter 3.2.1 such as confidentiality, availability, integrity and so forth. (Schieferdecker, Grossmann, & Schneider, 2012)

It is important to note that while security is considered in several phases of SDLC, no singular tool exists for detecting all defects. Thus, the logical solution for improving software security is to integrate security processes into SDLC processes. This solution was recognised over 15 years ago by organisations such as International Systems Security Engineering Association (ISSEA), the National Institute of Standards and Technology (NIST) and International Standards Organisation (ISO). In addition, Jones & Rastogi discovered that while organisations had SDLC processes, security was not included in them. (Jones & Rastogi, 2004)

Figure 12 SDLC with examples of integrated security processes



6.4 **Acceptance testing**

Acceptance testing begins with the assumption that the product is working in accordance with the requirements established during the requirements analysis. Unlike other types of testing, acceptance testing is not about finding defects but rather highlighting their absence or scarcity. The main thing to remember that the customer and/or end users absolutely must be involved in this testing phase since the aim is to obtain their acceptance of the product. On some occasions they go through the testing on their own or serve as witnesses to give feedback whether the product has reached an acceptable condition and thus has passed the test.

In this form of testing, the entire test object is being tested. It could be the program itself, a report, a use case and so forth. Acceptance testing can be conducted on various levels of software life cycle such as during the design, integration with another software, upon software acquirement by the customer or acceptance of new features. Alpha and beta tests that are described more in detail in Table 3 are forms of acceptance testing.(Hass, 2008, pp. 15–16; Homès, 2012, pp. 64–66; Tchier & Mili, 2015, pp. 24–32)

6.5 **Regression testing**

Regression testing is essentially re-testing modified software via re-execution of tests that were already successfully executed on the entire software. It is used to make sure that no side effects, or regressions, from the change were introduced into the system. The scale of regression testing can vary from rerunning all previously passed tests to testing a functionality of a component to none. Changes that constitute regression testing can include fixing defects, integrating new components or either adding or removing functionalities. In fact, any change to the system raises the need to do regression testing.

This is one of the testing types that is expected to be automated and occurs on every level of testing. For instance, version control software can test different version of the software, scripting software can obtain test inputs, compare them to outputs and generate test reports. It is for the tester to determine, which tests to include in regression testing. Including every test possible would result in a test set that is far too large to execute, even when automated. Tests added should test functionality on a reasonable level, so it can justify the resources used. If regression test fails to pass, first it must be decided, whether the software or test itself is broken. Even if all tests are passed, work still remains since the tests might not be compatible with a later version of the software so the regression test set must keep evolving with the software changes. (Hass, 2008, pp. 70–71; Homès, 2012, p. 72)

These are considered the necessary types of testing, so a product can be finished with optimal value. In addition, there are many more types of testing. They should be utilised at the organisation's discretion and within its policies to deliver the best possible product to the customer. The extant literature agrees that four main types of testing exist. These are unit or component testing, integration testing, system testing and acceptance testing. They are briefly examined here but for the purposes of this thesis, testing types relevant to the client, namely acceptance, regression and security testing will be further inspected with emphasis on security testing. Some examples of these types can be seen at Table 3.

Table 3: Different sub-types of testing

Method	Explanation
Accessibility Testing	Testing for example accessibility for disabled people regarding aspects such as font size, colour, and contrast
Alpha Testing	Combination of unit, integration, and system testing
Beta Testing	Sample of end-users test the application
Black Box Testing	Analysing functionality of the software without knowing internal design, comparing input with output, testing the system as a singular entity. Also known as behavioural testing
Compatibility Testing	Validating how the software operates in different environments, such as different web servers, hardware, and networks.
Compliance Testing	Whether the software is in line with internal or external standards
Graphical User Interface (GUI) Testing	Validating GUI regarding business requirements, such as size of buttons, alignment of text and so forth
Performance Testing	Testing things such as network delay, data rendering, database transaction processing, looking at speed, capacity, stability and scalability
Stress Testing	Testing the software under abnormal conditions. Applying extra load into the system like turning database on and off, consuming more resources such as CPU than normal
Usability Testing	Observing users through their usage and operation.

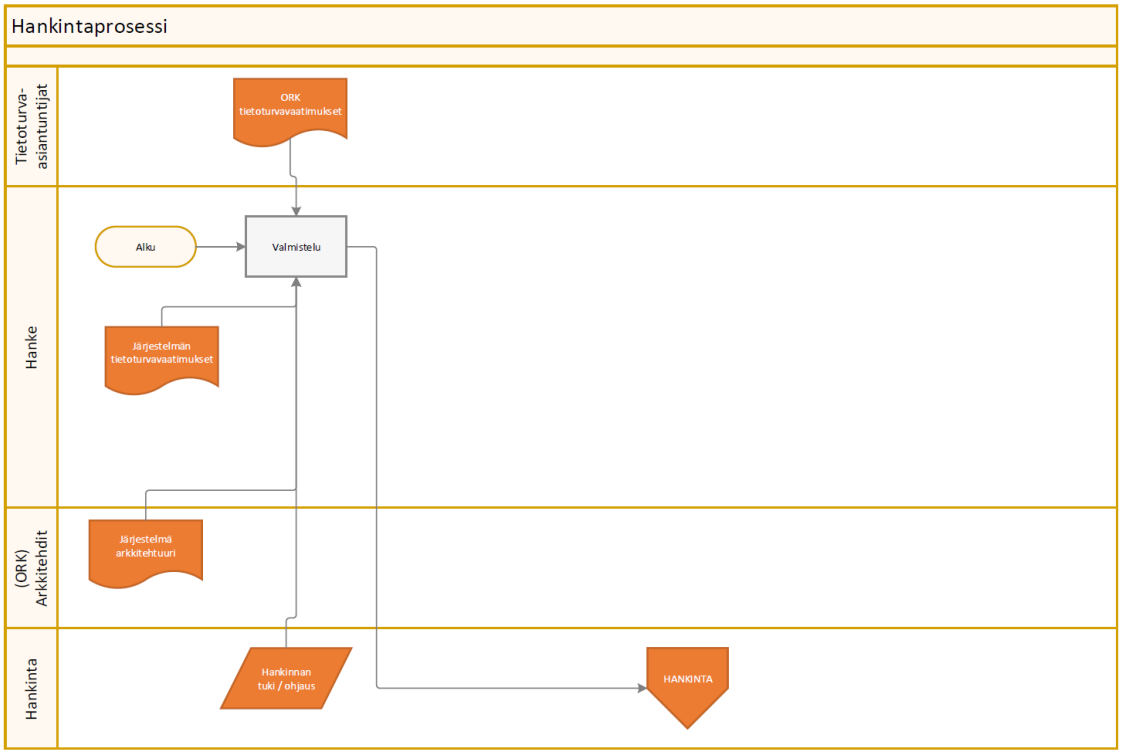
White Box Testing	Testing based on knowledge about the logic of the code, where testers get all necessary information about the system. Tests may include looking at code statements, branches, paths, conditions and so forth

7 INFORMATION SECURITY TESTING SOLUTION FOR THE CLIENT

In this chapter, the focus will be on defining the information security testing process of the client. First, the current process will be mapped using BPMN 2.0 to establish a firm overview of the present form of security testing within client organisation. Afterwards, new process will be established through previously mentioned methods. Documentation relevant to that process, such as testing plan and testing report with security as top priority for the use of the customer will be presented in this chapter. The models presented here are generalised versions of what the client uses to conform with protection levels of documents as stated before.

7.1 Information security process description

Figure 13 Description of the current process in use with client organization



As seen in **Error! Reference source not found.**, the current process begins with information security experts who determine the necessary information security related requirements for the project. These are then given to the project managers in preparation, who also provide project related security requirements. Next, the security

architects provide the framework for the project with support of the management who then proceed to initiate the order of the product. This process has many moving parts and client wishes it to be streamlined.

7.2 **Information security testing plan model**

The model for information security testing plan created for the purposes of the customer as the final product, to be utilised in various software projects of the organisation can be found in the appendix as a separate document. Its main purpose is to streamline process described in previous chapter so all can now be processed through one document instead of multi-departmental work which is much more time consuming.

8 CONCLUSIONS AND FUTURE RECOMMENDATIONS

During the process of creation, client was involved in regular meetings to make sure they stayed connected with the final product. Having the input from the client in line with agile methods of software development proved to be immensely valuable so that the product was delivered according to client expectations.

The testing plan itself is the first version delivered to the client, subject to further changes as relevant processes are being refined or modified based on the needs of the organisation. The client was reportedly happy with the first version, which is subject to further development as needed. The author learned a great deal about information security and how to apply it within existing testing frameworks to provide products that minimize security related threats which are especially important in today's environment, where security is paramount.

REFERENCES

A1QA. (2017). OWASP Top 10 2017. *Blog*. Retrieved from https://www.a1qa.com/blog/owasp-top-10-2017-whats-new/

Ahmad, M. O., Markkula, J., & Ovio, M. (2013). Kanban in software development: A systematic literature review BT - 39th Euromicro Conference Series on Software Engineering and Advanced Applications, SEAA 2013, September 4, 2013 - September 6, 2013, 9–16. https://doi.org/10.1109/SEAA.2013.28

Bertolino, A. (2007). Software testing research: Achievements, challenges, dreams. *FoSE 2007: Future of Software Engineering*, (September), 85–103. https://doi.org/10.1109/FOSE.2007.25

Brown, R. (2014). Computer security threats: A brief history - Direct2Dell. Retrieved May 19, 2019, from https://blog.dell.com/en-us/computer-security-threats-a-brief-history/

Chinosi, M., & Trombetta, A. (2012). BPMN: An introduction to the standard. *Computer Standards and Interfaces*, *34*(1), 124–134. https://doi.org/10.1016/j.csi.2011.06.002

Commission, E. CyberSecurity Act 2018 (2018).

Connolly, K. (2010). No Title. *The Guardian*. Retrieved from https://www.theguardian.com/world/2010/jan/06/2010-bug-millions-germans

Dalal, S., & Singh Chhillar, R. (2012). Case Studies of Most Common and Severe Types of Software System Failure. *International Journal of Advanced Research in Computer Science and Software Engineering*, *2*(8), 2277. Retrieved from www.ijarcsse.com

Firesmith, D. (2013). Using V Models for Testing. *SEI Blog*. Retrieved from https://insights.sei.cmu.edu/sei_blog/2013/11/using-v-models-for-testing.html

Hass, A. M. J. (2008). *Guide to Advanced Software Testing*.

Homès, B. (2012). *Fundamentals of Software Testing*.

Ikonen, M., Pirinen, E., Fagerholm, F., Kettunen, P., & Abrahamsson, P. (2011). On the Impact of Kanban on Software Development.

Jazequel, J.-M., & Meyer, B. (1997). Design by contract: the lessons of Ariane. *Computer*, *30*(1), 129–130. https://doi.org/10.1109/2.562936

Johnson, H. A. (2010). Trello Resource Review. *Journal of the Medical Library*

Association, *31*(2), 299. https://doi.org/10.1097/AUD.0b013e3181ce6231

Jones, R. L., & Rastogi, A. (2004). Secure coding: Building security into the software development life cycle. *Information Systems Security*, *13*(5), 29–39. https://doi.org/10.1201/1086/44797.13.5.20041101/84907.5

Leveson, N. G., & Turner, C. S. (1993). Therac_25.pdf.

Luukkonen, I., Savolainen, S., & Tamminen, M. (n.d.). *Toiminnan ja prosessien mallintaminen Tasot, näkökulmat ja esimerkit SOLEA-hanke Itä-Suomen yliopisto Aalto-yliopisto*.

Martinsuo, M., & Blomqvist, M. (2010). Prosessien mallintaminen osana toiminnan kehittämistä. *… of Technology. Faculty of Business and …*, 1–18. Retrieved from https://dspace.cc.tut.fi/dpub/handle/123456789/6825

Mendeley - Reference Management Software & Researcher Network. (n.d.). Retrieved May 19, 2019, from https://www.mendeley.com/?interaction_required=true

Ministry of Defense. (2015). *Katakri - Tietoturvallisuuden auditointityökalu viranomaisille - 2015*. Retrieved from www.defmin.fi

Ministry of Finance. (2010). *Implenting the Decree on Information Security in Central Government*.

Romanosky, S. (2016). Examining the costs and causes of cyber incidents. *Journal of Cybersecurity*, *2*(2), 121–135. https://doi.org/10.1093/cybsec/tyw001

Saini, H., Rao, Y. S., & Panda, T. C. (2012). Cyber-Crimes and their Impacts : A Review. *International Journal of Engineering Research and Applications*, *2*(2), 202–209.

Schieferdecker, I., Grossmann, J., & Schneider, M. (2012). Model-Based Security Testing. *Electronic Proceedings in Theoretical Computer Science*, *80*(Mbt), 1–12. https://doi.org/10.4204/eptcs.80.1

Soomro, Z. A., Shah, M. H., & Ahmed, J. (2016). Information security management needs more holistic approach: A literature review. *International Journal of Information Management*, *36*(2), 215–225. https://doi.org/10.1016/j.ijinfomgt.2015.11.009

Tchier, F., & Mili, A. (2015). *Software Testing: Concepts and Operations*.

Tugizimana, F. (2015, September). Why Mendeley has proven my ideal research companion - an author's story. Retrieved May 19, 2019, from https://www.elsevier.com/authors-update/story/publishing-tips/mendeley,-a-

companion-in-research

Tzu, S. (n.d.). *The Art of War*. *Online*.

Väestörekisterikeskus. (2017). *Julkisen hallinnon digitaalisen turvallisuuden johtoryhmä*.

Von Solms, R., & Van Niekerk, J. (2013). From information security to cyber security. *Computers and Security*, *38*, 97–102. https://doi.org/10.1016/j.cose.2013.04.004

Whitman, M. E., & Mattord, H. J. (2012). Principles of Information Security Fourth Edition. *Cengage Learning*, 658.

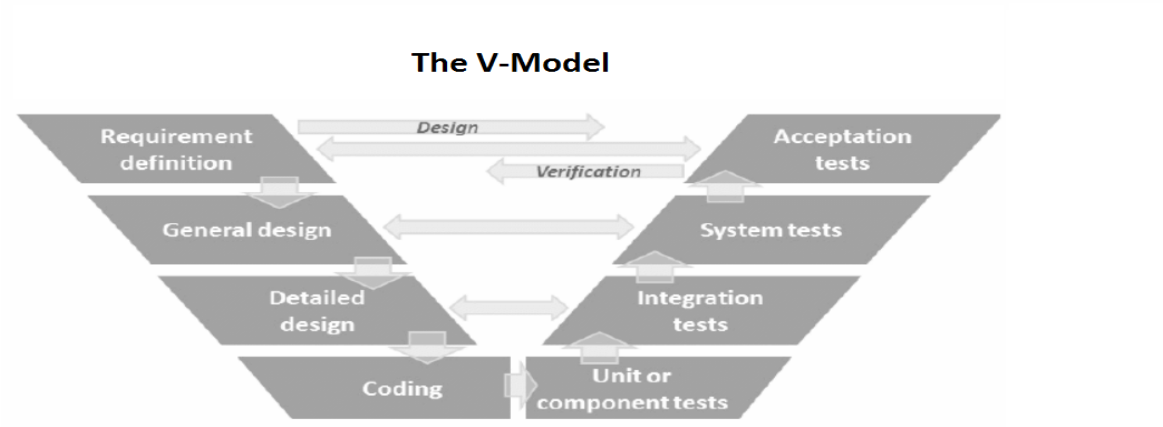
Appendix A: Processing rights of documents with different Protection Levels

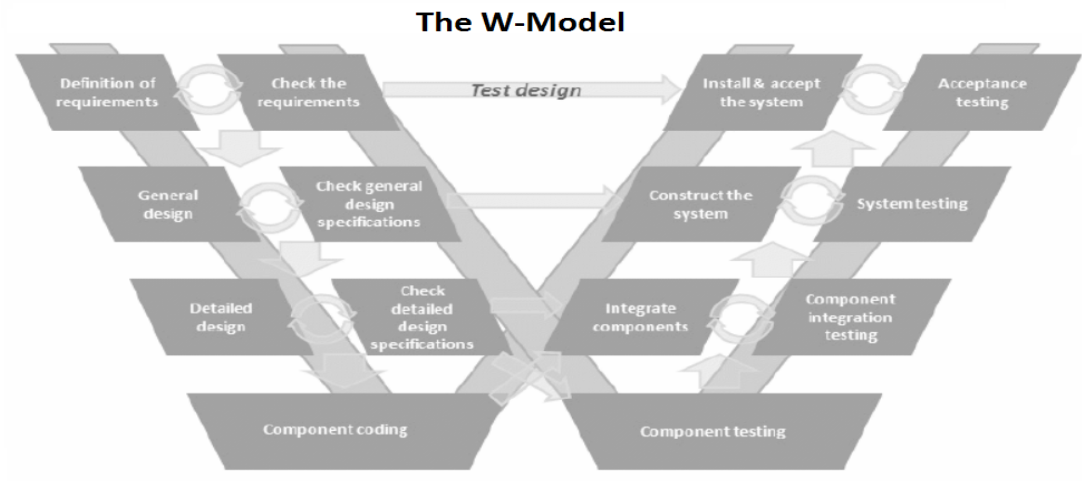
	Protection Level IV	**Protection Level III**	**Protection Level II**	**Protection Level I**
Processing right	Processing right granted	Processing right granted	Processing right granted	Mentioned in distribution, processing right granted
Distribution	According to tasks	According to tasks	Specified by author, based on tasks	Author specifies individual distribution
Recording of processing	Recording of processing events of documents containing information in personal data files of biometric data Recommended for other information	Recording of processing events of documents containing sensitive information in personal data files or biometric data Recommended for other information	-	-
Traceability	No monitoring	No monitoring	Document copy-specific traceability	Document copy-specific traceability
Transfer in open networks	Encrypted or otherwise protected	Encrypted or otherwise protected	Not permitted	Not permitted
Transfer in official networks	As clear text in base and higher information security level networks	As clear text in increased or high information security level networks	As clear text in controlled high information security level networks	Strongly encrypted or otherwise protected in controlled separate networks
Processing in workstation connected to open network	Permitted in base and higher information security level environments	Permitted in increased or high information security level environments	Permitted in controlled high information security level environments	Not permitted
Processing in workstation connected to official network	Permitted in base and higher information security level environments	Permitted in increased or high information security level environments	Permitted in a controlled high information security level environment	Permitted in a high information security level separate network, to which there is no connection from other information networks

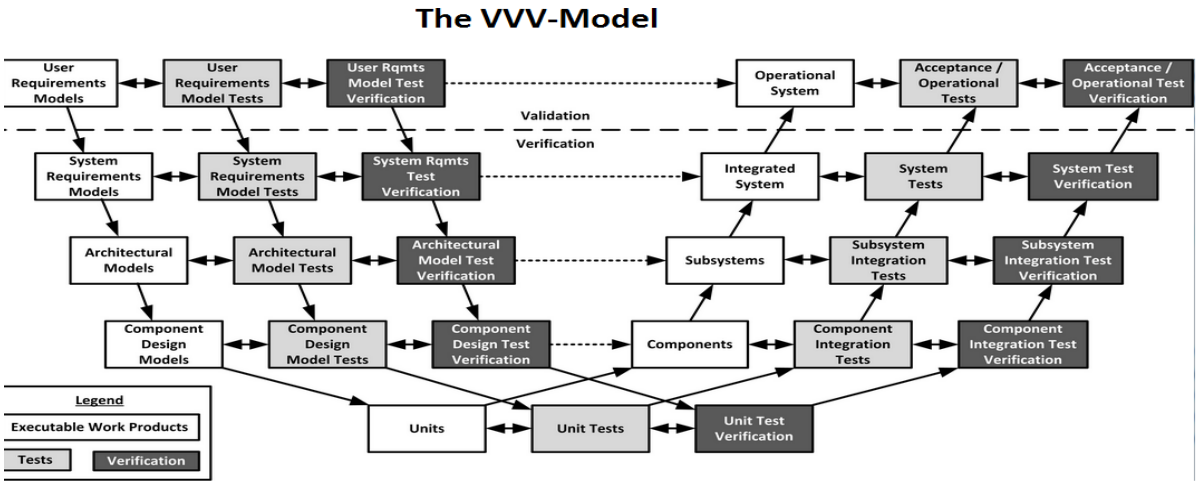
Saving in data storage medium (hard drive, transferable memory)	Protected	Encrypted or otherwise protected	Strongly encrypted or otherwise strongly protected	Strongly encrypted or otherwise strongly protected
Saving on official network server	Protected with user IDs	Encrypted or otherwise protected in an increased information security level environment	Encrypted or otherwise protected in a high information security level environment	Strongly encrypted or otherwise strongly protected if system fulfils high information security level requirements

Appendix B

Appendix B: The V-models of testing (Firesmith, 2013)







Appendix C

Appendix C: Types of static testing and their differences and similarities (Hass, 2008)

	Walk-through	Technical review	Management review	Inspection
Primary purpose	Finding defects	Finding defects	Finding defects	Finding defects
Secondary purpose	Sharing knowledge	Make decisions	Monitor and control progress	Process improvement
Preparation	Usually none	Familiarisation	Familiarisation	Formal preparation
Usage of basis	Rarely	Maybe	Maybe	Always
Leadership of meeting	Author	As appropriate	As appropriate	Trained moderator
Recommended group size	2-7	3 or more	3 or more	3-6
Formal procedure	Usually not	Sometimes	Sometimes	Always
Volume of material	Relatively low	Moderate to high	Moderate to high	Relatively low
Collection of metrics	Usually not	Sometimes	Sometimes	Always
Output	Sometimes an informal report	More or less formal report	More or less formal report	Defect list, measurement, and formal report